

ICN Research Group  
Internet-Draft  
Intended status: Experimental  
Expires: May 1, 2018

H. Asaeda  
X. Shao  
NICT  
T. Turletti  
Inria  
October 28, 2017

Contrace: Traceroute Facility for Content-Centric Network  
draft-asaeda-icnrg-contrace-04

Abstract

This document describes the traceroute facility for Content-Centric Network (CCN), named "Contrace". Contrace investigates: 1) the routing path information per name prefix, device name, and function/application, 2) the Round-Trip Time (RTT) between content forwarder and consumer, and 3) the states of in-network cache per name prefix. In addition, it discovers a gateway that supports different protocols such as CCN and NDN.

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 <https://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 May 1, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	6
2.1. Definitions . . . . .	6
3. Contrace Message Formats . . . . .	7
3.1. Request Message . . . . .	8
3.1.1. Request Block . . . . .	10
3.1.2. Report Block . . . . .	13
3.2. Reply Message . . . . .	14
3.2.1. Reply Block . . . . .	16
3.2.1.1. Reply Sub-Block . . . . .	16
4. Contrace User Behavior . . . . .	19
4.1. Sending Contrace Request . . . . .	19
4.1.1. Gateway Discovery . . . . .	19
4.1.2. Routing Path Information . . . . .	20
4.1.3. In-Network Cache Information . . . . .	20
4.2. Receiving Contrace Reply . . . . .	20
5. Router Behavior . . . . .	21
5.1. Receiving Contrace Request . . . . .	21
5.1.1. Request Packet Verification . . . . .	21
5.1.2. Request Normal Processing . . . . .	21
5.2. Forwarding Contrace Request . . . . .	22
5.3. Sending Contrace Reply . . . . .	23
5.4. Forwarding Contrace Reply . . . . .	24
6. Publisher Behavior . . . . .	24
7. Contrace Termination . . . . .	25
7.1. Arriving at Publisher or Gateway . . . . .	25
7.2. Arriving at Router Having Cache . . . . .	25
7.3. No Route . . . . .	25
7.4. No Information . . . . .	25
7.5. No Space . . . . .	25
7.6. Fatal Error . . . . .	25
7.7. Contrace Reply Timeout . . . . .	26
7.8. Non-Supported Node . . . . .	26
7.9. Administratively Prohibited . . . . .	26
8. Configurations . . . . .	26
8.1. Contrace Reply Timeout . . . . .	26
8.2. HopLimit in Fixed Header . . . . .	26
8.3. Access Control . . . . .	26
9. Diagnosis and Analysis . . . . .	27
9.1. Number of Hops . . . . .	27
9.2. Caching Router and Gateway Identification . . . . .	27

9.3.	TTL or Hop Limit . . . . .	27
9.4.	Time Delay . . . . .	27
9.5.	Path Stretch . . . . .	27
9.6.	Cache Hit Probability . . . . .	27
10.	Security Considerations . . . . .	28
10.1.	Policy-Based Information Provisioning for Request . . .	28
10.2.	Filtering of Contrace Users Located in Invalid Networks	28
10.3.	Topology Discovery . . . . .	29
10.4.	Characteristics of Content . . . . .	29
10.5.	Longer or Shorter Contrace Reply Timeout . . . . .	29
10.6.	Limiting Request Rates . . . . .	29
10.7.	Limiting Reply Rates . . . . .	29
10.8.	Adjacency Verification . . . . .	30
11.	Acknowledgements . . . . .	30
12.	References . . . . .	30
12.1.	Normative References . . . . .	30
12.2.	Informative References . . . . .	30
Appendix A.	Contrace Command and Options . . . . .	31
Authors' Addresses	. . . . .	33

## 1. Introduction

In Content-Centric Network (CCN) or Named-Data Network (NDN), publishers provide content through the network, and receivers retrieve content by name. In this network architecture, routers forward content requests by means of their Forwarding Information Bases (FIBs), which are populated by name-based routing protocols. CCN/NDN also enables receivers to retrieve content from an in-network cache.

In CCN/NDN, while consumers do not generally need to know which content forwarder is transmitting the content to them, operators and developers may want to identify the content forwarder and observe the routing path information per name prefix for troubleshooting or investigating the network conditions.

Traceroute [5] is a useful tool for analyzing the routing conditions in IP networks as it provides intermediate router addresses along the path between source and destination and the Round-Trip Time (RTT) for the path. However, this IP-based network tool cannot trace the name prefix paths used in CCN/NDN. Moreover, given a source-rooted routing path per name prefix, specifying a forwarding source (i.e., router or publisher) for any content is difficult, because we do not always know which branch of the source tree the consumer is on. Additionally, it is not feasible to flood the entire source-rooted tree to find the path from a source to a consumer. Furthermore, such IP-based network tool does not allow the states of the in-network cache to be discovered.

This document describes the specification of "Contrace", an active network measurement tool for investigating the path and caching condition in CCN. Contrace potentially discovers devices and functions/applications in CCN. Contrace is designed based on the work originally published in [4].

Contrace consists of the Contrace user command and the Contrace forwarding function implementation on a content forwarder (e.g., router). The Contrace user (e.g., consumer) invokes the `contrace` command (described in Appendix A) with the name prefix of the content, the device name, or the function (or application) name. The Contrace command initiates the Contrace "Request" message (described in Section 3.1). The Request message, for example, obtains routing path and cache information. When an appropriate adjacent neighbor router receives the Request message, it retrieves cache information. If the router is not the content forwarder for the request, it inserts its "Report" block (described in Section 3.1.2) into the Request message and forwards the Request message to its upstream neighbor router(s) decided by its FIB. These two message types, Contrace Request and Reply messages, are encoded in the CCNx TLV format [1].

In this way, the Contrace Request message is forwarded by routers toward the content publisher, and the Contrace Report record is inserted by each intermediate router. When the Request message reaches the content forwarder (i.e., a router or the publisher who has the specified cache or content), the content forwarder forms the Contrace "Reply" message (described in Section 3.2) and sends it to the downstream neighbor router. The Reply message is forwarded back toward the Contrace user in a hop-by-hop manner. This request-reply message flow, walking up the tree from a consumer toward a publisher, is inspired by the design of the IP multicast traceroute facility [6].

Contrace supports multipath forwarding. The Request messages can be forwarded to multiple neighbor routers. When the Request messages forwarded to multiple routers, the different Reply messages will be forwarded from different routers or publisher. To support this case, PIT entries initiated by Contrace remain until the defined timeout value is expired.

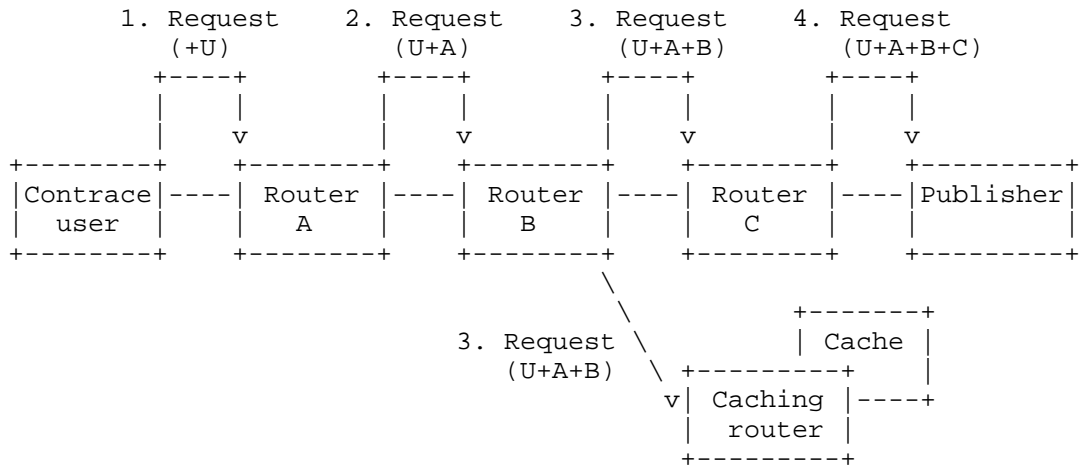


Figure 1: Request messages forwarded by consumer and routers. Contrace user and routers (i.e., Router A,B,C) insert their own Report blocks into the Request message and forward the message toward the content forwarder (i.e., caching router and publisher)

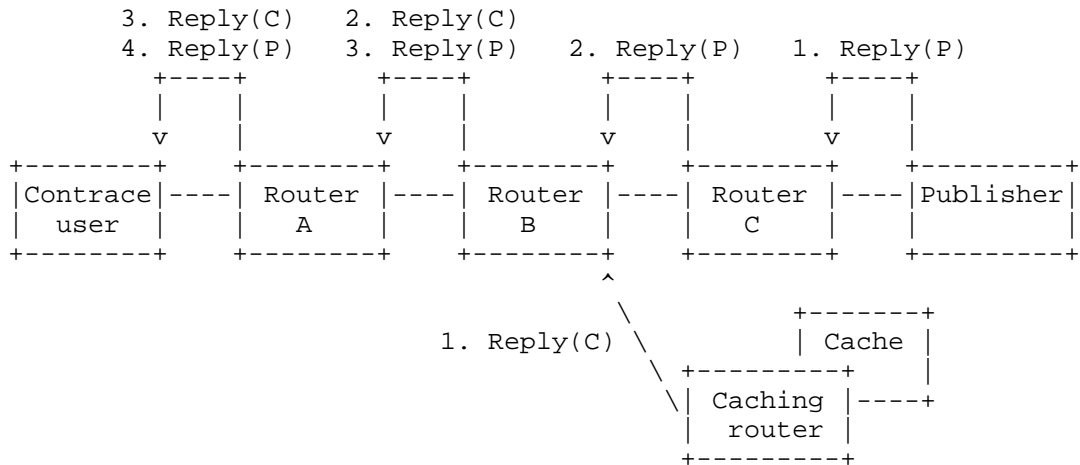


Figure 2: Reply messages forwarded by publisher and routers. Each router forwards the Reply message, and finally the Contrace user receives two Reply messages: one from the publisher and the other from the caching router.

Contrace facilitates the tracing of a routing path and provides: 1) the RTT between content forwarder (i.e., caching router or publisher) and consumer, 2) the states of in-network cache per name prefix, and 3) the routing path information per name prefix.

In addition, Contrace identifies the states of the cache, such as the following metrics for Content Store (CS) in the content forwarder: 1) size of the cached content, 2) number of the cached chunks of the content, 3) number of the accesses (i.e., received Interests) per cache or chunk, and 4) lifetime and expiration time per cache or chunk. The number of received Interests per cache or chunk on the routers indicates the popularity of the content.

Furthermore, Contrace implements policy-based information provisioning that enables administrators to "hide" secure or private information, but does not disrupt the forwarding of messages. This policy-based information provisioning reduces the deployment barrier faced by operators in installing and running Contrace on their routers.

## 2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [2], and indicate requirement levels for compliant Contrace implementations.

### 2.1. Definitions

Since Contrace requests flow in the opposite direction to the data flow, we refer to "upstream" and "downstream" with respect to data, unless explicitly specified.

#### Router

It is a router facilitating name-based content/device/function name or characteristic retrieval in the path between consumer and publisher.

#### Scheme name

It indicates a URI and protocol such as "ccnx:/" and "ndn:/" . This document considers the protocol for name-based content/device/function name or characteristic retrieval.

#### Gateway

It is a router supporting multiple scheme names in the path between consumer and publisher. The router has multiple FIBs for different protocols and establishes the connections with different neighbor routers for each protocol.

#### Node

It is a router, gateway, publisher, or consumer.

**Content forwarder**

It is either a caching router or a publisher that holds the cache (or content) and forwards it to consumers.

**Contrace user**

It is a node that invokes the `contrace` command and initiates the Contrace Request.

**Incoming face**

The face on which data is expected to arrive from the specified name prefix.

**Outgoing face**

The face to which data from the publisher or router is expected to transmit for the specified name prefix. It is also the face on which the Contrace Request messages are received.

**3. Contrace Message Formats**

Contrace uses two message types: Request and Reply. Both messages are encoded in the CCNx TLV format ([1], Figure 3). The Request message consists of a fixed header, Request block TLV Figure 7, and Report block TLV(s) Figure 11. The Reply message consists of a fixed header, Request block TLV, Report block TLV(s), and Reply block/sub-block TLV(s) Figure 14.

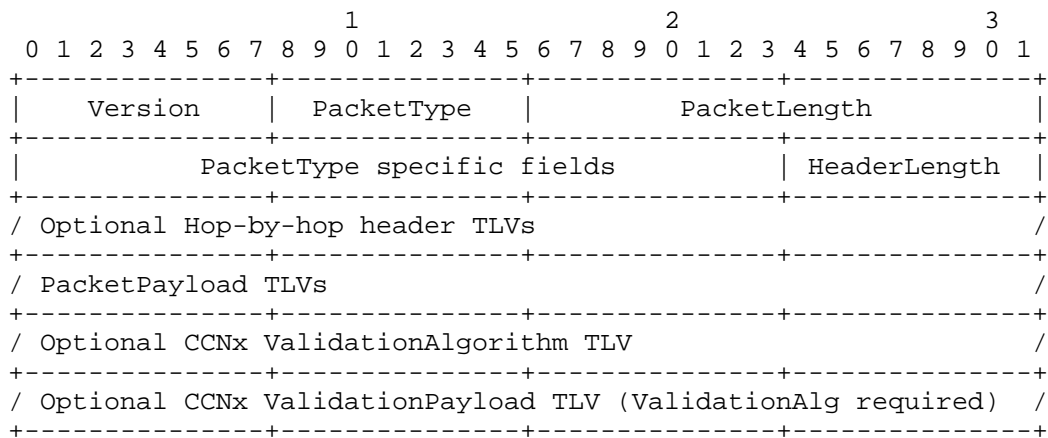


Figure 3: Packet format [1]

The Request and Reply Type values in the fixed header are `PT_REQUEST` and `PT_REPLY`, respectively (Figure 4). These messages are forwarded in a hop-by-hop manner. When the Request message reaches the content forwarder, the content forwarder turns the Request message into a

Reply message by changing the Type field value in the fixed header from PT\_REQUEST to PT\_REPLY and forwards back to the node that has initiated the Request message.

Code	Type name
=====	=====
1	PT_INTEREST [1]
2	PT_CONTENT [1]
3	PT_RETURN [1]
4	PT_REQUEST
5	PT_REPLY

Figure 4: Packet Type Namespace

Each Contrace message MUST begin with a fixed header with either a Request or Reply type value to specify whether it is a Request message or Reply message. Following a fixed header, there can be a sequence of optional hop-by-hop header TLV(s) for a Request message. In the case of a Request message, it is followed by a sequence of Report blocks, each from a router on the path toward the publisher or caching router.

At the beginning of PacketPayload TLVs, one top-level TLV type, T\_TRACE (Figure 5), exists at the outermost level of a CCNx protocol message. This TLV indicates that the Name segment TLV(s) and Reply block TLV(s) would follow in the Request or Reply message.

Code	Type name
=====	=====
1	T_INTEREST [1]
2	T_OBJECT [1]
3	T_VALIDATION_ALG [1]
4	T_VALIDATION_PAYLOAD [1]
5	T_PING
6	T_TRACE

Figure 5: Top-Level Type Namespace

### 3.1. Request Message

When a Contrace user initiates a trace request (e.g., by `contrace` command described in Appendix A), a Contrace Request message is created and forwarded to its upstream router through the Incoming face(s) determined by its FIB.

The Contrace Request message format is as shown in Figure 6. It consists of a fixed header, Request block TLV (Figure 7), Report block TLV(s) (Figure 11), and Name TLV. The Type value of Top-Level



type namespace is T\_TRACE (Figure 5). The Type value for the Report message is PT\_REQUEST.

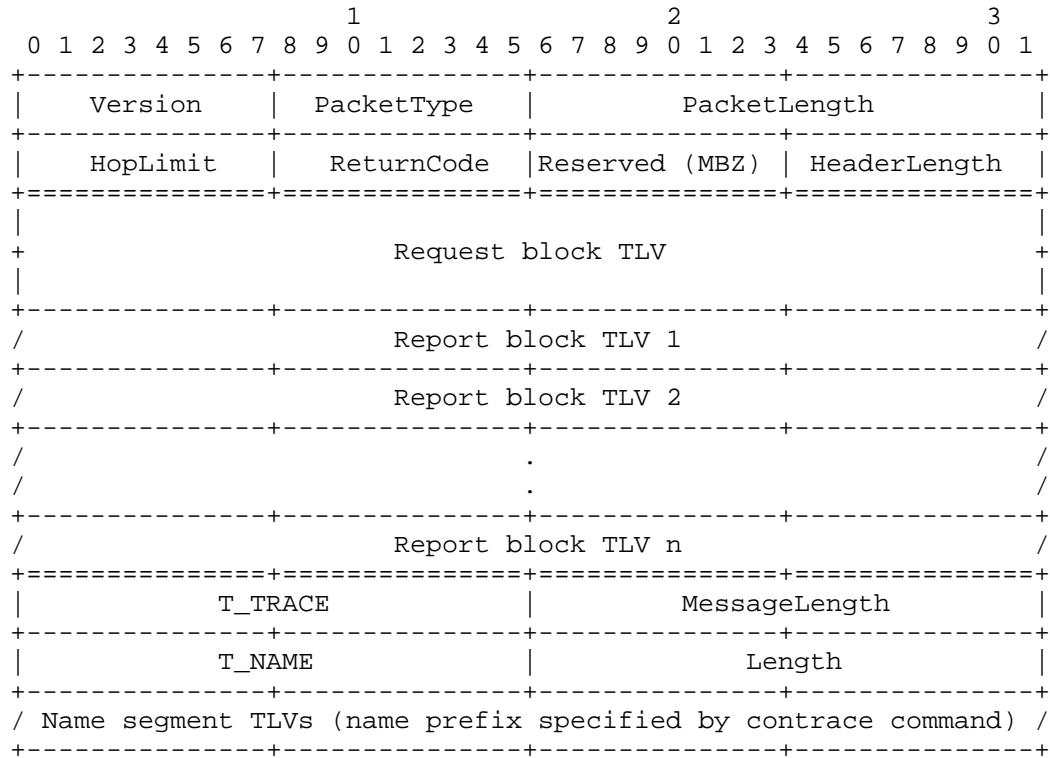


Figure 6: Request message consists of a fixed header, Request block TLV, Report block TLV(s), and Name TLV

HopLimit: 8 bits

HopLimit is a counter that is decremented with each hop. It limits the distance a Request may travel on the network.

ReturnCode: 8 bits

ReturnCode is used for the Reply message. This value is replaced by the content forwarder when the Request message is returned as the Reply message (see Section 3.2). Until then, this field MUST be transmitted as zeros and ignored on receipt.

Value	Name	Description
-----	-----	-----
0x00	NO_ERROR	No error
0x01	WRONG_IF	Contrace Request arrived on an interface to which this router would not forward for the specified name/function toward the publisher.
0x02	INVALID_REQUEST	Invalid Contrace Request is received.
0x03	NO_ROUTE	This router has no route for the name prefix and no way to determine a potential route.
0x04	NO_INFO	This router has no cache information for the specified name prefix, device information, or function.
0x05	NO_SPACE	There was not enough room to insert another Report block in the packet.
0x06	NO_GATAWAY	Contrace Request arrived on a non-gateway router.
0x07	INFO_HIDDEN	Information is hidden from this trace because of some policy.
0x0E	ADMIN_PROHIB	Contrace Request is administratively prohibited.
0x0F	UNKNOWN_REQUEST	This router does not support/recognize the Request message.
0x80	FATAL_ERROR	A fatal error is one where the router may know the upstream router but cannot forward the message to it.

Reserved (MBZ): 8 bits

The reserved fields in the Value field MUST be transmitted as zeros and ignored on receipt.

#### 3.1.1.1. Request Block

When a Contrace user transmits the Request message, it MUST insert the Request block TLV (Figure 7) and the Report block TLV (Figure 11) of its own to the Request message before sending it through the Incoming face(s).

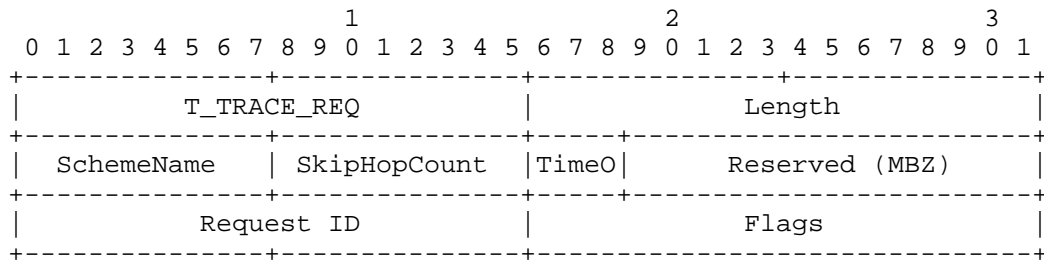


Figure 7: Request block TLV (hop-by-hop header)

Code	Type name
=====	=====
1	T_INTLIFE [1]
2	T_CACHETIME [1]
3	T_MSGHASH [1]
4 - 7	Reserved [1]
8	T_TRACE_REQ
9	T_TRACE_REPORT
%x0FFE	T_PAD [1]
%x0FFF	T_ORG [1]
%x1000-%x1FFF	Reserved [1]

Figure 8: Hop-by-Hop Type Namespace

Type: 16 bits

Format of the Value field. For the single Request block TLV, the type value MUST be T\_TRACE\_REQ. For all the available types for hop-by-hop type namespace, please see Figure 8.

Length: 16 bits

Length of Value field in octets. For the Request block, it MUST be 4 in the current specification.

SchemeName: 8 bits

Currently, the following scheme names are defined.

Code	Scheme name
=====	=====
0	ccnx:/
1	ndn:/
2	cefore:/
%x03-%FF	Not assigned

Figure 9: Scheme Names

SkipHopCount: 8 bits

Number of skipped routers. This value **MUST** be lower than the value of HopLimit at the fixed header.

TimeO: 3 bits

Timeout value (seconds). This Timeout value means a [Contrace Reply Timeout] value (seconds) requested by the Contrace user later described in Section 8.1. A Contrace user requests routers along the path to keep the PIT entry for the Request until this timeout value expires. Note that, because of some security concern (Section 10.5), a router along the path may configure the shorter timeout value than this requested timeout value. In that case, the Request may be timed out and the Contrace user may not receive the Reply as expected.

Request ID: 16 bits

This field is used as a unique identifier for this Contrace Request so that duplicate or delayed Reply messages can be detected.

Flags: 16 bits

The trace conditions specified as the `contrace` command options (described in Appendix A) are transferred in the Flags field. The trace conditions depend on the specified name (i.e., `name_prefix`, `device_name`, or `function_name`) as shown in Figure 10. Note that code `%x01` and `%x02` are exclusive options; that is, only one of them should be turned on at once.

Code	Type name
=====	=====
%x01	Cache retrieval allowing partial match (name_prefix)
%x02	No cache information required (name_prefix)
%x04	Publisher reachability (name_prefix and device_name)
%x08	Force trace. Request to multiple upstream routers simultaneously (name_prefix, device_name, and function_name)
%x16	Discovery of gateway supporting specified scheme name (name_prefix, device_name, and function_name)
%x32	Function's or application's version number retrieval (function_name)
%x64-%x32768	Not assigned

Figure 10: Codes and types specified in Flags field

### 3.1.2. Report Block

A Contrace user and each upstream router along the path would insert its own Report block TLV without changing the Type field of the fixed header of the Request message until one of these routers is ready to send a Reply. In the Report block TLV (Figure 11), the Request Arrival Time and the Node Identifier MUST be inserted.

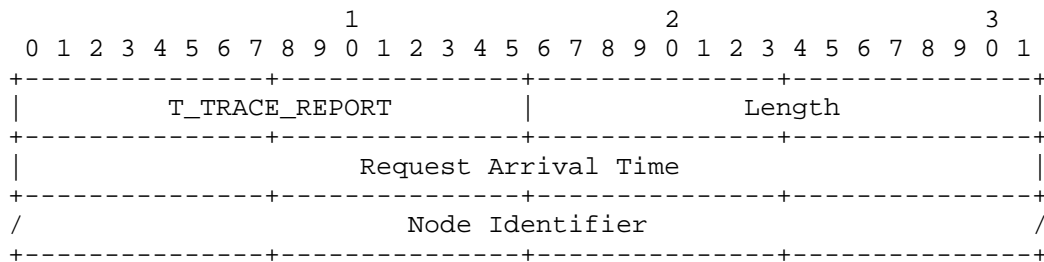


Figure 11: Report block TLV (hop-by-hop header)

Type: 16 bits

Format of the Value field. For the Request block TLV(s), the type value(s) MUST be T\_TRACE\_REPORT.

Length: 16 bits

Length of Value field in octets.

Request Arrival Time: 32 bits

The Request Arrival Time is a 32-bit NTP timestamp specifying the arrival time of the Contrace Request packet at this router. The 32-bit form of an NTP timestamp consists of the middle 32 bits of the full 64-bit form; that is, the low 16 bits of the integer part and the high 16 bits of the fractional part.

The following formula converts from a UNIX timeval to a 32-bit NTP timestamp:

$$\begin{aligned} &\text{request\_arrival\_time} \\ &= ((\text{tv.tv\_sec} + 32384) \ll 16) + ((\text{tv.tv\_nsec} \ll 7) / 1953125) \end{aligned}$$

The constant 32384 is the number of seconds from Jan 1, 1900 to Jan 1, 1970 truncated to 16 bits.  $((\text{tv.tv\_nsec} \ll 7) / 1953125)$  is a reduction of  $((\text{tv.tv\_nsec} / 1000000000) \ll 16)$ .

Note that Contrace does not require all the routers on the path to have synchronized clocks in order to measure one-way latency.

Node Identifier: variable length

This field specifies the Contrace user or the router identifier (e.g., IPv4 address) of the Incoming face on which packets from the publisher are expected to arrive, or all-zeros if unknown or unnumbered. Since we may not always rely on the IP addressing architecture, it would be necessary to define the identifier uniqueness (e.g., by specifying the protocol family) for this field. However, defining such uniqueness is out of scope of this document. Potentially, this field may be defined as a new TLV, which might be defined in the document for the CCNx TLV format[1].

### 3.2. Reply Message

When a content forwarder receives a Contrace Request message from the appropriate adjacent neighbor router, it would insert a Reply block TLV and Reply sub-block TLV(s) of its own to the Request message and turn the Request into the Reply by changing the Type field of the fixed header of the Request message from PT\_REQUEST to PT\_REPLY. The Reply message (see Figure 12) would then be forwarded back toward the Contrace user in a hop-by-hop manner.

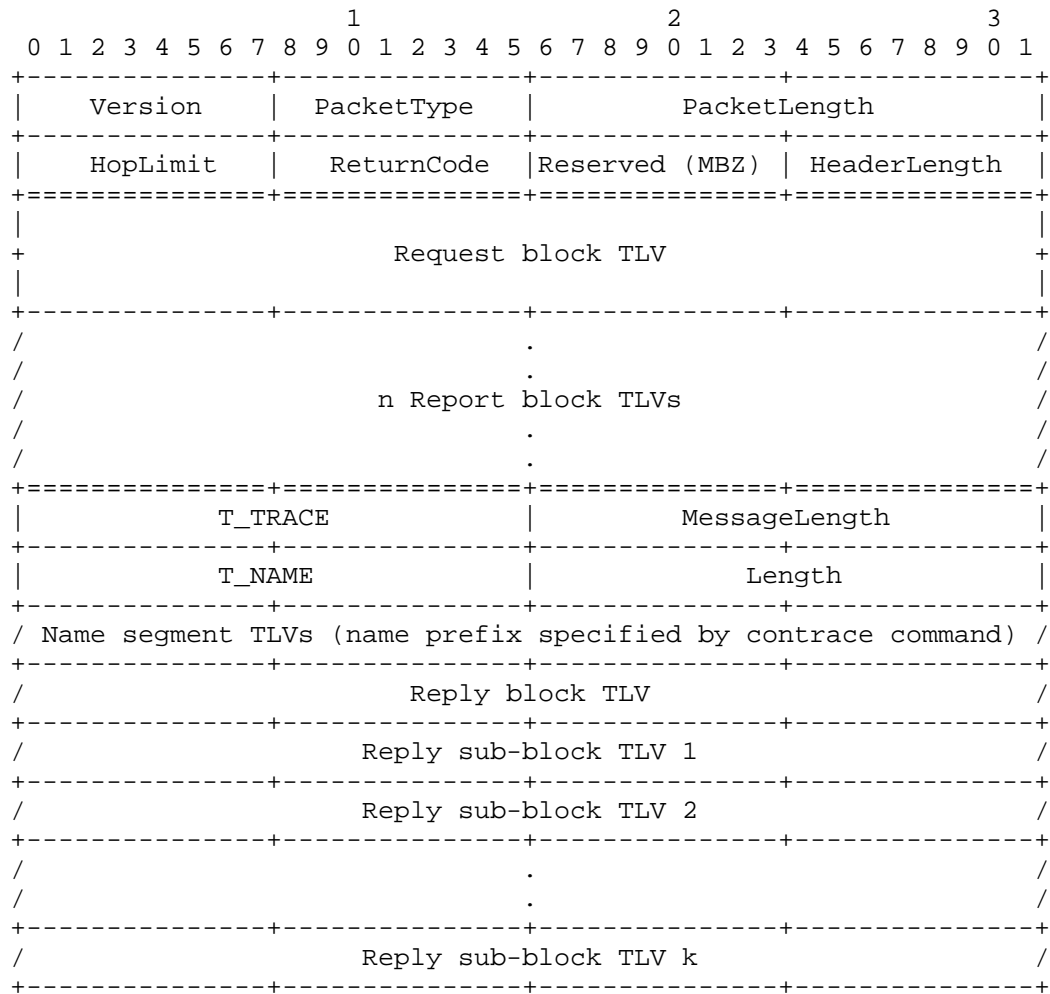


Figure 12: Reply message consists of a fixed header, Request block TLV, Report block TLV(s), Name TLV, and Reply block/sub-block TLV(s)

Code	Type name
=====	=====
0	T_NAME [1]
1	T_PAYLOAD [1]
2	T_KEYIDRESTR [1]
3	T_OBJHASHRESTR [1]
5	T_PAYLDTYPE [1]
6	T_EXPIRY [1]
8	T_TRACE_REPLY
9 - 12	Reserved [1]
%x0FFE	T_PAD [1]
%x0FFF	T_ORG [1]
%x1000-%x1FFF	Reserved [1]

Figure 13: CCNx Message Type Namespace

### 3.2.1. Reply Block

The Reply block TLV is an envelope for Reply sub-block TLV(s) (explained in Section 3.2.1.1).

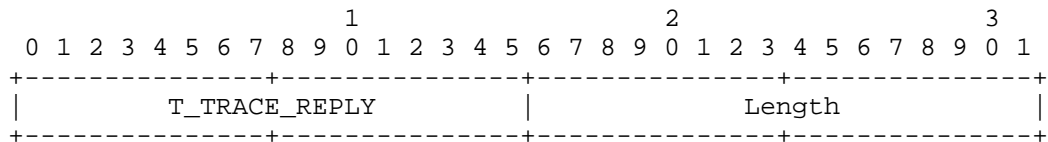


Figure 14: Reply block TLV (packet payload)

Type: 16 bits

Format of the Value field. For the Report block TLV, the type value MUST be T\_TRACE\_REPLY.

Length: 16 bits

Length of Value field in octets. This length is a total length of Reply sub-block(s).

#### 3.2.1.1. Reply Sub-Block

In addition to the Reply block, a router on the traced path will add one or multiple Reply sub-blocks followed by the Reply block before sending the Reply to its neighbor router.

The Reply sub-block is flexible for various purposes. For instance, operators and developers may want to obtain various characteristics of content such as content's ownership and copyright, or other cache



states and conditions. Various information about device or function (or application) may be also retrieved by the variety of Reply sub-blocks. In this document, Reply sub-block TLVs for T\_TRACE\_CONTENT and T\_TRACE\_CONTENT\_OWNER (Figure 15) and for T\_TRACE\_GATEWAY (Figure 16) are defined; other Reply sub-block TLVs will be defined in separate document(s).

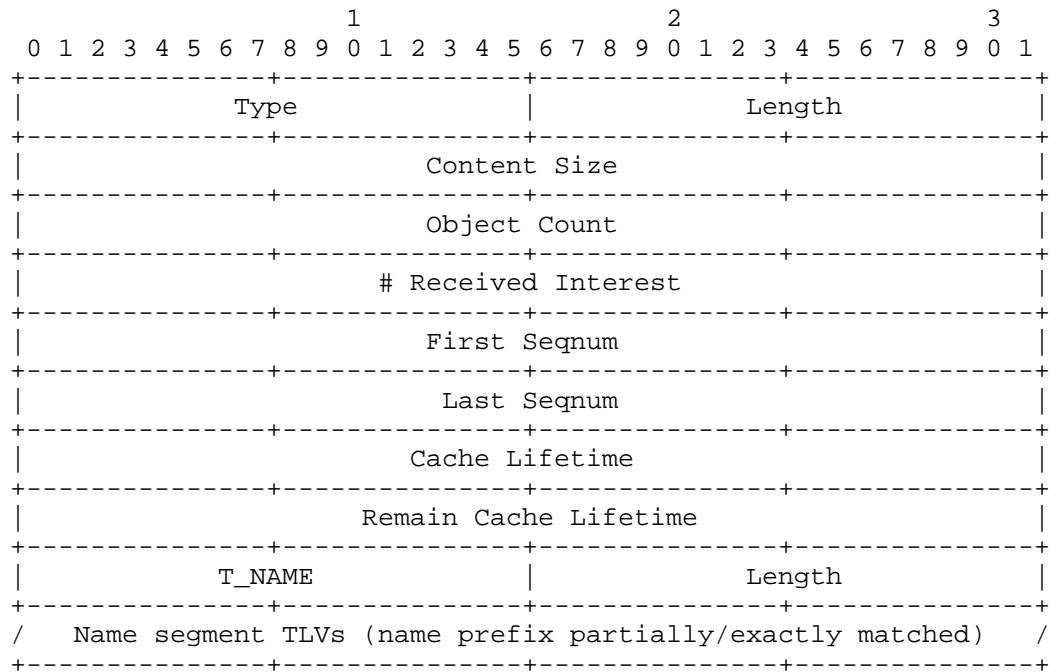


Figure 15: Reply sub-block TLV for T\_TRACE\_CONTENT and T\_TRACE\_CONTENT\_OWNER (packet payload)

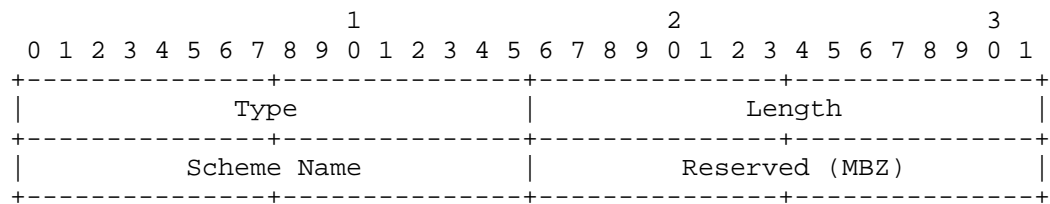


Figure 16: Reply sub-block TLV for T\_TRACE\_GATEWAY (packet payload)

Code	Type name
=====	=====
0	T_TRACE_CONTENT
1	T_TRACE_CONTENT_OWNER
2	T_TRACE_GATEWAY
3	T_TRACE_DEVICE
4	T_TRACE_FUNCTION
%x0FFF	T_ORG
%x1000-%x1FFF	Reserved (Experimental Use)

Figure 17: Contrace Reply Type Namespace

Type: 16 bits

Format of the Value field. For the Reply sub-block TLV, the type value MUST be one of the type value defined in the Contrace Reply Type Namespace (Figure 17). T\_TRACE\_CONTENT is specified when the cache information is replied from a caching router. T\_TRACE\_CONTENT\_OWNER is specified when the content information is replied from a publisher. T\_TRACE\_GATEWAY is used to discover a gateway that has a FIB for the specified scheme name.

Length: 16 bits

Length of Value field in octets.

Scheme Name: 8 bits

The code of the scheme name defined in Figure 9.

Content Size: 32 bits

The total size (MB) of the (cached) content objects. Note that the maximum size expressed by 32 bit field is 65 GB.

Object Count: 32 bits

The number of the (cached) content objects.

# Received Interest: 32 bits

The number of the received Interest messages to retrieve the content.

First Seqnum: 32 bits

The first sequential number of the (cached) content objects.

Last Seqnum: 32 bits

The last sequential number of the (cached) content objects. Above First Seqnum and this Last Seqnum do not guarantee the consecutiveness of the cached content objects.

Cache Lifetime: 32 bits

The elapsed time after the oldest content object in the cache is stored. The Cache Lifetime is a 32-bit NTP timestamp, and the formula converts from a UNIX timeval to a 32-bit NTP timestamp is same as that of Section 3.1.2.

Remain Cache Lifetime: 32 bits

The lifetime of a content object, which is removed first among the cached content objects. The Remain Cache Lifetime is a 32-bit NTP timestamp.

#### 4. Contrace User Behavior

##### 4.1. Sending Contrace Request

A Contrace user initiates a Contrace Request by sending the Request message to the adjacent neighbor router(s) of interest. As a typical example, a Contrace user invokes the `contrace` command (detailed in Appendix A) that forms a Request message and sends it to the user's adjacent neighbor router(s).

When the Contrace user's program initiates a Request message, it **MUST** insert the necessary values, the "Request ID" (in the Request block) and the "Node Identifier" (in the Report block), in the Request and Report blocks. Contrace user's program **MUST** also record the Request ID at the corresponding PIT entry. The Request ID is a unique identifier for the Contrace Request.

After the Contrace user's program sends the Request message, until the Reply times out, the Contrace user's program **MUST** keep the following information; Request ID and Flags specified in the Request block, Node Identifier and Request Arrival Time specified in the Report block, and HopLimit specified in the fixed header.

##### 4.1.1. Gateway Discovery

A Contrace Request can be used for gateway discovery; if a Contrace user invokes a Contrace Request with a scheme name (e.g., `ccnx:/` or `ndn:/`) and the "gateway discovery" flag value (i.e., "%x16" bit as seen in Figure 10), s/he could potentially discover a gateway that

supports different protocols such as CCN and NDN. The Contrace Request for gateway discovery only indicates the routing path information (see Section 4.1.2) and the scheme name whether the router is a gateway or not; it does not provide other information, e.g., cache information.

#### 4.1.2. Routing Path Information

A Contrace user can send a Contrace Request for investigating routing path information for the specified named content. By the Request, the legitimate user can obtain; 1) identifiers (e.g., IP addresses) of intermediate routers, 2) identifier of content forwarder, 3) number of hops between content forwarder and consumer, and 4) RTT between content forwarder and consumer, per name prefix. This Contrace Request is terminated when it reaches the content forwarder. The `contrace` command enables user to obtain both the routing path information and in-network cache information (see below) in a same time.

#### 4.1.3. In-Network Cache Information

A Contrace user can send a Contrace Request for investigating in-network cache information. By this Request, the legitimate user can obtain; 1) size of the cached content, 2) number of the cached chunks of the content, 3) number of the accesses (i.e., received Interests) per cache or chunk, and 4) lifetime and expiration time per cache or chunk, for Content Store (CS) in the content forwarder. This Contrace Request is terminated when it reaches the content forwarder.

#### 4.2. Receiving Contrace Reply

A Contrace user's program will receive one or multiple Contrace Reply messages from the adjacent neighbor router that has previously received and forwarded the Request message(s). When the program receives the Reply, it MUST compare the kept Request ID and the Request ID noted in the Reply. If they do not match, the Reply message SHOULD be silently discarded.

If the number of the Report blocks in the received Reply is more than the initial `HopLimit` value (which was inserted in the original Request) + 1, the Reply SHOULD be silently ignored.

After the Contrace user has determined that s/he has traced the whole path or as much as s/he can expect to, s/he might collect statistics by waiting a timeout. Useful statistics provided by Contrace can be seen in Section 9.

## 5. Router Behavior

### 5.1. Receiving Contrace Request

#### 5.1.1. Request Packet Verification

Upon receiving a Contrace Request message, a router MUST examine whether the message comes from a valid adjacent neighbor node. If it is invalid, the Request MUST be silently ignored. The router next examines the value of the "HopLimit" in the fixed header and the value of the "SkipHopCount" in the Request block (Figure 7). If SkipHopCount value is equal or more than the HopLimit value, the Request MUST be silently ignored.

#### 5.1.2. Request Normal Processing

When a router receives a Contrace Request message, it performs the following steps.

1. HopLimit and SkipHopCount are counters that are decremented with each hop. The router terminates the Contrace Request when the HopLimit value becomes zero. Until the SkipHopCount value becomes zero, the router forwards the Contrace Request messages to the upstream router(s) (if it knows) without adding its own Report block and without replying the Request. If the router does not know the upstream router(s), without depending on the SkipHopCount value, it replies the Contrace Reply message with NO\_ROUTE return code.
2. The router examines the Flags field of the Request block of received Contrace Request. If the flag value indicates "%x00" or "%x01" bit (as seen in Figure 10) for "cache information discovery", the router examines its FIB and CS. If the router caches the specified content, it inserts own Report block to the message and sends the Reply message with own Reply block and sub-block. If the router does not cache the specified content but knows the neighbor router(s) for the specified name prefix, it inserts own Report block and forwards the Request to the upstream neighbor(s). If the router does not cache the specified content and does not know the upstream neighbor router(s) for the specified name prefix, it replies the Contrace Reply message with NO\_ROUTE return code.
3. If the flag value indicates "%x02" bit for "routing path information discovery", the router examines its FIB and CS. If the router caches the specified content, it inserts own Report block to the message and sends the Reply message with own Reply block. The router does not insert any Reply sub-block here. If

the router does not cache the specified content but knows the neighbor router(s) for the specified name prefix, it inserts own Report block and forwards the Request to the upstream neighbor(s). If the router does not cache the specified content and does not know the upstream neighbor router(s) for the specified name prefix, it replies the Contrace Reply message with NO\_ROUTE return code.

4. If the flag value indicates "%x04" bit for "publisher discovery", the node receiving the Request message examines whether it owns the requested content as the publisher. If it is the publisher, it sends the Reply message with own Report block and sub-block. If the node is not the publisher but know the upstream neighbor router(s) for the specified name prefix, it adds the own Report block and forwards the Request to the neighbor(s). If the node is not the publisher and does not know the upstream neighbor router(s) for the specified name prefix, it replies the Contrace Reply message with NO\_ROUTE return code.
5. When a router receives a Contrace Request in which the "gateway discovery" flag (i.e., "%x16") is set in the Flags field and a scheme name is specified, the router examines whether it has the FIB for the specified scheme name and the connections with the neighbor router(s) for the scheme protocol. If the router is the gateway, it sends the Reply message back toward the Contrace user. If the router does not have the FIB for the specified scheme name or does not connect to any neighbor router for the specified scheme name, the router returns the Reply with NO\_GATEWAY return code.

## 5.2. Forwarding Contrace Request

When a router decides to forward a Request message with its Report block to its upstream router(s), it specifies the Request Arrival Time and Node Identifier in the Report block of the Request message. The router then forwards the Request message upstream toward the publisher or caching router based on the FIB entry.

When the router forwards the Request message, it MUST record the Request ID at the corresponding PIT entry. The router can later decide the PIT entry to correctly forward back the Reply message even if it receives multiple Reply messages within the same timeout period. (See below.)

Contrace supports multipath forwarding. The Request messages can be forwarded to multiple neighbor routers. Some router may have strategy for multipath forwarding; when it sends Interest messages to multiple neighbor routers, it may delay or prioritize to send the

message to the upstream routers. The Contrace Request, as the default, complies with such strategy; a Contrace user could trace the actual forwarding path based on the strategy. On the other hand, there may be the case that a Contrace user wants to discover all potential forwarding paths based on routers' FIBs. If a Contrace user invokes a Contrace Request with the force flag value (i.e., "%x08" bit as seen in Figure 10), the forwarding strategy will be ignored and the router sends Requests to multiple upstream routers simultaneously, and the Contrace user could trace the all potential forwarding paths.

When the Request messages forwarded to multiple routers, the different Reply messages will be forwarded from different routers or publisher. To support this case, PIT entries initiated by Contrace remain until the configured Contrace Reply Timeout (Section 8.1) passes. In other words, unlike the ordinary Interest-Data communications in CCN, the router SHOULD NOT remove the PIT entry created by the Contrace Request before the timeout value expires, even if the router receives the Contrace Reply.

Contrace Requests SHOULD NOT result in PIT aggregation in routers during the Request message transmission.

### 5.3. Sending Contrace Reply

When a router decides to send a Reply message to its downstream neighbor router or the Contrace user with NO\_ERROR return code, it inserts a Report block having the Request Arrival Time and Node Identifier to the hop-by-hop TLV header of the Request message. And then the router inserts the corresponding Reply block and Reply sub-block to the payload. The router does not insert any Reply block/sub-block if there is an error. The router finally changes the Type field in the fixed header from PT\_REQUEST to PT\_REPLY and forwards the message back as the Reply toward the Contrace user in a hop-by-hop manner.

When a router decides to send the Reply message for the Request for the cache or routing path information discovery, it forms the Reply message including a Reply block and a Reply sub-block with the T\_TRACE\_CONTENT type value (Figure 15) and various cache information. After the router puts the NO\_ERROR return code in the fixed header, it sends the Reply back toward the Contrace user.

When a router decides to send the Reply message for the Request for the publisher discovery, it forms the Reply message including a Reply block and a Reply sub-block with the T\_TRACE\_CONTENT\_OWNER type value (Figure 15) and various cache information. After the router puts the

NO\_ERROR return code in the fixed header, it sends the Reply back toward the Contrace user.

When a router decides to send the Reply message for the Request for the gateway discovery, it forms the Reply message including a Reply block and a Reply sub-block with the T\_TRACE\_GATEWAY type value (Figure 16) and the scheme name (Figure 9). After the router puts the NO\_ERROR return code in the fixed header, it sends the Reply back toward the Contrace user.

If a router cannot continue the Request, it MUST put an appropriate ReturnCode in the Request message, change the Type field value in the fixed header from PT\_REQUEST to PT\_REPLY, and forward the Reply message back toward the Contrace user, to terminate the request. See Section 7.

#### 5.4. Forwarding Contrace Reply

When a router receives a Contrace Reply whose Request ID matches the one in the original Contrace Request block TLV from a valid adjacent neighbor node, it MUST relay the Contrace Reply back to the Contrace user. If the router does not receive the corresponding Reply within the [Contrace Reply Timeout] period, then it removes the corresponding PIT entry and terminates the trace.

Contrace Replies MUST NOT be cached in routers upon the Reply message transmission.

#### 6. Publisher Behavior

Upon receiving a Contrace Request message, a publisher MUST examine whether the message comes from a valid adjacent neighbor node. If it is invalid, the Request SHOULD be silently ignored.

If a publisher cannot accept the Request, it will note an appropriate ReturnCode in the Request message, change the Type field value in the fixed header from PT\_REQUEST to PT\_REPLY, and forward the message as the Reply back to the Contrace user. See Section 7 for details.

If a publisher accepts the Request forwarded by a valid adjacent neighbor node, it retrieves the local content information. The Reply message having a Reply block and Reply sub-block is transmitted back to the neighbor node that had forwarded the Request message.



## 7. Contrace Termination

When performing an expanding hop-by-hop trace, it is necessary to determine when to stop expanding. There are several cases an intermediate router might return a Reply before a Request reaches the caching router or the publisher.

### 7.1. Arriving at Publisher or Gateway

A Contrace Request can be determined to have arrived at the publisher or gateway.

### 7.2. Arriving at Router Having Cache

A Contrace Request can be determined to have arrived at the router having the specified content cache within the specified HopLimit.

### 7.3. No Route

If the router cannot determine the routing paths or neighbor routers for the specified name prefix, device name, or function within the specified HopLimit, the router MUST note a ReturnCode of NO\_ROUTE in the fixed header of the message, and forwards the message as the Reply back to the Contrace user.

### 7.4. No Information

If the router does not have any information about the specified name prefix, device name, or function within the specified HopLimit, the router MUST note a ReturnCode of NO\_INFO in the fixed header of the message, and forwards the message as the Reply back to the Contrace user.

### 7.5. No Space

If appending the Report block would make the Contrace Request packet longer than the MTU of the Incoming face, or longer than 1280 bytes (especially in the situation supporting IPv6 as the payload [3]), the router MUST note a ReturnCode of NO\_SPACE in the fixed header of the message, and forwards the message as the Reply back to the Contrace user.

### 7.6. Fatal Error

A Contrace Request has encountered a fatal error if the last ReturnCode in the trace has the 0x80 bit set (see Section 3.1).

### 7.7. Contrace Reply Timeout

If a Contrace user or a router encounters the Request or Reply message whose expires its own [Contrace Reply Timeout] value (Section 8.1), which is used to time out a Contrace Reply such as the case of Section 7.8.

### 7.8. Non-Supported Node

Cases will arise in which a router or a publisher along the path does not support Contrace. In such cases, a Contrace user and routers that forward the Contrace Request will time out the Contrace request.

### 7.9. Administratively Prohibited

If Contrace is administratively prohibited, a router or a publisher rejects the Request message, and the router or the publisher, or its downstream router will reply the Contrace Reply with the ReturnCode of ADMIN\_PROHIB.

## 8. Configurations

### 8.1. Contrace Reply Timeout

The [Contrace Reply Timeout] value is used to time out a Contrace Reply. Both Contrace users and routers can configure their own Contrace Reply Timeout values. Contrace users, for example, can configure the timeout value by the `contrace` command. The default [Contrace Reply Timeout] value is 4 (seconds). Routers may want to configure the short timeout values because of some security concern, e.g., Section 10.5. However, the [Contrace Reply Timeout] value SHOULD NOT be larger than 6 (seconds) and SHOULD NOT be lower than 3 (seconds).

### 8.2. HopLimit in Fixed Header

If a Contrace user does not specify the HopLimit value in a fixed header for a Request message as the HopLimit, the HopLimit is set to 32. Note that a Contrace user specifies 0 as the HopLimit, it is an invalid Request and discarded.

### 8.3. Access Control

A router MAY configure the valid or invalid networks to enable an access control. The access control can be defined per name prefix, such as "who can retrieve which name prefix". See Section 10.2.

## 9. Diagnosis and Analysis

### 9.1. Number of Hops

A Contrace Request message is forwarded in a hop-by-hop manner and each forwarding router appended its own Report block. We can then verify the number of hops to reach the content forwarder or the publisher.

### 9.2. Caching Router and Gateway Identification

It is possible to identify the caching routers or a gateway in the path from the Contrace user to the content forwarder, while some routers may hide their identifier (with all-zeros) in the Report blocks (Section 10.1).

### 9.3. TTL or Hop Limit

By taking the HopLimit from the content forwarder and forwarding TTL threshold over all hops, it is possible to discover the TTL or hop limit required for the content forwarder to reach the Contrace user.

### 9.4. Time Delay

If the routers have synchronized clocks, it is possible to estimate propagation and queuing delay from the differences between the timestamps at successive hops. However, this delay includes control processing overhead, so is not necessarily indicative of the delay that data traffic would experience.

### 9.5. Path Stretch

By getting the path stretch " $d / P$ ", where " $d$ " is the hop count of the data and " $P$ " is the hop count from the consumer to the publisher, we can measure the improvement in path stretch in various cases, such as different caching and routing algorithms. We can then facilitate investigation of the performance of the protocol.

### 9.6. Cache Hit Probability

Contrace can show the number of received interests per cache or chunk on a router. By this, Contrace measures the content popularity (i.e., the number of accesses for each content/cache), and you can investigate the routing/caching strategy in networks.

## 10. Security Considerations

This section addresses some of the security considerations.

### 10.1. Policy-Based Information Provisioning for Request

Although Contrace gives excellent troubleshooting cues, some network administrators or operators may not want to disclose everything about their network to the public, or may wish to securely transmit private information to specific members of their networks. Contrace provides policy-based information provisioning allowing network administrators to specify their response policy for each router.

The access policy regarding "who is allowed to retrieve" and/or "what kind of information" can be defined for each router. For the former access policy, routers having the specified content can examine the signature enclosed in the Request message and decide whether they should notify the content information in the Reply or not. If the routers decide to not notify the content information, they reply the Contrace Reply with the ReturnCode of ADMIN\_PROHIB without appending any Reply (sub-)block TLV. For the latter policy, the permission, whether (1) All (all cache information is disclosed), (2) Partial (cache information with the particular name prefix can (or cannot) be disclosed), or (3) Deny (no cache information is disclosed), is defined at routers.

On the other hand, we entail that each router does not disrupt forwarding Contrace Request and Reply messages. When a Request message is received, the router SHOULD insert Report block. Here, according to the policy configuration, the Node Identifier field in the Report block MAY be null (i.e., all-zeros), but the Request Arrival Time field SHOULD NOT be null. At last, the router SHOULD forward the Request message to the upstream router toward the content forwarder if no fatal error occurs.

### 10.2. Filtering of Contrace Users Located in Invalid Networks

A router MAY support an access control mechanism to filter out Requests from invalid Contrace users. For it, invalid networks (or domains) could, for example, be configured via a list of allowed/disallowed networks (as seen in Section 8.3). If a Request is received from the disallowed network (according to the Node Identifier in the Request block), the Request SHOULD NOT be processed and the Reply with the ReturnCode of INFO\_HIDDEN may be used to note that. The router MAY, however, perform rate limited logging of such events.

### 10.3. Topology Discovery

Contrace can be used to discover actively-used topologies. If a network topology is a secret, Contrace Requests may be restricted at the border of the domain, using the ADMIN\_PROHIB return code.

### 10.4. Characteristics of Content

Contrace can be used to discover what publishers are sending to what kinds of contents. If this information is a secret, Contrace Requests may be restricted at the border of the domain, using the ADMIN\_PROHIB return code.

### 10.5. Longer or Shorter Contrace Reply Timeout

Routers can configure the Contrace Reply Timeout (Section 8.1), which is the allowable timeout value to keep the PIT entry. If routers configure the longer timeout value, there may be an attractive attack vector against PIT memory. Moreover, especially when the force option (Section 5.2) is specified for the Contrace Request, a number of Reply messages may come back and cause a response storm. (See Section 10.7 for rate limiting to avoid the storm). In order to avoid DoS attacks, routers may configure the shorter timeout value than the user-configured Contrace timeout value. However, if it is too short, the Request may be timed out and the Contrace user does not receive the all Replies and only retrieves the partial path information (i.e., information about part of the tree).

There may be the way to allow for incremental exploration (i.e., to explore the part of the tree the previous operation did not explore), whereas discussing such mechanism is out of scope of this document.

### 10.6. Limiting Request Rates

A router may limit Contrace Requests by ignoring some of the consecutive messages. The router MAY randomly ignore the received messages to minimize the processing overhead, i.e., to keep fairness in processing requests, or prevent traffic amplification. No error is returned. The rate limit is left to the router's implementation.

### 10.7. Limiting Reply Rates

Contrace supporting multipath forwarding may result in one Request returning multiple Reply messages. In order to prevent abuse, the routers in the traced path MAY need to rate-limit the Replies. No error is returned. The rate limit function is left to the router's implementation.

## 10.8. Adjacency Verification

Contrace Request and Reply messages MUST be forwarded by adjacent neighbor nodes or routers. Forwarding Contrace messages given from non-adjacent neighbor nodes/routers MUST be prohibited. Such invalid messages SHOULD be silently discarded. Note that defining the secure way to verify the adjacency cannot rely on the way specified in CCNx message format or semantics. An adjacency verification mechanism and the corresponding TLV for adjacency verification using hop-by-hop TLV header will be defined in a separate document.

## 11. Acknowledgements

The authors would like to thank Spyridon Mastorakis, Ilya Moiseenko, and David Oran for their valuable comments and suggestions on this document.

## 12. References

### 12.1. Normative References

- [1] Mosko, M., Solis, I., and C. Wood, "CCNx Messages in TLV Format", draft-irtf-icnrg-ccnxmessages-04 (work in progress), March 2017.
- [2] Bradner, S., "Key words for use in RFCs to indicate requirement levels", RFC 2119, March 1997.
- [3] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

### 12.2. Informative References

- [4] Asaeda, H., Matsuzono, K., and T. Turlatti, "Contrace: A Tool for Measuring and Tracing Content-Centric Networks", IEEE Communications Magazine, Vol.53, No.3, pp.182-188, March 2015.
- [5] Malkin, G., "Traceroute Using an IP Option", RFC 1393, January 1993.
- [6] Asaeda, H., Mayer, K., and W. Lee, "Mtrace Version 2: Traceroute Facility for IP Multicast", draft-ietf-mboned-mtrace-v2-17 (work in progress), March 2017.

## Appendix A. Contrace Command and Options

The `contrace` command enables the Contrace user to investigate the routing path based on the name prefix of the content (e.g., `ccnx:/news/today`), device name, and function (or application) name. The name prefix, device name, and function name (or application name) are mandatory but exclusive options; that is, only one of them should be used with the `contrace` command at once.

The usage of `contrace` command is as follows:

Usage: `contrace [-P] [-g] [-f] [-n] [-o] [-r hop_count] [-s hop_count] [-w wait_time] name_prefix; or,`

Usage: `contrace [-r hop_count] [-s hop_count] [-w wait_time] device_name | function_name (or application_name)`

**name\_prefix**

Name prefix of the content (e.g., `ccnx:/news/today`) the Contrace user wants to trace. If the Contrace user specifies only a scheme name, e.g., `"ccnx:/"`, s/he must specify `"-g"` option (i.e., `contrace -g ccnx:/`). In that case, the Contrace user discovers the router having the FIB of the specified scheme name and the RTT between Contrace user and the router. The `"-P"` option allows a partial match for the name prefix; otherwise, an exact match is required.

**device\_name**

Device name (e.g., `ccnx:/%device/server-A`, `ccnx:/%device/sensor-123`) the Contrace user wants to trace. Here, we assume the `contrace` command with the `"%device"` prefix indicates the trace request for specified device/server/node, but defining the syntax of device name specification is [TBD].

**function\_name (or application\_name)**

Function name (e.g., `ccnx:/%function/firewall`, `ccnx:/%function/transcoding/mpeg2-h.264`) or application name (e.g., `ccnx:/%application/mplayer`) the Contrace user wants to trace. Here, we assume the `contrace` command with the `"%function"` or `"%application"` prefix indicates the trace request for specified function or application, but defining the syntax of function or application name specification is [TBD].

**g option**

This option enables to discover a gateway that supports specified scheme name and has multiple FIBs. When a Contrace user specifies only a scheme name, e.g., `"ccnx:/"`, this option must be specified and other content name prefix is ignored.

**f option**

This option enables to ignore the forwarding strategy and send Contrace Requests to multiple upstream routers simultaneously. The Contrace user could then trace the all potential forwarding paths.

**n option**

This option can be specified if a Contrace user only needs the routing path information to the specified content/cache and RTT between Contrace user and content forwarder (i.e., cache information is not given).

**o option**

This option enables to trace the path to the content publisher. If this option is specified, each router along the path to the publisher only forwards the Request message; it inserts each Report block but does not send Reply even if it caches the specified content. The publisher (who has the complete set of content and is not a caching router) replies the Reply message. Specifying only a scheme name is not allowed with this option.

**r option**

Number of traced routers. If the Contrace user specifies this option, only the specified number of hops from the Contrace user trace the Request; each router inserts its own Report block and forwards the Request message to the upstream router(s), and the last router stops the trace and sends the Reply message back to the Contrace user. This value is set in the "HopLimit" field located in the fixed header of the Request. For example, when the Contrace user invokes the Contrace command with this option such as "-r 3", only three routers along the path examine their path and cache information. If there is a caching router within the hop count along the path, the caching router sends back the Reply message and terminates the trace request. If the last router does not have the corresponding cache, it replies the Reply message with NO\_INFO return code (described in Section 3.1) with no Reply block TLV inserted. The Request messages are terminated at publishers; therefore, although the maximum value for this option a Contrace user can specify is 255, the Request messages should be in general reached at the publisher within significantly lower than 255 hops.

**s option**

Number of skipped routers. If the Contrace user specifies this option, the number of hops from the Contrace user simply forward the Contrace Request messages without adding its own Report block and without replying the Request, and the next upstream router starts the trace. This value is set in the "SkipHopCount" field



located in the Request block TLV. For example, when the Contrace user invokes the Contrace command with this option such as "-s 3", the three upstream routers along the path only forwards the Request message, but does not append their Report blocks in the hop-by-hop headers and does not send the Reply messages even though they have the corresponding cache. The Request messages are terminated at publishers; therefore, although the maximum value for this option a Contrace user can specify is 255, if the Request messages reaches the publisher, the publisher silently discards the Request message and the request will be timed out.

#### w option

This option defines the Contrace timeout value (in seconds) that the Contrace user will wait for the Reply. After the timeout, the Contrace user terminates the Request and silently discards the Reply message even if s/he receives the Reply. Note that routers along the path can configure the Contrace Reply Timeout Section 8.1, which is the allowable timeout value to keep the PIT entry. In order to avoid DoS attacks Section 10, routers MAY configure the shorter timeout value than the user-configured Contrace timeout value. If it is shorter, the Request may be timed out and the Contrace user may not receive the Reply as expected.

#### Authors' Addresses

Hitoshi Asaeda  
National Institute of Information and Communications Technology  
4-2-1 Nukui-Kitamachi  
Koganei, Tokyo 184-8795  
Japan

Email: asaeda@nict.go.jp

Xun Shao  
National Institute of Information and Communications Technology  
4-2-1 Nukui-Kitamachi  
Koganei, Tokyo 184-8795  
Japan

Email: x-shao@nict.go.jp

Thierry Turletti  
Inria  
2004 Route des Lucioles  
Sophia Antipolis 06902  
France

Email: [thierry.turletti@inria.fr](mailto:thierry.turletti@inria.fr)

ICN Research Group  
Internet-Draft  
Intended status: Experimental  
Expires: January 17, 2019

C. Gundogan  
T. Schmidt  
HAW Hamburg  
M. Waehlich  
link-lab & FU Berlin  
C. Scherb  
C. Marxer  
C. Tschudin  
University of Basel  
July 16, 2018

ICN Adaptation to LowPAN Networks (ICN LoWPAN)  
draft-gundogan-icnrg-ccnlowpan-02

Abstract

In this document, a convergence layer for CCNx and NDN over IEEE 802.15.4 LoWPAN networks is defined. A new frame format is specified to adapt CCNx and NDN packets to the small MTU size of IEEE 802.15.4. For that, syntactic and semantic changes to the TLV-based header formats are described. To support compatibility with other LoWPAN technologies that may coexist on a wireless medium, the dispatching scheme provided by 6LoWPAN is extended to include new dispatch types for CCNx and NDN. Additionally, the link fragmentation component of the 6LoWPAN dispatching framework is applied to ICN chunks. Basic improvements in efficiency are advised by stateless and stateful compression schemes.

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 <https://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 January 17, 2019.

## Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Overview of ICN LowPAN . . . . .	5
3.1. Link-Layer Convergence . . . . .	5
3.2. Stateless Header Compression . . . . .	5
3.3. Stateful Header Compression . . . . .	6
4. IEEE 802.15.4 Adaptation . . . . .	8
4.1. LowPAN Encapsulation . . . . .	8
4.2. Link Fragmentation . . . . .	9
4.3. Integrating Stateful Header Compression . . . . .	10
5. ICN LowPAN for NDN . . . . .	11
5.1. TLV Encoding . . . . .	11
5.2. Name TLV Compression . . . . .	13
5.3. Interest Messages . . . . .	14
5.4. Data Messages . . . . .	16
6. ICN LowPAN for CCNx . . . . .	18
6.1. TLV Encoding . . . . .	18
6.2. Name TLV Compression . . . . .	18
6.3. Interest Messages . . . . .	18
6.4. Content Objects . . . . .	24
7. Security Considerations . . . . .	27
8. IANA Considerations . . . . .	27
8.1. Page Switch Dispatch Type . . . . .	27
9. References . . . . .	27
9.1. Normative References . . . . .	27
9.2. Informative References . . . . .	28
Appendix A. Estimated Size Reduction . . . . .	30
A.1. NDN . . . . .	30
A.1.1. Interest . . . . .	30
A.1.2. Data . . . . .	31
A.2. CCNx . . . . .	33
A.2.1. Interest . . . . .	33
A.2.2. Data . . . . .	34
Acknowledgments . . . . .	35

Authors' Addresses	35
--------------------	----

## 1. Introduction

The Internet of Things (IoT) has been identified as a promising deployment area for Information Centric Networks (ICN), as infrastructureless access to content, resilient forwarding, and in-network data replication have shown notable advantages over the traditional host-to-host approach on the Internet [NDN-EXP]. Recent studies [NDN-MAC] have shown that an appropriate mapping to link layer technologies has a large impact on the practical performance of an ICN. This will be even more relevant in the context of IoT communication where nodes often exchange messages via low-power wireless links under lossy conditions. In this memo, we address the base adaptation of data chunks to such link layers for the ICN flavors NDN [NDN] and CCNx.

The IEEE 802.15.4 [ieee802.15.4] link layer is used in low-power and lossy networks (see "LLN" in [RFC7228]), in which devices are typically battery-operated and constrained in resources. Characteristics of LLNs include an unreliable environment, low bandwidth transmissions, and increased latencies. IEEE 802.15.4 admits a maximum physical layer packet size of 127 octets. The maximum frame header size is 25 octets, which leaves 102 octets for the payload. IEEE 802.15.4 security features further reduce this payload length by up to 21 octets, yielding a net of 81 octets for CCNx or NDN packet headers, signatures and content.

6LoWPAN [RFC4944][RFC6282] is a convergence layer that provides frame formats, header compression and link fragmentation for IPv6 packets in IEEE 802.15.4 networks. The 6LoWPAN adaptation introduces a dispatching framework that prepends further information to 6LoWPAN packets, including a protocol identifier for IEEE 802.15.4 payload and meta information about link fragmentation.

Prevalent Type-Length-Value (TLV) based packet formats such as in CCNx and NDN are designed to be generic and extensible. This leads to header verbosity which is inappropriate in constrained environments of IEEE 802.15.4 links. This document presents ICN LoWPAN, a convergence layer for IEEE 802.15.4 motivated by 6LoWPAN that compresses packet headers of CCNx as well as NDN and allows for an increased payload size per packet. Additionally by reusing the dispatching framework defined by 6LoWPAN, compatibility between coexisting wireless networks of competing technologies is enabled. This also allows to reuse the link fragmentation scheme specified by 6LoWPAN for ICN LoWPAN.

ICN LoWPAN utilizes a more space efficient representation of CCNx and NDN packet formats. This syntactic change is described for CCNx and NDN separately, as the header formats and TLV encodings differ largely. For further reductions, default header values suitable for constrained IoT networks are selected in order to elide corresponding TLVs.

In a typical IoT scenario (see Figure 1), embedded devices are interconnected via quasi-stationary infrastructure with a border router (BR) interconnecting the constrained LoWPAN networks via some Gateway with the public Internet. In ICN based IoT networks, Interest and Data messages transparently travel through the BR up and down between a Gateway and the embedded devices within the constrained LoWPANs.

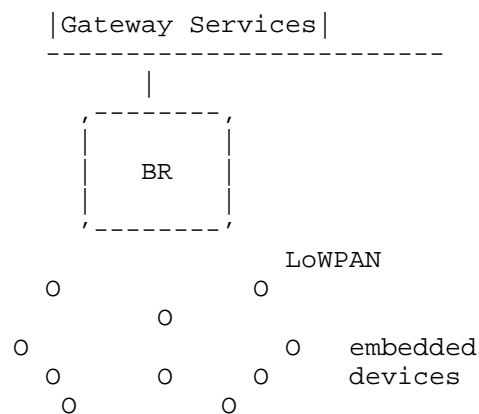


Figure 1: IoT Stub Network

## 2. 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 RFC 2119 [RFC2119]. The use of the term, "silently ignore" is not defined in RFC 2119. However, the term is used in this document and can be similarly construed.

This document uses the terminology of [RFC7476], [RFC7927], and [RFC7945] for ICN entities.

The following terms are used in the document and defined as follows:

ICN LoWPAN: Information-Centric Networking over Low-power Wireless Personal Area Network

LLN                    Low-Power and Lossy Network

CCNx:                Content-Centric Networking Architecture

NDN:                Named Data Networking

### 3. Overview of ICN LoWPAN

#### 3.1. Link-Layer Convergence

ICN LoWPAN provides a convergence layer that maps ICN packets onto constrained link-layer technologies. This includes features such as link-layer fragmentation, protocol separation on the link-layer level, and link-layer address mappings. The stack traversal is visualized in Figure 2.

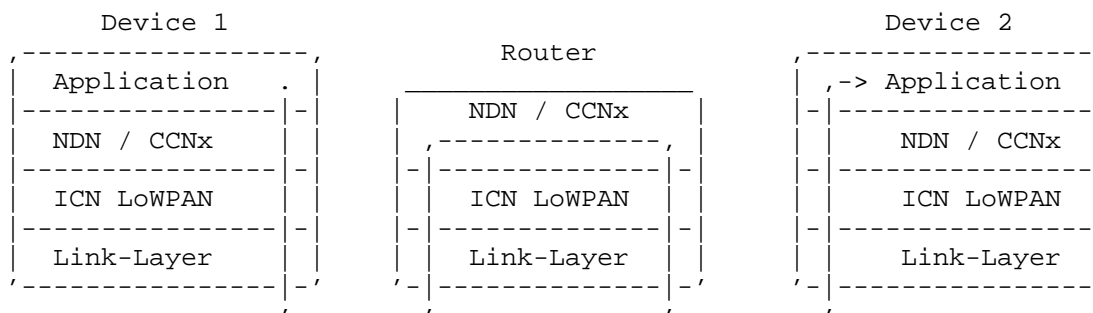


Figure 2: ICN LoWPAN convergence layer for IEEE 802.15.4

Section 4 of this document defines the convergence layer for IEEE 802.15.4.

#### 3.2. Stateless Header Compression

ICN LoWPAN also defines a stateless header compression scheme with the main purpose of reducing header overhead of ICN packets. This is of particular importance for link-layers with small MTUs. The stateless compression does not require pre-configuration of global state.

The CCNx and NDN header formats are composed of Type-Length-Value (TLV) fields to encode header data. The advantage of TLVs is its native support of variable-sized data. The main disadvantage of TLVs is the verbosity that results from storing the type and length of the encoded data.

The stateless header compression scheme makes use of compact bit fields to indicate the presence of mandatory and optional TLVs in the uncompressed packet. The order of set bits in the bit fields corresponds to the order of each TLV in the packet. Further compression is achieved by specifying default values and reducing the codomain of certain header fields.

Figure 3 demonstrates the stateless header compression idea. In this example, the first type of the first TLV is removed and the corresponding bit in the bit field is set. The second TLV represents a fixed-length TLV (e.g. the Nonce TLV in NDN), so that the type and the length fields are removed. The third TLV represents a boolean TLV (e.g. the MustBeFresh selector in NDN) and is missing the type, length and the value field.

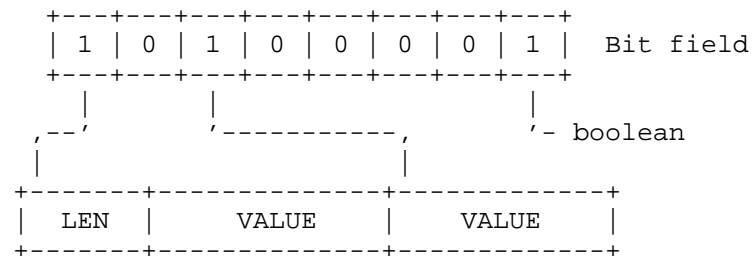


Figure 3: Compression using a compact bit field to encode context information.

### 3.3. Stateful Header Compression

ICN LowPAN further employs 2 stateful compression schemes to enhance size reductions. These mechanisms rely on shared contexts that are either distributed and maintained in the whole LowPAN, or are generated on-demand for a particular Interest-data path.

#### 3.3.1. LowPAN-local State

A context identifier (CID) is a 1-octet wide number that refers to a particular conceptual context between network devices and MAY be used to replace frequently appearing information, like name prefixes, suffixes, or meta information, such as Interest lifetime.

The initial distribution and maintenance of shared context is out of scope. Frames containing unknown or invalid CIDs are silently discarded.



### 3.3.2. En-route State

In CCNx and NDN, Name TLVs are included in Interest messages, and they return in data messages. Returning Name TLVs either equal to the original Name TLV, or they contain the original Name TLV as a prefix. ICN LoWPAN reduces this duplication in responses by replacing Name TLVs with 1-octet wide HopIDs. While an Interest is forwarded, each hop generates an ephemeral HopID that is tied to a PIT entry. Each HopID MUST be unique within the local PIT and only exist during the lifetime of a PIT entry. To maintain HopIDs, the local PIT is extended by two new columns: HIDi (inbound HopIDs) and HIDO (outbound HopIDs).

HopIDs are included in Interests and stored on the next hop with the resulting PIT entry in the HIDi column. The HopID is replaced with a newly generated local HopID before the Interest is forwarded. This new HopID is stored in the HIDO column of the local PIT (see Figure 4).

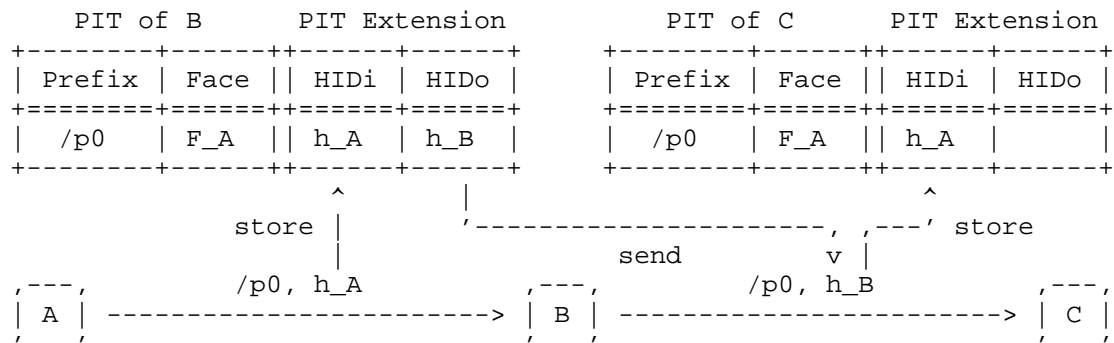


Figure 4: Setting compression state en-route (Interest).

Responses include HopIDs that were obtained from Interests. If the returning Name TLV equals the original Name TLV, then the name is elided fully. Otherwise, the distinct suffix is included along with the HopID. When a response is forwarded, the contained HopID is extracted and used to match against the correct PIT entry by performing a lookup on the HIDO column. The HopID is then replaced with the corresponding HopID from the HIDi column before forwarding the response (Figure 5).

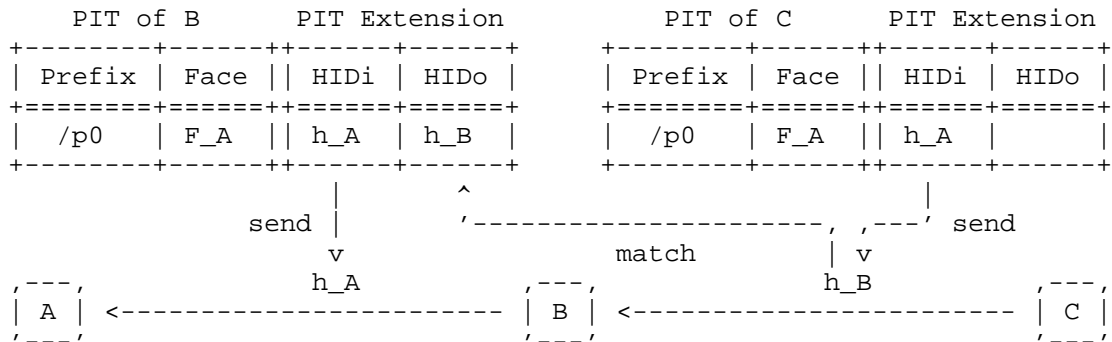


Figure 5: Eliding Name TLVs using en-route state (data).

#### 4. IEEE 802.15.4 Adaptation

##### 4.1. LoWPAN Encapsulation

The IEEE 802.15.4 frame header does not provide a protocol identifier for its payload. This causes problems of misinterpreting frames when several networks coexist on the same link layer. To mitigate errors, 6LoWPAN defines dispatches as encapsulation headers for IEEE 802.15.4 frames (see Section 5 of [RFC4944]). Multiple LoWPAN encapsulation headers can prepend the actual payload and each encapsulation header is identified by a dispatch type.

[RFC8025] further specifies dispatch pages to switch between different contexts. When a LoWPAN parser encounters a "Page switch" LoWPAN encapsulation header, then all following encapsulation headers are interpreted by using a dispatch table as specified by the "Page switch" header. Page 0 and page 1 are reserved for 6LoWPAN. This document uses page 2 ("1111 0010 (0xF2)") for NDN and page 3 ("1111 0011 (0xF3)") for CCNx.

The base dispatch format (Figure 6) is used and extended by CCNx and NDN in Section 5 and Section 6.

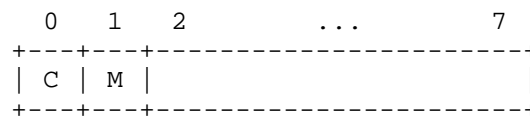


Figure 6: Base dispatch format for NDN

C: Compression

0: The message is uncompressed.

1:           The message is compressed.

M: Message Type

0:           The payload contains a Interest message.

1:           The payload contains a Data message.

The encapsulation format for ICN LoWPAN identifying an NDN Interest message is exemplarily displayed in Figure 7.

```
+-----+-----+-----+-----+-----+
| IEEE 802.15.4 | Dispatches | Page 2 | NDN Dispatches | Payl. /
+-----+-----+-----+-----+-----+
```

Figure 7: LoWPAN Encapsulation of NDN Interest with ICN LoWPAN

IEEE 802.15.4: The IEEE 802.15.4 header.

Dispatches:   Optional additional dispatch types.

Page 2:       Page Switch 2 (0xF2) for NDN.

NDN Dispatches: NDN dispatches as defined in Section 5.

Payload:      The actual (un-)compressed NDN Interest.

#### 4.2. Link Fragmentation

Section 5.3 of [RFC4944] defines a protocol independent fragmentation dispatch type, a fragmentation header for the first fragment and a separate fragmentation header for subsequent fragments. ICN LoWPAN adopts the fragmentation handling of [RFC4944].

The Fragmentation LoWPAN header can encapsulate other dispatch headers. The order of dispatch types is adopted from [RFC4944]. Figure 8 shows the fragmentation scheme. The reassembled ICN LoWPAN frame does not contain any fragmentation headers and is depicted in Figure 9.

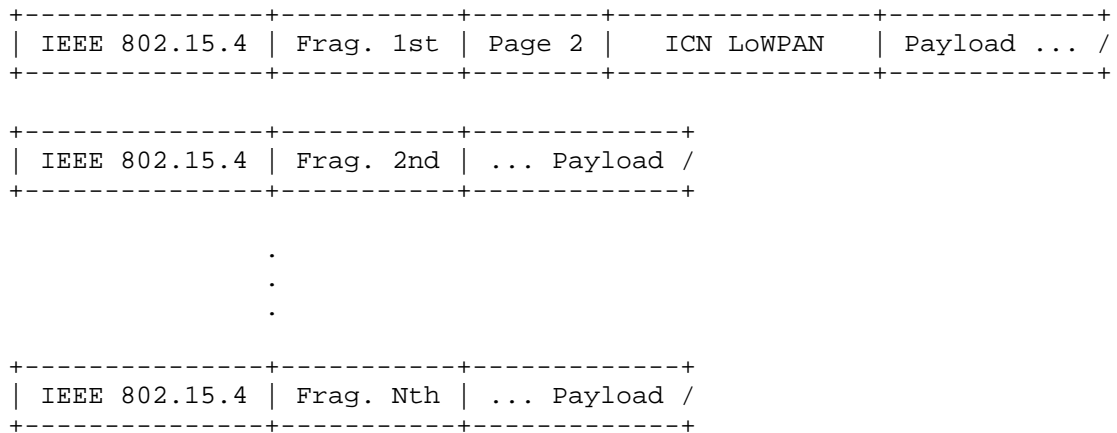


Figure 8: Fragmentation scheme

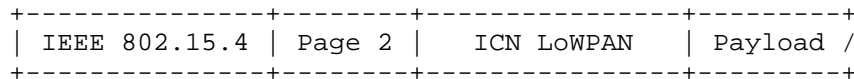


Figure 9: Reassembled ICN LoWPAN frame

### 4.3. Integrating Stateful Header Compression

#### 4.3.1. LoWPAN-Local State

A CID is appended to the last ICN LoWPAN dispatch octet. Multiple CIDs are chained together, whereas the most significant bit indicates the presence of a subsequent CID (Figure 10).

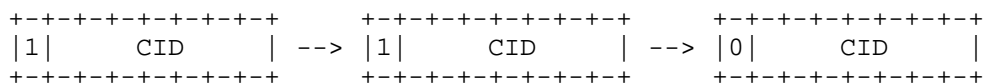


Figure 10: Multiple 1-octet wide context identifiers.

#### 4.3.2. En-Route State

The HopID is included as the very first CID. To distinguish the HopID from a typical LoWPAN-local CID, the 1st bit MUST be set (Figure 11). This yields 64 distinct HopIDs. If this range (0..63) is exhausted, the messages MUST be sent without en-route state compression until new HopIDs are available.

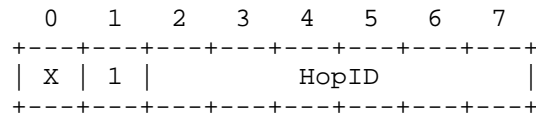


Figure 11: Context Identifier as HopID.

## 5. ICN LowPAN for NDN

### 5.1. TLV Encoding

The NDN packet format consists of TLV fields using the TLV encoding that is described in [NDN-PACKET-SPEC]. Type and length fields are of variable size, where numbers greater than 252 are encoded using multiple octets. Figure 12 shows the NDN TLV encoding scheme.

If the type or length number is less than "253", then that number is encoded into the actual type or length field (Figure 12 a). If the number is greater or equals "253" and fits into 2 octets, then the type or length field is set to "253" and the number is encoded in the next following 2 octets in network byte order, i.e., from the most significant byte (MSB) to the least significant byte (LSB) (Figure 12 b). If the number is greater than 2 octets and fits into 4 octets, then the type or length field is set to "254" and the number is encoded in the subsequent 4 octets in network byte order (Figure 12 c). For greater numbers, the type or length field is set to "255" and the number is encoded in the subsequent 8 octets in network byte order (Figure 12 d).

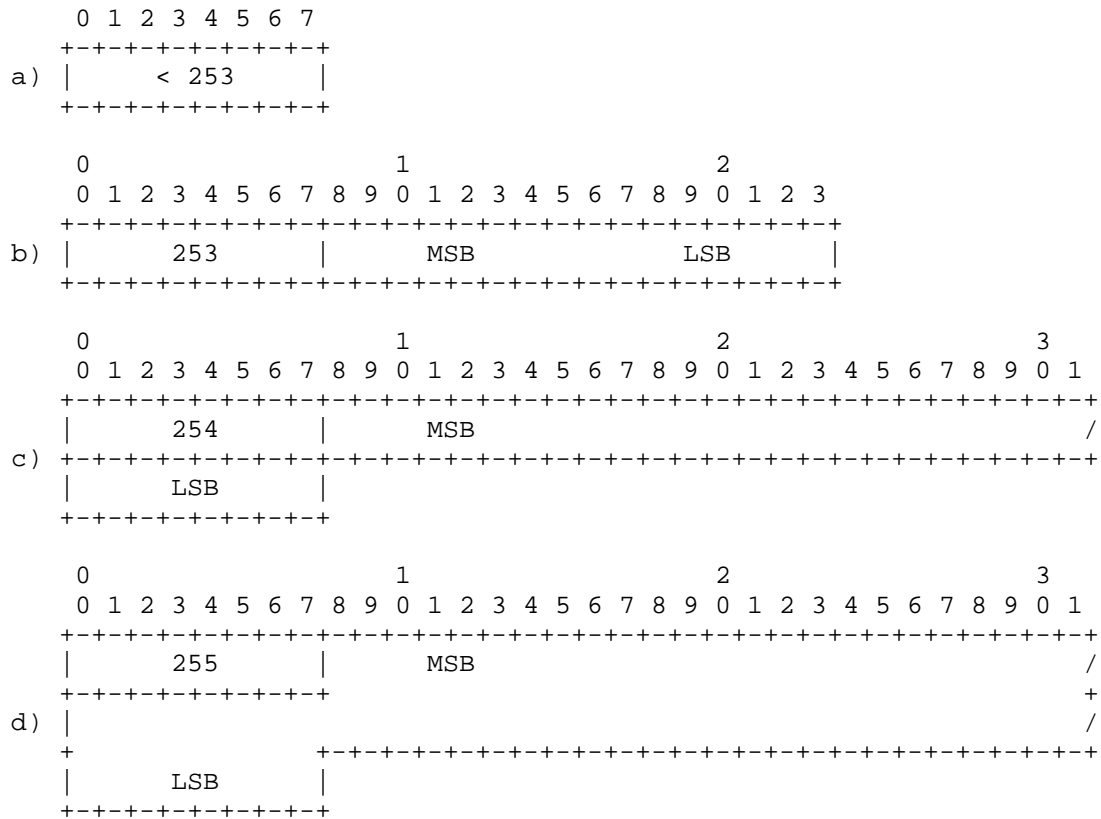


Figure 12: NDN TLV encoding scheme

In this document, compressed NDN TLVs make use of a different TLV scheme that puts more emphasis on size reduction. Instead of using the first octet as a marker for the number of following octets, the compressed NDN TLV scheme uses a method to chain a variable number of octets together. If an octet equals "255 (0xFF)", then the following octet will also be interpreted. The actual value of a chain equals the sum of all links.

If the type or length number is less than "255", then that number is encoded into the actual type or length field (Figure 13 a). If the type or length number (X) fits into 2 octets, then the first octet is set to "255" and the subsequent octet equals "X mod 255" (Figure 13 b). Following this scheme, a variable-sized number (X) is encoded using multiple octets of "255" with a trailing octet containing "X mod 255" (Figure 13 c).

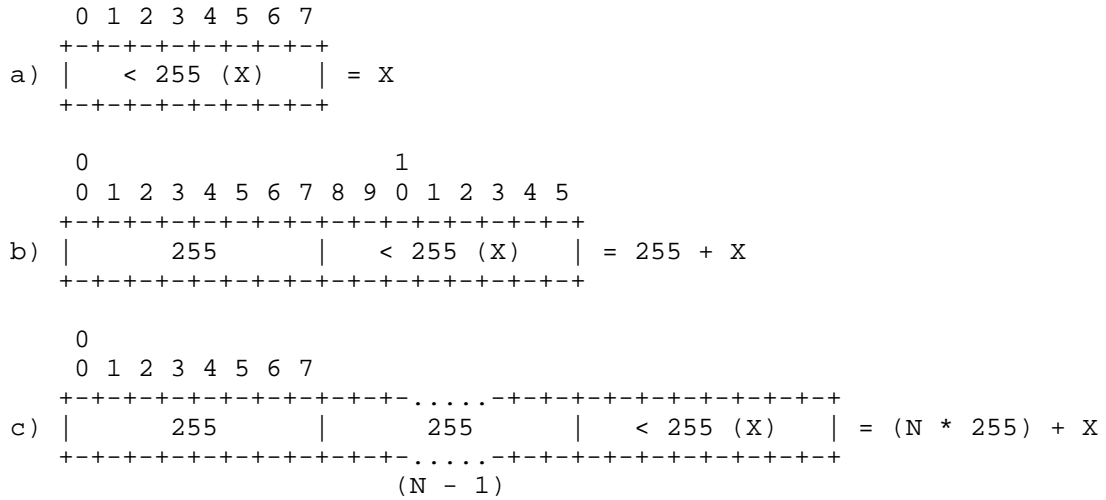


Figure 13: Compressed NDN TLV encoding scheme

## 5.2. Name TLV Compression

This Name TLV compression encodes length fields of two consecutive NameComponent TLVs into one octet, using 4 bits each. This process limits the length of a NameComponent TLV to 15 octets. A length of 0 marks the end of the compressed Name TLV.

Name: /HAW/Room/481/Humid/99

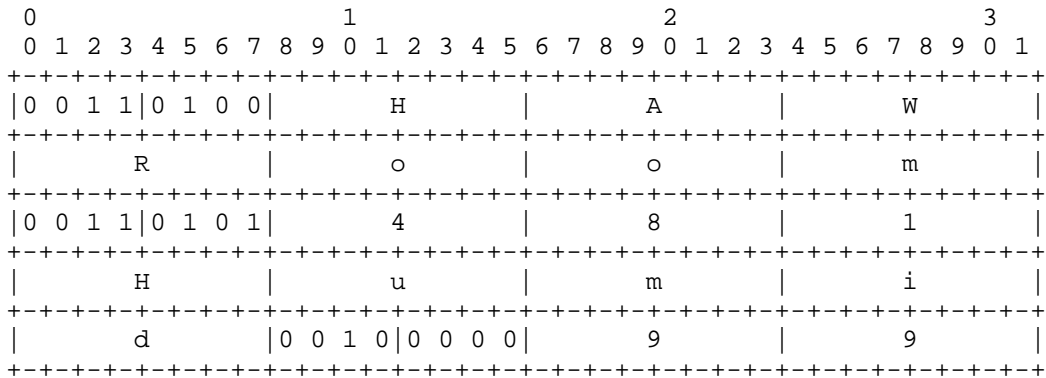


Figure 14: Name TLV compression for /HAW/Room/481/Humid/99

### 5.3. Interest Messages

#### 5.3.1. Uncompressed Interest Messages

An uncompressed Interest message uses the base dispatch format (see Figure 6) and sets the C as well as the M flag to "0" (Figure 15). "resv" MUST be set to 0. The Interest message is handed to the NDN network stack without modifications.

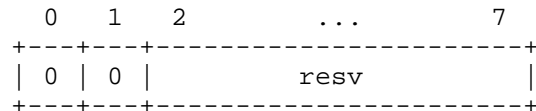


Figure 15: Dispatch format for uncompressed NDN Interest messages

#### 5.3.2. Compressed Interest Messages

The compressed Interest message uses the base dispatch format and sets the C flag to "1" and the M flag to "0". By default, the Interest message is compressed with the following base rule set:

1. The "Type" field of the outermost MessageType TLV is removed.
2. The Name TLV is compressed according to Section 5.2. For this, all NameComponents are expected to be of type GenericNameComponent. Otherwise, the message MUST be sent uncompressed.
3. The InterestLifetime TLV length is set to 2. Messages with lifetimes that require more than 2 octets MUST be sent uncompressed.
4. The Nonce TLV, InterestLifetime TLV and HopLimit TLV MUST be moved to the end of the compressed Interest, keeping the order 1) Nonce TLV, 2) InterestLifetime TLV and 3) HopLimit TLV.
5. The Type and Length fields of Nonce TLV, InterestLifetime TLV and HopLimit TLV are elided. The presence of each TLV is deduced from the remaining length to parse. The Nonce TLV has a fixed length of 4, the InterestLifetime TLV has a fixed length of 2 and the HopLimit TLV has a fixed length of 1. Any combination yields a distinct value that matches the remaining length to parse.

Further TLV compression is indicated by the ICN LoWPAN dispatch in Figure 16.



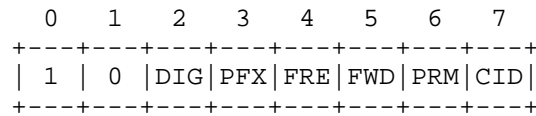


Figure 16: Dispatch format for compressed NDN Interest messages

DIG: ImplicitSha256DigestComponent TLV

- 0:           The name does not include an  
ImplicitSha256DigestComponent as the last TLV.
- 1:           The name does include an  
ImplicitSha256DigestComponent as the last TLV. The  
Type and Length fields are omitted.

PFX: CanBePrefix TLV

- 0:           The uncompressed message does not include a  
CanBePrefix TLV.
- 1:           The uncompressed message does include a CanBePrefix  
TLV and is removed from the compressed message.

FRE: MustBeFresh TLV

- 0:           The uncompressed message does not include a  
MustBeFresh TLV.
- 1:           The uncompressed message does include a MustBeFresh  
TLV and is removed from the compressed message.

FWD: ForwardingHint TLV

- 0:           The uncompressed message does not include a  
ForwardingHint TLV.
- 1:           The uncompressed message does include a  
ForwardingHint TLV. The Type field is removed from  
the compressed message.

PRM: Parameters TLV

- 0:           The uncompressed message does not include a  
Parameters TLV.

- 1: The uncompressed message does include a Parameters TLV. The Type field is removed from the compressed message.

CID: Context Identifiers

- 0: CID(s) are not appended to the dispatch octet.
- 1: CID(s) are appended to the dispatch octet.

#### 5.4. Data Messages

##### 5.4.1. Uncompressed Data Messages

An uncompressed Data message uses the base dispatch format and sets the C flag to "0" and the M flag to "1" (Figure 17). "resv" MUST be set to 0. The Data message is handed to the NDN network stack without modifications.

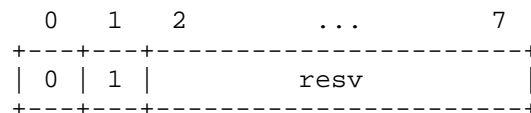


Figure 17: Dispatch format for uncompressed NDN Data messages

##### 5.4.2. Compressed Data Messages

The compressed Data message uses the base dispatch format and sets the C flag as well as the M flag to "1". By default, the Data message is compressed with the following base rule set:

1. The "Type" field of the outermost MessageType TLV is removed.
2. The Name TLV is compressed according to Section 5.2. For this, all NameComponents are expected to be of type GenericNameComponent. Otherwise, the message MUST be sent uncompressed.
3. The MetaInfo Type and Length fields are elided from the compressed Data message.
4. If present, the FinalBlockId TLV is encoded according to Section 5.2.
5. The ContentType TLV length is set to 1. Messages with ContentTypes that require more than 1 octet MUST be sent uncompressed.

6. The FreshnessPeriod TLV length is set to 2. Messages with FreshnessPeriods that require more than 2 octets MUST be sent uncompressed.
7. The FreshnessPeriod TLV and Content Type TLV MUST be moved to the end of the compressed Data, keeping the order 1) FreshnessPeriod TLV and 2) Content Type TLV.
8. The Type and Length fields of Content Type TLV and FreshnessPeriod TLV are elided. The presence of each TLV is deduced from the remaining length to parse. The FreshnessPeriod TLV has a fixed length of 2 and the Content Type TLV has a fixed length of 1. Any combination yields a distinct value that matches the remaining length to parse.

Further TLV compression is indicated by the ICN LoWPAN dispatch in Figure 18.

```

      0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| 1 | 1 | DIG | FBI | CON |   SIG   | CID |
+---+---+---+---+---+---+---+---+

```

Figure 18: Dispatch format for compressed NDN Data messages

DIG: ImplicitSha256DigestComponent TLV

- 0: The name does not include an ImplicitSha256DigestComponent as the last TLV.
- 1: The name does include an ImplicitSha256DigestComponent as the last TLV. The Type and Length fields are omitted.

FBI: FinalBlockId TLV

- 0: The uncompressed message does not include a FinalBlockId TLV.
- 1: The uncompressed message does include a FinalBlockId.

CON: Content TLV

- 0: The uncompressed message does not include a Content TLV.

- 1: The uncompressed message does include a Content TLV. The Type field is removed from the compressed message.

SIG: Signature TLV

- 00: The Type fields of the SignatureInfo TLV, SignatureType TLV and SignatureValue TLV are removed.
- 01: Reserved.
- 10: Reserved.
- 11: Reserved.

CID: Context Identifiers

- 0: CID(s) are not appended to the dispatch octet.
- 1: CID(s) are appended to the dispatch octet.

## 6. ICN LowPAN for CCNx

### 6.1. TLV Encoding

The CCNx TLV encoding is described in [I-D.irtf-icnrg-ccnxmessages]. Type and Length fields are of fixed length of 2 octets each.

In this document, the TLV encoding is changed to the more space efficient encoding described in Section 5.1. Type and Length fields MUST be encoded as in Figure 13.

### 6.2. Name TLV Compression

Name TLVs are compressed using the same approach outlined in Section 5.2.

### 6.3. Interest Messages

#### 6.3.1. Uncompressed Interest Messages

An uncompressed Interest message uses the base dispatch format (see Figure 6) and sets the C as well as the M flag to "0" (Figure 19). "resv" MUST be set to 0. The Interest message is handed to the CCNx network stack without modifications.

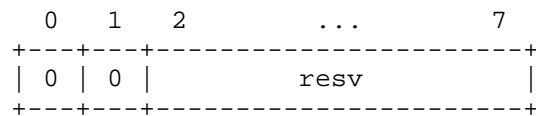


Figure 19: Dispatch format for uncompressed CCNx Interest messages

### 6.3.2. Compressed Interest Messages

The compressed Interest message uses the base dispatch format and sets the C flag to "1" and the M flag to "0". By default, the Interest message is compressed with the following base rule set:

1. The Type and Length fields of the CCNx Message TLV are elided and are obtained from the Fixed Header on decompression.

Further TLV compression is indicated by the ICN LoWPAN dispatch in Figure 20.

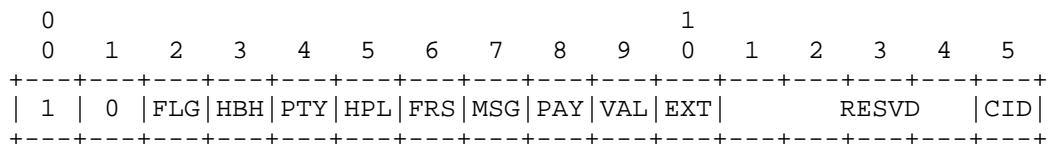


Figure 20: Dispatch format for compressed CCNx Interest messages

FLG: Flags field in the Fixed Header

- 0: The Flags field equals 0 and is removed from the Interest message.
- 1: The Flags field is carried in-line.

HBH: Optional Hop-By-Hop Header TLVs

- 0: No Hop-By-Hop Header TLVs are present in the Interest message. Also, the HeaderLength field in the fixed header is elided from the Interest message and assumed to be "8".
- 1: Hop-By-Hop Header TLVs are present in the Interest message. An additional octet follows immediately that handles Hop-By-Hop Header TLV compressions and is described in Section 6.3.3.

PTY: PacketType field in the fixed header

0: The PacketType field is elided and assumed to be "PT\_INTEREST"

1: The PacketType field is elided and assumed to be "PT\_RETURN"

HPL: HopLimit field in the fixed header

0: The HopLimit field is carried in-line

1: The HopLimit field is elided and assumed to be "1"

FRS: Reserved field in the fixed header

0: The Reserved field is carried in-line

1: The Reserved field is elided and assumed to be "0"

MSG: Optional Interest Message TLVs

0: No Interest Message TLVs are present in the Interest message.

1: Interest Message TLVs are present in the Interest message. An additional octet follows immediately that handles Interest Message TLV compressions and is described in Section 6.3.4.

PAY: Optional Payload TLV

0: The Payload TLV is absent.

1: The Payload TLV is present and the type field is elided.

VAL: Optional ValidationAlgorithm and ValidationPayload TLVs

0: No validation related TLVs are present in the Interest message.

1: Validation related TLVs are present in the Interest message. An additional octet follows immediately that handles validation related TLV compressions and is described in Section 6.3.5.

EXT: Extension

0: No extension octet follows.

- 1: An extension octet follows immediately. Extension octets are used to extend the compression scheme, but are out of scope of this document.

CID: Context Identifiers

- 0: CID(s) are not appended to the last dispatch octet.
- 1: CID(s) are appended to the last dispatch octet.

### 6.3.3. Hop-By-Hop Header TLVs Compression

Hop-By-Hop Header TLVs are unordered. For an Interest message, two optional Hop-By-Hop Header TLVs are defined in [I-D.irtf-icnrg-ccnxmessages], but several more can be defined in higher level specifications. For better compression, an ordering of Hop-By-Hop TLVs is enforced as follows:

1. Interest Lifetime TLV
2. Message Hash TLV

With this ordering in place, Type fields are elided from the Interest Lifetime TLV and the Message Hash TLV.

Note: If the original Interest message includes Hop-By-Hop Header TLVs with a different ordering, then they remain uncompressed.

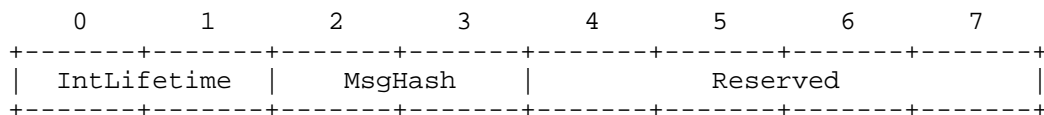


Figure 21: Dispatch for HBH Compression

IntLifetime: InterestLifetime Hop-By-Hop Header TLV

- 00: The Interest Lifetime TLV is absent.
- 01: The Interest Lifetime TLV is present and the type field is removed.
- 10: The Interest Lifetime TLV is absent and a default value of 0 seconds is assumed.
- 11: The Interest Lifetime TLV is absent and a default value of 10 minutes is assumed.

MsgHash: Message Hash Hop-By-Hop Header TLV

- 00: The Message Hash TLV is absent.
- 01: The Message Hash TLV is present and uncompressed.
- 10: A T\_SHA-256 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 32 octets. The outer Message Hash TLV is omitted.
- 11: A T\_SHA-512 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 64 octets. The outer Message Hash TLV is omitted.

#### 6.3.4. Interest Message TLVs Compression

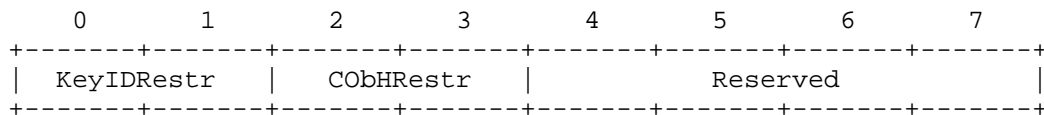


Figure 22: Dispatch for Interest Messages

KeyIDRestr: Optional KeyIdRestriction TLV within a CCNx Message TLV

- 00: The KeyIdRestriction TLV is absent.
- 01: The KeyIdRestriction TLV is present and uncompressed.
- 10: A T\_SHA-256 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 32 octets. The outer KeyIdRestriction TLV is omitted.
- 11: A T\_SHA-512 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 64 octets. The outer KeyIdRestriction TLV is omitted.

CObHRestr: Optional ContentObjectHashRestriction TLV within a CCNx Message TLV

- 00: The ContentObjectHashRestriction TLV is absent.
- 01: The ContentObjectHashRestriction TLV is present and uncompressed.



- 10: A T\_SHA-256 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 32 octets. The outer ContentObjectHashRestriction TLV is omitted.
- 11: A T\_SHA-512 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 64 octets. The outer ContentObjectHashRestriction TLV is omitted.

#### 6.3.5. Validation

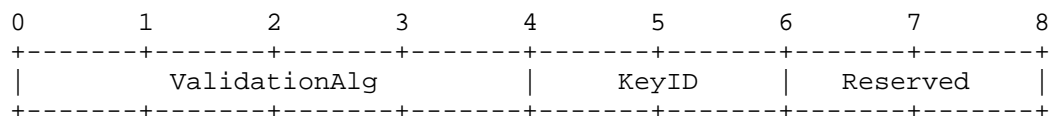


Figure 23: Dispatch for Intersect Validations

ValidationAlg: Optional ValidationAlgorithm TLV

- 0000: An uncompressed ValidationAlgorithm TLV is included.
- 0001: A T\_CRC32C ValidationAlgorithm TLV is assumed, but no ValidationAlgorithm TLV is included.
- 0010: A T\_CRC32C ValidationAlgorithm TLV is assumed, but no ValidationAlgorithm TLV is included. Additionally, a Sigtime TLV is inlined without a type and a length field.
- 0011: A T\_HMAC-SHA256 ValidationAlgorithm TLV is assumed, but no ValidationAlgorithm TLV is included.
- 0100: A T\_HMAC-SHA256 ValidationAlgorithm TLV is assumed, but no ValidationAlgorithm TLV is included. Additionally, a Sigtime TLV is inlined without a type and a length field.
- 0101: Reserved.
- 0110: Reserved.
- 0111: Reserved.
- 1000: Reserved.
- 1001: Reserved.
- 1010: Reserved.

1011: Reserved.  
 1100: Reserved.  
 1101: Reserved.  
 1110: Reserved.  
 1111: Reserved.

KeyID: Optional KeyID TLV within the ValidationAlgorithm TLV

00: The KeyId TLV is absent.  
 01: The KeyId TLV is present and uncompressed.  
 10: A T\_SHA-256 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 32 octets. The outer KeyId TLV is omitted.  
 11: A T\_SHA-512 TLV is present and the type field as well as the length fields are removed. The length field is assumed to represent 64 octets. The outer KeyId TLV is omitted.

The ValidationPayload TLV is present if the ValidationAlgorithm TLV is present. The type field is omitted.

## 6.4. Content Objects

### 6.4.1. Uncompressed Content Objects

An uncompressed Content object uses the base dispatch format (see Figure 6) and sets the C flag to "0" and the M flag to "1" (Figure 24). "resv" MUST be set to 0. The Content object is handed to the CCNx network stack without modifications.

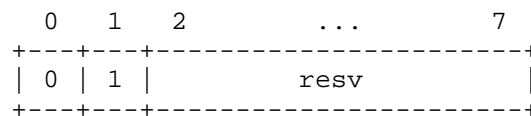


Figure 24: Dispatch format for uncompressed CCNx Content objects

### 6.4.2. Compressed Content Objects

The compressed Content object uses the base dispatch format and sets the C flag as well as the M flag to "1". By default, the Content object is compressed with the following base rule set:

1. The PacketType field is elided from the Fixed Header.
2. The Type and Length fields of the CCNx Message TLV are elided and are obtained from the Fixed Header on decompression.

Further TLV compression is indicated by the ICN LoWPAN dispatch in Figure 25.

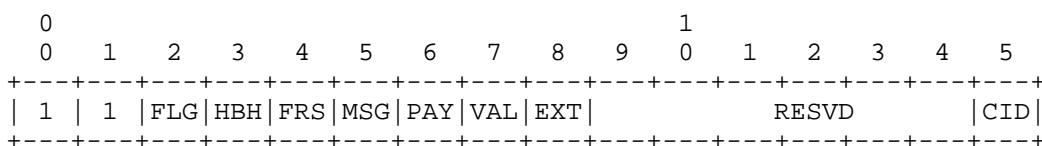


Figure 25: Dispatch format for compressed CCNx Content objects

FLG: Flags field in the fixed header See Section 6.3.2.

## HBH: Optional Hop-By-Hop Header TLVs

- ```

0:      No Hop-By-Hop Header TLVs are present in the Content
      Object message.  Also, the HeaderLength field in the
      fixed header is elided from the Content Object message
      and assumed to be "8".

1:      Hop-By-Hop Header TLVs are present in the Content Object
      message.  An additional octet follows immediately that
      handles Hop-By-Hop Header TLV compressions and is
      described in Section 6.4.3.

```

FRS: Reserved field in the Fixed Header See Section 6.3.2.

MSG: Optional Content Object Message TLVs

- ```

0:      No Content Object Message TLVs are present in the Content
      Object message.

1:      Content Object Message TLVs are present in the Content
      Object message.  An additional octet follows immediately
      that handles Content Object Message TLV compressions and
      is described in Section 6.4.4.

```

PAY: Optional Payload TLV See Section 6.3.2.

VAL: Optional ValidationAlgorithm and ValidationPayload TLVs See Section 6.3.2.

EXT: Extension See Section 6.3.2.

CID: Context Identifiers

0: CID(s) are not appended to the last dispatch octet.

1: CID(s) are appended to the last dispatch octet.

#### 6.4.3. Hop-By-Hop Header TLVs Compression

Hop-By-Hop Header TLVs are unordered. For a Content Object message, two optional Hop-By-Hop Header TLVs are defined in [I-D.irtf-icnrg-ccnxmessages], but several more can be defined in higher level specifications. For better compression, an ordering of Hop-By-Hop TLVs is enforced as follows:

1. Recommended Cache Time TLV

2. Message Hash TLV

With this ordering in place, Type fields are elided from the Recommended Cache Time TLV and Message Hash TLV.

Note: If the original Content Object message includes Hop-By-Hop Header TLVs with a different ordering, then they remain uncompressed.

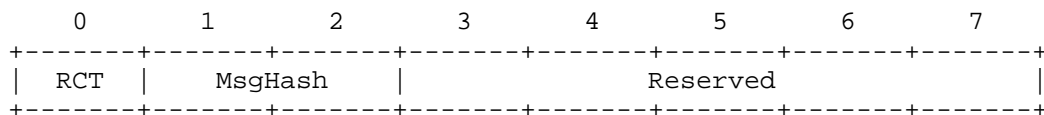


Figure 26: Dispatch for HBH Compression

RCT: Recommended Cache Time Hop-By-Hop Header TLV

0: The Recommended Cache Time TLV is absent.

1: The Recommended Cache Time TLV is present and the type as well as the length fields are elided.

MsgHash: Message Hash Hop-By-Hop Header TLV See Section 6.3.3.

## 6.4.4. Content Object Message TLVs Compression

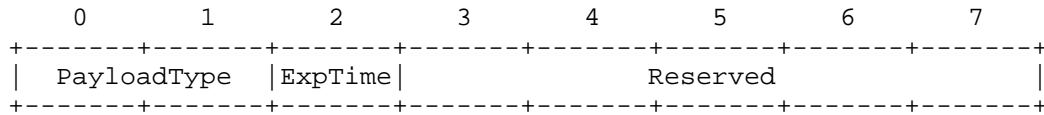


Figure 27: Dispatch for Message TLVs

PayloadType: Optional PayloadType TLV within a CCNx Message TLV

- 00: The PayloadType TLV is absent and T\_PAYLOADTYPE\_DATA is assumed.
- 01: The PayloadType TLV is absent and T\_PAYLOADTYPE\_KEY is assumed.
- 10: The PayloadType TLV is absent and T\_PAYLOADTYPE\_LINK is assumed.
- 11: The PayloadType TLV is present and uncompressed.

ExpTime: Optional ExpiryTime TLV within a CCNx Message TLV

- 0: The ExpiryTime TLV is absent.
- 1: The ExpiryTime TLV is present and the type as well as the length fields are elided.

## 7. Security Considerations

TODO

## 8. IANA Considerations

## 8.1. Page Switch Dispatch Type

This document makes use of "Page 2" from the existing paging dispatches in [RFC8025].

## 9. References

## 9.1. Normative References

- [ieee802.15.4]  
IEEE Computer Society, "IEEE Std. 802.15.4-2015", April 2016, <<https://standards.ieee.org/findstds/standard/802.15.4-2015.html>>.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007, <<https://www.rfc-editor.org/info/rfc4944>>.
- [RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, DOI 10.17487/RFC6282, September 2011, <<https://www.rfc-editor.org/info/rfc6282>>.

## 9.2. Informative References

- [CCN-LITE] "CCN-lite: A lightweight CCNx and NDN implementation", <<http://ccn-lite.net/>>.
- [I-D.irtf-icnrg-ccnxmessages] Mosko, M., Solis, I., and C. Wood, "CCNx Messages in TLV Format", draft-irtf-icnrg-ccnxmessages-08 (work in progress), July 2018.
- [I-D.irtf-icnrg-ccnxsemantics] Mosko, M., Solis, I., and C. Wood, "CCNx Semantics", draft-irtf-icnrg-ccnxsemantics-09 (work in progress), June 2018.
- [NDN] Jacobson, V., Smetters, D., Thornton, J., and M. Plass, "Networking Named Content", 5th Int. Conf. on emerging Networking Experiments and Technologies (ACM CoNEXT), 2009, <<https://doi.org/10.1145/1658939.1658941>>.
- [NDN-EXP] Baccelli, E., Mehlis, C., Hahm, O., Schmidt, TC., and M. Waehlich, "Information Centric Networking in the IoT: Experiments with NDN in the Wild", Proc. of 1st ACM Conf. on Information-Centric Networking (ICN-2014) ACM DL, pp. 77-86, September 2014, <<http://dx.doi.org/10.1145/2660129.2660144>>.

- [NDN-MAC] Kietzmann, P., Gundogan, C., Schmidt, T., Hahn, O., and M. Waehlich, "The Need for a Name to MAC Address Mapping in NDN: Towards Quantifying the Resource Gain", Proc. of 4th ACM Conf. on Information-Centric Networking (ICN-2017) ACM DL, pp. 36-42, September 2017, <<https://doi.org/10.1145/3125719.3125737>>.
- [NDN-PACKET-SPEC] "NDN Packet Format Specification", <<http://named-data.net/doc/NDN-packet-spec/0.3/>>.
- [RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.
- [RFC7476] Pentikousis, K., Ed., Ohlman, B., Corujo, D., Boggia, G., Tyson, G., Davies, E., Molinaro, A., and S. Eum, "Information-Centric Networking: Baseline Scenarios", RFC 7476, DOI 10.17487/RFC7476, March 2015, <<https://www.rfc-editor.org/info/rfc7476>>.
- [RFC7927] Kutscher, D., Ed., Eum, S., Pentikousis, K., Psaras, I., Corujo, D., Saucez, D., Schmidt, T., and M. Waehlich, "Information-Centric Networking (ICN) Research Challenges", RFC 7927, DOI 10.17487/RFC7927, July 2016, <<https://www.rfc-editor.org/info/rfc7927>>.
- [RFC7945] Pentikousis, K., Ed., Ohlman, B., Davies, E., Spirou, S., and G. Boggia, "Information-Centric Networking: Evaluation and Security Considerations", RFC 7945, DOI 10.17487/RFC7945, September 2016, <<https://www.rfc-editor.org/info/rfc7945>>.
- [RFC8025] Thubert, P., Ed. and R. Cragie, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch", RFC 8025, DOI 10.17487/RFC8025, November 2016, <<https://www.rfc-editor.org/info/rfc8025>>.

## Appendix A. Estimated Size Reduction

In the following a theoretical evaluation is given to estimate the gains of ICN LoWPAN compared to uncompressed CCNx and NDN messages.

We assume that "n" is the number of name components, "comps\_n" denotes the sum of n name component lengths. We also assume that the length of each name component is lower than 16 bytes. The length of the content is given by "clen". The lengths of TLV components is specific to the CCNx or NDN encoding and outlined below.

### A.1. NDN

The NDN TLV encoding has variable-sized TLV fields. For simplicity, the 1 octet form of each TLV component is assumed. A typical TLV component therefore is of size 2 (type field + length field) + the actual value.

#### A.1.1. Interest

Figure 28 depicts the size requirements for a basic, uncompressed NDN Interest containing a CanBePrefix TLV, a MustBeFresh TLV, a InterestLifetime TLV set to 4 seconds and a HopLimit TLV set to 6. Numbers below represent the amount of octets.

Interest TLV	= 2	
Name	2 +	
NameComponents	= 2n +	
	comps_n	
		= 21 + 2n + comps_n
CanBePrefix	= 2	
MustBeFresh	= 2	
Nonce	= 6	
InterestLifetime	= 4	
HopLimit	= 3	

Figure 28: Estimated size of an uncompressed NDN Interest

Figure 29 depicts the size requirements after compression.



Dispatch Page Switch	= 1		
NDN Interest Dispatch	= 1		
Interest TLV	= 1		
-----,			
Name			= 9 + n/2 + comps_n
NameComponents	= n/2 +		
	comps_n		
-----,			
Nonce	= 4		
InterestLifetime	= 2		

Figure 29: Estimated size of a compressed NDN Interest

The size difference is:  
 $12 + 1.5n$  octets.

For the name "/DE/HH/HAW/BT7", the total size gain is 18 octets,  
 which is 46% of the uncompressed packet.

#### A.1.2. Data

Figure 30 depicts the size requirements for a basic, uncompressed NDN Data containing a FreshnessPeriod as MetaInfo. A FreshnessPeriod of 1 minute is assumed. The value is thereby encoded using 2 octets. An HMACWithSha256 is assumed as signature. The key locator is assumed to contain a Name TLV of length klen.

Data TLV	= 2	
Name	2 +	
NameComponents	= 2n +	
	comps_n	
MetaInfo		
FreshnessPeriod	= 6	= 53 + 2n + comps_n +
		clen + klen
Content	= 2 + clen	
SignatureInfo		
SignatureType		
KeyLocator	= 41 + klen	
SignatureValue		
DigestSha256		

Figure 30: Estimated size of an uncompressed NDN Data

Figure 31 depicts the size requirements for the compressed version of the above Data packet.

Dispatch Page Switch	= 1	
NDN Data Dispatch	= 1	
Name		
NameComponents	= n/2 +	= 38 + n/2 + comps_n +
	comps_n	clen + klen
Content	= 1 + clen	
KeyLocator	= 1 + klen	
DigestSha256	= 32	
FreshnessPeriod	= 2	

Figure 31: Estimated size of a compressed NDN Data

The size difference is:  
 $15 + 1.5n$  octets.

For the name "/DE/HH/HAW/BT7", the total size gain is 21 octets.

## A.2. CCNx

The CCNx TLV encoding defines a 2-octet encoding for type and length fields, summing up to 4 octets in total without a value.

## A.2.1. Interest

Figure 32 depicts the size requirements for a basic, uncompressed CCNx Interest. No Hop-By-Hop TLVs are included and the protocol version as well as the reserved field are assumed to be 0. A KeyIdRestriction TLV with T\_SHA-256 is included to limit the responses to Content Objects containing the specific key.

Fixed Header	= 8	
Message	= 4	
Name	4 +	= 56 + 4n + comps_n
NameSegments	= 4n +	
	comps_n	
KeyIdRestriction	= 40	

Figure 32: Estimated size of an uncompressed CCNx Interest

Figure 33 depicts the size requirements after compression.

Dispatch Page Switch	= 1	
CCNx Interest Dispatch	= 3	
Fixed Header	= 3	
Name		= 39 + n/2 + comps_n
NameSegments	= n/2 +	
	comps_n	
T_SHA-256	= 32	

Figure 33: Estimated size of a compressed CCNx Interest

The size difference is:  
17 + 3.5n octets.

For the name "/DE/HH/HAW/BT7", the total size gain is 31 octets, which is 38% of the uncompressed packet.

## A.2.2. Data

Figure 34 depicts the size requirements for a basic, uncompressed CCNx Data containing an ExpiryTime Message TLV, an HMAC\_SHA-256 signature, the signature time and a hash of the shared secret key.

Fixed Header	= 8	
Message	= 4	
-----		
Name	4 +	
NameSegments	= 4n +	
	comps_n	
-----		
ExpiryTime	= 12	= 124 + 4n + comps_n + clen
Payload	= 4 + clen	
-----		
ValidationAlgorithm		
T_HMAC-256	= 56	
KeyId		
SignatureTime		
-----		
ValidationPayload	= 36	
-----		

Figure 34: Estimated size of an uncompressed CCNx Data Object

Figure 35 depicts the size requirements for a basic, compressed CCNx Data.

Dispatch Page Switch	= 1	
CCNx Content Dispatch	= 4	
Fixed Header	= 2	
-----		
Name		
NameSegments	= n/2 +	
	comps_n	= 91 + n/2 + comps_n + clen
-----		
ExpiryTime	= 8	
Payload	= 1 + clen	
T_HMAC-SHA256	= 32	
SignatureTime	= 8	
ValidationPayload	= 34	
-----		

Figure 35: Estimated size of a compressed CCNx Data Object

The size difference is:  
33 + 3.5n octets.

For the name "/DE/HH/HAW/BT7", the total size gain is 47 octets.

#### Acknowledgments

#### Authors' Addresses

Cenk Gundogan  
HAW Hamburg  
Berliner Tor 7  
Hamburg D-20099  
Germany

Phone: +4940428758067  
EMail: [cenk.guendogan@haw-hamburg.de](mailto:cenk.guendogan@haw-hamburg.de)  
URI: <http://inet.haw-hamburg.de/members/cenk-gundogan>

Thomas C. Schmidt  
HAW Hamburg  
Berliner Tor 7  
Hamburg D-20099  
Germany

EMail: [t.schmidt@haw-hamburg.de](mailto:t.schmidt@haw-hamburg.de)  
URI: <http://inet.haw-hamburg.de/members/schmidt>

Matthias Waehlich  
link-lab & FU Berlin  
Hoenower Str. 35  
Berlin D-10318  
Germany

EMail: [mw@link-lab.net](mailto:mw@link-lab.net)  
URI: <http://www.inf.fu-berlin.de/~waehl>

Christopher Scherb  
University of Basel  
Spiegelgasse 1  
Basel CH-4051  
Switzerland

EMail: [christopher.scherb@unibas.ch](mailto:christopher.scherb@unibas.ch)

Claudio Marxer  
University of Basel  
Spiegelgasse 1  
Basel CH-4051  
Switzerland

EMail: [claudio.marxer@unibas.ch](mailto:claudio.marxer@unibas.ch)

Christian Tschudin  
University of Basel  
Spiegelgasse 1  
Basel CH-4051  
Switzerland

EMail: [christian.tschudin@unibas.ch](mailto:christian.tschudin@unibas.ch)

ICNRG  
Internet-Draft  
Intended status: Informational  
Expires: March 6, 2020

A. Rahman  
InterDigital Communications, LLC  
D. Trossen  
InterDigital Europe, Ltd  
D. Kutscher  
University of Applied Sciences Emden/Leer  
R. Ravindran  
Futurewei  
September 3, 2019

Deployment Considerations for Information-Centric Networking (ICN)  
draft-irtf-icnrg-deployment-guidelines-07

Abstract

Information-Centric Networking (ICN) is now reaching technological maturity after many years of fundamental research and experimentation. This document provides a number of deployment considerations in the interest of helping the ICN community move forward to the next step of live deployments. First, the major deployment configurations for ICN are described including the key overlay and underlay approaches. Then proposed deployment migration paths are outlined to address major practical issues such as network and application migration. Next, selected ICN trial experiences are summarized. Finally, protocol areas that require further standardization are identified to facilitate future interoperable ICN deployments. This document is a product of the Information-Centric Networking Research Group (ICNRG).

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 <https://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 March 6, 2020.

## Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Acronyms List . . . . .	5
4. Deployment Configurations . . . . .	8
4.1. Clean-slate ICN . . . . .	8
4.2. ICN-as-an-Overlay . . . . .	8
4.3. ICN-as-an-Underlay . . . . .	9
4.3.1. Edge Network . . . . .	9
4.3.2. Core Network . . . . .	10
4.4. ICN-as-a-Slice . . . . .	10
4.5. Composite-ICN Approach . . . . .	11
5. Deployment Migration Paths . . . . .	12
5.1. Application and Service Migration . . . . .	12
5.2. Content Delivery Network Migration . . . . .	13
5.3. Edge Network Migration . . . . .	13
5.4. Core Network Migration . . . . .	14
6. Deployment Trial Experiences . . . . .	14
6.1. ICN-as-an-Overlay . . . . .	15
6.1.1. FP7 PURSUIT Efforts . . . . .	15
6.1.2. FP7 SAIL Trial . . . . .	15
6.1.3. NDN Testbed . . . . .	15
6.1.4. ICN2020 Efforts . . . . .	16
6.1.5. UMOBILE Efforts . . . . .	16
6.2. ICN-as-an-Underlay . . . . .	17
6.2.1. H2020 POINT and RIFE Efforts . . . . .	17
6.2.2. H2020 FLAME Efforts . . . . .	18
6.2.3. CableLabs Content Delivery System . . . . .	18
6.2.4. NDN IoT Trials . . . . .	19
6.2.5. NREN ICN Testbed . . . . .	19
6.2.6. Doctor Testbed . . . . .	19
6.3. Composite-ICN Approach . . . . .	20



6.4. Summary of Deployment Trials . . . . .	20
7. Deployment Issues Requiring Further Standardization . . . . .	21
7.1. Protocols for Application and Service Migration . . . . .	21
7.2. Protocols for Content Delivery Network Migration . . . . .	21
7.3. Protocols for Edge and Core Network Migration . . . . .	22
7.4. Summary of ICN Protocol Gaps and Potential Protocol Efforts . . . . .	23
8. Conclusion . . . . .	24
9. IANA Considerations . . . . .	25
10. Security Considerations . . . . .	25
11. Acknowledgments . . . . .	26
12. Informative References . . . . .	26
Appendix A. Change Log . . . . .	35
Authors' Addresses . . . . .	37

## 1. Introduction

The ICN RG charter identifies deployment guidelines as an important topic area for the ICN community. Specifically, the charter states that defining concrete migration paths for ICN deployments which avoid forklift upgrades, and defining practical ICN interworking configurations with the existing Internet paradigm, are key topic areas that require further investigation [ICN RG Charter]. Also, it is well understood that results and conclusions from any mid to large-scale ICN experiments in the live Internet will also provide useful guidance for deployments.

So far, outside of some preliminary investigations such as [I-D.paik-icn-deployment-considerations], there has not been much progress on this topic. This document attempts to fill some of these gaps by defining clear deployment configurations for ICN, and associated migration pathways for these configurations. Also, selected deployment trial experiences of ICN technology are summarized. Recommendations are also made for potential future IETF standardization of key protocol functionality that will facilitate interoperable ICN deployments going forward.

The major configurations of possible ICN deployments are identified in this document as (1) Clean-slate ICN replacement of existing Internet infrastructure; (2) ICN-as-an-Overlay; (3) ICN-as-an-Underlay; (4) ICN-as-a-Slice; and (5) Composite-ICN. Existing ICN trial systems primarily fall under the ICN-as-an-Overlay, ICN-as-an-Underlay and Composite-ICN configurations. Each of these deployment configurations have their respective strengths and weaknesses. This document will aim to provide guidance for current and future members of the ICN community when they consider deployment of ICN technologies.

This document represents the consensus of the Information-Centric Networking Research Group (ICNRG). It has been reviewed extensively by the Research Group (RG) members active in the specific areas of work covered by the document.

## 2. Terminology

This document assumes readers are, in general, familiar with the terms and concepts that are defined in [RFC7927] and [I-D.irtf-icnrg-terminology]. In addition, this document defines the following terminology:

Deployment - In the context of this document, deployment refers to the final stage of the process of setting up an ICN network that is (1) ready for useful work (e.g., transmission of end user video and text) in a live environment, and (2) integrated and interoperable with the Internet. We consider the Internet in its widest sense where it encompasses various access networks (e.g., WiFi, Mobile radio network), service edge networks (e.g., for edge computing), transport networks, CDNs, core networks (e.g., Mobile core network), and back-end processing networks (e.g., Data Centres). However, throughout the document we typically limit the discussion to edge networks, core networks and CDNs for simplicity.

Information-Centric Networking (ICN) - A data-centric network architecture where accessing data by name is the essential network primitive. See [I-D.irtf-icnrg-terminology] for further information.

Network Functions Virtualization (NFV): A networking approach where network functions (e.g., firewalls, load balancers) are modularized as software logic that can run on general purpose hardware, and thus are specifically decoupled from the previous generation of proprietary and dedicated hardware. See [I-D.irtf-nfvrg-gaps-network-virtualization] for further information.

Software-Defined Networking (SDN) - A networking approach where the control and data plane for switches are separated, allowing for realizing capabilities such as traffic isolation and programmable forwarding actions. See [RFC7426] for further information.

### 3. Acronyms List

API - Application Programming Interface

BIER - Bit Indexed Explicit Replication

BoF - Birds of a Feather (session)

CCN - Content Centric Networking

CCNx - Content Centric Networking

CDN - Content Distribution Network

CoAP - Constrained Application Protocol

DASH - Dynamic Adaptive Streaming over HTTP

DiffServ - Differentiated Services

DoS - Denial of Service

DTN - Delay Tolerant Networking

ETSI - European Telecommunication Standards Institute

EU - European Union

FP7 - 7th Framework Programme for Research and Technological Development

HLS - HTTP Live Streaming

HTTP - Hyper Text Transfer Protocol

HTTPS- Hyper Text Transfer Protocol Secure

H2020- Horizon 2020 (research program)

ICN - Information-Centric Networking

ICNRG- Information-Centric Networking Research Group

IETF - Internet Engineering Task Force

IntServ - Integrated Services

IoT - Internet of Things

IP - Internet Protocol

IPv4 - Internet Protocol Version 4

IPv6 - Internet Protocol Version 6

IPTV - Internet Protocol Television

ISIS - Intermediate System to Intermediate System

ISP - Internet Service Provider

k - kilo (1000)

L2 - Layer 2

LTE - Long Term Evolution (or 4th generation cellular system)

MANO - Management and Orchestration

MEC - Mobile Edge Computing

Mbps - Megabits per second

M2M - Machine-to-Machine

NAP - Network Attachment Point

NDN - Named Data Networking

NETCONF - Network Configuration Protocol

NetInf - Network of Information

NFD - Named Data Networking Forwarding Daemon

NFV - Network Functions Virtualization

NICT - National Institute of Information and Communications Technology of Japan

NR - New Radio (access network for 5G)

OAM - Operations and Maintenance

ONAP - Open Network Automation Platform

OSPF - Open Shortest Path First

PoC - Proof of Concept (demo)

POINT- IP Over ICN - the better IP (project)

qMp - Quick Mesh Project

QoS - Quality of Service

RAM - Random Access Memory

RAN - Radio Access Network

REST - Representational State Transfer (architecture)

RESTCONF - Representational State Transfer Configuration (protocol)

RIFE - Architecture for an Internet For Everybody (project)

RIP - Routing Information Protocol

ROM - Read Only Memory

RSVP - Resource Reservation Protocol

RTP - Real-time Transport Protocol

SDN - Software-Defined Networking

SFC - Service Function Chaining

SLA - Service Level Agreement

TCL - Transport Convergence Layer

TCP - Transmission Control Protocol

UDP - User Datagram Protocol

UMOBILE - Universal Mobile-centric and Opportunistic Communications Architecture

US - United States

USA - United States of America

VoD - Video on Demand

VPN - Virtual Private Network

WG - Working Group

YANG - Yet Another Next Generation (data modeling language)

5G - Fifth Generation (cellular network)

6LoWPAN - IPv6 over Low-Power Wireless Personal Area Networks

#### 4. Deployment Configurations

In this section, we present various deployment options for ICN. These are presented as "configurations" that allow for studying these options further. While this document will outline experiences with various of these configurations (in Section 6), we will not provide an in-depth technical or commercial evaluation for any of them - for this we refer to existing literature in this space such as [Tateson].

##### 4.1. Clean-slate ICN

ICN has often been described as a "clean-slate" approach with the goal to renew or replace the complete IP infrastructure of the Internet. As such, existing routing hardware as well as ancillary services such as existing applications which are typically tied directly to the TCP/IP protocol stack are not taken for granted. For instance, a Clean-slate ICN deployment would see existing IP routers being replaced by ICN-specific forwarding and routing elements, such as NFD [NFD], CCN routers [Jacobson] or PURSUIT forwarding nodes [IEEE\_Communications].

While such clean-slate replacement could be seen as exclusive for ICN deployments, some ICN approaches (e.g., [POINT]) also rely on the deployment of general infrastructure upgrades, in this case SDN switches. Different proposals have been made for various ICN approaches to enable the operation over an SDN transport [Reed][CONET][C\_FLOW].

##### 4.2. ICN-as-an-Overlay

Similarly to other significant changes to the Internet routing fabric, particularly the transition from IPv4 to IPv6 or the introduction of IP multicast, this deployment configuration foresees the creation of an ICN overlay. Note that this overlay approach is sometimes, informally, also referred to as a tunneling approach. The overlay approach can be implemented directly such as ICN-over-UDP as described in [CCNx\_UDP]. Alternatively, the overlay can be accomplished via ICN-in-L2-in-IP as in [IEEE\_Communications] which

describes a recursive layering process. Another approach used in the Network of Information (NetInf) is to define a convergence layer to map NetInf semantics to HTTP [I-D.kutscher-icnrg-netinf-proto]. Finally, [Overlay\_ICN] describes an incremental approach to deploying an ICN architecture particularly well-suited to SDN based networks by also segregating ICN user and control plane traffic.

Regardless of the flavor, however, the overlay approach results in islands of ICN deployments over existing IP-based infrastructure. Furthermore, these ICN islands are typically connected to each other via ICN/IP tunnels. In certain scenarios this requires interoperability between existing IP routing protocols (e.g., OSPF, RIP, ISIS) and ICN based ones. ICN-as-an-Overlay can be deployed over the IP infrastructure in either edge or core networks. This overlay approach is thus very attractive for ICN experimentation and testing as it allows rapid and easy deployment of ICN over existing IP networks.

#### 4.3. ICN-as-an-Underlay

Proposals such as [POINT] and [White] outline the deployment option of using an ICN underlay that would integrate with existing (external) IP-based networks by deploying application layer gateways at appropriate locations. The main reasons for such a configuration option is the introduction of ICN technology in given islands (e.g., inside a CDN or edge IoT network) to reap the benefits of native ICN in terms of underlying multicast delivery, mobility support, fast indirection due to location independence, in-network computing and possibly more. The underlay approach thus results in islands of native ICN deployments which are connected to the rest of the Internet through protocol conversion gateways or proxies. Routing domains are strictly separated. Outside of the ICN island, normal IP routing protocols apply. Within the ICN island, ICN based routing schemes apply. The gateways transfer the semantic content of the messages (i.e., IP packet payload) between the two routing domains.

##### 4.3.1. Edge Network

Native ICN networks may be located at the edge of the network where the introduction of new network architectures and protocols is easier in so-called greenfield deployments. In this context ICN is an attractive option for scenarios such as IoT [I-D.irtf-icnrg-icniot]. The integration with the current IP protocol suite takes place at an application gateway/proxy at the edge network boundary, e.g., translating incoming CoAP request/response transactions [RFC7252] into ICN message exchanges or vice versa.

The work in [VSER] positions ICN as an edge service gateway driven by a generalized ICN based service orchestration system with its own compute and network virtualization controllers to manage an ICN infrastructure. The platform also offers service discovery capabilities to enable user applications to discover appropriate ICN service gateways. To exemplify a use case scenario, the [VSER] platform shows the realization of a multi-party audio/video conferencing service over such a edge cloud deployment of ICN routers realized over commodity hardware platforms. This platform has also been extended to offer seamless mobility and mobility as a service [VSER-Mob] features.

#### 4.3.2. Core Network

In this sub-option, a core network would utilize edge-based protocol mapping onto the native ICN underlay. For instance, [POINT] proposes to map HTTP transactions, or some other IP based transactions such as CoAP, directly onto an ICN-based message exchange. This mapping is realized at the NAP, such as realized in access points or customer premise equipment, which in turn provides a standard IP interface to existing user devices. The NAPs thus provides the apparent perception of an IP-based core network towards any external peering network.

The work in [White] proposes a similar deployment configuration. There, the goal is to use ICN for content distribution within CDN server farms. Specifically, the protocol mapping is realized at the ingress of the server farm where the HTTP-based retrieval request is served, while the response is delivered through a suitable egress node translation.

#### 4.4. ICN-as-a-Slice

The objective of Network slicing [NGMN-5G] is to multiplex a general pool of compute, storage and bandwidth resources among multiple service networks with exclusive SLA requirements on transport and compute level QoS and security. This is enabled through NFV and SDN technology functions that enables functional decomposition hence modularity, independent scalability of control and/or the user-plane functions, agility and service driven programmability. Network slicing is often associated with 5G but is clearly not limited to such systems. However, from a 5G perspective, the definition of slicing includes access network enabling dynamic slicing the spectrum resources among various services hence naturally extending itself to end points and also cloud resources across multiple domains, to offer end-to-end guarantees. These slices once instantiated could include a mix of connectivity services like LTE-as-a-service or OTT services like VoD or other IoT services through composition of a group of



virtual and/or physical network functions at control, user and service plane level. Such a framework can also be used to realize ICN slices with its own control and forwarding plane over which one or more end-user services can be delivered [NGMN-Network-Slicing].

The 5G next generation architecture [fiveG-23501] provides the flexibility to deploy the ICN-as-a-Slice over either the edge (RAN), Mobile core network, or the ICN-as-a-Slice may be deployed end-to-end. Further discussions on extending the architecture presented in [fiveG-23501] and the corresponding procedures in [fiveG-23502] to support ICN has been provided in [I-D.ravi-icnrg-5gc-icn]. The draft elaborates on two possible approaches to enable ICN. First, as an edge service using the local data network (LDN) feature in 5G using UPF classification functions to fast handover to the ICN forwarder; the other is as a native deployment using the non-IP PDU support that would allow new network layer PDU to be handed over to ICN UPFs collocated with the gNB functions, without invoking any IP functions. While the former deployment would still rely on 3GPP based mobility functions, the later would allow mobility to be handled natively by ICN. However both these deployment modes should benefit from other ICN features such as in-network caching and computing. Associated with this ICN user plan enablement, control plane extensions are also proposed leveraging 5GC's interface to other application functions (AF) to allow new network service level programmability. Such a generalized network slicing framework should be able to offer service slices over both IP and ICN. Coupled with the view of ICN functions as being "chained service functions" [RFC7665], an ICN deployment within such a slice could also be realized within the emerging control plane that is targeted for adoption in future (e.g., 5G mobile) network deployments. Finally, it should be noted that ICN is not creating the network slice, but instead that the slice is created to run an 5G-ICN instance [Ravindran].

At the level of the specific technologies involved, such as ONAP [ONAP] that can be used to orchestrate slices, the 5G-ICN slice requires compatibility for instance at the level of the forwarding/data plane depending on if it is realized as an overlay or using programmable data planes. With SDN emerging for new network deployments, some ICN approaches will need to integrate with SDN as a data plane forwarding function, as briefly discussed in Section 4.1. Further cross domain ICN slices can also be realized using frameworks such as [ONAP].

#### 4.5. Composite-ICN Approach

Some deployments do not clearly correspond to any of the previously defined basic configurations of (1) Clean-slate ICN; (2) ICN-as-an-Overlay; (3) ICN-as-an-Underlay; and (4) ICN-as-a-Slice. Or, a

deployment may contain a composite mixture of the properties of these basic configurations. For example, the Hybrid ICN [H-ICN\_1] approach carries ICN names in existing IPv6 headers and does not have distinct gateways or tunnels connecting ICN islands, or any other distinct feature identified in the previous basic configurations. So we categorize Hybrid ICN, and other approaches that do not clearly correspond to one of the other basic configurations, as a Composite-ICN approach.

## 5. Deployment Migration Paths

We now focus on the various migration paths that will have importance to the various stakeholders that are usually involved in the deployment of ICN networks. We can identify these stakeholders as:

- o Application providers
- o ISPs and service providers, both as core as well as access network providers, and also ICN network providers
- o CDN providers (due to the strong relation of the ICN proposition to content delivery)
- o End device manufacturers and users

Our focus is on technological aspects of such migration. Economic or regulatory aspects, such as studied in [Tateson], [Techno\_Economic] and [Internet\_Pricing] are left out of our discussion.

### 5.1. Application and Service Migration

The Internet supports a multitude of applications and services using the many protocols defined over the packet level IP service. HTTP provides one convergence point for these services with many Web development frameworks based on the semantics provided by it. In recent years, even services such as video delivery have been migrating from the traditional RTP-over-UDP delivery to the various HTTP-level streaming solutions, such as DASH [DASH] and others. Nonetheless, many non-HTTP services exist, all of which need consideration when migrating from the IP-based Internet to an ICN-based one.

The underlay deployment configuration option presented in Section 4.3 aims at providing some level of compatibility to the existing ecosystem through a proxy based message flow mapping mechanism (e.g., mapping of existing HTTP/TCP/IP message flows to HTTP/ICN message flows). A related approach of mapping TCP/IP to TCP/ICN message flows is described in [Moiseenko]. Another approach described in

[Marchal] uses HTTP/NDN gateways and focuses in particular on the right strategy to map HTTP to NDN to guarantee a high level of compatibility with HTTP while enabling an efficient caching of Data in the ICN island. The choice of approach is a design decision based on how to configure the protocol stack. For example, the approach described in [Moiseenko] carries the TCP layer into the ICN underlay. While the [Marchal] approach terminates both HTTP and TCP at the edge of the ICN underlay and maps these functionalities onto existing ICN functionalities.

Alternatively, ICN as an overlay (Section 4.2), as well as ICN-as-a-Slice (Section 4.4), allow for the introduction of the full capabilities of ICN through new application/service interfaces as well as operations in the network. With that, these approaches of deployment are likely to aim at introducing new application/services capitalizing on those ICN capabilities, such as in-network multicast and/or caching.

Finally, [I-D.irtf-icnrg-icn-lte-4g] outlines a dual-stack end user device approach that is applicable for all deployment configurations. Specifically, it introduces middleware layers (called the TCL) in the device that will dynamically adapt existing applications to either an underlying ICN protocol stack or standard IP protocol stack. This involves end device signalling with the network to determine which protocol stack instance and associated middleware adaptation layers to utilize for a given application transaction.

## 5.2. Content Delivery Network Migration

A significant number of services and applications are devoted to content delivery in some form, either as video delivery, social media platforms, and many others. CDNs are deployed to assist these services through localizing the content requests and therefore reducing latency and possibly increase utilization of available bandwidth as well as reducing the load on origin servers. Similar to the previous sub-section, the underlay deployment configuration presented in Section 4.3 aim at providing a migration path for existing CDNs. This is also highlighted in a BIER use case document [I-D.ietf-bier-multicast-http-response], specifically with potential benefits in terms of utilizing multicast in the delivery of content but also reducing load on origin as well as delegation server. We return to this benefit in the trial experiences in Section 6.

## 5.3. Edge Network Migration

Edge networks often see the deployment of novel network level technology, e.g., in the space of IoT. Such IoT deployments have for many years relied, and often still do, on proprietary protocols for

reasons such as increased efficiency, lack of standardization incentives and others. Utilizing the underlay deployment configuration in Section 4.3.1, application gateways/proxies can integrate such edge deployments into IP-based services, e.g., utilizing CoAP [RFC7252] based M2M platforms such as oneM2M [oneM2M] or others.

Another area of increased edge network innovation is that of mobile (access) networks, particularly in the context of the 5G Mobile networks. With the proliferation of network softwarization (using technologies like service orchestration frameworks leveraging NFV and SDN concepts) access networks and other network segments, the ICN-as-a-Slice deployment configuration in Section 4.4 provides a suitable migration path for integration non-IP-based edge networks into the overall system through virtue of realizing the relevant (ICN) protocols in an access network slice.

With the advent of SDN and NFV capabilities, so-called campus or site-specific deployments could see the introduction of ICN islands at the edge for scenarios such as gaming or AR/VR-based deployments for, e.g., smart cities or theme parks.

#### 5.4. Core Network Migration

Migrating core networks of the Internet or Mobile networks requires not only significant infrastructure renewal but also the fulfillment of the key performance requirements, particularly in terms of throughput. For those parts of the core network that would migrate to an SDN-based optical transport the ICN-as-a-Slice deployment configuration in Section 4.4 would allow the introduction of native ICN solutions within slices. This would allow for isolating the ICN traffic while addressing the specific ICN performance benefits, such as in-network multicast or caching, and constraints, such as the need for specific network elements within such isolated slices. For ICN solutions that natively work on top of SDN, the underlay deployment configuration in Section 4.3.2 provides an additional migration path, preserving the IP-based services and applications at the edge of the network, while realizing the core network routing through an ICN solution (possibly itself realized in a slice of the SDN transport network).

#### 6. Deployment Trial Experiences

In this section, we will outline trial experiences, often conducted within collaborative project efforts. Our focus here is on the realization of the various deployment configurations identified in Section 4, and we therefore categorize the trial experiences according to these deployment configurations. While a large body of

work exists at the simulation or emulation level, we specifically exclude these studies from our analysis to retain the focus on real life experiences.

## 6.1. ICN-as-an-Overlay

### 6.1.1. FP7 PURSUIT Efforts

Although the FP7 PURSUIT [IEEE\_Communications] efforts were generally positioned as a Clean-slate ICN replacement of IP (Section 4.1), the project realized its experimental test bed as an L2 VPN-based overlay between several European, US as well as Asian sites, following the overlay deployment configuration presented in Section 4.2. Software-based forwarders were utilized for the ICN message exchange, while native ICN applications, e.g., for video transmissions, were showcased. At the height of the project efforts, about 70+ nodes were active in the (overlay) network with presentations given at several conferences as well as to the ICNRG.

### 6.1.2. FP7 SAIL Trial

The Network of Information (NetInf) is the approach to ICN developed by the EU FP7 SAIL project [SAIL]. NetInf provides both name-based forwarding with CCNx-like semantics and name resolution (for indirection and late-binding). The NetInf architecture supports different deployment options through its convergence layer such as using UDP, HTTP, and even DTN underlays. In its first prototypes and trials, NetInf was deployed mostly in an HTTP embedding and in a UDP overlay following the overlay deployment configuration in Section 4.2. Reference [SAIL\_Prototyping] describes several trials including a stadium environment and a multi-site testbed, leveraging NetInf's Routing Hint approach for routing scalability [SAIL\_Content\_Delivery].

### 6.1.3. NDN Testbed

The Named Data Networking (NDN) is one of the research projects of the National Science Foundation (NSF) of the USA as part of the Future Internet Architecture (FIA) Program. The original NDN proposal was positioned as a Clean-slate ICN replacement of IP (Section 4.1). However, in several trials, NDN generally follows the overlay deployment configuration of Section 4.2 to connect institutions over the public Internet across several continents. The use cases covered in the trials include real-time video-conferencing, geo-locating, and interfacing to consumer applications. Typical trials involve up to 100 NDN enabled nodes [NDN-testbed] [Jangam].

#### 6.1.4. ICN2020 Efforts

ICN2020 is an ICN related project of the EU H2020 research program and NICT [ICN2020-overview]. ICN2020 has a specific focus to advance ICN towards real-world deployments through applications such as video delivery, interactive videos and social networks. The federated testbed spans the USA, Europe and Japan. Both NDN and CCN approaches are within the scope of the project.

ICN2020 has released a set of interim public technical reports [ICN2020]. The report [ICN2020-Experiments] contains a detailed description of the progress made in both local testbeds as well as federated testbeds. The plan for the federated testbed includes integrating the NDN testbed, the CUTEi testbed [RFC7945] [CUTEi] and the GEANT testbed [GEANT] to create an overlay deployment configuration of Section 4.2 over the public Internet. The total network contains 37 nodes. Since video was an important application typical throughput was measured in certain scenarios and found to be in the order of 70 Mbps per node.

#### 6.1.5. UMOBILE Efforts

UMOBILE is another of the ICN research projects under the H2020 research program [UMOBILE-overview]. The UMOBILE architecture integrates the principles of DTN and ICN in a common framework to support edge computing and mobile opportunistic wireless environments (e.g., post-disaster scenarios and remote areas). The UMOBILE architecture [UMOBILE-2] was developed on top of the NDN framework by following the overlay deployment configuration of Section 4.2. UMOBILE aims to extend Internet functionally by combining ICN and DTN technologies.

One of the key aspects of UMOBILE was the extension of the NDN framework to locate network services (e.g., mobility management, intermittent connectivity support) and user services (e.g., pervasive content management) as close as possible to the end-users to optimize bandwidth utilization and resource management. Another aspect was the evolution of the NDN framework to operate in challenging wireless networks, namely in emergency scenarios [UMOBILE-3] and environments with intermittent connectivity. To achieve this, the NDN framework was leveraged with a new messaging application called Oi! [UMOBILE-4] [UMOBILE-5] that supports intermittent wireless networking. UMOBILE also implements a new data-centric wireless routing protocol, DABBER [UMOBILE-6] [I-D.mendes-icnrg-dabber], which was designed based on data reachability metrics that take into consideration availability of adjacent wireless nodes and different data sources. The contextual-awareness of the wireless network

operation is obtained via a machine learning agent running within the wireless nodes [UMOBILE-7].

The consortium has completed several ICN deployment trails. In a post disaster scenario trial [UMOBILE-8], a special DTN face was created to provide reachability to remote areas where there is no typical Internet connection. Another trail was the ICN deployment over the "Guifi.net" community network in the Barcelona region. This trial focused on the evaluation of ICN edge computing platform, called PiCasso [UMOBILE-9]. In this trial, ten (10) raspberry Pis were deployed across Barcelona to create an ICN overlay network on top of the existing IP routing protocol (e.g., qMp routing). This trial showed that ICN can play a key role in improving data delivery QoS as well as reducing the traffic in intermittent connectivity environments (e.g., wireless community network). A third trial in Italy was focused on displaying the capability of the UMOBILE architecture to reach disconnected areas and assist responsible authorities in emergencies, corresponding to an infrastructure scenario. The demonstration encompassed seven (7) end-user devices, one (1) access-point, and one (1) gateway.

## 6.2. ICN-as-an-Underlay

### 6.2.1. H2020 POINT and RIFE Efforts

POINT and RIFE are two more ICN related research projects of the H2020 research program. The efforts in the H2020 POINT+RIFE projects follow the underlay deployment configuration in Section 4.3.2, edge-based NAPs provide the IP/HTTP-level protocol mapping onto ICN protocol exchanges, while the SDN underlay (or the VPN-based L2 underlay) is used as a transport network.

The multicast as well as service endpoint surrogate benefits in HTTP-based scenarios, such as for HTTP-level streaming video delivery, have been demonstrated in the deployed POINT test bed with 80+ nodes being utilized. Demonstrations of this capability have been given to the ICNRG, and public demonstrations were also provided at events [MWC\_Demo]. The trial has also been accepted by the ETSI MEC group as a public proof-of-concept demonstration.

While the afore-mentioned demonstrations all use the overlay deployment, H2020 also has performed ICN underlay trials. One such trial involved commercial end users located in the Primetel network in Cyprus with the use case centered on IPTV and HLS video dissemination. Another trial was performed over the "Guifi.net" community network in the Barcelona region, where the solution was deployed in 40 households, providing general Internet connectivity to the residents. Standard IPTV STBs as well as HLS video players were

utilized in accordance with the aim of this deployment configuration, namely to provide application and service migration.

#### 6.2.2. H2020 FLAME Efforts

The H2020 FLAME efforts concentrate on providing an experimental ground for the aforementioned POINT/RIFE solution in initially two city-scale locations, namely in Bristol and Barcelona. This trial followed the underlay deployment configuration in Section 4.3.2 as per POINT/RIFE approach. Experiments were conducted with the city/university joint venture Bristol-is-Open (BIO), to ensure the readiness of the city-scale SDN transport network for such experiments. Another trial was for the ETSI MEC PoC. This trial showcased operational benefits provided by the ICN underlay for the scenario of a location-based game. These benefits aim at reduced network utilization through improved video delivery performance (multicast of all captured videos to the service surrogates deployed in the city at six locations) as well as reduced latency through the playout of the video originating from the local NAP, collocated with the WiFi AP instead of a remote server, i.e., the playout latency was bounded by the maximum single hop latency.

Twenty three (23) large-scale media service experiments are planned as part of the H2020 FLAME efforts in the area of Future Media Internet (FMI). The platform, which includes the ICN capabilities integrated with NFV and SDN capabilities of the infrastructure. The ultimate goal of these platform efforts is the full integration of ICN into the overall media function platform for the provisioning of advanced (media-centric) Internet services.

#### 6.2.3. CableLabs Content Delivery System

The Cablelabs ICN work reported in [White] proposes an underlay deployment configuration based on Section 4.3.2. The use case is ICN for content distribution within complex CDN server farms to leverage ICN's superior in-network caching properties. This "island of ICN" based CDN is then used to service standard HTTP/IP-based content retrieval request coming from the general Internet. This approach acknowledges that whole scale replacement (see Section 4.1) of existing HTTP/IP end user applications and related Web infrastructure is a difficult proposition. [White] is clear that the architecture proposed had not yet been tested experimentally but that implementations were in process and expected in the 3-5 year time frame.



#### 6.2.4. NDN IoT Trials

[Baccelli] summarizes the trial of an NDN system adapted specifically for a wireless IoT scenario. The trial was run with 60 nodes distributed over several multi-story buildings in a university campus environment. The NDN protocols were optimized to run directly over 6LoWPAN wireless link layers. The performance of the NDN based IoT system was then compared to an equivalent system running standard IP based IoT protocols. It was found that the NDN based IoT system was superior in several respects including in terms of energy consumption, and for RAM and ROM footprints [Baccelli] [Anastasiades]. For example, the binary file size reductions for NDN protocol stack versus standard IP based IoT protocol stack on given devices were up to 60% less for ROM size and up to 80% less for RAM size.

#### 6.2.5. NREN ICN Testbed

The National Research and Education Network (NREN) ICN Testbed is a project sponsored by Cisco, Internet2, and the US Research and Education community. Participants include universities and US federal government entities that connect via a nation-wide VPN-based L2 underlay. The testbed uses the CCN approach and is based on the [CICN] open source software. There are approximately 15 nodes spread across the USA which connect to the testbed. The project's current focus is to advance data-intensive science and network research by improving data movement, searchability, and accessibility.

#### 6.2.6. Doctor Testbed

The Doctor project is a French research project meaning "Deployment and Securitisation of new Functionalities in Virtualized Networking Environments". The project aims to run NDN over virtualized NFV infrastructure [Doctor] (based on Docker technology) and focuses on the NFV MANO aspects to build an operational NDN network focusing on important performance criteria such as security, performance and interoperability.

The data-plane relies on a HTTP/NDN gateway [Marchal] that processes HTTP traffic and transports it in an optimized way over NDN to benefit from the properties of the NDN-island (i.e., by mapping HTTP semantics to NDN semantics within the NDN-island). The testbed carries real Web traffic of users, and has been currently evaluated with the top-1000 most popular Web sites. The users only need to set the gateway as the Web proxy. The control-plane relies on a central manager which uses machine learning based detection methods [Mai-1] from the data gathered by distributed probes and applies orchestrated counter-measures against NDN attacks [Nguyen-1] [Nguyen-2] [Mai-2] or

performance issues. A remediation can be, for example, the scale-up of a bottleneck component, or the deployment of a security function like a firewall or a signature verification module. Test results thus far have indicated that key attacks can be detected accurately. For example, content poisoning attacks can be detected at up to over 95% accuracy (with less than 0.01% false positives) [Nguyen-3].

### 6.3. Composite-ICN Approach

Hybrid ICN [H-ICN\_1] [H-ICN\_2] is an approach where the ICN names are mapped to IPv6 addresses, and other ICN information is carried as payload inside the IP packet. This allows standard (ICN-unaware) IP routers to forward packets based on IPv6 info, but enables ICN-aware routers to apply ICN semantics. The intent is to enable rapid hybrid deployments and seamless interconnection of IP and Hybrid ICN domains. Hybrid ICN uses [CICN] open source software. Initial tests have been done with 150 clients consuming DASH videos which showed good scalability properties at the Server Side using the Hybrid ICN transport [H-ICN\_3] [H-ICN\_2].

### 6.4. Summary of Deployment Trials

In summary, there have been significant trials over the years with all the major ICN protocol flavors (e.g., CCN, NDN, POINT) using both the ICN-as-an-Overlay and ICN-as-an-Underlay deployment configurations. The major limitations of the trials include the fact that only a limited number of applications have been tested. However, the tested applications include both native ICN and existing IP based applications (e.g., video-conferencing and IPTV). Another limitation of the trials is that all of them involve less than 1k users.

The ICN-as-a-Slice configuration has just started being trialled by Huawei and China Unicom to demonstrate ICN features of security, mobility and bandwidth efficiency over a wired infrastructure using video conferencing as the application scenario [Chakraborti], also this prototype has been extended to demonstrate this over a 5G-NR access.

The Clean-slate ICN approach has obviously never been trialled as complete replacement of Internet infrastructure (e.g., existing applications, TCP/IP protocol stack, IP routers, etc.) is no longer considered a viable alternative.

Finally, Hybrid ICN is a Composite-ICN approach that offers an interesting alternative as it allows ICN semantics to be embedded in standard IPv6 packets so the packets can be routed through either IP routers or Hybrid ICN routers. Note that some other trials such as

the Doctor testbed (Section 6.2.6) could also be characterized as a Composite-ICN approach because it contains both ICN gateways (as in ICN-as-an-Underlay) and virtualized infrastructure (as in ICN-as-a-Slice). However, for the Doctor testbed we have chosen to characterize it as an ICN-as-an-Underlay configuration because that is a dominant characteristic.

## 7. Deployment Issues Requiring Further Standardization

The ICN Research Challenges [RFC7927] describes key ICN principles and technical research topics. As the title suggests, [RFC7927] is research oriented without a specific focus on deployment or standardization issues. This section addresses this open area by identifying key protocol functionality that that may be relevant for further standardization effort in IETF. The focus is specifically on identifying protocols that will facilitate future interoperable ICN deployments correlating to the scenarios identified in the deployment migration paths in Section 5. The identified list of potential protocol functionality is not exhaustive.

### 7.1. Protocols for Application and Service Migration

End user applications and services need a standardized approach to trigger ICN transactions. For example, in Internet and Web applications today, there are established socket APIs, communication paradigms such as REST, common libraries, and best practices. We see a need to study application requirements in an ICN environment further and, at the same time, develop new APIs and best practices that can take advantage of ICN communication characteristics.

### 7.2. Protocols for Content Delivery Network Migration

A key issue in CDNs is to quickly find a location of a copy of the object requested by an end user. In ICN, a Named Data Object (NDO) is typically defined by its name. [RFC6920] defines a mechanism that is suitable for static naming of ICN data objects. Other ways of encoding and representing ICN names have been described in [I-D.irtf-icnrg-ccnxmessages] and [I-D.irtf-icnrg-ccnxsemantics]. Naming dynamically generated data requires different approaches (e.g., hash digest based names would normally not work), and there is lack of established conventions and standards.

Another CDN issue for ICN is related to multicast distribution of content. Existing CDNs have started using multicast mechanisms for certain cases such as for broadcast streaming TV. However, as discussed in Section 6.2.1, certain ICN approaches provide substantial improvements over IP multicast, such as the implicit support for multicast retrieval of content in all ICN flavours.

Caching is an implicit feature in many ICN architectures that can improve performance and availability in several scenarios. The ICN in-network caching can augment managed CDN and improve its performance. The details of the interplay between ICN caching and managed CDN need further consideration.

### 7.3. Protocols for Edge and Core Network Migration

ICN provides the potential to redesign current edge and core network computing approaches. Leveraging ICN's inherent security and its ability to make name data and dynamic computation results available independent of location, can enable a light-weight insertion of traffic into the network without relying on redirection of DNS requests. For this, proxies that translate from commonly used protocols in the general Internet to ICN message exchanges in the ICN domain could be used for the migration of application and services within deployments at the network edge but also in core networks. This is similar to existing approaches for IoT scenarios where a proxy translates CoAP request/responses to other message formats. For example, [RFC8075] specifies proxy mapping between CoAP and HTTP protocols. Also, [RFC8613] is an example of how to pass end-to-end encrypted content between HTTP and COAP by an application layer security mechanism. Further work is required to identify if an [RFC8613]-like approach, or some other approach, is suitable to preserve ICN message security through future protocol translation functions of gateways/proxies.

Interaction and interoperability between existing IP routing protocols (e.g., OSPF, RIP, ISIS) and ICN routing approaches (e.g., NFD, CCN routers) are expected especially in the overlay approach. Another important topic is the integration of ICN into networks that support virtualized infrastructure in the form of NFV/SDN and most likely utilizing SFC as a key protocol. Further work is required to validate this idea and document best practices.

There are several existing approaches to supporting QoS in IP networks including DiffServ, IntServ and RSVP. Some initial ideas for QoS support in ICN networks are outlined in [I-D.moiseenko-icnrg-flowclass] which proposes a flow classification based approach to enable functions such ICN rate control and cache control. Also [I-D.anilj-icnrg-icn-qos] proposes how to use DiffServ DSCP codes to support QoS for ICN based data path delivery. Further work is required to identify the best approaches for support of QoS in ICN networks.

OAM is a crucial area that has not yet been fully addressed by the ICN research community, but which is obviously critical for future deployments of ICN. Potential areas that need investigation include

whether the YANG data modelling approach and associated NETCONF/RESTCONF protocols need any specific updates for ICN support. Another open area is how to measure and benchmark performance of ICN networks comparable to the sophisticated techniques that exist for standard IP networks, virtualized networks and data centers. It should be noted that some initial progress has been made in the area of ICN network path traceroute facility with approaches such as CCNinfo [I-D.irtf-icnrg-ccninfo] [Contrace].

#### 7.4. Summary of ICN Protocol Gaps and Potential Protocol Efforts

Without claiming completeness, Table 1 maps the open ICN issues identified in this document to potential protocol efforts that could address some aspects of the gap.

ICN Gap	Potential Protocol Effort
1-Support of REST APIs	HTTP/CoAP support of ICN semantics
2-Naming	Dynamic naming of ICN data objects
3-Routing	Interactions between IP and ICN routing protocols
4-Multicast distribution	Multicast enhancements for ICN
5-In-network caching	ICN Cache placement and sharing
6-NFV/SDN support	Integration of ICN with NFV/SDN and including possible impacts to SFC
7-ICN mapping	Mapping of HTTP and other protocols onto ICN message exchanges (and vice-versa) while preserving ICN message security
8-QoS support	Support of ICN QoS via mechanisms such as DiffServ and flow classification
9-OAM support	YANG models, NETCONF/RESTCONF protocols, and network performance measurements

Table 1: Mapping of ICN Gaps to Potential Protocol Efforts

## 8. Conclusion

This document provides high level deployment considerations for current and future members of the ICN community. Specifically, the major configurations of possible ICN deployments are identified as (1) Clean-slate ICN replacement of existing Internet infrastructure; (2) ICN-as-an-Overlay; (3) ICN-as-an-Underlay; (4) ICN-as-a-Slice; and (5) Composite-ICN. Existing ICN trial systems primarily fall under the ICN-as-an-Overlay, ICN-as-an-Underlay and Composite-ICN configurations.

In terms of deployment migration paths, ICN-as-an-Underlay offers a clear migration path for CDN, edge or core networks to go to an ICN paradigm (e.g., for an IoT deployment) while leaving the critical mass of existing end user applications untouched. ICN-as-an-Overlay is the easiest configuration to deploy rapidly as it leaves the underlying IP infrastructure essentially untouched. However, its applicability for general deployment must be considered on a case by case basis (e.g., can it support all required user applications). ICN-as-a-Slice is an attractive deployment option for up coming 5G systems (i.e., for 5G radio and core networks) which will naturally support network slicing, but this still has to be validated through more trial experiences. Composite-ICN, by its nature, can combine some of the best characteristics of the other configurations, but its applicability for general deployment must again be considered on a case by case basis (e.g., can enough IP routers be upgraded to support Composite-ICN functionality to provide sufficient performance benefits).

There has been significant trial experience with all the major ICN protocol flavors (e.g., CCN, NDN, POINT). However, only a limited number of applications have been tested so far, and the maximum number of users in any given trial has been less than 1k users. It is recommended that future ICN deployments scale their users gradually and closely monitor network performance as they go above 1k users. A logical approach would be to increase the number of users in a slowly increasing linear manner and monitor network performance and stability especially at every multiple of 1k users.

Finally, this document describes a set of technical features in ICN that warrant potential future IETF specification work. This will aid initial and incremental deployments to proceed in an interoperable manner. The fundamental details of the potential protocol specification effort, however, are best left for future study by the appropriate IETF WGs and/or BoFs. The ICNRG can aid this process in the near and mid-term by continuing to examine key system issues like QoS mechanisms, flexible naming schemes and OAM support for ICN.

## 9. IANA Considerations

This document requests no IANA actions.

## 10. Security Considerations

ICN was purposefully designed from the start to have certain intrinsic security properties. The most well known of which are authentication of delivered content and (optional) encryption of the content. [RFC7945] has an extensive discussion of various aspects of ICN security including many which are relevant to deployments. Specifically, [RFC7945] points out that ICN access control, privacy, security of in-network caches, and protection against various network attacks (e.g., DoS) have not yet been fully developed due to the lack of a sufficient mass of deployments. [RFC7945] also points out relevant advances occurring in the ICN research community that hold promise to address each of the identified security gaps. Lastly, [RFC7945] points out that as secure communications in the existing Internet (e.g., HTTPS) becomes the norm, that major gaps in ICN security will inevitably slow down the adoption of ICN.

In addition to the security findings of [RFC7945], this document has highlighted that all anticipated ICN deployment configurations will involve co-existence with existing Internet infrastructure and applications. Thus even the basic authentication and encryption properties of ICN content will need to account for interworking with non-ICN content to preserve end-to-end security. For example, in the edge network underlay deployment configuration described in Section 4.3.1, the gateway/proxy that translates HTTP or CoAP request/responses into ICN message exchanges will need to support a security model to preserve end-to-end security. One alternative would be to consider an approach similar to [RFC8613] which is used to pass end-to-end encrypted content between HTTP and COAP by an application layer security mechanism. Further investigation is required to see if this approach is suitable to preserve ICN message security through future protocol translation functions (e.g., ICN to HTTP, or COAP to ICN) of gateways/proxies.

Finally, the Doctor project discussed in Section 6.2.6 is an example of an early deployment that is looking at specific attacks against ICN infrastructure. In this case, looking at Interest Flooding Attacks [Nguyen-2] and Content Poisoning Attacks [Nguyen-1] [Mai-2] [Nguyen-3] and evaluation of potential counter-measures based on MANO orchestrated actions on the virtualized infrastructure [Mai-1] .

## 11. Acknowledgments

The authors want to thank Alex Afanasyev, Hitoshi Asaeda, Giovanna Carofiglio, Xavier de Foy, Guillaume Doyen, Hannu Flinck, Anil Jangam, Michael Kowal, Adisorn Lertsinsruttavee, Paulo Mendes, Luca Muscariello, Thomas Schmidt, Jan Seedorf, Eve Schooler, Samar Shailendra, Milan Stolic, Prakash Suthar, Atsushi Mayutan, and Lixia Zhang for their very useful reviews and comments to the document.

Special thanks to Dave Oran (ICNRG co-chair) and Marie-Jose Montpetit for their extensive and thoughtful reviews of the document. Their reviews helped to immeasurably improve the document quality.

## 12. Informative References

### [Anastasiades]

Anastasiades, C., "Information-centric communication in mobile and wireless networks", PhD Dissertation, 2016, <[http://boris.unibe.ch/83683/1/16anastasiades\\_c.pdf](http://boris.unibe.ch/83683/1/16anastasiades_c.pdf)>.

### [Baccelli]

Baccelli, E. and et al., "Information Centric Networking in the IoT: Experiments with NDN in the Wild", ACM 20164, Paris, France, 2014, <<http://conferences2.sigcomm.org/acm-icn/2014/papers/p77.pdf>>.

### [C\_FLOW]

Suh, J. and et al., "C\_FLOW: Content-Oriented Networking over OpenFlow", Open Networking Summit, April, 2012, <<http://opennetsummit.org/archives/apr12/site/pdf/snu.pdf>>.

### [CCNx\_UDP]

PARC, "CCNx Over UDP", 2015, <<https://www.ietf.org/proceedings/interim-2015-icnrg-04/slides/slides-interim-2015-icnrg-4-5.pdf>>.

### [Chakraborti]

Chakraborti, A. and et al., "Design and Evaluation of a Multi-source Multi-destination Real-time Application on Content Centric Network", IEEE, HoT ICN, 2018 , 2018.

### [CICN]

CICN, "Community Information-Centric Networking (CICN)", 2017, <<https://wiki.fd.io/view/Cicn>>.



- [CONET] Veltri, L. and et al., "CONET Project: Supporting Information-Centric Functionality in Software Defined Networks", Workshop on Software Defined Networks, , 2012, <[http://netgroup.uniroma2.it/Stefano\\_Salsano/papers/salsano-iccl2-wshop-sdn.pdf](http://netgroup.uniroma2.it/Stefano_Salsano/papers/salsano-iccl2-wshop-sdn.pdf)>.
- [Contrace] Asaeda, H. and et al., "Contrace: A Tool for Measuring and Tracing Content-Centric Networks", IEEE Communications Magazine, Vol.53, No.3 , 2015.
- [CUTEi] Asaeda, H. and N. Choi, "Container-Based Unified Testbed for Information Centric Networking", IEEE Network, Vol.28, No.6 , 2014.
- [DASH] DASH, "DASH Industry Forum", 2017, <<http://dashif.org/>>.
- [Doctor] Doctor, "Deployment and Securisation of new Functionalities in Virtualized Networking Environments (Doctor)", 2017, <<http://www.doctor-project.org/index.htm>>.
- [fiveG-23501] 3gpp-23.501, "Technical Specification Group Services and System Aspects; System Architecture for the 5G System (Rel.15)", 3GPP , 2017.
- [fiveG-23502] 3gpp-23.502, "Technical Specification Group Services and System Aspects; Procedures for the 5G System (Rel.15)", 3GPP , 2017.
- [GEANT] GEANT, "GEANT Overview", 2016, <<https://www.geant.org/>>.
- [H-ICN\_1] Cisco, "Hybrid ICN: Cisco Announces Important Steps toward Adoption of Information-Centric Networking", 2017, <<http://blogs.cisco.com/sp/cisco-announces-important-steps-toward-adoption-of-information-centric-networking>>.
- [H-ICN\_2] Cisco, "Mobile Video Delivery with Hybrid ICN: IP-Integrated ICN Solution for 5G", 2017, <<https://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/ultra-services-platform/mwcl7-hicn-video-wp.pdf>>.

- [H-ICN\_3] Muscariello, L. and et al., "Hybrid Information-Centric Networking: ICN inside the Internet Protocol", 2018, <<https://datatracker.ietf.org/meeting/interim-2018-icnrg-01/materials/slides-interim-2018-icnrg-01-sessa-hybrid-icn-hicn-luca-muscariello>>.
- [H-ICN\_4] Sardara, M. and et al., "(h)ICN Socket Library for HTTP: Leveraging (h)ICN socket library for carrying HTTP messages", 2018, <<https://datatracker.ietf.org/meeting/interim-2018-icnrg-01/materials/slides-interim-2018-icnrg-01-sessa-hicn-socket-library-for-http-luca-muscariello>>.
- [I-D.anilj-icnrg-icn-qos]  
Jangam, A., suthar, P., and M. Stolic, "Supporting QoS aware Data Delivery in Information Centric Networks", draft-anilj-icnrg-icn-qos-00 (work in progress), July 2018.
- [I-D.ietf-bier-multicast-http-response]  
Trossen, D., Rahman, A., Wang, C., and T. Eckert, "Applicability of BIER Multicast Overlay for Adaptive Streaming Services", draft-ietf-bier-multicast-http-response-01 (work in progress), June 2019.
- [I-D.irtf-icnrg-ccninfo]  
Asaeda, H., Ooka, A., and X. Shao, "CCNinfo: Discovering Content and Network Information in Content-Centric Networks", draft-irtf-icnrg-ccninfo-02 (work in progress), July 2019.
- [I-D.irtf-icnrg-ccnxmessages]  
Mosko, M., Solis, I., and C. Wood, "CCNx Messages in TLV Format", draft-irtf-icnrg-ccnxmessages-09 (work in progress), January 2019.
- [I-D.irtf-icnrg-ccnxsemantics]  
Mosko, M., Solis, I., and C. Wood, "CCNx Semantics", draft-irtf-icnrg-ccnxsemantics-10 (work in progress), January 2019.
- [I-D.irtf-icnrg-icn-lte-4g]  
suthar, P., Stolic, M., Jangam, A., Trossen, D., and R. Ravindran, "Native Deployment of ICN in LTE, 4G Mobile Networks", draft-irtf-icnrg-icn-lte-4g-03 (work in progress), March 2019.

- [I-D.irtf-icnrg-icniot]  
Ravindran, R., Zhang, Y., Grieco, L., Lindgren, A., Burke, J., Ahlgren, B., and A. Azgin, "Design Considerations for Applying ICN to IoT", draft-irtf-icnrg-icniot-03 (work in progress), May 2019.
- [I-D.irtf-icnrg-terminology]  
Wissingh, B., Wood, C., Afanasyev, A., Zhang, L., Oran, D., and C. Tschudin, "Information-Centric Networking (ICN): CCN and NDN Terminology", draft-irtf-icnrg-terminology-04 (work in progress), June 2019.
- [I-D.irtf-nfvrg-gaps-network-virtualization]  
Bernardos, C., Rahman, A., Zuniga, J., Contreras, L., Aranda, P., and P. Lynch, "Network Virtualization Research Challenges", draft-irtf-nfvrg-gaps-network-virtualization-10 (work in progress), September 2018.
- [I-D.kutscher-icnrg-netinf-proto]  
Kutscher, D., Farrell, S., and E. Davies, "The NetInf Protocol", draft-kutscher-icnrg-netinf-proto-01 (work in progress), February 2013.
- [I-D.mendes-icnrg-dabber]  
Mendes, P., Sofia, R., Tsaoussidis, V., Diamantopoulos, S., Sarros, C., Borrego, C., and J. Borrell, "Information-centric Routing for Opportunistic Wireless Networks", draft-mendes-icnrg-dabber-02 (work in progress), February 2019.
- [I-D.moiseenko-icnrg-flowclass]  
Moiseenko, I. and D. Oran, "Flow Classification in Information Centric Networking", draft-moiseenko-icnrg-flowclass-04 (work in progress), July 2019.
- [I-D.paik-icn-deployment-considerations]  
Paik, E., Yun, W., Kwon, T., and h. hgchoi@mmlab.snu.ac.kr, "Deployment Considerations for Information-Centric Networking", draft-paik-icn-deployment-considerations-00 (work in progress), July 2013.
- [I-D.ravi-icnrg-5gc-icn]  
Ravindran, R., suthar, P., Trossen, D., Wang, C., and G. White, "Enabling ICN in 3GPP's 5G NextGen Core Architecture", draft-ravi-icnrg-5gc-icn-04 (work in progress), May 2019.

- [ICN2020] ICN2020, "ICN2020 Deliverables", 2017,  
<<http://www.icn2020.org/dissemination/deliverables-public/>>.
- [ICN2020-Experiments]  
ICN2020, "Deliverable D4.1: 1st yearly report on Testbed  
and Experiments (WP4)", 2017,  
<<http://www.icn2020.org/dissemination/deliverables-public/>>.
- [ICN2020-overview]  
ICN2020, "ICN2020 Project Overview", 2016,  
<<http://www.icn2020.org/>>.
- [ICNRGCharter]  
NDN, "Information-Centric Networking Research Group  
Charter", 2013,  
<<https://datatracker.ietf.org/doc/charter-irtf-icnrg/>>.
- [IEEE\_Communications]  
Trossen, D. and G. Parisis, "Designing and Realizing an  
Information-Centric Internet", Information-Centric  
Networking, IEEE Communications Magazine Special Issue,  
2012.
- [Internet\_Pricing]  
Trossen, D. and G. Biczok, "Not Paying the Truck Driver:  
Differentiated Pricing for the Future Internet", ReArch  
Workshop in conjunction with ACM Context, December, 2010.
- [Jacobson]  
Jacobson, V. and et al., "Networking Named Content",  
Proceedings of ACM Context, , 2009.
- [Jangam]  
Jangam, A. and et al., "Porting and Simulation of Named-  
data Link State Routing Protocol into ndnSIM", ACM  
DIVANet'17, Miami Beach, USA, 2017,  
<<https://dl.acm.org/citation.cfm?id=3132351>>.
- [Mai-1]  
Mai, H. and et al., "Implementation of Content Poisoning  
Attack Detection and Reaction in Virtualized NDN  
Networks", 21st Conference on Innovation in Clouds,  
Internet and Networks, ICIN 2018 (demo paper) IEEE, 2018,  
<<http://www.mallouli.com/recherche/publications/noms2018-1.pdf>>.

- [Mai-2] Mai, H. and et al., "Towards a Security Monitoring Plane for Named Data Networking: Application to Content Poisoning Attack", Proceedings of the 2018 IEEE/IFIP Symposium on Network Operations and Management (NOMS) IEEE, 2018.
- [Marchal] Marchal, X. and et al., "Leveraging NFV for the Deployment of NDN: Application to HTTP Traffic Transport", Proceedings of the 2018 IEEE/IFIP Symposium on Network Operations and Management (NOMS), 2018, <<http://www.mallouli.com/recherche/publications/noms2018-1.pdf>>.
- [Moiseenko] Moiseenko, I. and D. Oran, "TCP/ICN : Carrying TCP over Content Centric and Named Data Networks", 2016, <<http://conferences2.sigcomm.org/acm-icn/2016/proceedings/p112-moiseenko.pdf>>.
- [MWC\_Demo] InterDigital, "InterDigital Demo at Mobile World Congress (MWC)", 2016, <<http://www.interdigital.com/download/56d5c71bd616f892ba001861>>.
- [NDN-testbed] NDN Testbed, "Named Data Networking (NDN) Testbed", 2010, <<https://named-data.net/ndn-testbed/>>.
- [NFD] NDN, "NFD - Named Data Networking Forwarding Daemon", 2017, <<https://named-data.net/doc/NFD/current/>>.
- [NGMN-5G] NGMN, "NGMN 5G White Paper", 2015, <[https://www.ngmn.org/fileadmin/ngmn/content/images/news/ngmn\\_news/NGMN\\_5G\\_White\\_Paper\\_V1\\_0.pdf](https://www.ngmn.org/fileadmin/ngmn/content/images/news/ngmn_news/NGMN_5G_White_Paper_V1_0.pdf)>.
- [NGMN-Network-Slicing] NGMN, "NGMN Description of Network Slicing Concept", 2016, <[https://www.ngmn.org/fileadmin/user\\_upload/160113\\_Network\\_Slicing\\_v1\\_0.pdf](https://www.ngmn.org/fileadmin/user_upload/160113_Network_Slicing_v1_0.pdf)>.
- [Nguyen-1] Nguyen, T. and et al., "Content Poisoning in Named Data Networking: Comprehensive Characterization of real Deployment", Proceedings of the 15th IEEE/IFIP International Symposium on Integrated Network Management, 2017.

- [Nguyen-2] Nguyen, T., Cogranne, R., and G. Doyen, "An Optimal Statistical Test for Robust Detection against Interest Flooding Attacks in CCN", Proceedings of the 14th IEEE/IFIP International Symposium on Integrated Network Management, 2015.
- [Nguyen-3] Nguyen, T. and et al., "A Security Monitoring Plane for Named Data Networking Deployment", IEEE Communications Magazine, Nov 2018.
- [ONAP] ONAP, "Open Network Automation Platform", 2017, <<https://www.onap.org/>>.
- [oneM2M] OneM2M, "oneM2M Service Layer Standards for M2M and IoT", 2017, <<http://www.onem2m.org/>>.
- [Overlay\_ICN] Shailendra, S. and et al., "A Novel Overlay Architecture for F Networking", 2016, <[https://www.researchgate.net/publication/282779666\\_A\\_novel\\_overlay\\_architecture\\_for\\_Information\\_Centric\\_Networking](https://www.researchgate.net/publication/282779666_A_novel_overlay_architecture_for_Information_Centric_Networking)>.
- [POINT] Trossen, D. and et al., "POINT: IP Over ICN - The Better IP?", European Conference on Networks and Communications (EuCNC), , 2015.
- [Ravindran] Ravindran, R. and et al., "5G-ICN : Delivering ICN Services over 5G using Network Slicing", IEEE Communication Magazine, May, 2016, <<https://arxiv.org/abs/1610.01182>>.
- [Reed] Reed, M. and et al., "Stateless Multicast Switching in Software Defined Networks", ICC 2016, Kuala Lumpur, Malaysia, 2016.
- [RFC6920] Farrell, S., Kutscher, D., Dannewitz, C., Ohlman, B., Keranen, A., and P. Hallam-Baker, "Naming Things with Hashes", RFC 6920, DOI 10.17487/RFC6920, April 2013, <<https://www.rfc-editor.org/info/rfc6920>>.
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <<https://www.rfc-editor.org/info/rfc7252>>.

- [RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S., Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-Defined Networking (SDN): Layers and Architecture Terminology", RFC 7426, DOI 10.17487/RFC7426, January 2015, <<https://www.rfc-editor.org/info/rfc7426>>.
- [RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <<https://www.rfc-editor.org/info/rfc7665>>.
- [RFC7927] Kutscher, D., Ed., Eum, S., Pentikousis, K., Psaras, I., Corujo, D., Saucez, D., Schmidt, T., and M. Waehlich, "Information-Centric Networking (ICN) Research Challenges", RFC 7927, DOI 10.17487/RFC7927, July 2016, <<https://www.rfc-editor.org/info/rfc7927>>.
- [RFC7945] Pentikousis, K., Ed., Ohlman, B., Davies, E., Spirou, S., and G. Boggia, "Information-Centric Networking: Evaluation and Security Considerations", RFC 7945, DOI 10.17487/RFC7945, September 2016, <<https://www.rfc-editor.org/info/rfc7945>>.
- [RFC8075] Castellani, A., Loreto, S., Rahman, A., Fossati, T., and E. Dijk, "Guidelines for Mapping Implementations: HTTP to the Constrained Application Protocol (CoAP)", RFC 8075, DOI 10.17487/RFC8075, February 2017, <<https://www.rfc-editor.org/info/rfc8075>>.
- [RFC8613] Selander, G., Mattsson, J., Palombini, F., and L. Seitz, "Object Security for Constrained RESTful Environments (OSCORE)", RFC 8613, DOI 10.17487/RFC8613, July 2019, <<https://www.rfc-editor.org/info/rfc8613>>.
- [SAIL] SAIL, "Scalable and Adaptive Internet Solutions (SAIL)", 2013, <<http://www.sail-project.eu/>>.
- [SAIL\_Content\_Delivery]  
FP7, "SAIL Content Delivery and Operations", 2013, <[https://sail-project.eu/wp-content/uploads/2012/06/SAIL\\_DB2\\_v1\\_0\\_final-Public.pdf](https://sail-project.eu/wp-content/uploads/2012/06/SAIL_DB2_v1_0_final-Public.pdf)>.
- [SAIL\_Prototyping]  
FP7, "SAIL Prototyping and Evaluation", 2013, <[http://www.sail-project.eu/wp-content/uploads/2013/05/SAIL\\_DB4\\_v1.1\\_Final\\_Public.pdf](http://www.sail-project.eu/wp-content/uploads/2013/05/SAIL_DB4_v1.1_Final_Public.pdf)>.

- [Tateson] Tateson, J. and et al., "Final Evaluation Report on Deployment Incentives and Business Models", 2010, <[http://www.psirp.org/files/Deliverables/FP7-INFISO-ICT-216173-PSIRP-D4.6\\_FinalReportOnDeplIncBusinessModels.pdf](http://www.psirp.org/files/Deliverables/FP7-INFISO-ICT-216173-PSIRP-D4.6_FinalReportOnDeplIncBusinessModels.pdf)>.
- [Techno\_Economic] Trossen, D. and A. Kostopolous, "Techno-Economics Aspects of Information-Centric Networking", Journal for Information Policy, Volume 2, 2012.
- [UMOBILE-2] Sarros, C. and et al., "Connecting the Edges: A Universal, Mobile-Centric, and Opportunistic Communications Architecture", IEEE Communications Magazine, vol. 56, February 2018.
- [UMOBILE-3] Tavares, M., Aponte, O., and P. Mendes, "Named-data Emergency Network Services", Proc. of ACM MOBISYS, Munich, Germany, June 2018.
- [UMOBILE-4] Lopes, L. and et al., "Oi! - Opportunistic Data Transmission Based on Wi-Fi Direct", Proc. of IEEE INFOCOM, San Francisco, USA, April 2016.
- [UMOBILE-5] Dynierowicz, S. and P. Mendes, "Named-Data Networking in Opportunistic Networks", Proc. of ACM ICN, Berlin, Germany, September 2017.
- [UMOBILE-6] Mendes, P. and et al., "Information-centric Routing for Opportunistic Wireless Networks", Proc. of ACM ICN, Boston, USA, September 2018.
- [UMOBILE-7] Sofia, R., "The UMOBILE Contextual Manager Service. Technical Report. Technical Report Senception 001, 2018 (base for UMOBILE deliverable D4.5 - Report on Data Collection and Inference Models", 2018.
- [UMOBILE-8] Sarros, C. and et al., "ICN-based edge service deployment in challenged networks", Proceedings of the 4th ACM Conference on Information-Centric Networking (ICN '17). ACM, New York, NY, USA, 2017 .



## [UMOBILE-9]

Lertsinsrubeetavee, A. and et al., "Information-Centric Multi-Access Edge Computing Platform for Community Mesh Networks", Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies (COMPASS '18). ACM, New York, NY, USA, 2018 .

## [UMOBILE-overview]

UMOBILE, "Universal Mobile-centric and Opportunistic Communications Architecture (UMOBILE)", 2018, <<http://www.umobile-project.eu/>>.

## [VSER]

Ravindran, R. and et al., "Towards software defined ICN based edge-cloud services", CloudNetworking(CloudNet), IEEE International Conference on, IEEE International Conference on CloudNetworking(CloudNet), 2013.

## [VSER-Mob]

Azgin, A. and et al., "Seamless Mobility as a Service in Information-centric Networks", ACM ICN Sigcomm, IC5G Workshop, 2016.

## [White]

White, G. and G. Rutz, "Content Delivery with Content Centric Networking, CableLabs White Paper", 2016, <<http://www.cablelabs.com/wp-content/uploads/2016/02/Content-Delivery-with-Content-Centric-Networking-Feb-2016.pdf>>.

## Appendix A. Change Log

[Note to RFC Editor: Please remove this section before publication.]

Changes from draft-irtf-rev-06 to draft-irtf-rev-07:

- o Added reference to OSCORE (RFC 8613) which is a way of passing end-to-end encrypted content between HTTP and COAP without invalidating encryption. Thus it can be a potential model for HTTP to ICN, or COAP to ICN, to consider in the future.
- o Updated affiliation information for author Ravi Ravindran.

Changes from draft-irtf-rev-05 to draft-irtf-rev-06:

- o Various updates to ensure that draft complies with RFC 5743 (Definition of an Internet Research Task Force (IRTF) Document Stream) section 2.1.

Changes from draft-irtf-rev-04 to draft-irtf-rev-05:

- o Addressed detailed review comments from Marie-Jose Montpetit.

Changes from draft-irtf-rev-03 to draft-irtf-rev-04:

- o Added text from Paulo Mendes and Adisorn Lertsinsruttavee on UMOBILE Trial Experiences.
- o Incorporated off-line editorial comments from Hitoshi Asaeda and Anil Jangam.

Changes from draft-irtf-rev-02 to draft-irtf-rev-03:

- o Editorial update of description and references of Doctor testbed as per comments from Guillaume Doyen.
- o Ran IETF spell checker tool and corrected various spelling errors.

Changes from draft-irtf-rev-01 to draft-irtf-rev-02:

- o Updated description of Doctor testbed as per comments from Guillaume Doyen. Also referenced Doctor testbed from the Security Considerations section.
- o Added "Composite-ICN" configuration to cover the Hybrid ICN and similar configurations which do not clearly fit in one of the other basic configurations.
- o Updated description of the ICN-as-a-Slice configuration to clarify that it may also apply to non-5G systems.

Changes from draft-irtf-rev-00 to draft-irtf-rev-01:

- o Added text from Michael Kowal describing NREN ICN Testbed.
- o Added text from Guillaume Doyen describing Doctor Project.
- o Updated text on Hybrid ICN based on input from Luca Muscariello.

Changes from draft-rahman-rev-05 to draft-irtf-rev-00:

- o Changed draft status from individual draft-rahman-icnrg-deployment-guidelines-05 to RG adopted draft-irtf-icnrg-deployment-guidelines-00.

Changes from rev-04 to rev-05:

- o Added this Change Log in Appendix A.
- o Removed references to Hybrid ICN from section 3.2 (ICN-as-an-Overlay definition). Instead, consolidated all Hybrid ICN info in the Deployment Trial Experiences under a new subsection 5.3 (Other Configurations).
- o Updated ICN2020 description in Section 5.1.4 with text received from Mayutan Arumaithurai and Hitoshi Asaeda.
- o Clarified in ICN-as-a-Slice description (section 3.4) that it may be deployed on either the Edge (RAN) or Core Network, or the ICN-as-a-Slice may be deployed end-to-end through the entire Mobile network.
- o Added several new references in various sections.
- o Various minor editorial updates.

#### Authors' Addresses

Akbar Rahman  
InterDigital Communications, LLC  
1000 Sherbrooke Street West, 10th floor  
Montreal H3A 3G4  
Canada

Email: Akbar.Rahman@InterDigital.com  
URI: <http://www.InterDigital.com/>

Dirk Trossen  
InterDigital Europe, Ltd  
64 Great Eastern Street, 1st Floor  
London EC2A 3QR  
United Kingdom

Email: Dirk.Trossen@InterDigital.com  
URI: <http://www.InterDigital.com/>

Dirk Kutscher  
University of Applied Sciences Emden/Leer  
Constantiapl. 4  
Emden 26723  
Germany

Email: [ietf@dkutscher.net](mailto:ietf@dkutscher.net)  
URI: <https://www.hs-emden-leer.de/en/>

Ravi Ravindran  
Future Technologies  
2330 Central Expressway  
Santa Clara 95050  
USA

Email: [ravi.ravindran@futurewei.com](mailto:ravi.ravindran@futurewei.com)

ICN Research Group  
Internet-Draft  
Intended status: Experimental  
Expires: 22 September 2022

Prakash Suthar  
Google Inc.  
Milan Stolic  
Anil Jangam, Ed.  
Cisco Systems Inc.  
Dirk Trossen  
Huawei Technologies  
Ravi Ravindran  
F5 Networks  
21 March 2022

Experimental Scenarios of ICN Integration in 4G Mobile Networks  
draft-irtf-icnrg-icn-lte-4g-12

Abstract

4G mobile network uses IP-based transport for the control plane to establish the data session at the user plane for the actual data delivery. In the existing architecture, IP-based unicast is used for the delivery of multimedia content to a mobile terminal, where each user is receiving a separate stream from the server. From a bandwidth and routing perspective, this approach is inefficient. Evolved multimedia broadcast and multicast service (eMBMS) provides capabilities for delivering contents to multiple users simultaneously, but its deployment is very limited or at an experimental stage due to numerous challenges. The focus of this draft is to list the options for use of Information centric technology (ICN) in 4G mobile networks and elaborate the experimental setups for its further evaluation. The experimental setups discussed provide for using ICN either natively or with existing mobility protocol stack. With further investigations based on the listed experiments, ICN with its inherent capabilities such as, network-layer multicast, anchorless mobility, security, and optimized data delivery using local caching at the edge may provide a viable alternative to IP transport in 4G mobile networks.

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 <https://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 22 September 2022.

#### Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

#### Table of Contents

1. Introduction . . . . .	3
2. 3GPP Terminology and Concepts . . . . .	3
3. 4G Mobile Network Architecture . . . . .	7
3.1. Network Overview . . . . .	7
3.2. Mobile Network Quality of Service . . . . .	9
3.3. Data Transport Using IP . . . . .	10
3.4. Virtualized Mobile Networks . . . . .	11
4. Data Transport Using ICN . . . . .	11
5. Experimental Scenarios for ICN Deployment . . . . .	14
5.1. General Considerations . . . . .	14
5.2. Scenarios of ICN Integration . . . . .	15
5.3. Integration of ICN in 4G Control Plane . . . . .	18
5.4. Integration of ICN in 4G User Plane . . . . .	20
5.4.1. Dual Transport (IP/ICN) Mode in Mobile Terminal . . . . .	20
5.4.2. Using ICN in Mobile Terminal . . . . .	24
5.4.3. Using ICN in eNodeB . . . . .	25
5.4.4. Using ICN in Packet Core (SGW, PGW) Gateways . . . . .	27
5.5. An Experimental Test Setup . . . . .	29
6. Expected Outcomes from Experimentation . . . . .	30
6.1. Feeding into ICN Research . . . . .	30
6.2. Use of Results Beyond Research . . . . .	31
7. Security and Privacy Considerations . . . . .	31
7.1. Security Considerations . . . . .	32
7.2. Privacy Considerations . . . . .	33
8. Summary . . . . .	35

9. Acknowledgements . . . . .	36
10. References . . . . .	36
10.1. Normative References . . . . .	36
10.2. Informative References . . . . .	37
Authors' Addresses . . . . .	42

## 1. Introduction

4G mobile technology is built as an all-IP network using routing protocols (OSPF, ISIS, BGP, etc.) to establish network routes. Stickiness of an IP address to a device is the key to get connected to a mobile network. The same IP address is maintained through the session until the device gets detached or moves to another network.

Key protocols used in 4G networks are GPRS Tunneling protocol (GTP), DIAMETER, and other protocols that are built on top of IP. One of the biggest challenges with IP-based routing in 4G is that it is not optimized for data transport. As an alternative to IP routing, this draft presents and list the possible options for integration of Information Centric Networking (ICN) in 3GPP 4G mobile network, offering an opportunity to leverage inherent ICN capabilities such as in-network caching, multicast, anchorless mobility management, and authentication. This draft also discuss how those options affect mobile providers and end users.

The goal of the proposed experiments is to present possibilities to create simulated environments for evaluation of the benefits of ICN protocol deployment in a 4G mobile network in different scenarios that have been analyzed in this document. The consensus of the Information-Centric Networking Research Group (ICNRG) is to publish this document in order to facilitate experiments to show deployment options and qualitative and quantitative benefits of ICN protocol deployment in 4G mobile networks.

## 2. 3GPP Terminology and Concepts

### 1. Access Point Name

The Access Point Name (APN) is a Fully Qualified Domain Name (FQDN) and resolves to a set of gateways in an operator's network. APN identifies the packet data network (PDN) with which a mobile data user wants to communicate. In addition to identifying a PDN, an APN may also be used to define the type of service, QoS, and other logical entities inside GGSN, PGW.

### 2. Control Plane

The control plane carries signaling traffic and is responsible for routing between eNodeB and MME, MME and HSS, MME and SGW, SGW and PGW, etc. Control plane signaling is required to authenticate and authorize the mobile terminal and establish a mobility session with mobile gateways (SGW/PGW). Control plane functions also include system configuration and management.

### 3. Dual Address PDN/PDP Type

The dual address Packet Data Network/Packet Data Protocol (PDN/PDP) Type (IPv4v6) is used in 3GPP context, in many cases as a synonym for dual stack, i.e., a connection type capable of serving IPv4 and IPv6 simultaneously.

### 4. eNodeB

The eNodeB is a base station entity that supports the Long-Term Evolution (LTE) air interface.

### 5. Evolved Packet Core

The Evolved Packet Core (EPC) is an evolution of the 3GPP GPRS system characterized by a higher-data-rate, lower-latency, packet-optimized system. The EPC comprises some sub components of the EPS core such as Mobility Management Entity (MME), Serving Gateway (SGW), Packet Data Network Gateway (PDN-GW), and Home Subscriber Server (HSS).

### 6. Evolved Packet System

The Evolved Packet System (EPS) is an evolution of the 3GPP GPRS system characterized by a higher-data-rate, lower-latency, packet-optimized system that supports multiple Radio Access Technologies (RATs). The EPS comprises the EPC together with the Evolved Universal Terrestrial Radio Access (E-UTRA) and the Evolved Universal Terrestrial Radio Access Network (E-UTRAN).

### 7. Evolved UTRAN

The E-UTRAN is a communications network sometimes referred to as 4G, and consists of eNodeB (4G base stations). The E-UTRAN allows connectivity between the User Equipment and the core network.

### 8. GPRS Tunneling Protocol



The GPRS Tunneling Protocol (GTP) [TS29.060] [TS29.274] [TS29.281] is a tunneling protocol defined by 3GPP. It is a network-based mobility protocol, working similar to Proxy Mobile IPv6 (PMIPv6). However, GTP also provides functionality beyond mobility, such as in-band signaling related to QoS and charging, among others.

#### 9. Gateway GPRS Support Node

The Gateway GPRS Support Node (GGSN) is a gateway function in the GPRS and 3G network that provides connectivity to the Internet or other PDNs. The host attaches to a GGSN identified by an APN assigned to it by an operator. The GGSN also serves as the topological anchor for addresses/prefixes assigned to the User Equipment.

#### 10. General Packet Radio Service

The General Packet Radio Service (GPRS) is a packet-oriented mobile data service available to users of the 2G and 3G cellular communication systems--the GSM--specified by 3GPP.

#### 11. Home Subscriber Server

The Home Subscriber Server (HSS) is a database for a given subscriber and was introduced in 3GPP Release-5. It is the entity containing subscription-related information to support the network entities that handle calls/sessions.

#### 12. Mobility Management Entity

The Mobility Management Entity (MME) is a network element responsible for control plane functionalities, including authentication, authorization, bearer management, layer-2 mobility, and so on. The MME is essentially the control plane part of the SGSN in the GPRS. The user plane traffic bypasses the MME.

#### 13. Public Land Mobile Network

The Public Land Mobile Network (PLMN) is a network operated by a single administration. A PLMN (and, therefore, also an operator) is identified by the Mobile Country Code (MCC) and the Mobile Network Code (MNC). Each (telecommunications) operator providing mobile services has its own PLMN.

#### 14. Policy and Charging Control

The Policy and Charging Control (PCC) framework is used for QoS policy and charging control. It has two main functions: flow-based charging (including online credit control), and policy control (for example, gating control, QoS control, and QoS signaling). It is optional to 3GPP EPS but needed if dynamic policy and charging control by means of PCC rules based on user and services are desired.

#### 15. Packet Data Network

The Packet Data Network (PDN) is a packet-based network that either belongs to the operator or is an external network such as the Internet or a corporate intranet. The user eventually accesses services in one or more PDNs. The operator's packet core networks are separated from packet data networks either by GGSNs or PDN Gateways (PGWs).

#### 16. Serving Gateway

The Serving Gateway (SGW) is a gateway function in the EPS, which terminates the interface towards the E-UTRAN. The SGW is the Mobility Anchor point for layer-2 mobility (inter-eNodeB handovers). For each mobile terminal connected with the EPS, there is only one SGW at any given point in time. The SGW is essentially the user plane part of the GPRS's SGSN.

#### 17. Packet Data Network Gateway

The Packet Data Network Gateway (PGW) is a gateway function in the Evolved Packet System (EPS), which provides connectivity to the Internet or other PDNs. The host attaches to a PGW identified by an APN assigned to it by an operator. The PGW also serves as the topological anchor for addresses/prefixes assigned to the User Equipment.

#### 18. Packet Data Protocol Context

A Packet Data Protocol (PDP) context is the equivalent of a virtual connection between the mobile terminal (MT) and a PDN using a specific gateway.

#### 19. Packet Data Protocol Type

A Packet Data Protocol Type (PDP Type) identifies the used/allowed protocols within the PDP context. Examples are IPv4, IPv6, and IPv4v6 (dual-stack).

#### 20. Serving GPRS Support Node

The Serving GPRS Support Node (SGSN) is a network element located between the radio access network (RAN) and the gateway (GGSN). A per-MT point-to-point (p2p) tunnel between the GGSN and SGSN transports the packets between the mobile terminal and the gateway.

#### 21. Mobile Terminal/User Equipment

The terms User Equipment (UE), Mobile Station (MS), Mobile Node (MN), and mobile refer to the devices that are hosts with the ability to obtain Internet connectivity via a 3GPP network. An MS comprises the Terminal Equipment (TE) and a Mobile Terminal (MT). The terms MT, MS, MN, and mobile are used interchangeably within this document.

#### 22. User Plane

The user plane refers to data traffic and the required bearers for the data traffic. In practice, IP is the only data traffic protocol used in the user plane.

### 3. 4G Mobile Network Architecture

This section provide a high-level overview of typical 4G mobile network architecture and their key functions related to a possibility of using of ICN technology.

#### 3.1. Network Overview

4G mobile networks are designed to use IP transport for communication among different elements such as eNodeB, MME, SGW/PGW, HSS, PCRF, etc. [GRAYSON]. For backward compatibility with 3G, it has support for legacy Circuit Switch features such as voice and SMS through transitional CS fallback and flexible IMS deployment. For each mobile device attached to the radio (eNodeB), there is a separate overlay tunnel (GPRS Tunneling Protocol, GTP) between eNodeB and Mobile gateways (i.e., SGW, PGW).

When any mobile terminal is powered up, it attaches to a mobile network based on its configuration and subscription. After a successful attachment procedure, the mobile terminal registers with the mobile core network using IPv4 and/or IPv6 address based on request and capabilities offered by mobile gateways.

The GTP tunnel is used to carry user traffic between gateways and mobile terminal, therefore using the unicast delivery for any data transfer. It is also important to understand the overhead of GTP and IPsec protocols. All mobile backhaul traffic is encapsulated using a

GTP tunnel, which has overhead of 8 bytes on top of IP and UDP [NGMN]. Additionally, if IPsec is used for security (which is often required if the Service Provider is using a shared backhaul), it adds overhead based on the IPsec tunneling model (tunnel or transport) as well as the encryption and authentication header algorithm used. If we consider as an example an Advanced Encryption Standard (AES) encryption, the overhead can be significant [OLTEANU], particularly for smaller payloads.

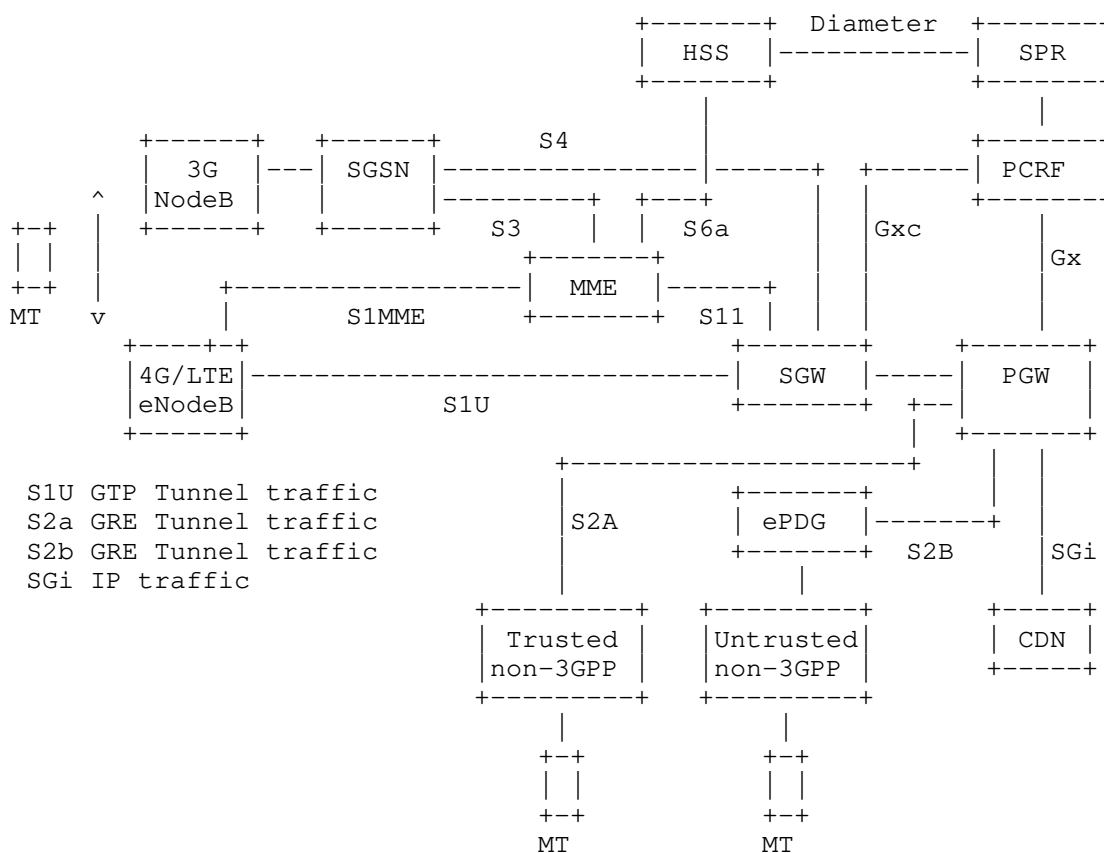


Figure 1: 4G Mobile Network Overview

If we consider the combined impact of GTP, IPsec and unicast traffic, the data delivery is not efficient because of overhead. The IETF has developed various header compression algorithms to reduce the overhead associated with IP packets. Some techniques are robust header compression (ROHC) and enhanced compression of the real-time transport protocol (ECRTP) so that the impact of overhead created by

GTP, IPsec, etc., is reduced to some extent [BROWER]. For commercial mobile networks, 3GPP has adopted different mechanisms for header compression to achieve efficiency in data delivery [TS25.323]; those solutions can be adapted to other data protocols, such as ICN, too [ICNLOWPAN] [TLVCOMP].

### 3.2. Mobile Network Quality of Service

During the mobile terminal attachment procedure, a default bearer is created for each mobile terminal and it is assigned to the default Access Point Name (APN), which provides the default transport. For any QoS-aware application, one or more new dedicated bearers are established between eNodeB and Mobile Gateway. Dedicated bearer can be requested either by mobile terminal or mobile gateway based on direction of first data flow. There are many bearers (logical paths) established between eNodeB and mobile gateway for each mobile terminal catering to different data flow simultaneously.

While all traffic within a certain bearer receives the same treatment, QoS parameters supporting these requirements can be very granular in different bearers. These values vary for the control, management and user traffic, and can be very different depending on application key parameters such as latency, jitter (important for voice and other real-time applications), packet loss, and queuing mechanism (strict priority, low-latency, fair, and so on).

Implementation of QoS for mobile networks is done at two stages: at content prioritization/marketing and transport marking, and congestion management. From the transport perspective, QoS is defined at layer 2 as class of service (CoS) and at layer 3 as Differentiated Services (DS). The mapping of DSCP to CoS takes place at layer 2/3 switching and routing elements. 3GPP has specified a QoS Class Identifier (QCI), which represents different types of content and equivalent mappings to the DSCP at transport layer [TS23.401]. However, this requires manual configuration at different elements and is therefore prone to possible misconfigurations.

In summary, QoS configuration in mobile networks for user plane traffic requires synchronization of parameters among different platforms. Normally, QoS in IP is implemented using DiffServ, which uses hop-by-hop QoS configuration at each router. Any inconsistency in IP QoS configuration at routers in the forwarding path can result in a poor subscriber experience (e.g., packet classified as high priority can go to a lower priority queue). By deploying ICN, we intend to enhance the subscriber experience using policy-based configuration, which can be associated with the named contents [ICNQoS] at the ICN forwarder. Further investigation is underway to understand how QoS in ICN [I-D.anilj-icnrg-dnc-qos-icn] can be implemented with reference to the ICN QoS guidelines [RFC9064] to meet the QoS requirements [RFC4594].

### 3.3. Data Transport Using IP

The data delivered to mobile devices is sent in unicast semantic inside the GTP tunnel from an eNodeB to a PDN gateway (PGW), as described in 3GPP specifications [TS23.401]. While the technology exists to address the issue of possible multicast delivery, there are many difficulties related to multicast protocol implementations on the RAN side of the network. By using eMBMS [EMBMS], multicast routing can be enabled in mobile backhaul between eNodeB and Mobile Gateways (SGW) however for radio interface it requires broadcast which implies that we need dedicated radio channel. Implementation of eMBMS in RAN is still lagging behind due to complexities related to client mobility, handovers, and the fact that the potential gain to Service Providers may not justify the investment, which explains the prevalence of unicast delivery in mobile networks. Techniques to handle multicast (such as LTE-M or eMBMS) have been designed to handle pre-planned content delivery, such as live content, which contrasts user behavior today, largely based on content (or video) on demand model.

To ease the burden on the bandwidth of the SGi interface, caching is introduced in a similar manner as with many Enterprises. In mobile networks, whenever possible, cached data is delivered. Caching servers are placed at a centralized location, typically in the Service Provider's Data Center, or in some cases lightly distributed in Packet Core locations with the PGW nodes close to the Internet and IP services access (SGi interface). This is a very inefficient concept because traffic must traverse the entire backhaul path for the data to be delivered to the end user. Other issues, such as out-of-order delivery, contribute to this complexity and inefficiency, which needs to be addressed at the application level.

### 3.4. Virtualized Mobile Networks

The Mobile gateways deployed in a major Service Provider network are either based on dedicated hardware or, commercially off the shelf (COTS) based x86 technology. With the adoption of Mobile Virtual Network Operators (MVNO), public safety networks, and enterprise mobility networks, elastic mobile core architecture are needed. By deploying the mobile packet core on COTS platform, using a virtualized infrastructure (NFVI) framework and end-to-end orchestration, new deployments can be simplified to provide optimized total cost of ownership (TCO).

While virtualization is growing, and many mobile providers use a hybrid architecture that consists of dedicated and virtualized infrastructures, the control, and data planes are still the same. There is also work under way to separate the control and user plane for the network to scale better. Virtualized mobile networks and network slicing with control and user plane separation provide a mechanism to evolve the GTP-based architecture towards an OpenFlow SDN-based signaling for 4G and proposed 5G core. Some early architecture work for 5G mobile technologies provides a mechanism for control and user plane separation and simplifies the mobility call flow by introducing OpenFlow-based signaling [ICN5G]. This has been considered by 3GPP [EPCCUPS] and is also described in [SDN5G].

## 4. Data Transport Using ICN

For mobile devices, the edge connectivity is between mobile terminal and a router or mobile edge computing (MEC) [MECSPEC] element. Edge computing has the capability of processing client requests and segregating control and user traffic at the edge of radio, rather than sending all requests to the mobile gateway.

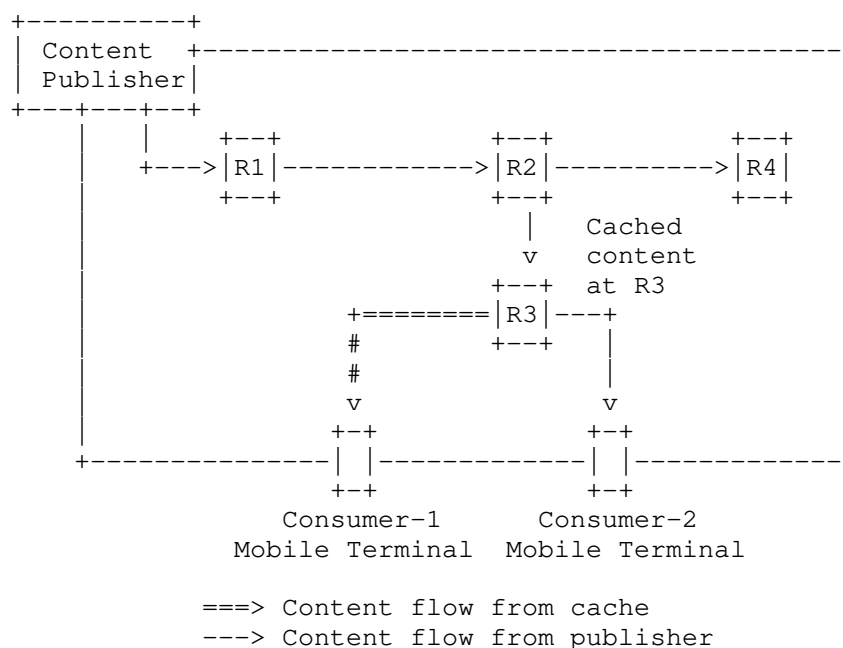


Figure 2: ICN Architecture

Edge computing transforms radio access network into an intelligent service edge capable of delivering services directly from the edge of the network, while providing the best possible performance to the client. Edge computing can be an ideal candidate for implementing ICN forwarders in addition to its usual function of managing mobile termination. In addition to edge computing, other transport elements, such as routers, can work as ICN forwarders.

Data transport using ICN is different to IP-based transport by introducing uniquely named-data as a core design principle. Communication in ICN takes place between the content provider (producer) and the end user (consumer), as described in Figure 2.

Every node in a physical path between a client and a content provider is called the ICN forwarder or router. It can route the request intelligently and cache content so it can be delivered locally for subsequent requests from any other client. For mobile networks, transport between a client and a content provider consists of radio network + mobile backhaul and IP core transport + Mobile Gateways + Internet + content data network (CDN).



To understand the suitability of ICN for mobile networks, we will discuss the ICN framework by describing its protocols architecture and different types of messages to then consider how we can use this in mobile networks for delivering content more efficiently. ICN uses two types of packets called "interest packet" and "data packet" as described in Figure 3.

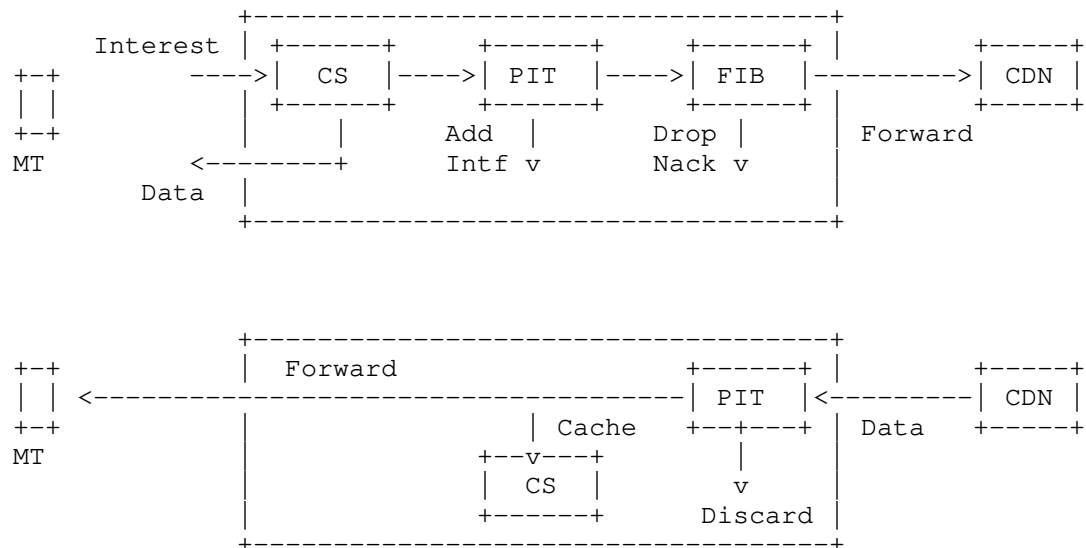


Figure 3: ICN Interest, Data Packet and Forwarder

In an 4G network, when a mobile device wants to receive certain content, it will send an Interest message to the closest eNodeB. Interest packets follow the TLV format [RFC8609] and contain mandatory fields, such as name of the content and content matching restrictions (KeyIdRestr and ContentObjectHashRestr), expressed as a tuple [RFC8569]. The content matching tuple uniquely identifies the matching data packet for the given Interest packet. Another attribute called HopLimit is used to detect looping Interest messages.

An ICN router will receive an Interest packet and lookup if a request for such content has arrived earlier from another client. If so, it may be served from the local cache; otherwise, the request is forwarded to the next-hop ICN router. Each ICN router maintains three data structures: Pending Interest Table (PIT), Forwarding Information Base (FIB), and Content Store (CS). The Interest packet travels hop-by-hop towards the content provider. Once the Interest packet reaches the content provider, it will return a Data packet containing information such as content name, signature, and the actual data.

The data packet travels in reverse direction following the same path taken by the Interest packet, maintaining routing symmetry. Details about algorithms used in PIT, FIB, CS, and security trust models are described in various resources [CCN]; here, we have explained the concept and its applicability to the 4G network.

## 5. Experimental Scenarios for ICN Deployment

In 4G mobile networks, both user and control plane traffic have to be transported from the edge to the mobile packet core via IP transport. The evolution of the existing mobile packet core using Control and User Plane Separation (CUPS) [TS23.714] enables flexible network and operations by distributed deployment and the independent scaling of control plane and user plane functions - while not affecting the functionality of existing nodes subject to this split.

In this section, we analyze the potential impact of ICN on control and user plane traffic for centralized and disaggregated CUPS-based mobile network architecture. We list various experimental options and opportunities to study the feasibility of the deployment of ICN in 4G networks. The proposed experiments would help the network and OEM designers to understand various issues, optimizations, and advantages of deployment of ICN in 4G networks.

### 5.1. General Considerations

In the CUPS architecture, there is an opportunity to shorten the path for user plane traffic by deploying offload nodes closer to the edge [OFFLOAD]. With this major architecture change, a User Plane Function (UPF) node is placed close to the edge so traffic no longer needs to traverse the entire backhaul path to reach the EPC. In many cases, where feasible, the UPF can be collocated with the eNodeB, which is also a business decision based on user demand. Placing a Publisher close to the offload site, or at the offload site, provides for a significant improvement in user experience, especially with latency-sensitive applications. This capability allows for the introduction of ICN and amplifies its advantages.

## 5.2. Scenarios of ICN Integration

The integration of ICN provides an opportunity to further optimize the existing data transport in 4G mobile networks. The various opportunities from the coexistence of ICN and IP transport in the mobile network are somewhat analogous to the deployment scenarios when IPv6 was introduced to interoperate with IPv4 except, with ICN, the whole IP stack can be replaced. We have reviewed [RFC6459] and analyzed the impact of ICN on control plane signaling and user plane data delivery. In general, ICN can be used natively by replacing IP transport (IPv4 and IPv6) or as an overlay protocol. Figure 4 describes a proposal to modify the existing transport protocol stack to support ICN in 4G mobile network.

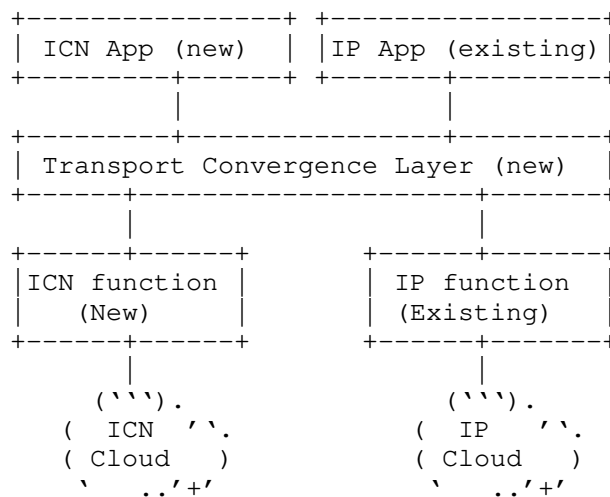


Figure 4: IP/ICN Convergence Scenarios

As shown in Figure 4, for applications - running either in the mobile terminal or in the content provider system - to use the ICN transport option, we propose a new transport convergence layer (TCL). The TCL helps determine the type of transport (such as ICN or IP), as well as the type of radio interface (LTE or WiFi or both) used to send and receive traffic based on preference (e.g., content location, content type, content publisher, congestion, cost, QoS). It helps configure and determine the type of connection (native IP or ICN) or the overlay mode (ICNoIP or IPoICN) between application and the protocol stack (IP or ICN).

Combined with the existing IP function, the ICN function provides support for either native ICN and/or the dual transport (ICN/IP) transport functionality. See Section 5.4.1 for elaborate descriptions of these functional layers.

The TCL can use several mechanisms for transport selection. It can use a per-application configuration through a management interface, possibly a user-facing setting realized through a user interface, like those used to select cellular over WiFi. In another option, it might use a software API, which an adapted IP application could use to specify the type of transport option (such as ICN) to take advantage of its benefits.

Another potential application of TCL is in implementation of network slicing, with a slice management capability locally or through an interface to an external slice manager via an API [GALIS]. This solution can enable network slicing for IP and ICN transport selection from the mobile terminal itself. The TCL could apply slice settings to direct certain applications traffic over one slice and others over another slice, determined by some form of 'slicing policy'. Slicing policy can be obtained externally from the slice manager or configured locally on the mobile terminal.

From the perspective of applications either on the mobile terminal or at a content provider, the following options are possible for potential use of ICN natively and/or with IP.

1. IP over IP

In this scenario, the mobile terminal applications are tightly integrated with the existing IP transport infrastructure. The TCL has no additional function because packets are forwarded directly using an IP protocol stack, which sends packets over the IP transport.

2. ICN over ICN

Similar to case 1, ICN applications tightly integrate with the ICN transport infrastructure. The TCL has no additional responsibility because packets are forwarded directly using the native ICN protocol stack, which sends packets over the ICN transport.

3. ICN over IP (ICNoIP)

In this scenario, the underlying IP transport infrastructure is not impacted (that is, ICN is implemented as an IP overlay between mobile terminal and content provider). IP routing is

used from the Radio Access Network (eNodeB) to the mobile backhaul, the IP core, and the Mobile Gateway (SGW/PGW). The mobile terminal attaches to the Mobile Gateway (SGW/PGW) using an IP address. Also, the data transport between Mobile Gateway (SGW/PGW) and content publisher uses IP. The content provider can serve content either using IP or ICN, based on the mobile terminal request.

One of the approaches to implement ICN in mobile backhaul networks is described in [MBICN]. It implements a GTP-U extension header option to encapsulate ICN payload in a GTP tunnel. However, as this design runs ICN as an IP overlay, the mobile backhaul can be deployed using native IP. The proposal describes a mechanism where the GTP-U tunnel can be terminated by hairpinning the packet before it reaches SGW, if an ICN-enabled node is deployed in the mobile backhaul (that is, between eNodeB and SGW). This could be useful when an ICN data packet is stored in the ICN node (such as repositories, caches) in the tunnel path so that the ICN node can reply without going all the way through the mobile core. While a GTP-U extension header is used to carry mobile terminal specific ICN payload, they are not visible to the transport, including SGW. On the other hand, the PGW can use the mobile terminal-specific ICN header extension and ICN payload to set up an uplink transport towards a content provider in the Internet. In addition, the design assumes a proxy function at the edge, to perform ICN data retrieval on behalf of a non-ICN end device.

#### 4. IP over ICN (IPoICN)

[IPoICN] provides an architectural framework for running IP as an overlay over ICN protocol. Implementing IP services over ICN provides an opportunity to leverage the benefits of ICN in the transport infrastructure while there is no impact on end devices (MT and access network) as they continue to use IP. IPoICN however, will require an inter-working function (IWF/Border Gateway) to translate various transport primitives. The IWF function will provide a mechanism for protocol translation between IPoICN and the native IP. After reviewing [IPoICN], we understand and interpret that ICN is implemented in the transport natively, however, IP is implemented in MT, eNodeB, and Mobile gateway (SGW/PGW), which is also called as a network attach point (NAP).

For this, said NAP receives an incoming IP or HTTP packet (the latter through TCP connection termination) and publishes the packet under a suitable ICN name (i.e., the hash over the destination IP address for an IP packet or the hash over the FQDN

of the HTTP request for an HTTP packet) to the ICN network. In the HTTP case, the NAP maintains a pending request mapping table to map returning responses to the terminated TCP connection.

## 5. Hybrid ICN (hICN)

An alternative approach to implement ICN over IP is provided in Hybrid ICN [HICN]. It describes a novel approach to integrate ICN into IPv6 without creating overlays with a new packet format as an encapsulation. hICN addresses the content by encoding a location-independent name in an IPv6 address. It uses two name components--name prefix and name suffix--that identify the source of data and the data segment within the scope of the name prefix, respectively.

At application layer, hICN maps the name into an IPv6 prefix and, thus, uses IP as transport. As long as the name prefixes, which are routable IP prefixes, point towards a mobile GW (PGW or local breakout, such as CUPS), there are potentially no updates required to any of the mobile core gateways (for example, SGW/PGW). The IPv6 backhaul routes the packets within the mobile core. hICN can run in the mobile terminal, in the eNodeB, in the mobile backhaul, or in the mobile core. Finally, as hICN itself uses IPv6, it cannot be considered as an alternative transport layer.

### 5.3. Integration of ICN in 4G Control Plane

In this section, we analyze signaling messages that are required for different procedures, such as attach, handover, tracking area update, and so on. The goal of this analysis is to see if there are any benefits to replacing IP-based protocols with ICN for 4G signaling in the current architecture. It is important to understand the concept of point of attachment (POA). When mobile terminal connects to a network, it has the following POAs:

1. eNodeB managing location or physical POA
2. Authentication and Authorization (MME, HSS) managing identity or authentication POA
3. Mobile Gateways (SGW, PGW) managing logical or session management POA

In the current architecture, IP transport is used for all messages associated with the control plane for mobility and session management. IP is embedded very deeply into these messages utilizing TLV syntax for carrying additional attributes such as a layer 3

transport. The physical POA in the eNodeB handles both mobility and session management for any mobile terminal attached to 4G network. The number of mobility management messages between different nodes in an 4G network per signaling procedure is shown in Table 1.

Normally, two types of mobile terminals attach to the 4G network: SIM based (need 3GPP mobility protocol for authentication) or non-SIM based (which connect to WiFi network). Both device types require authentication. For non-SIM based devices, AAA is used for authentication. We do not propose to change mobile terminal authentication or mobility management messaging for user data transport using ICN. A separate study would be required to analyze the impact of ICN on mobility management messages structures and flows. We are merely analyzing the viability of implementing ICN as a transport for control plane messages.

It is important to note that if MME and HSS do not support ICN transport, they still need to support mobile terminal capable of dual transport or native ICN. When mobile terminal initiates an attach request using the identity as ICN, MME must be able to parse that message and create a session. MME forwards mobile terminal authentication to HSS, so HSS must be able to authenticate an ICN-capable mobile terminal and authorize create session [TS23.401].

4G Signaling Procedures	MME	HSS	SGW	PGW	PCRF
Attach	10	2	3	2	1
Additional default bearer	4	0	3	2	1
Dedicated bearer	2	0	2	2	0
Idle-to-connect	3	0	1	0	0
Connect-to-Idle	3	0	1	0	0
X2 handover	2	0	1	0	0
S1 handover	8	0	3	0	0
Tracking area update	2	2	0	0	0
Total	34	2	14	6	3

Table 1: Signaling Messages in 4G Gateways

Anchorless mobility [ALM] provides a fully decentralized, control-plane agnostic solution to handle producer mobility in ICN. Mobility management at layer-3 level makes it access agnostic and transparent to the end device or the application. The solution discusses handling mobility without having to depend on core network functions (e.g. MME); however, a location update to the core network may still be required to support legal compliance requirements such as lawful intercept and emergency services. These aspects are open for further study.

One of the advantages of ICN is in the caching and reusing of the content, which does not apply to the transactional signaling exchange. After analyzing 4G signaling call flows [TS23.401] and messages inter-dependencies (see Table 1), our recommendation is that it is not beneficial to use ICN for control plane and mobility management functions. Among the features of ICN design, Interest aggregation and content caching are not applicable to control plane signaling messages. Control plane messages are originated and consumed by the applications and they cannot be shared.

#### 5.4. Integration of ICN in 4G User Plane

We will consider Figure 1 to discuss different mechanisms to integrate ICN in mobile networks. In Section 5.2, we discussed generic experimental setups of ICN integration. In this section, we discuss the specific options of possible use of native ICN in 4G user plane. We consider the following options:

1. Dual transport (IP/ICN) mode in Mobile Terminal
2. Using ICN in Mobile Terminal
3. Using ICN in eNodeB
4. Using ICN in mobile gateways (SGW/PGW)

##### 5.4.1. Dual Transport (IP/ICN) Mode in Mobile Terminal

The control and user plane communications in 4G mobile networks are specified in 3GPP documents [TS23.203] and [TS23.401]. It is important to understand that mobile terminal can be either consumer (receiving content) or publisher (pushing content for other clients). The protocol stack inside the mobile terminal (MT) is complex because it must support multiple radio connectivity access to eNodeB(s).



Figure 5 provides a high-level description of a protocol stack, where IP is used at two layers: (1) user plane communication and (2) UDP encapsulation. User plane communication takes place between Packet Data Convergence Protocol (PDCP) and Application layer, whereas UDP encapsulation is at GTP protocol stack.

The protocol interactions and impact of supporting tunneling of ICN packet into IP or to support ICN natively are described in Figure 5 and Figure 6, respectively.

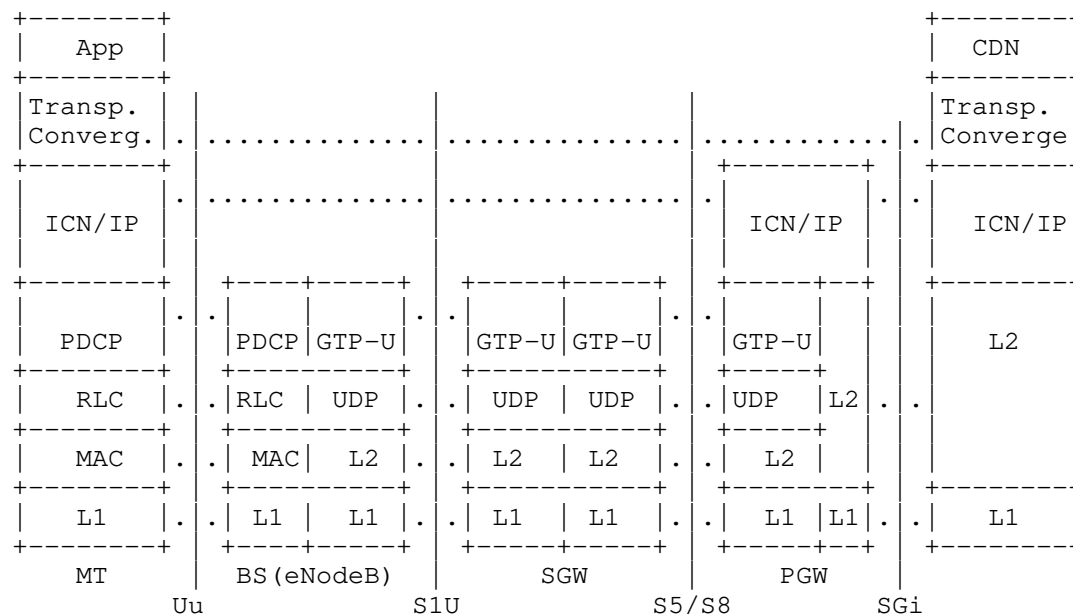


Figure 5: Dual Transport (IP/ICN) mode in Mobile Terminal

The protocols and software stack used inside 4G capable mobile terminal support both 3G and 4G software interworking and handover. 3GPP Rel.13 onward specifications describe the use of IP and non-IP protocols to establish logical/session connectivity. We can leverage the non-IP protocol-based mechanism to deploy ICN protocol stack in the mobile terminal, as well as in eNodeB and mobile gateways (SGW, PGW). The following paragraphs describe per-layer considerations of supporting tunneling of ICN packet into IP or to support ICN natively.

1. An existing application layer can be modified to provide options for a new ICN-based application and existing IP-based applications. The mobile terminal can continue to support

existing IP-based applications or can develop new applications to support native ICN, ICNoIP, or IPoICN-based transport. The application layer can be provided with an option of selecting either ICN or IP transport, as well as radio interface, to send and receive data traffic.

Our proposal is to provide an Application Programming Interface (API) to the application developers so they can choose either ICN or IP transport for exchanging the traffic with the network. As mentioned in Section 5.2, the transport convergence layer (TCL) function handles the interaction of applications with multiple transport options.

2. The transport convergence layer helps determine the type of transport (such as ICN, hICN, or IP) and type of radio interface (LTE or WiFi, or both) used to send and receive traffic. Application layer can make the decision to select a specific transport based on preference, such as content location, content type, content publisher, congestion, cost, QoS, and so on. There can be an Application Programming Interface (API) to exchange parameters required for transport selection. Southbound interactions of Transport Convergence Layer (TCL) will be either to IP or ICN at the network layer.

When selecting the IPoICN mode, the TCL is responsible for receiving an incoming IP or HTTP packet and publishing the packet to the ICN network under a suitable ICN name (that is, the hash over the destination IP address for an IP packet, or the hash over the FQDN of the HTTP request for an HTTP packet).

In the HTTP case, the TCL can maintain a pending request mapping table to map returning responses to the originating HTTP request. The common API will provide a 'connection' abstraction for this HTTP mode of operation, returning the response over said connection abstraction, akin to the TCP socket interface, while implementing a reliable transport connection semantic over the ICN from the mobile terminal to the receiving mobile terminal or the PGW. If the HTTP protocol stack remains unchanged, therefore utilizing the TCP protocol for transfer, the TCL operates in local TCP termination mode, retrieving the HTTP packet through said local termination.

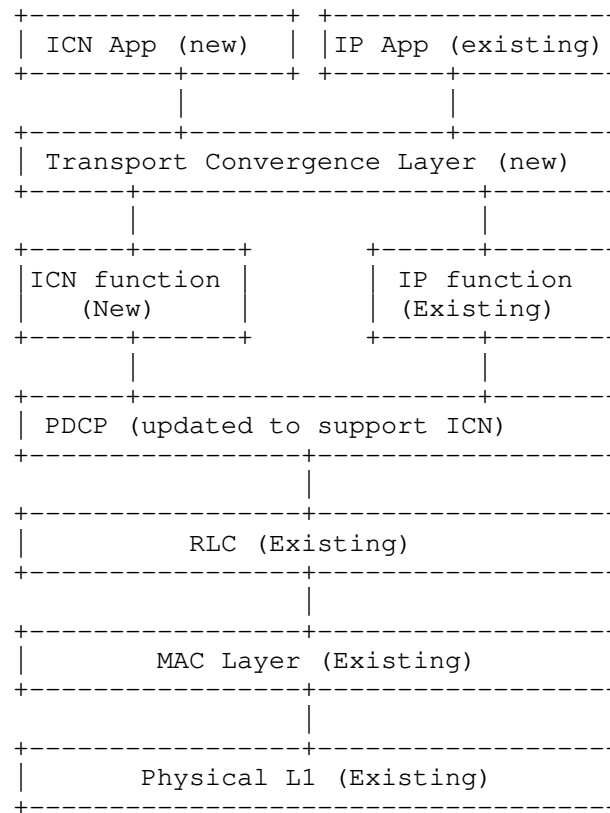


Figure 6: Dual Stack ICN Protocol Interactions

3. The ICN function (forwarder) is proposed to run in parallel to the existing IP layer. The ICN forwarder forwards the ICN packets, such as an Interest packet to eNodeB or a response "data packet" from eNodeB to the application.
4. For the dual-transport scenario, when mobile terminal is not supporting ICN as transport, the TCL can use the IP underlay to tunnel the ICN packets. The ICN function can use the IP interface to send Interest and Data packets for fetching or sending data respectively. This interface can use the ICN overlay over IP.

5. To support ICN at network layer in mobile terminal, the PDCP layer should be aware of ICN capabilities and parameters. PDCP is located in the Radio Protocol Stack in the LTE Air interface, between IP (Network layer) and Radio Link Control Layer (RLC). PDCP performs the following functions [TS36.323]:
  1. Data transport by listening to upper layer, formatting and pushing down to Radio Link Layer (RLC)
  2. Header compression and decompression using Robust Header Compression (ROHC)
  3. Security protections such as ciphering, deciphering, and integrity protection
  4. Radio layer messages associated with sequencing, packet drop detection and re-transmission, and so on.
6. No changes are required for lower layer such as RLC, MAC, and Physical (L1) as they are not IP aware.

One key point to understand in this scenario is that ICN is deployed as an overlay on top of IP.

#### 5.4.2. Using ICN in Mobile Terminal

We can implement ICN natively in mobile terminal by modifying the PDCP layer in 3GPP protocols. Figure 7 provides a high-level protocol stack description where ICN can be used at the following different layers:

1. User plane communication
2. Transport layer

ICN transport would be a substitute of the GTP protocol. The removal of the GTP protocol stack is a significant change in the mobile architecture and requires a thorough study mainly because it is used not just for routing but for mobility management functions, such as billing, mediation, and policy enforcement.

The implementation of ICN natively in the mobile terminal leads to a changed communication model between mobile terminal and eNodeB. Also, we can avoid tunneling the user plane traffic from eNodeB to the mobile packet core (SGW, PGW) through a GTP tunnel.

For native ICN use, an application can be configured to use ICN forwarder and it does not need the TCL layer. Also, to support ICN at the network layer, the existing PDCP layer may need to be changed to be aware of ICN capabilities and parameters.

The native implementation can provide new opportunities to develop new use cases leveraging ICN capabilities, such as seamless mobility, mobile terminal to mobile terminal content delivery using radio network without traversing the mobile gateways, and more.

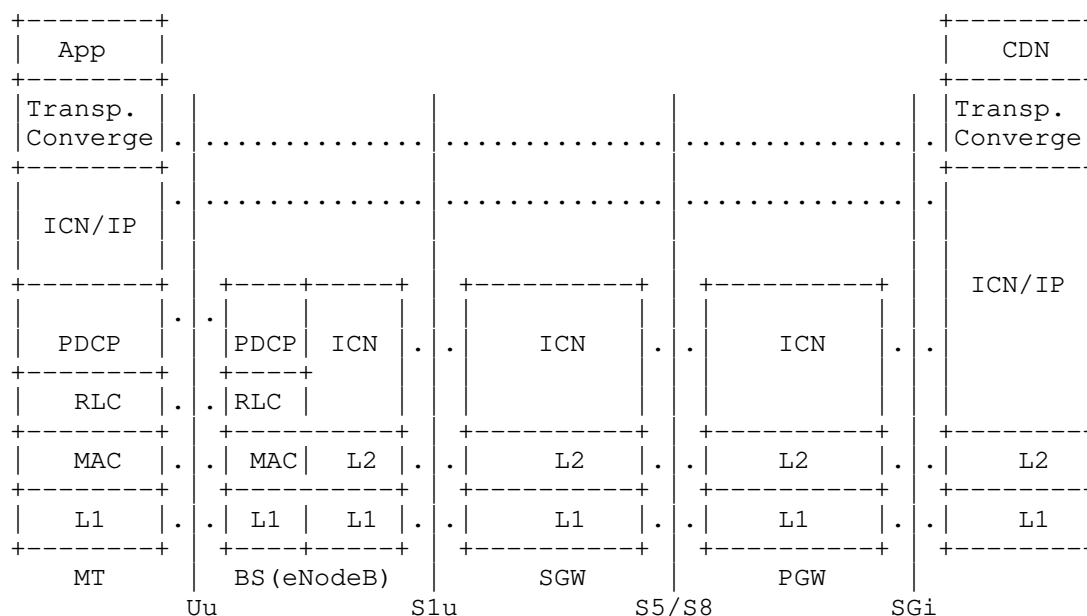


Figure 7: Using Native ICN in Mobile Terminal

#### 5.4.3. Using ICN in eNodeB

The eNodeB is a physical point of attachment for the mobile terminal, where radio protocols are converted into IP transport protocol for dual transport/overlay and native ICN, respectively (see Figure 6 and Figure 7). When a mobile terminal performs an attach procedure, it would be assigned an identity either as IP or dual transport (IP and ICN), or ICN endpoint. Mobile terminal can initiate data traffic using any of the following options:

1. Native IP (IPv4 or IPv6)
2. Native ICN

### 3. Dual transport IP (IPv4/IPv6) and ICN

The mobile terminal encapsulates a user data transport request into PDCP layer and sends the information on the air interface to eNodeB, which in turn receives the information and, using PDCP [TS36.323], de-encapsulates the air-interface messages and converts them to forward to core mobile gateways (SGW, PGW). As shown in Figure 8, to support ICN natively in eNodeB, it is proposed to provide transport convergence layer (TCL) capabilities in eNodeB (similar to as provided in MT), which provides the following functions:

1. It decides the forwarding strategy for a user data request coming from mobile terminal. The strategy can decide based on preference indicated by the application, such as congestion, cost, QoS, and so on.
2. eNodeB to provide open Application Programming Interface (API) to external management systems, to provide capability to eNodeB to program the forwarding strategies.

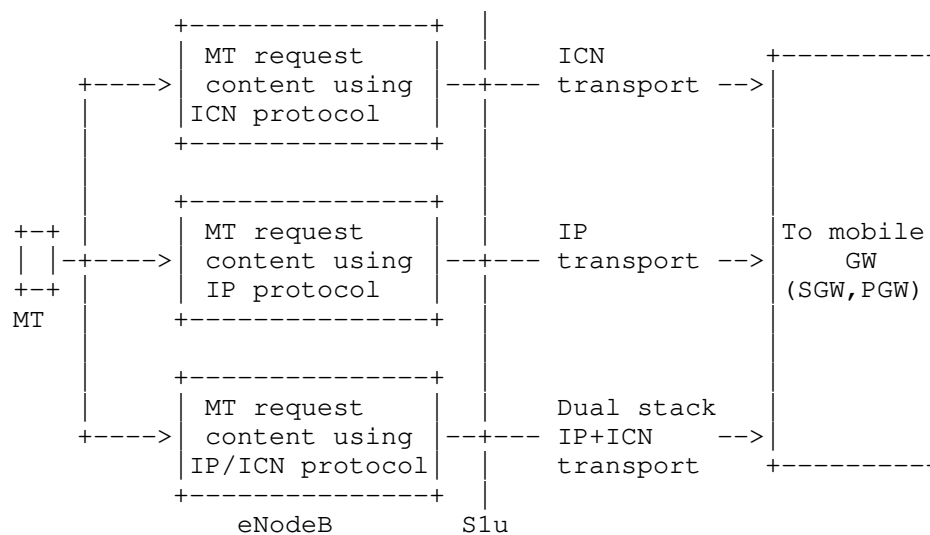


Figure 8: Integration of Native ICN in eNodeB

3. eNodeB can be upgraded to support three different types of transport: IP, ICN, and dual transport IP+ICN towards mobile gateways, as depicted in Figure 8. It is also proposed to deploy IP and/or ICN forwarding capabilities into eNodeB, for efficient transfer of data between eNodeB and mobile gateways. Following are choices for forwarding a data request towards mobile gateways:
  1. Assuming eNodeB is IP enabled and the MT requests an IP transfer, eNodeB forwards data over IP.
  2. Assuming eNodeB is ICN enabled and the MT requests an ICN transfer, eNodeB forwards data over ICN.
  3. Assuming eNodeB is IP enabled and the MT requests an ICN transfer, eNodeB overlays ICN on IP and forwards user plane traffic over IP.
  4. Assuming eNodeB is ICN enabled and the MT requests an IP transfer, eNodeB overlays IP on ICN and forwards user plane traffic over ICN [IPoICN].

#### 5.4.4. Using ICN in Packet Core (SGW, PGW) Gateways

Mobile gateways (a.k.a. Evolved Packet Core (EPC)) include SGW, PGW, which perform session management for MT from the initial attach to disconnection. When MT is powered on, it performs NAS signaling and attaches to PGW after successful authentication. PGW is an anchoring point for MT and responsible for service creations, authorization, maintenance, and so on. The Entire functionality is managed using IP address(es) for MT.

To implement ICN in EPC, the following functions are proposed:

1. Insert ICN attributes in session management layer as additional functionality with IP stack. Session management layer is used for performing attach procedures and assigning logical identity to user. After successful authentication by HSS, MME sends a create session request (CSR) to SGW and SGW to PGW.

2. When MME sends Create Session Request message (Step 12 in [TS23.401]) to SGW or PGW, it includes a Protocol Configuration Option Information Element (PCO IE) containing MT capabilities. We can use PCO IE to carry ICN-related capabilities information from MT to PGW. This information is received from MT during the initial attach request in MME. Details of available TLV, which can be used for ICN, are given in subsequent sections. MT can support either native IP, ICN+IP, or native ICN. IP is referred to as both IPv4 and IPv6 protocols.
3. For ICN+IP-capable MT, PGW assigns the MT both an IP address and ICN identity. MT selects either of the identities during the initial attach procedures and registers with the network for session management. For ICN-capable MT, it will provide only ICN attachment. For native IP-capable MT, there is no change.
4. To support ICN-capable attach procedures and use ICN for user plane traffic, PGW needs to have full ICN protocol stack functionalities. Typical ICN capabilities include functions such as content store (CS), Pending Interest Table (PIT), Forwarding Information Base (FIB) capabilities, and so on. If MT requests ICN in PCO IE, then PGW registers MT with ICN names. For ICN forwarding, PGW caches content locally using CS functionality.
5. PCO IE described in [TS24.008] (see Figure 10.5.136 on page 598) and [TS24.008] (see Table 10.5.154 on page 599) provide details for different fields.
  1. Octet 3 (configuration protocols define PDN types), which contains details about IPv4, IPv6, both or ICN.
  2. Any combination of Octet 4 to Z can be used to provide additional information related to ICN capability. It is most important that PCO IE parameters are matched between MT and mobile gateways (SGW, PGW) so they can be interpreted properly and the MT can attach successfully.
6. The ICN functionalities in SGW and PGW should be matched with MT and eNodeB because they will exchange ICN protocols and parameters.
7. Mobile gateways SGW, PGW will also need ICN forwarding and caching capability. This is especially important if CUPS is implemented. User Plane Function (UPF), comprising the SGW and PGW user plane, will be located at the edge of the network and close to the end user. ICN-enabled gateway means that this UPF would serve as a forwarder and should be capable of caching, as is the case with any other ICN-enabled node.



8. The transport between PGW and CDN provider can be either IP or ICN. When MT is attached to PGW with ICN identity and communicates with an ICN-enabled CDN provider, it will use ICN primitives to fetch the data. On the other hand, for a MT attached with an ICN identity, if PGW must communicate with an IP enabled CDN provider, it will have to use an ICN-IP interworking gateway to perform conversion between ICN and IP primitives for data retrieval. In the case of CUPS implementation with an offload close to the edge, this interworking gateway can be collocated with the UPF at the offload site to maximize the path optimization. Further study is required to understand how this ICN-to-IP (and vice versa) interworking gateway would function.

#### 5.5. An Experimental Test Setup

This section proposes an experimental lab setup and discusses the open issues and questions that use of ICN protocol is intended to address. To further test the modifications proposed in different scenarios, a simple lab can be set up, as shown in Figure 9.

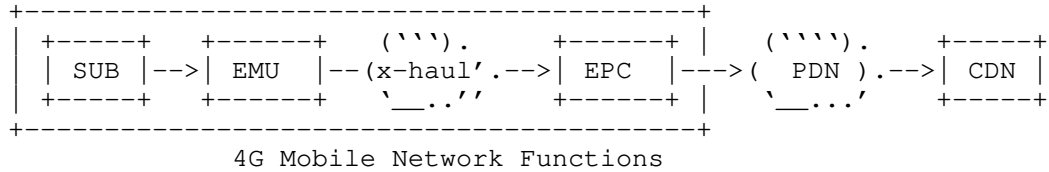


Figure 9: Native ICN Deployment Lab Setup

The following test scenarios can be set up with VM-based deployment:

1. SUB: ICN simulated client (using ndnSIM), a client application on workstation requesting content.
2. EMU: test unit emulating eNodeB. This will be a test node allowing for UE attachment and routing traffic subsequently from the Subscriber to the Publisher.
3. EPC: Evolved Packet Core in a single instance (such as 5GOpenCore [Open5GCore]).
4. CDN: content delivery by a Publisher server.

For the purpose of this testing, ICN emulating code can be inserted in the test code in EMU to emulate ICN-capable eNodeB. An example of the code to be used is NS3 in its LTE model. Effect of such traffic on EPC and CDN can be observed and documented. In a subsequent phase, EPC code supporting ICN can be tested when available.

Another option is to simulate the UE/eNodeB and EPC functions using NS3's LTE [NS3LTE] and EPC [NS3EPC] models respectively. LTE model includes the LTE Radio Protocol stack, which resides entirely within the UE and the eNodeB nodes. This capability provides the simulation of UE and eNodeB deployment use cases. Similarly, EPC model includes core network interfaces, protocols, and entities, which reside within the SGW, PGW and MME nodes, and partially within the eNodeB nodes.

Even with its current limitations (such as IPv4 only, lack of integration with ndnSIM, no support for UE idle state), LTE simulation may be a very useful tool. In any case, both control and user plane traffic should be tested independently according to the deployment model discussed in Section 5.4.

## 6. Expected Outcomes from Experimentation

The experimentations explained in Section 5 can be categorized in three broader scopes as follows. Note that, a further research and study is required to fully understand and document the impact.

1. Architecture scope: to study the aspect of use of ICN at user plane to reduce the complexities in current transport protocols, while also evaluating its use in the control plane.
2. Performance scope: to evaluate the gains through multicast, caching, and other ICN features.
3. Deployment scope: to check the viability of the ICN inclusion in 3GPP protocol stack and its viability in real-world deployments.

### 6.1. Feeding into ICN Research

Specifically, we have identified the following open questions, from the architectural and performance perspective, that the proposed experiments with ICN implementation scenarios in 4G mobile networks could address in further research:

1. Efficiency gains in terms of the amount of traffic in multicast scenarios (i.e., quantify the possible gains along different use cases) and the efficiency gained in terms of latency for cached content, mainly in the CDN use case.

2. How the new transport would coexist or replace the legacy transport protocols (e.g., IPv4, IPv6, MPLS, RSVP, etc.) and related services (e.g., bandwidth management, QoS handling, etc.).
3. To what extent the simplification in the IP-based transport protocols will be achieved. The multiple overlays (e.g., the MPLS, VPN, VPLS, Ethernet VPN, etc.) of services in the current IP-based transport adds to the complexity on top of basic IP transport. This makes the troubleshooting extremely challenging.
4. How the new transport can become service-aware such that it brings in more simplicity in the system.
5. Confirm how (in)adequate would be ICN implementation in control plane (which this draft discourages). Given that the 5G system, as specified in [TS23.501] (Appendix G.4), encourages the use of name-based routing in (5G) control plane for realizing the 5G-specific service-based architecture for control plane services (so-called network function service), it would be worthwhile to investigate whether the 4G control plane would benefit similarly from such use or whether specific 4G architectural constraints would prevent ICN from providing any notable benefit.

#### 6.2. Use of Results Beyond Research

With the experiments and their outcomes outlined in this draft, we believe that this technology is ready for a wider use and adoption, providing additional insights. Specifically, we expect to study the following:

1. Viability of ICN inclusion in the 3GPP protocol stack, i.e., investigate how realistic it would be to modify the stack, considering the scenarios explained in Section 5.4, and complete the user session without feature degradation?
2. Viability of utilizing solutions in greenfield deployments, i.e., deploying the ICN-based extensions and solutions proposed in this draft in greenfield 4G deployments in order to assess real-world benefits when doing so.

#### 7. Security and Privacy Considerations

This section will cover some security and privacy considerations in mobile and 4G network because of introduction of ICN.

### 7.1. Security Considerations

To ensure only authenticated mobile terminals are connected to the network, 4G mobile network implements various security mechanisms. From the perspective of using ICN in the user plane, it needs to take care of the following security aspects:

1. MT authentication and authorization
2. Radio or air interface security
3. Denial of service attacks on the mobile gateway, services either by the MT or by external entities in the Internet
4. Content poisoning either in transport or servers
5. Content cache pollution attacks
6. Secure naming, routing, and forwarding
7. Application security

Security over the LTE air interface is provided through cryptographic techniques. When MT is powered up, it performs a key exchange between MT's USIM and HSS/Authentication Center using NAS messages, including ciphering and integrity protections between MT and MME. Details for secure MT authentication, key exchange, ciphering, and integrity protections messages are given in the 3GPP call flow [TS23.401]. With ICN we are modifying protocol stack for user plane and not control plane. The NAS signaling is exchanged between MT and mobile gateways e.g. MME, using control plane, therefore there is no adverse impact of ICN on MT.

4G uses IP transport in its mobile backhaul (between eNodeB and core network). In case of provider-owned backhaul, service provider may require implementing a security mechanism in the backhaul network. The native IP transport continues to leverage security mechanism such as Internet key exchange (IKE) and the IP security protocol (IPsec). More details of mobile backhaul security are provided in 3GPP network security specifications [TS33.310] and [TS33.320]. When mobile backhaul is upgraded to support dual transport (IP+ICN) or native ICN, it is required to implement security techniques that are deployed in the mobile backhaul. When ICN forwarding is enabled on mobile transport routers, we need to deploy security practices based on [RFC7476] and [RFC7927].

4G mobile gateways (SGW, PGW) perform some of key functions such as content based online/offline billing and accounting, deep packet inspection (DPI), and lawful interception (LI). When ICN is deployed in user plane , we need to integrate ICN security for sessions between MT and gateway. If we encrypt user plane payload metadata then it might be difficult to perform routing based on contents and it may not work because we need decryption keys at every forwarder to route the content. The content itself can be encrypted between publisher and consumer to ensure privacy. Only the user with right decryption key shall be able to access the content. We need further research for ICN impact on LI, online/offline charging and accounting.

## 7.2. Privacy Considerations

In 4G networks, two main privacy issues are [MUTHANA]

1. User Identity Privacy Issues. The main privacy issue within the 4G is the exposure of the IMSI. The IMSI can be intercepted by adversaries. Such attacks are commonly referred to as "IMSI catching".
2. Location Privacy Issues. IMSI Catching is closely related to the issue of location privacy. Knowing IMSI of user allows the attacker to track the user's movements and create profile about the user and thus breaches the user's location privacy.

In any network, caching implies a trade-off between network efficiency and privacy. The activity of users is exposed to the scrutiny of cache owners with whom they may not have any relationship. By monitoring the cache transactions, an attacker could obtain significant information related to the objects accessed, topology and timing of the requests [RFC7945]. Privacy concerns are amplified by the introduction of new network functions such as Information lookup and Network storage, and different forms of communication [FOTIOU]. Privacy risks in ICN can be broadly divided in the following categories [TOURANI]:

1. Timing attack
2. Communication monitoring attack
3. Censorship and anonymity attack
4. Protocol attack
5. Naming-signature privacy

Introduction of TCL effectively enables ICN at the application and/or transport layer, depending on the scenario described in section 5. Enabling ICN in 4G networks is expected to increase efficiency by taking advantage of ICN's inherent characteristics. This approach would potentially leave some of the above-mentioned privacy concerns open as a consequence of using ICN transport and ICN inherent privacy vulnerabilities.

1. IPoIP Section 5.2 would not be affected as TCL has no role in it and ICN does not apply
2. ICNoICN scenario Section 5.2 has increased risk of a privacy attack, and that risk is applicable to ICN protocol in general rather than specifically to the 4G implementation. Since this scenario describes communication over ICN transport, every forwarder in the path could be a potential risk for privacy attack
3. ICNoIP scenario Section 5.2 uses IP for transport, so the only additional ICN-related potential privacy risk areas are the endpoints (consumer and publisher) where, at the application layer, content is being served
4. IPoICN scenario Section 5.2 could have potentially increased risk due to possible vulnerability of the forwarders in the path of ICN transport

Privacy issues already identified in 4G remain a concern if ICN is introduced in any of the scenarios described earlier and compound to the new, ICN-related privacy issues. Many research papers have been published proposing solutions to the privacy issues listed above. For LTE-specific privacy issues, some of the proposed solutions [MUTHANA] are IMSI encryption by a MT, mutual authentication, concealing the real IMSI within a random bit stream of certain size where only the subscriber and HSS could extract the respective IMSI, IMSI replacement with a changing pseudonym that only the HSS server can map it the UE's IMSI, and others. Similarly, some of the proposed ICN-specific privacy concerns mitigation methods, applicable where ICN transport is introduced as specified earlier in this section, include [FOTIOU]:

- \* Delay for the first, or first k interests on edge routers (timing attack)
- \* Creating a secure tunnel or clients flagging the requests as non-cacheable for privacy (communication monitoring attack)

- \* Encoding interest by mixing content and cover file or using hierarchical DNS-based brokering model (censorship and anonymity attack)
- \* Use of rate-limiting requests for a specific namespace (protocol attack)
- \* Cryptographic content hash-based naming or digital identity in an overlay network (naming-signature privacy)

Further research in this area is needed. Detailed discussion of privacy is beyond the scope of this document.

## 8. Summary

In this draft, we have discussed the 4G networks and the experimental setups to study the advantages of potential use of ICN for efficient delivery of contents to mobile terminals. We have discussed different options to try and test the ICN and dependencies such as ICN functionalities and changes required in different 4G network elements. In order to further explore potential use of ICN one can devise an experimental set-up consisting of 4G network elements and deploy ICN data transport in user plane. Different options can be either overlay, dual transport (IP + ICN), hICN, or natively (by integrating ICN with CDN, eNodeB, SGW, PGW and transport network). Note that, for the scenarios discussed above, additional study is required for lawful interception, billing/mediation, network slicing, and provisioning APIs.

Edge Computing [CHENG] provides capabilities to deploy functionalities such as Content Delivery Network (CDN) caching and mobile user plane functions (UPF) [TS23.501]. Recent research for delivering real-time video content [MPVCICN] using ICN has also been proven to be efficient [NDNRTC] and can be used towards realizing the benefits of using ICN in eNodeB, edge computing, mobile gateways (SGW, PGW) and CDN. The key aspect for ICN is in its seamless integration in 4G and 5G networks with tangible benefits so we can optimize content delivery using a simple and scalable architecture. The authors will continue to explore how ICN forwarding in edge computing could be used for efficient data delivery from the mobile edge.

Based on our study of control plane signaling, it is not beneficial to deploy ICN with existing protocols unless further changes are introduced in the control protocol stack itself.

As a starting step towards use of ICN in user plane, it is proposed to incorporate protocol changes in MT, eNodeB, SGW/PGW for data transport. ICN has inherent capabilities for mobility and content caching, which can improve the efficiency of data transport for unicast and multicast delivery. The authors welcome contributions and suggestions, including those related to further validations of the principles by implementing prototype and/or proof of concept in the lab and in the production environment.

## 9. Acknowledgements

We thank all contributors, reviewers, and the chairs for the valuable time in providing comments and feedback that helped improve this draft. We specially want to mention the following members of the IRTF Information-Centric Networking Research Group (ICNRG), listed in alphabetical order: Kashif Islam, Thomas Jagodits, Luca Muscariello, David R. Oran, Akbar Rahman, Martin J. Reed, Thomas C. Schmidt, and Randy Zhang.

The IRSG review was provided by Colin Perkins.

## 10. References

### 10.1. Normative References

- [TS24.008] 3GPP, "Mobile radio interface Layer 3 specification; Core network protocols; Stage 3", 3GPP TS 24.008 3.20.0, 15 December 2005, <<http://www.3gpp.org/ftp/Specs/html-info/24008.htm>>.
- [TS25.323] 3GPP, "Packet Data Convergence Protocol (PDCP) specification", 3GPP TS 25.323 3.10.0, 18 September 2002, <<http://www.3gpp.org/ftp/Specs/html-info/25323.htm>>.
- [TS29.274] 3GPP, "3GPP Evolved Packet System (EPS); Evolved General Packet Radio Service (GPRS) Tunneling Protocol for Control plane (GTPv2-C); Stage 3", 3GPP TS 29.274 10.11.0, 25 June 2013, <<http://www.3gpp.org/ftp/Specs/html-info/29274.htm>>.
- [TS29.281] 3GPP, "General Packet Radio System (GPRS) Tunneling Protocol User Plane (GTPv1-U)", 3GPP TS 29.281 10.3.0, 26 September 2011, <<http://www.3gpp.org/ftp/Specs/html-info/29281.htm>>.
- [TS36.323] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification", 3GPP TS 36.323 10.2.0, 3 January 2013, <<http://www.3gpp.org/ftp/Specs/html-info/36323.htm>>.



## 10.2. Informative References

- [ALM] Augé, J., Carofiglio, G., Grassi, G., Muscariello, L., Pau, G., and X. Zeng, "Anchor-Less Producer Mobility in ICN", Proceedings of the 2Nd ACM Conference on Information-Centric Networking, ACM-ICN'15, ACM DL, pp.189-190, 30 September 2013, <<https://dl.acm.org/citation.cfm?id=2812601>>.
- [BROWER] Brower, E., Jeffress, L., Pezeshki, J., Jasani, R., and E. Ertekin, "Integrating Header Compression with IPsec", MILCOM 2006 - 2006 IEEE Military Communications conference IEEE Xplore DL, pp.1-6, 23 October 2006, <<https://ieeexplore.ieee.org/document/4086687>>.
- [CCN] "Content Centric Networking", <<http://www.ccnx.org>>.
- [CHENG] Liang, C., Yu, R., and X. Zhang, "Information-centric network function virtualization over 5g mobile wireless networks", IEEE Network Journal vol. 29, number 3, pp. 68-74, 1 June 2015, <<https://ieeexplore.ieee.org/document/7113228>>.
- [EMBMS] Zahoor, K., Bilal, K., Erbad, A., and A. Mohamed, "Service-Less Video Multicast in 5G: Enablers and Challenges", IEEE Network vol. 34, no. 3, pp. 270-276, May 2020, <<https://ieeexplore.ieee.org/document/9105941>>.
- [EPCCUPS] Schmitt, P., Landais, B., and F. Yong Yang, "Control and User Plane Separation of EPC nodes (CUPS)", 3GPP The Mobile Broadband Standard, 3 July 2017, <<http://www.3gpp.org/news-events/3gpp-news/1882-cups>>.
- [FOTIOU] Fotiou, N. and G. Polyzos, "ICN privacy and name based security", ACM-ICN '14: Proceedings of the 1st ACM Conference on Information-Centric Networking ACM Digital Library, pp. 5-6, September 2014, <<https://dl.acm.org/doi/10.1145/2660129.2666711>>.
- [GALIS] Galis, A., Makhijani, K., Yu, D., and B. Liu, "Autonomic Slice Networking", Work in Progress, Internet-Draft, draft-galis-anima-autonomic-slice-networking-05, 26 September 2018, <<http://www.ietf.org/internet-drafts/draft-galis-anima-autonomic-slice-networking-05.txt>>.

- [GRAYSON] Grayson, M., Shatzkamer, M., and S. Wainner, "Cisco Press book "IP Design for Mobile Networks"", Cisco Press Networking Technology series, 15 June 2009, <<http://www.ciscopress.com/store/ip-design-for-mobile-networks-9781587058264>>.
- [HICN] Muscariello, L., Carofiglio, G., Auge, J., and M. Papalini, "Hybrid Information-Centric Networking", Work in Progress, Internet-Draft, draft-muscariello-intarea-hicn-04, 20 May 2020, <<https://www.ietf.org/id/draft-muscariello-intarea-hicn-04.txt>>.
- [I-D.anilj-icnrg-dnc-qos-icn] Jangam, A., suthar, P., and M. Stolic, "QoS Treatments in ICN using Disaggregated Name Components", Work in Progress, Internet-Draft, draft-anilj-icnrg-dnc-qos-icn-02, 9 March 2020, <<http://www.ietf.org/internet-drafts/draft-anilj-icnrg-dnc-qos-icn-02.txt>>.
- [ICN5G] Ravindran, R., suthar, P., Trossen, D., and G. White, "Enabling ICN in 3GPP's 5G NextGen Core Architecture", Work in Progress, Internet-Draft, draft-ravi-icnrg-5gc-icn-04, 10 January 2021, <<https://www.ietf.org/id/draft-irtf-icnrg-5gc-icn-04.txt>>.
- [ICNLOWPAN] Gundogan, C., Schmidt, T., Waehlich, M., Scherb, C., Marxer, C., and C. Tschudin, "ICN Adaptation to LowPAN Networks (ICN LoWPAN)", Work in Progress, Internet-Draft, draft-irtf-icnrg-icnlowpan-10, 10 February 2021, <<https://www.ietf.org/id/draft-irtf-icnrg-icnlowpan-10.txt>>.
- [ICNQoS] Al-Naday, M.F., Bontozoglou, A., Vassilakis, G., and M. J. Reed, "Quality of Service in an Information-Centric Network", 2014 IEEE Global Communications Conference IEEE Xplore DL, pp. 1861-1866, 8 December 2014, <<https://ieeexplore.ieee.org/document/7037079>>.
- [IPoICN] Trossen, D., Read, M J., Riihijarvi, J., Georgiades, M., Fotiou, N., and G. Xylomenos, "IP over ICN - The better IP?", 2015 European Conference on Networks and Communications (EuCNC) IEEE Xplore DL, pp. 413-417, 29 June 2015, <<https://ieeexplore.ieee.org/document/7194109>>.

- [MBICN] Carofiglio, G., Gallo, M., Muscariello, L., and D. Perino, "Scalable mobile backhauling via information-centric networking", The 21st IEEE International Workshop on Local and Metropolitan Area Networks, Beijing, pp. 1-6, 22 April 2015, <<https://ieeexplore.ieee.org/document/7114719>>.
- [MECSPEC] "Mobile Edge Computing (MEC); Framework and Reference Architecture", ETSI European Telecommunication Standards Institute (ETSI) MEC specification, March 2016, <[https://www.etsi.org/deliver/etsi\\_gs/MEC/001\\_099/003/01.01.01\\_60/gs\\_MEC003v010101p.pdf](https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf)>.
- [MPVCICN] Jangam, A., Ravindran, R., Chakraborti, A., Wan, X., and G. Wang, "Realtime multi-party video conferencing service over information centric network", IEEE International Conference on Multimedia and Expo Workshops (ICMEW) Turin, Italy, pp. 1-6, 29 June 2015, <<https://ieeexplore.ieee.org/document/7169810>>.
- [MUTHANA] Muthana, A. and M. Saeed, "Analysis of User Identity Privacy in LTE and Proposed Solution", International Journal of Computer Network and Information Security(IJCNIS) MECS Press, pp. 54-63, January 2017, <<http://www.mecs-press.org/ijcnis/ijcnis-v9-n1/v9n1-7.html>>.
- [NDNRTC] Gusev, P., Wang, Z., Burke, J., Zhang, L., Yoneda, T., Ohnishi, R., and E. Muramoto, "Real-time Streaming Data Delivery over Named Data Networking,", IEICE Transactions on Communications vol. E99.B, pp. 974-991, 1 May 2016, <<https://doi.org/10.1587/transcom.2015AMI0002>>.
- [NGMN] Robson, J., "Backhaul Provisioning for LTE-Advanced and Small Cells", Next Generation Mobile Networks, LTE-Advanced Transport Provisioning, V0.0.14, 20 October 2015, <[https://www.ngmn.org/wp-content/uploads/Publications/2015/150929\\_NGMN\\_P-SmallCells\\_Backhaul\\_for\\_LTE-Advanced\\_and\\_Small\\_Cells.pdf](https://www.ngmn.org/wp-content/uploads/Publications/2015/150929_NGMN_P-SmallCells_Backhaul_for_LTE-Advanced_and_Small_Cells.pdf)>.
- [NS3EPC] Baldo, N., "The ns-3 EPC module", NS3 EPC Model, <<https://www.nsnam.org/docs/models/html/lte-design.html#epc-model>>.
- [NS3LTE] Baldo, N., "The ns-3 LTE module", NS3 LTE Model, <<https://www.nsnam.org/docs/models/html/lte-design.html#lte-model>>.

- [OFFLOAD] Rebecchi, F., Dias de Amorim, M., Conan, V., Passarella, A., Bruno, R., and M. Conti, "Data Offloading Techniques in Cellular Networks: A Survey", IEEE Communications Surveys and Tutorials, IEEE Xplore DL, vol:17, issue:2, pp.580-603, 11 November 2014, <<https://ieeexplore.ieee.org/document/6953022>>.
- [OLTEANU] Olteanu, A. and P. Xiao, "Fragmentation and AES Encryption Overhead in Very High-speed Wireless LANs", Proceedings of the 2009 IEEE International Conference on Communications ICC'09, ACM DL, pp.575-579, 14 June 2009, <<http://dl.acm.org/citation.cfm?id=1817271.1817379>>.
- [Open5GCore] Open5GCore, M., "Open5GCore - Fundamental 4G Core Network Functionality", Open5GCore, <<https://www.open5gcore.org>>.
- [RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, DOI 10.17487/RFC4594, August 2006, <<https://www.rfc-editor.org/info/rfc4594>>.
- [RFC6459] Korhonen, J., Ed., Soininen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459, DOI 10.17487/RFC6459, January 2012, <<https://www.rfc-editor.org/info/rfc6459>>.
- [RFC7476] Pentikousis, K., Ed., Ohlman, B., Corujo, D., Boggia, G., Tyson, G., Davies, E., Molinaro, A., and S. Eum, "Information-Centric Networking: Baseline Scenarios", RFC 7476, DOI 10.17487/RFC7476, March 2015, <<https://www.rfc-editor.org/info/rfc7476>>.
- [RFC7927] Kutscher, D., Ed., Eum, S., Pentikousis, K., Psaras, I., Corujo, D., Saucez, D., Schmidt, T., and M. Waehlis, "Information-Centric Networking (ICN) Research Challenges", RFC 7927, DOI 10.17487/RFC7927, July 2016, <<https://www.rfc-editor.org/info/rfc7927>>.
- [RFC7945] Pentikousis, K., Ed., Ohlman, B., Davies, E., Spirou, S., and G. Boggia, "Information-Centric Networking: Evaluation and Security Considerations", RFC 7945, DOI 10.17487/RFC7945, September 2016, <<https://www.rfc-editor.org/info/rfc7945>>.

- [RFC8569] Mosko, M., Solis, I., and C. Wood, "Content-Centric Networking (CCNx) Semantics", RFC 8569, DOI 10.17487/RFC8569, July 2019, <<https://www.rfc-editor.org/info/rfc8569>>.
- [RFC8609] Mosko, M., Solis, I., and C. Wood, "Content-Centric Networking (CCNx) Messages in TLV Format", RFC 8609, DOI 10.17487/RFC8609, July 2019, <<https://www.rfc-editor.org/info/rfc8609>>.
- [RFC9064] Oran, D., "Considerations in the Development of a QoS Architecture for CCNx-Like Information-Centric Networking Protocols", RFC 9064, DOI 10.17487/RFC9064, June 2021, <<https://www.rfc-editor.org/info/rfc9064>>.
- [SDN5G] Page, J. and J. Dricot, "Software-defined networking for low-latency 5G core network", 2016 International Conference on Military Communications and Information Systems (ICMCIS) IEEE Xplore DL, pp. 1-7, May 2016, <<https://ieeexplore.ieee.org/document/7496561>>.
- [TLVCOMP] Mosko, M., "Header Compression for TLV-based Packets", ICNMG Buenos Aires IETF 95, 3 April 2016, <<https://datatracker.ietf.org/meeting/interim-2016-icnrg-02/materials/slides-interim-2016-icnrg-2-7>>.
- [TOURANI] Tourani, R., Misra, S., Mick, T., and G. Panwar, "Security, Privacy, and Access Control in Information-Centric Networking: A Survey", IEEE Communications Surveys and Tutorials Volume 20, Issue 1, pp 566-600, September 2017, <<https://ieeexplore.ieee.org/document/8027034>>.
- [TS23.203] 3GPP, "Policy and charging control architecture", 3GPP TS 23.203 10.9.0, 12 September 2013, <<http://www.3gpp.org/ftp/Specs/html-info/23203.htm>>.
- [TS23.401] 3GPP, "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access", 3GPP TS 23.401 10.10.0, 7 March 2013, <<http://www.3gpp.org/ftp/Specs/html-info/23401.htm>>.
- [TS23.501] 3GPP, "System Architecture for the 5G System", 3GPP TS 23.501 15.2.0, 15 June 2018, <<http://www.3gpp.org/ftp/Specs/html-info/23501.htm>>.

- [TS23.714] 3GPP, "Technical Specification Group Services and System Aspects: Study on control and user plane separation of EPC nodes", 3GPP TS 23.714 0.2.2, 4 June 2016,  
<<http://www.3gpp.org/ftp/Specs/html-info/23714.htm>>.
- [TS29.060] 3GPP, "General Packet Radio Service (GPRS); GPRS Tunneling Protocol (GTP) across the Gn and Gp interface", 3GPP TS 29.060 3.19.0, 24 March 2004,  
<<http://www.3gpp.org/ftp/Specs/html-info/29060.htm>>.
- [TS33.310] 3GPP, "Network Domain Security (NDS); Authentication Framework (AF)", 3GPP TS 33.310 10.7.0, 21 December 2012,  
<<http://www.3gpp.org/ftp/Specs/html-info/33310.htm>>.
- [TS33.320] 3GPP, "Security of Home Node B (HNB) / Home evolved Node B (HeNB)", 3GPP TS 33.320 10.5.0, 29 June 2012,  
<<http://www.3gpp.org/ftp/Specs/html-info/33320.htm>>.

## Authors' Addresses

Prakash Suthar  
Google Inc.  
Mountain View, California 94043  
United States of America  
Email: [psuthar@google.com](mailto:psuthar@google.com)

Milan Stolic  
Cisco Systems Inc.  
Naperville, Illinois 60540  
United States of America  
Email: [mistolic@cisco.com](mailto:mistolic@cisco.com)

Anil Jangam (editor)  
Cisco Systems Inc.  
San Jose, California 95134  
United States of America  
Email: [anjangam@cisco.com](mailto:anjangam@cisco.com)

Dirk Trossen  
Huawei Technologies  
Riesstrasse 25  
80992 Munich  
Germany  
Email: [dirk.trossen@huawei.com](mailto:dirk.trossen@huawei.com)

Ravi Ravindran  
F5 Networks  
3545 North First Street  
San Jose, 95134  
United States of America  
Email: r.ravindran@f5.com

ICN Research Group  
Internet-Draft  
Intended status: Informational  
Expires: January 3, 2019

J. Hong  
ETRI  
L. Dong  
Huawei  
T. You  
ETRI  
C. Westphal  
Huawei  
Y-G. Hong  
ETRI  
GQ. Wang  
Huawei  
J. Wang  
City University Hong Kong  
July 2, 2018

Requirements for Name Resolution Service in ICN  
draft-jhong-icnrg-nrs-requirements-04

Abstract

This document discusses the motivation and requirements for Name Resolution Service (NRS) in ICN. The NRS in ICN is to translate an object name into some other information such as locator and another name which is used for forwarding the object request.

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 <https://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 January 3, 2019.



## Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Conventions and Terminology . . . . .	4
3. Name Resolution Service in ICN . . . . .	4
3.1. Standalone name resolution approach . . . . .	4
3.2. Name based routing approach . . . . .	4
3.3. Hybrid approach . . . . .	5
3.4. Comparisons of name resolution approaches . . . . .	5
4. Motivation of NRS in ICN . . . . .	6
4.1. Heterogeneous names in ICN . . . . .	6
4.2. Dynamism in ICN . . . . .	7
4.3. Routing system in ICN . . . . .	8
4.4. Use cases of NRS . . . . .	8
4.4.1. Flat name based routing support . . . . .	8
4.4.2. Producer mobility support . . . . .	9
4.4.3. Scalable routing support . . . . .	9
4.4.4. Off-Path cache support . . . . .	10
4.4.5. Nameless object support . . . . .	10
4.4.6. Manifest support . . . . .	10
5. Requirements for NRS in ICN . . . . .	11
5.1. Requirements as a service . . . . .	11
5.1.1. Delay sensitivity . . . . .	11
5.1.2. Accuracy . . . . .	11
5.1.3. Resolution guarantee . . . . .	11
5.2. Requirements as a system . . . . .	11
5.2.1. Scalability . . . . .	12
5.2.2. Manageability . . . . .	12
5.2.3. Deployability . . . . .	12
5.2.4. Fault tolerance . . . . .	12
5.3. Requirements on Security aspect . . . . .	12
5.3.1. Accessibility . . . . .	12
5.3.2. Authentication . . . . .	12

5.3.3. Data confidentiality . . . . .	13
5.3.4. Data privacy . . . . .	13
6. IANA Considerations . . . . .	13
7. Security Considerations . . . . .	13
8. Acknowledgements . . . . .	13
9. References . . . . .	13
9.1. Normative References . . . . .	13
9.2. Informative References . . . . .	13
Authors' Addresses . . . . .	17

## 1. Introduction

The current Internet is a host-centric networking, where hosts are uniquely identified with IP addresses and communication is possible between any pair of hosts. Thus, information in the current Internet is identified by the name of host where the information is stored. In contrast to the host-centric networking, the primary communication objects in Information-centric networking (ICN) are the named data objects (NDOs) and they are uniquely identified by the location-independent names. Thus, ICN aiming to the efficient dissemination and retrieval of the NDOs in a global scale has been identified and acknowledged as a promising technology for the future Internet architecture to overcome the limitations of the current Internet such as scalability, mobility, etc.[Ahlgren] [Xylomenos]. ICN also has been emerged as a candidate architecture for IoT environment since IoT focuses on data and information rather than end-to-end communications [Baccelli] [Amadeo] [Quevedo] [Amadeo2] [ID.Zhang2].

Since naming data independently from the current location where it is stored is a primary concept of ICN, how to find the NDO using the location-independent name is one of the most important design challenges in ICN. Such ICN routing may comprise three steps [RFC7927] :

- o Name resolution : matches/translate a content name to locators of providers/sources that can provide the content.
- o Content discovery : routes the content request towards the content either based on its name or locator.
- o Content delivery : transfers the content to the requester.

In three steps of ICN routing, this document focuses only the name resolution step which translates a content name to its locators. In addition, this document considers all other types of name resolution in ICN such as name to name, name to manifest.

Thus, this document presents the definition of the Name Resolution Service (NRS) in ICN and discusses the motivation and the requirements in designing the NRS for ICN.

## 2. Conventions and 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 [RFC2119].

## 3. Name Resolution Service in ICN

The Name Resolution Service (NRS) in ICN is defined as the service that provides the name resolution function translating an object name into some other information such as locator and another name that is used for forwarding the object request. In other words, the NRS is the service that shall be provided by ICN infrastructure to help a consumer to reach a specific piece of content, service, or host using a persistent name when the name resolution is needed.

The name resolution is a necessary process in ICN routing although the name resolution either can be separated from the content discovery as a standalone process or can be integrated with the content discovery as one combined process. The former is referred as standalone name resolution approach, the latter is referred as name based routing approach in this document.

### 3.1. Standalone name resolution approach

The NRS could take the standalone name resolution approach to return the client with the locators of the content, which will be used by the underlying network as the identifier to route the client's request to one of the producers. There are several ICN projects that use the standalone name resolution approach such as DONA[Koponen], PURSUIT [PURSUIT], SAIL [SAIL], MobilityFirst [MF], IDNet [Jung], etc.

### 3.2. Name based routing approach

The NRS could take the name based routing approach, which integrates the name resolution with the content request message routing as in NDN [NDN]/CCN [CCN].

In the case that the content request also specifies the reverse path, as in NDN/CCN, the name resolution mechanism also determines the routing path for the data. This adds a requirement on the name resolution service to propagate request in a way that is consistent with the subsequent data forwarding. Namely, the request must select

a path for the data based upon the finding the copy of the content, but also properly delivering the data.

### 3.3. Hybrid approach

The NRS could also take hybrid approach which can perform name based routing approach from the beginning, when it fails at certain router, the router can go back to the standalone name resolution approach. The alternative hybrid NRS approach also works, which can perform standalone name resolution approach to find locators of routers which can carry out the name based routing of the client's request.

A hybrid approach would combine name resolution as a subset of routers on the path with some tunneling in between (say, across an administrative domain) so that only a few of the nodes in the architecture perform name resolution in the name-based routing approach.

### 3.4. Comparisons of name resolution approaches

The following compares the standalone name resolution and name based routing approaches from different aspects:

- o Update message overhead : The update message overhead is due to the change of content reachability, which may include content caching or expiration, content producer mobility etc. The name based routing approach may require to flood part of the network for update propagation. In the worst case, the name based routing approach may flood the whole network (but mitigating techniques may be used to scope the flooding). The standalone name resolution approach only requires to update propagation in part of the name resolution overlay.
- o Resolution capability : The standalone name resolution approach can guarantee the resolution of any content in the network if it is registered to the name resolution overlay (assuming the content is being broadcast in the overlay after it is registered). On the other hand, the name based routing approach can only promise a high probability of content resolution, depending on the flooding scope of the content availability information (i.e. content publishing message and name based routing table).
- o Node failure impact : Nodes involved in the standalone name resolution approach are the name resolution overlay servers (e.g. Resolution Handlers in DONA), while the nodes involved in the name based routing approach are routers which route messages based on locally maintained name based routing tables (e.g. NDN routers). Node failures in the standalone name resolution approach may cause

some content resolution to fail even though the content is available. This problem does not exist in the name based routing approach because other alternative paths can be discovered to bypass the failed ICN routers, given the assumption that the network is still connected.

- o Maintained databases : The storage usage for the standalone name resolution approach is different from that of the name based routing approach. The standalone name resolution approach typically needs to maintain two databases: name to locator mapping in the name resolution overlay and routing tables in the routers on the data forwarding plane. The name based routing approach needs to maintain different databases: name routing table and optionally breadcrumbs for reverse routing of content back to the requester.

#### 4. Motivation of NRS in ICN

This section presents the motivation and use cases of NRS in ICN.

##### 4.1. Heterogeneous names in ICN

In ICN design, a name is used to identify an entity, such as named data content, a device, an application, a service. ICN requires uniqueness and persistency of the name of any entity to ensure the reachability of the entity within certain scope and with proper authentication and trust guarantees. The name does not change with the mobility and multi-home of the corresponding entity. A client can always use this name to retrieve the content from network and verify the binding of the content and the name.

Ideally, a name can include any form of identifier, which can be flat, hierarchical, and human readable or non-readable.

There are heterogeneous content naming schemes [ID.Zhang] [RFC1498] and name resolution approaches from different ICN architectures. For example:

- o Names in DONA [Koponen] consist of the cryptographic hash of the principal's public key P and a label L uniquely identifying the information with respect to the principal. Name resolution in DONA is provided by specialized servers called Resolution Handlers (RHs).
- o Content in PURSUIT [PURSUIT] is identified by a combination of a scope ID and a rendezvous ID. The scope ID represents the boundaries of a defined dissemination strategy for the content it contain. The rendezvous ID is the actual identity for a

particular content. Name resolution in PURSUIT is handled by a collection of Rendezvous Nodes (RNs), which are implemented as a hierarchical Dynamic Hash Table (DHT)[Rajahalme] [Katsaros].

- o Names in NDN [NDN] and CCN [CCN] are hierarchical and may be similar to URLs. Each name component can be anything, including a dotted human-readable string or a hash value. NDN/CCN adopts the name based routing. The NDN router forwards the request by doing the longest-match lookup in the Forwarding Information Base (FIB) based on the content name and the request is stored in the Pending Interest Table (PIT).
- o In MobilityFirst [MF], every network entity, content has a Global Unique Identifier (GUID). GUIDs are flat 160-bit strings with no semantic structure. Name Resolution in MobilityFirst is carried out via a Global Name Resolution Service (GNRS).

Although the existing naming schemes are different, they all need to provide basic functions for identifying a content, supporting trust provenance, content lookup and routing. The NRS may combine the advantages of different mechanisms. The NRS may be able to provide a generic naming schema to resolve any type of content name, either it is flat or hierarchical.

#### 4.2. Dynamism in ICN

In ICN literature, it is said that mobility can be achieved in fundamental feature of ICN. Especially, consumer or client mobility can be achieved by allowing information requests to basic procedure from different interfaces or through attachment point of the new network. Moreover, seamless mobility service in ICN ensures that content reception continues without any disruption in ICN application, so in consumer point of view, seamless mobility can be easily supported.

However, producer or publisher mobility in ICN is more complicated to be supported. If a publisher moves into different authority domain or network location, then the request for a content published by the moving publisher with origin name would be hardly forwarded to the moving publisher. Especially in a hierarchical name scheme, publisher mobility support is much harder than in a flat name scheme since the routing tables related in broad area should be updated according to the publisher movement. Therefore, various ICN literatures would adopt NRS to achieve the publisher mobility, where NRS can be implemented in different ways such as rendezvous mechanism, mapping, etc.

Besides mobility, ICN has challenge to support the dynamism features like multi-homing, migration, and replication of named resources such as content, devices, services, etc. and NRS may help to support the dynamism features.

#### 4.3. Routing system in ICN

In ICN, data objects must be identified by names regardless their location or container [RFC7927] and the names are divided into two types of schemes: hierarchical and flat namespaces. A hierarchical scheme used in CCN and NDN architectures has a structure similar to current URIs, where the hierarchy improves scalability of routing system. It is because the hierarchy enables aggregation of the name resulting in reducing the size of RIB or FIB as similar to IP routing system. In a flat scheme, on the other hand, name routing is not easy since names in a flat namespace cannot be aggregated anymore, which would cause more the scalability problem in routing system. In order to address such problem, a flat name can be resolved to some information which is routable through NRS.

In ICN, application names identifying contents are used directly for packet delivery, so ICN routers run a name-based routing protocol to build name-based routing and forwarding tables. Regardless of name scheme, if non-aggregated name prefixes are injected to the Default Route Free Zone (DFZ) of ICN, then they would be driving the growth of the DFZ routing table size, which is the same as the scalability issue of IP routing. Thus a solution to keep the routing table size under control is needed, which can be done by defining indirection layer.

#### 4.4. Use cases of NRS

This subsection describes NRS used in many other ways in ICN literature.

##### 4.4.1. Flat name based routing support

In PURSUIT [PURSUIT], names are flat and the rendezvous functions are defined for NRS, which is implemented by a set of Rendezvous Nodes (RNs), the Rendezvous Network (RENE). Thus a name consisted of a sequence of scope IDs and a single rendezvous ID is routed by RNs in RENE. Thus, PURSUIT decouples name resolution and data routing, where NRS is performed by the RENE.

In MobilityFirst [MF], a name called a global unique Identifier (GUID) derived from a human-readable name via a global naming service is flat typed 160-bits strings with self-certifying function. Thus, MobilityFirst defines a global name resolution service (GNRS) which

resolves GUIDs to network addresses and decouples name resolution and data routing as similar to PURSUIT.

#### 4.4.2. Producer mobility support

In NDN [Zhang2], for producer mobility support, rendezvous mechanisms have been proposed to build interests rendezvous (RV) with data generated by a mobile producer (MP). There can be classified two approaches such as chase mobile producer and rendezvous data. Regarding MP chasing, rendezvous acts as a mapping service that provides the mapping from the name of the data produced by the MP to the MP's current point of attachment (PoA) name. Alternatively, the RV serves as a home agent like as IP mobility support, so the RV enables consumer's interest message to tunnel towards the MP at the PoA. Regarding rendezvous data, moving the data produced by the MP have been hosting at data depot instead of forwarding interest messages. Thus a consumer's interest message can be forwarded to stationary place as called data rendezvous, so it would either return the data or fetch it using another mapping solution. Therefore, RV or other mapping functions are in the role of NRS in NDN.

In [Ravindran], forwarding-label (FL) object is referred to enable identifier (ID) and locator (LID) namespaces to be split in ICN. Generally, IDs are managed by applications, while locators are managed by a network administrator, so that IDs are mapping to heterogeneous name schemes and LIDs are mapping to network domains or specific network elements. Thus the proposed FL object acts as a locator (LID) and provides the flexibility to forward Interest messages through mapping service between IDs and LIDs. Therefore, the mapping service in control plane infrastructure can be considered as NRS in this draft.

In MobilityFirst [MF], both consumer and publisher mobility can be primarily handled by the global name resolution service (GNRS) which resolves GUIDs to network addresses. Thus, the GNRS must be updated for mobility support when a network attached object changes its point of attachment, which differs from NDN/CCN.

#### 4.4.3. Scalable routing support

In [Afanasyev], in order to address the routing scalability problem in NDN's DFZ, a well-known concept of Map-and-Encap is applied to provide a simple and secure namespace mapping solution. In the proposed map-and-encap design, data whose name prefixes do not exist in the DFZ forwarding table can be retrieved by a distributed mapping system called NDNS, which maintains and lookups the mapping information from a name to its globally routed prefixes, where NDNS is a kind of NRS.



#### 4.4.4. Off-Path cache support

Caching in-network is considered to be a basic architectural component of an ICN architecture. It may be used to provide a Quality-of-Service (QoS) experience to users, reduce the overall network traffic, prevent network congestion and Denial-of-Service (DoS) attacks and increase availability. Caching approaches can be categorized into off-path caching and on-path caching based on the location of caches in relation to the forwarding path from a original server to a consumer. Off-path caching, also referred as content replication or content storing, aims to replicate content within a network in order to increase availability, regardless of the relationship of the location to the forwarding path. Thus, finding off-path cached objects is not trivial in name based routing of ICN. In order to support off-path caches, replicas are usually advertised into a name- based routing system or into NRS.

In [Bayhan], a NRS used to find off-path copies in the network, which may not be accessible via content discovery mechanisms. Such capability is essential for an Autonomous System (AS) to avoid the costly inter-AS traffic for external content, to yield higher bandwidth efficiency for intra-AS traffic, and to decrease the data access latency for a pleasant user experience.

#### 4.4.5. Nameless object support

In CCNx 1.0 [Mosko2], the concept of "Nameless Objects" that are a Content Object without a Name is introduced to provide a means to move Content between storage replicas without having to rename or re-sign the content objects for the new name. Nameless Objects can be addressed by the ContentObjectHash that is to restrict Content Object matching by using SHA-256 hash.

An Interest message would still carry a Name and a ContentObjectHash, where a Name is used for routing, while a ContentObjectHash is used for matching. However, on the reverse path, if the Content Object's name is missing, it is a "Nameless Object" and only matches against the ContentObjectHash. Therefore, a consumer needs to resolve proper name and hashes through an outside system, which can be considered as NRS.

#### 4.4.6. Manifest support

In collection of data objects which were organized as large and file like contents [FLIC], the manifests are used as data structures to transport this information. Thus, the manifests may contain hash digests of signed content objects or other manifests, so that large

content objects which represent large piece of application data can be collected by using the manifest.

In order to request content objects, a consumer needs to know a manifest root name to acquire the manifest. In case of FLIC, a manifest name can be represented by a nameless root manifest, so that outside system may be involved to give this information to the consumer. Therefore, NRS can be considered as a kind of mapping database system.

## 5. Requirements for NRS in ICN

This section presents the requirements for designing NRS in ICN in terms of service, system and security aspects, respectively.

### 5.1. Requirements as a service

This subsection presents the requirements for NRS as a service.

#### 5.1.1. Delay sensitivity

The name resolution process provided by the NRS must be completed within a minimum delay. If the name resolution takes too long, then the content request packet may get dropped or it will yield the high content retrieval time for content requestor. Thus, the content retrieval time has to be content requestor-tolerant.

#### 5.1.2. Accuracy

The NRS must provide accurate and up-to-date information on how to discover the requested content with minimum overhead in propagating the update information. For example, a content can be moved from one domain to another domain due to the mobility of the producer, then the old name record should be deleted from the NRS system and a new name record should be added and updated with minimum delay.

#### 5.1.3. Resolution guarantee

The NRS must ensure the name resolution success if the matching content exists in the network, regardless of its popularity, number of cached copies.

### 5.2. Requirements as a system

This subsection presents the requirements for NRS as a system.

#### 5.2.1. Scalability

The NRS system must be extremely scalable to support a large number of content objects as well as billions of users, who may access the system through various connection methods and devices. Especially in IoT applications, the data size is small but frequently generated by sensors. Message forwarding and processing, routing table building-up and name records propagation must be efficient and scalable.

#### 5.2.2. Manageability

The NRS system must be manageable since some parts of the system may grow or shrink dynamically and a NRS system node may be added or deleted.

#### 5.2.3. Deployability

The NRS system must be deployable since deployability is important for a real world system. If the NRS system can be deployed from the edges, then the deployment can be simplified.

#### 5.2.4. Fault tolerance

The NRS system must ensure resilience to node failures. After a NRS node fails, the NRS system must be able to restore the name records stored in the NRS node.

### 5.3. Requirements on Security aspect

This subsection presents the requirements for NRS on security aspect for both node and data in the NRS system.

#### 5.3.1. Accessibility

The name records must have proper access rights such that the information contained in the name record would not be revealed to unauthorized users. In other words, The NRS system must be prevented from the malicious users attempting to hijack or corrupt name records.

#### 5.3.2. Authentication

Users/nodes that register themselves in the NRS system must require the authentication to ensure who claims to be. For example, the attacker can act as a fake NRS server which causes disruption or intercepts the data.

### 5.3.3. Data confidentiality

NRS must keep the data confidentiality to prevent a lot of sensitive data from reaching unauthorized data requestor such as in IoT environment.

### 5.3.4. Data privacy

When a private data is registered in the system, the NRS system must support the privacy to avoid the information leaking. Otherwise, unauthorized entity may disclose the privacy.

## 6. IANA Considerations

There are no IANA considerations related to this document.

## 7. Security Considerations

[TBD]

## 8. Acknowledgements

[TBD]

## 9. References

### 9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC7927] Kutscher, D., Ed., Eum, S., Pentikousis, K., Psaras, I., Corujo, D., Saucez, D., Schmidt, T., and M. Waehlich, "Information-Centric Networking (ICN) Research Challenges", RFC 7927, DOI 10.17487/RFC7927, July 2016, <<https://www.rfc-editor.org/info/rfc7927>>.

### 9.2. Informative References

- [Ahlgren] Ahlgren, B., Dannewitz, C., Imbrenda, C., Kutscher, D., and B. Ohlman, "A Survey of Information-Centric Networking", IEEE Communications Magazine Vol.50, Issue 7, 2012.

- [Xylomenos] Xylomenos, G., Ververidis, C., Siris, V., Fotiou, N., Tsilopoulos, C., Vasilako, X., Katsaros, K., and G. Polyzos, "A Survey of Information-Centric Networking Research, Communications Surveys and Tutorials", IEEE Communications Surveys and Tutorials vol. 16, no. 2, 2014.
- [Baccelli] Baccelli, E., Mehlis, C., Hahm, O., Schmidt, T., and M. Wahlisch, "Information Centric Networking in the IoT: Experiments with NDN in the Wild", ACM ICN 2014, 2014.
- [Amadeo] Amadeo, M., Campolo, C., Iera, A., and A. Molinaro, "Named data networking for IoT: An architectural perspective", European Conference on Networks and Communications (EuCNC) , 2014.
- [Quevedo] Quevedo, J., Corujo, D., and R. Aguiar, "A case for ICN usage in IoT environments", IEEE GLOBECOM , 2014.
- [Amadeo2] Amadeo, M. et al., "Information-centric networking for the internet of things: challenges and opportunities", IEEE Network vol. 30, no. 2, July 2016.
- [ID.Zhang2] Zhang, Y., "Design Considerations for Applying ICN to IoT", draft-zhang-icnrg-icniot-01 , June 2017.
- [Koponen] Koponen, T., Chawla, M., Chun, B., Ermolinskiy, A., Kim, K., Shenker, S., and I. Stoica, "A Data-Oriented (and Beyond) Network Architecture", ACM SIGCOMM 2007 pp. 181-192, 2007.
- [PURSUIT] "FP7 PURSUIT project.", <http://www.fp7-pursuit.eu/PursuitWeb/> .
- [SAIL] "FP7 SAIL project.", <http://www.sail-project.eu/> .
- [NDN] "NSF Named Data Networking project.", <http://www.named-data.net> .
- [CCN] "Content Centric Networking project.", <https://wiki.fd.io/view/Cicn> .
- [MF] "NSF Mobility First project.", <http://mobilityfirst.winlab.rutgers.edu/> .

- [Jung] Jung, H. et al., "IDNet: Beyond All-IP Network", ETRI Journal vol. 37, no. 5, October 2015.
- [Jacobson] Jacobson, V., Smetters, D., Thornton, J., Plass, M., Briggs, N., and R. Braynard, "Networking Named Content", ACM CoNEXT , 2009.
- [Baid] Baid, A., Vu, T., and D. Raychaudhuri, "Comparing Alternative Approaches for Networking of Named Objects in the Future Internet", IEEE Workshop on Emerging Design Choices in Name-Oriented Networking (NOMEN) , 2012.
- [Bari] Bari, M., Chowdhury, S., Ahmed, R., Boutaba, R., and B. Mathieu, "A Survey of Naming and Routing in Information-Centric Networks", IEEE Communications Magazine Vol. 50, No. 12, P.44-53, 2012.
- [Rajahalme] Rajahalme, J., Sarela, M., Visala, K., and J. Riihijarvi, "On Name-based Inter-domain Routing", Computer Networks Vol. 55, No. 4, P. 975-986, March 2011.
- [Katsaros] Katsaros, K., Fotiou, N., Vasilakos, X., Ververidis, C., Tsilopoulos, C., Xylomenos, G., and G. Polyzos, "On Inter-Domain Name Resolution for Information-Centric Networks", Proc.IFIP-TC6 Networking Conference , 2012.
- [ID.Wang] Wang, J., Li, S., and C. Wetphal, "Namespace Resolution in Future Internet Architectures", draft-wang-fia-namespace-01 , October 2015.
- [ID.Zhang] Zhang, X., Ravindran, R., Xie, H., and G. Wang, "PID: A Generic Naming Schema for Information-centric Network", draft-zhang-icnrg-pid-naming-scheme-03 , August 2013.
- [D.Zhang] Zhang, D. and H. Liu, "Routing and Name Resolution in Information-Centric Networks", 22nd International Conference on Computer Communications and Networks (ICCCN) , 2013.
- [Sevilla] Sevilla, S., Mahadevan, P., and J. Garcia-Luna-Aceves, "iDNS: Enabling Information Centric Networking Through The DNS", Name Oriented Mobility (workshop co-located with Infocom 2014) , 2014.

- [RFC1498] Saltzer, J., "On the Naming and Binding of Network Destinations", RFC 1498, DOI 10.17487/RFC1498, August 1993, <<https://www.rfc-editor.org/info/rfc1498>>.
- [oneM2M] "oneM2M Functional Architecture TS 0001.", <http://www.onem2m.org/technical/published-documents>. .
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <<https://www.rfc-editor.org/info/rfc7252>>.
- [ID.Shelby] Shelby, Z., "CoRE Resource Directory", draft-ietf-core-resource-directory-10 , March 2017.
- [CoRE] "Constrained RESTful Environments, CoRE", <https://datatracker.ietf.org/wg/core/charter/> , March 2013.
- [Westphal] Westphal, C. and E. Demirors, "An IP-based Manifest Architecture for ICN", ACM ICN , 2015.
- [Mosko] Mosko, M., Scott, G., Solis, I., and C. Wood, "CCNx Manifest Specification", draft-wood-icnrg-ccnxmanifests-00 , July 2015.
- [RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <<https://www.rfc-editor.org/info/rfc6830>>.
- [RFC6833] Fuller, V. and D. Farinacci, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, DOI 10.17487/RFC6833, January 2013, <<https://www.rfc-editor.org/info/rfc6833>>.
- [Zhang] Zhang, L. et al., "Named data networking", ACM SIGCOMM Computer Communication Review vol. 44, no. 3, July 2014.
- [Zhang2] Zhang, Y., "A Survey of Mobility Support in Named Data Networking", NAMED-ORIENTED MOBILITY: ARCHITECTURES, ALGORITHMS, AND APPLICATIONS(NOM) , 2016.

- [Dannewitz]  
Dannewitz, C. et al., "Network of Information (NetInf)-An information centric networking architecture", Computer Communications vol. 36, no. 7, April 2013.
- [Seskar] Seskar, I., Nagaraja, K., Nelson, S., and D. Raychaudhuri, "MobilityFirst Future Internet Architecture Project", 7th Asian Internet Engineering Conference , November 2011.
- [Dannewitz2]  
Dannewitz, C., D'Ambrosio, M., and V. Vercellone, "Hierarchical DHT-based name resolution for Information-Centric Networks", Computer Communications vol. 36, no. 7, April 2013.
- [Vu] Vu, T. et al., "DMap: A Shared Hosting Scheme for Dynamic Identifier to Locator Mapping in the Global Internet", IEEE 32nd International Conference on Distributed Computing Systems , 2012.
- [Hong] Hong, J., Chun, W., and H. Jung, "Demonstrating a Scalable Name Resolution System for Information-Centric Networking", ACM ICN , September 2015.
- [Ravindran]  
Ravindran, R. et al., "Forwarding-Label support in CCN Protocol", draft-ravi-icnrg-ccn-forwarding-label-01 , July 2017.
- [Afanasyev]  
Afanasyev, A. et al., "SNAMP: Secure Namespace Mapping to Scale NDN Forwarding", IEEE Global Internet Symposium , April 2015.
- [Mosko2] Mosko, M., "Nameless Objects", , July 2015.
- [Bayhan] Bayhan, S. et al., "On Content Indexing for Off-Path Caching in Information-Centric Networks", ACM ICN , September 2016.
- [FLIC] Tschudin, C. and C. Wood, "File-Like ICN Collection (FLIC)", draft-irtf-icnrg-flic-01, , June 2018.

Authors' Addresses



Jungha Hong  
ETRI  
218 Gajeong-ro, Yuseung-Gu  
Daejeon 34129  
Korea

Phone: +82 42 860 0926  
Email: jhong@etri.re.kr

Lijun Dong  
Huawei  
10180 Telesis Court  
San Diego, CA 92121  
USA

Email: lijun.dong@huawei.com

Tae-Wan You  
ETRI  
218 Gajeong-ro, Yuseung-Gu  
Daejeon 34129  
Korea

Phone: +82 42 860 0642  
Email: twyou@etri.re.kr

Cedric Westphal  
Huawei  
2330 Central Expressway  
Santa Clara, CA 95050  
USA

Email: cedric.westphal@huawei.com

Yong-Geun Hong  
ETRI  
218 Gajeong-ro, Yuseung-Gu  
Daejeon 34129  
Korea

Phone: +82 42 860 6557  
Email: yghong@etri.re.kr

GQ Wang  
Huawei  
2330 Central Expressway  
Santa Clara, CA 95050  
USA

Email: [gq.wang@huawei.com](mailto:gq.wang@huawei.com)

Jianping Wang  
City University Hong Kong

Email: [jianwang@cityu.edu.hk](mailto:jianwang@cityu.edu.hk)

ICNRG  
Internet-Draft  
Intended status: Informational  
Expires: December 2, 2019

R. Ravindran  
Futurewei  
P. Suthar  
Cisco  
D. Trossen  
C. Wang  
InterDigital Inc.  
G. White  
CableLabs  
May 31, 2019

Enabling ICN in 3GPP's 5G NextGen Core Architecture  
draft-ravi-icnrg-5gc-icn-04

Abstract

The proposed 3GPP's 5G core nextgen architecture (5GC) offers flexibility to introduce new user and control plane function, considering the support for network slicing functions, that allows greater flexibility to handle heterogeneous devices and applications. In this draft, we provide a short description of the proposed 5GC architecture, including recent efforts to provide cellular Local Area Network (LAN) connectivity, followed by extensions to 5GC's control and user plane to support Packet Data Unit (PDU) sessions from Information-Centric Networks (ICN). In addition, ICN over 5GLAN is also described.

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 <https://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 December 2, 2019.

## Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	4
3. 5G NextGen Core Design Principles . . . . .	5
4. 5GC Architecture with ICN Support . . . . .	6
4.1. 5G NextGen Core Architecture . . . . .	6
4.2. ICN over 5GC . . . . .	8
4.2.1. Control Plane Extensions . . . . .	10
4.2.2. User Plane Extensions . . . . .	13
4.2.3. Dual Stack ICN Deployment . . . . .	16
5. 5GLAN Architecture with ICN Support . . . . .	23
5.1. 5GC Architecture Extensions for 5GLAN Support . . . . .	23
5.1.1. Realization of Nx Interface . . . . .	24
5.1.2. Bitfield-based Forwarding in Existing Transport Networks . . . . .	25
5.2. ICN over 5GLAN . . . . .	26
6. Deployment Considerations . . . . .	27
7. Conclusion . . . . .	28
8. IANA Considerations . . . . .	28
9. Security Considerations . . . . .	28
10. Acknowledgments . . . . .	28
11. Informative References . . . . .	29
Authors' Addresses . . . . .	31

## 1. Introduction

The objective of this draft is to propose an architecture to enable information-centric networking (ICN) in the proposed 5G Next-generation Core network architecture (5GC) by leveraging its flexibility to allow new user and associated control plane functions. The reference architectural discussions in the 5G core network 3GPP specifications [TS23.501][TS23.502] form the basis of our

discussions. This draft also complements the discussions related to various ICN deployment opportunities explored in [I-D.irtf-icnrg-deployment-guidelines], where 5G technology is considered as one of the promising alternatives.

Though ICN is a general networking technology, it would benefit 5G particularly from the perspective of mobile edge computing (MEC). The following ICN features shall benefit MEC deployments in 5G:

- o **Edge Computing:** Multi-access Edge Computing (MEC) is located at the edge of the network and aids several latency sensitive applications such as augmented and virtual reality (AR/VR), as well as the ultra reliable and low latency class (URLLC) of applications such as autonomous vehicles. Enabling edge computing over an IP converged 5GC comes with the challenge of application level reconfiguration required to re-initialize a session whenever it is being served by a non-optimal service instance topologically. In contrast, named-based networking, as considered by ICN, naturally supports service-centric networking, which minimizes network related configuration for applications and allows fast resolution for named service instances.
- o **Edge Storage and Caching :** A principal design feature of ICN is the secured content (or named-data) object, which allows location independent data replication at strategic storage points in the network, or data dissemination through ICN routers by means of opportunistic caching. These features benefit both realtime and non-realtime applications whenever there is spatial and temporal correlation among content accessed by these users, thereby advantageous to both high-bandwidth and low-latency applications such as conferencing, AR/VR, and non-real time applications such as Video-on-Demand (VOD) and IoT transactions.
- o **Session Mobility:** Existing long-term evolution (LTE) deployments handle session mobility using centralized routing using the MME function, IP anchor points at Packet Data Network Gateway (PDN-GW) and service anchor point called Access Point Name (APN) functionality hosted in PDN-GW. LTE uses tunnel between radio edge (eNodeB) and PDN-GW for each mobile device attached to network. This design fails when service instances are replicated close to radio access network (RAN) instances, requiring new techniques to handle session mobility. In contrast, application-bound identifier and name resolution split principle considered for the ICN is shown to handle host mobility quite efficiently [ICNMOB].

In this document, we first discuss 5GC's design principals that allows the support of new network architectures. Then we summarize

the 5GC proposal, followed by control and user plane extensions required to support ICN PDU sessions. This is followed by discussions on enabling IP over ICN over 3GPP proposed 5GLAN service framework. We then discuss deployment considerations for both ICN over 5GC and IP over ICN over 5GLAN.

## 2. Terminology

Following are terminologies relevant to this draft:

**5G-NextGen Core (5GC):** Refers to the new 5G core network architecture being developed by 3GPP, we specifically refer to the architectural discussions in [TS23.501][TS23.502].

**5G-New Radio (5G-NR):** This refers to the new radio access interface developed to support 5G wireless interface [TS38.300].

**User Plane Function (UPF):** UPF is the generalized logical data plane function with context of the UE PDU session. UPFs can play many role, such as, being an flow classifier (UL-CL) (defined next), a PDU session anchoring point, or a branching point.

**Uplink Classifier (UL-CL):** This is a functionality supported by an UPF that aims at diverting traffic (locally) to local data networks based on traffic matching filters applied to the UE traffic.

**Packet Data Network (PDN or DN):** This refers to service networks that belong to the operator or third party offered as a service to the UE.

**Unified Data Management (UDM):** Manages unified data management for wireless, wireline and any other types of subscribers for M2M, IOT applications, etc. UDM reports subscriber related vital information e.g. virtual edge region, list of location visits, sessions active etc. UDM works as a subscriber anchor point so that means OSS/BSS systems will have centralized monitoring-of/access-to of the system to get/set subscriber information.

**Authentication Server Function (AUSF):** Provides mechanism for unified authentication for subscribers related to wireless, wireline and any other types of subscribers such as M2M and IOT applications. The functions performed by AUSF are similar to HSS with additional functionalities to related to 5G.

**Session Management Function (SMF):** Performs session management functions for attached users equipment (UE) in the 5G Core. SMF

can thus be formed by leveraging the CUPS (discussed in the next section) feature with control plane session management.

**Access Mobility Function (AMF):** Perform access mobility management for attached user equipment (UE) to the 5G core network. The function includes, network access stratus (NAS) mobility functions such as authentication and authorization.

**Application Function (AF):** Helps with influencing the user plane routing state in 5GC considering service requirements.

**Network Slicing:** This conceptualizes the grouping for a set of logical or physical network functions with its own or shared control, data and service plane to meet specific service requirements.

**5GLAN Service:** A service over the 5G system offering private communication using IP and/or non-IP type communications.

### 3. 5G NextGen Core Design Principles

The 5GC architecture is based on the following design principles that allows it to support new service networks like ICN efficiently compared to LTE networks:

- o **Control and User plane split (CUPS):** This design principle moves away from LTE's vertically integrated control/user plane design (i.e., Serving Gateway, S-GW, and Packet Data Network Gateway, P-GW) to one espousing an NFV framework with network functions separated from the hardware for service-centricity, scalability, flexibility and programmability. In doing so, network functions can be implemented both physically and virtually, while allowing each to be customized and scaled based on their individual requirements, also allowing the realization of multi-slice co-existence. This feature also allows the introduction of new user plane functions (UPF) in 5GC. UPFs can play many roles, such as, being an uplink flow classifier (UL-CL), a PDU session anchor point, a branching point function, or one based on new network architectures like ICN with new control functions, or re-using/ extending the existing ones to manage the new user plane realizations.
- o **Decoupling of RAT and Core Network :** Unlike LTE's unified control plane for access and the core, 5GC offers control plane separation of the RAN from the core network. This allows the introduction of new radio access technologies (RAT) along with slices based on new network architectures, offering the ability to map heterogeneous

RAN flows to arbitrary core network slices based on service requirements.

- o Non-IP PDU Session Support : A PDU session is defined as the logical connection between the UE and the data network (DN). 5GC offers a scope to support both IP and non-IP PDU (termed as "unstructured" payload), and this feature can potentially allow the support for ICN PDUs by extending or re-using the existing control functions. More discussions on taking advantage of this feature in ICN's context is presented in Section 4.2.2.2.
- o Service Centric Design: 5GC's service orchestration and control functions, such as naming, addressing, registration/authentication and mobility, will utilize API design similar to those used in cloud technologies. Doing so enables opening up interfaces for authorized service function interaction and creating service level extensions to support new network architectures. These APIs include the well accepted Get/Response and Pub/Sub approaches, while not precluding the use of point-to-point procedural approach among 5GC functional units (where necessary).
- o Distributed LAN Support: utilizing the aforementioned unstructured PDU session support, 5GC offers the capability to expose a Layer 2 LAN service to cellular user equipment. Such distributed LAN targets to complement those in fixed broadband, including local WLAN fanouts. Through such LAN capability, services can be realized by being virtually embedded into an intranet deployment with dedicated Internet-facing packet gateway functionality. Examples for such services, among others, are those related to Industrial IoT, smart city services and others. Utilizing this capability for ICN-based services is presented in Section 5.1.

#### 4. 5GC Architecture with ICN Support

##### 4.1. 5G NextGen Core Architecture

In this section, for brevity purposes, we restrict the discussions to the control and user plane functions relevant to an ICN deployment discussion in Section 4.2. More exhaustive discussions on the various architecture functions, such as registration, connection and subscription management, can be found in [TS23.501][TS23.502].



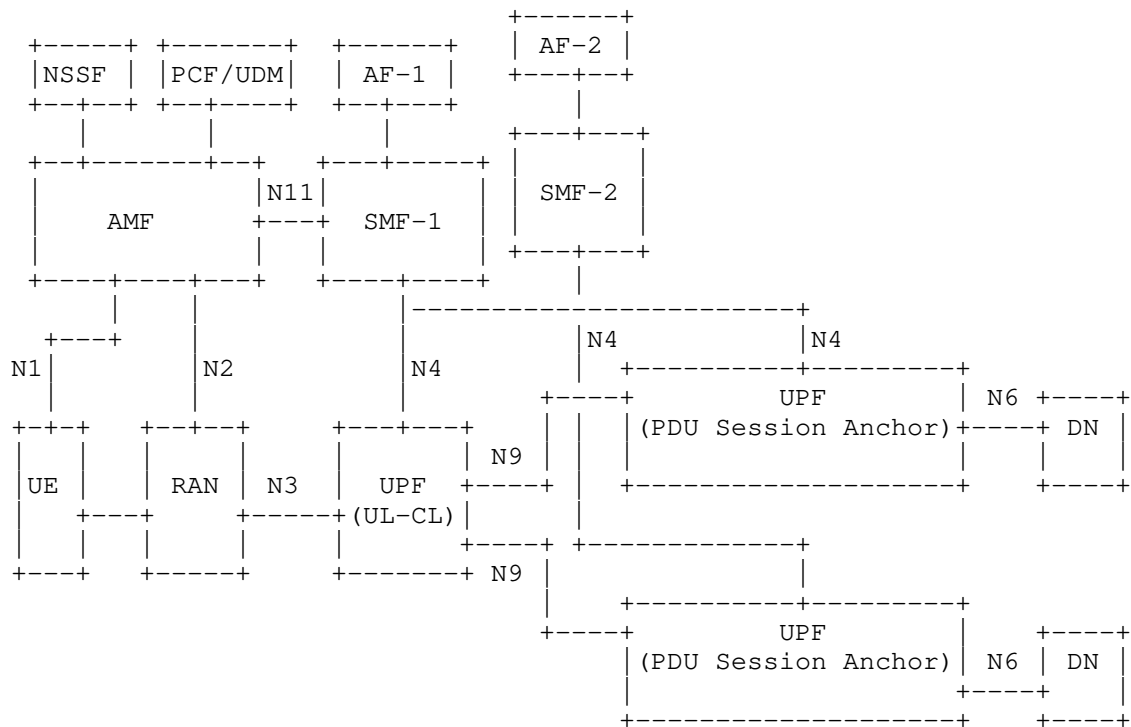


Figure 1: 5G Next Generation Core Architecture

In Figure 1, we show one variant of a 5GC architecture from [TS23.501], for which the functions of UPF's branching point and PDU session anchoring are used to support inter-connection between a UE and the related service or packet data networks (or PDNs) managed by the signaling interactions with control plane functions. In 5GC, control plane functions can be categorized as follows:

- o Common control plane functions that are common to all slices and which include the Network Slice Selection Function (NSSF), Policy Control Function (PCF), and Unified Data Management (UDM) among others.
- o Shared or slice specific control functions, which include the Access and Mobility Function (AMF), Session and Management Function (SMF) and the Application Function (AF).

AMF serves multiple purposes: (i) device authentication and authorization; (ii) security and integrity protection to non-access stratum (NAS) signaling; (iii) tracking UE registration in the

operator's network and mobility management functions as the UE moves among different RANs, each of which might be using different radio access technologies (RAT).

NSSF handles the selection of a particular slice for the PDU session request from the user entity (UE) using the Network Slice Selection Assistance Information (NSSAI) parameters provided by the UE and the configured user subscription policies in PCF and UDM functions. Compared to LTE's evolved packet core (EPC), where PDU session states in RAN and core are synchronized with respect to management, 5GC decouples this using NSSF by allowing PDU sessions to be defined prior to a PDU session request by a UE (for other differences see [lteversus5g]). This decoupling allows policy based inter-connection of RAN flows with slices provisioned in the core network. This functionality is useful particularly towards new use cases related to M2M and IOT devices requiring pre-provisioned network resources to ensure appropriate SLAs.

SMF is used to handle IP anchor point selection and addressing functionality, management of the user plane state in the UPFs (such as in uplink classifier (UL-CL), IP anchor point and branching point functions) during PDU session establishment, modification and termination, and interaction with RAN to allow PDU session forwarding in uplink/downlink (UL/DL) to the respective DN. SMF decisions are also influenced by AF to serve application requirements, for e.g., actions related to introducing edge computing functions.

In the data plane, UE's PDUs are tunneled to the RAN using the 5G RAN protocol [TS38.300]. From the RAN, the PDU's five tuple header information (IP source/destination, port, protocol etc.) is used to map the flow to an appropriate tunnel from RAN to UPF. Though the current 5GC proposal [TS23.501] follows LTE on using GPRS tunneling protocol (GTP) tunnel from NR to the UPF to carry data PDUs and another one for the control messages to serve the control plane functions; there are ongoing discussions to arrive upon efficient alternatives to GTP.

#### 4.2. ICN over 5GC

In this section, we focus on control and user plane enhancements required to enable ICN within 5GC, and identify the interfaces that require extensions to support ICN PDU sessions. Explicit support for ICN PDU sessions within access and 5GC networks will enable applications to leverage the core ICN features while offering it as a service to 5G users.

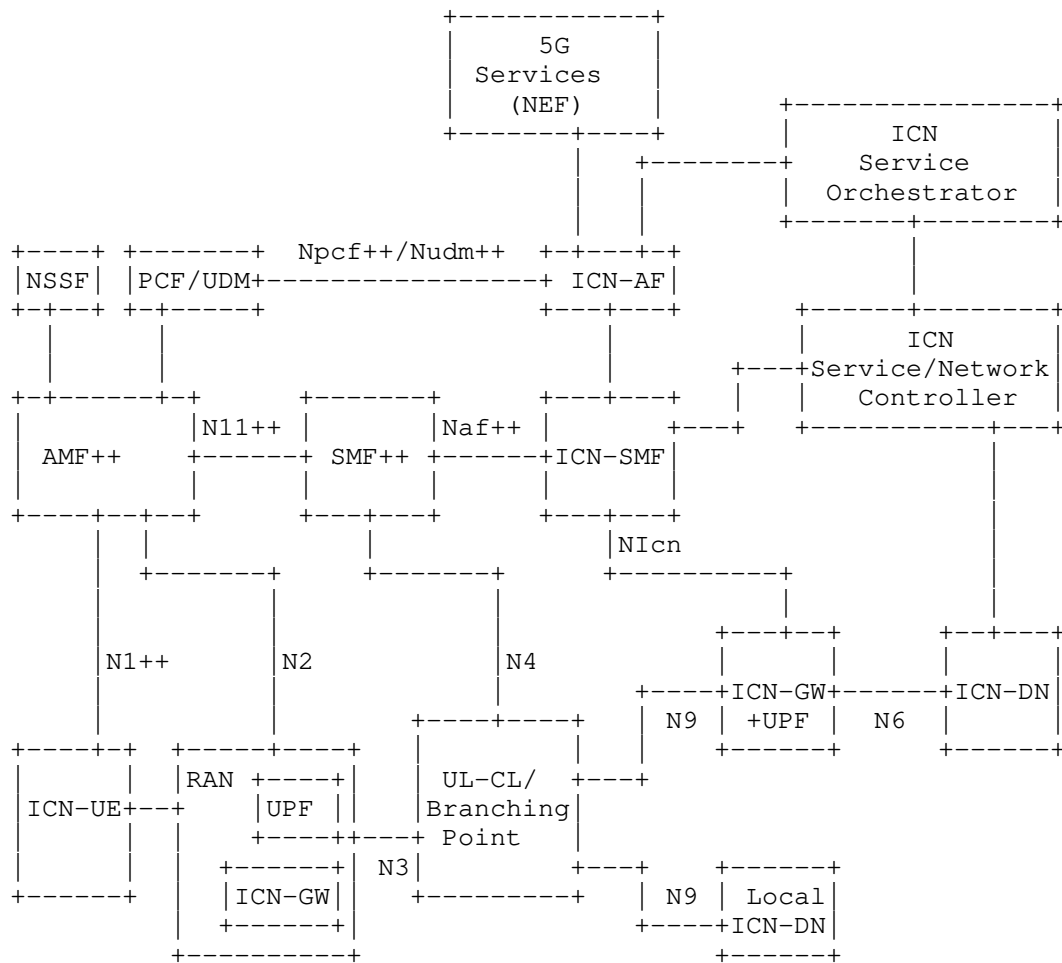


Figure 2: 5G Next Generation Core Architecture with ICN support

For an ICN-enabled 5GC network, the assumption is that the UE may have applications that can run over ICN or IP, for instance, UE's operating system offering applications to operate over ICN [Jacobson] or IP-based networking sockets. There may also be cases where UE is exclusively based on ICN. In either case, we identify an ICN enabled UE as ICN-UE. Different options exist to implement ICN in UE as described in [I-D.irtf-icnrg-icn-lte-4g] which is also applicable for 5G UE to enable formal ICN session handling, such as, using a Transport Convergence Layer (TCL) above 5G-NR, through IP address assignment from 5GC or using 5GC provision of using unstructured PDU session mode during the PDU session establishment process, which is

discussed in Section 4.2.2.2. Such convergence layer would implement necessary IP over ICN mappings, such as those described in [TROSSEN], for IP-based applications that are assigned to be transported over an ICN network. 5G UE can also be non-mobile devices or an IOT device using radio specification which can operate based on [TS38.300].

5GC will take advantage of network slicing function to instantiate heterogeneous slices, the same framework can be extended to create ICN slices as well [Ravindran]. This discussion also borrows ideas from [TS23.799], which offers a wide range of architectural discussions and proposals on enabling slices and managing multiple PDU sessions with local networks (with MEC) and its associated architectural support (in the service, control and data planes) and procedures within the context of 5GC.

Figure 2 shows the proposed ICN-enabled 5GC architecture. In the figure, the new and modified functional components are identified that interconnects an ICN-DN with 5GC. The interfaces and functions that require extensions to enable ICN as a service in 5GC can be identified in the figure with a '++' symbol. We next summarize the control, user plane and normative interface extensions that help with the formal ICN support.

#### 4.2.1. Control Plane Extensions

To support interconnection between ICN UEs and the appropriate ICN DN instances, we consider the following additional control plane extensions to orchestrate ICN services in coordination with 5GC's control components.

- o Authentication and Mobility Function (AMF++): ICN applications in the UEs have to be authorized to access ICN DNs. For this purpose, as in [TS23.501], operator enables ICN as a DN offering ICN services. As a network service, ICN-UE should also be subscribed to it and this is imposed using the PCF and UDM, which may interface with the ICN Application Function (ICN-AF) for subscription and session policy management of ICN PDU sessions. To enable ICN stack in the UE, AMF++ function has to be enabled with the capability of authenticating UE's attach request for ICN resources in 5GC. The request can be incorporated in NSSI parameter to request either ICN specific slice or using ICN in existing IP network slice when the UE is dual stacked. AMF++ can potentially be extended to also support ICN specific bootstrapping (such as naming and security) and forwarding functions to configure UE's ICN layer. These functions can also be handled by the ICN-AF and ICN control function in the UE after setting PDU session state in 5GC. Here, the recommendation is not about redefining the 5G UE attach procedures, but to extend the attach

procedures messages to carry ICN capabilities extensions in addition to supporting existing IP based services. The extensions should allow a 5G UE to request authentication to 5GC either in ICN, IP or dual-stack (IP and ICN) modes. Further research is required to optimize 5G attach procedures so that an ICN capable UE can be bootstrapped by minimizing the number of control plane messages. One possibility is to leverage existing 5G UE attach procedures as described in 3GPP's [TS23.502], where the UE can provide ICN identity in the LTE equivalent protocol configuration option information element (PCO-IE) message during the attach request as described in [I-D.irtf-icnrg-icn-lte-4g]. In addition, such requirement can be also be provided by the UE in NSSI parameters during initial attach procedures. Alternately, ICN paradigm offers name-based control plane messaging and security which one can leverage during the 5G UE attach procedures, however this requires further research.

- o Session Management Function (SMF++): Once a UE is authenticated to access ICN service in network, SMF manages to connect UE's ICN PDU sessions to the ICN DN in the UL/DL. SMF++ should be capable to manage both IP, ICN or dual stack UE with IP and ICN capabilities. To support ICN sessions, SMF++ creates appropriate PDU session policies in the UPF, which include UL-CL and ICN gateway (ICN-GW) (discussed in Section 4.2.2) through the ICN-SMF. For centrally delivered services, ICN-GW could also multiplex as an IP anchor point for IP applications. If MEC is enabled, these two functions would be distributed, as the UL-CL will re-route the flow to a local ICN-DN. 3GPP has defined IP based session management procedures to handle UE PDU sessions in TS23.502. For ICN UE we can either leverage same procedures when ICN is deployed as an overlay protocol. Towards this, SMF++ interfaces with AMF++ over N11++ to enable ICN specific user plane functions, which include tunnel configuration and traffic filter policy to inter-connect UE with the appropriate radio and the core slice. Furthermore, AMF++ sets appropriate state in the RAN and the UE that directs ICN flows to the chosen ICN UL-CL in the core network, and towards the right UE in the downlink.
- o ICN Session Management Function (ICN-SMF): ICN-SMF serves as control plane for the ICN state managed in ICN-GW. This function can be either incorporated as part of SMF++ or as a stand-alone one. This function interacts with SMF++ to obtain and also push ICN PDU session management information for the creation, modification and deletion of ICN PDU sessions in ICN-GW. For instance, when new ICN slices are provisioned by the ICN service orchestrator, ICN-SMF requests a new PDU session to the SMF that extends to the RAN. While SMF++ manages the tunnels to interconnect ICN-GW to UL-CL, ICN-SMF creates the appropriate

forwarding state in ICN-GW (using the forwarding information base or FIB) to enable ICN flows over appropriate tunnel interfaces managed by the SMF++. In addition, it also signals resource management rules to share compute, bandwidth, storage/cache resources among multiple slice instances co-located in the ICN-GW.

- o ICN Application Function (ICN-AF): ICN-AF represents the application controller function that interfaces with ICN-SMF and PCF/UDM function in 5GC. In addition to transferring ICN forwarding rules to ICN-SMF, ICN-AF also interfaces with PCF/UDM to transfer user profile and subscription policies along with session management requirement to UE's ICN PDU session in the 5GC network. ICN-AF is an extension of the ICN service orchestration function, which can influence both ICN-SMF and indirectly SMF++ to steer traffic based on ICN service requirements. ICN-AF can also interact with the northbound 5G operator's service functions, such as network exposure function (NEF) that exposes network capabilities, for e.g. location based services, that can be used by ICN-AF for proactive ICN PDU session and slice management and offer additional capabilities to the ICN slices.

#### 4.2.1.1. Normative Interface Extensions

- o N1++/N11++: This extension enables ICN specific control functions to support ICN authentication, configuration and programmability of an ICN-UE via AMF++ and SMF++, and also impose QoS requirements, handle mobility management of an ICN PDU session in 5GC based on service requirements.
- o N4: Though this signaling is service agnostic, as discussed in Section 4.2.2, future extensions may include signaling to enable ICN user plane features in these network functions. The extension of N4 to RAN is to handle the case when UPF function collocates with the RAN instance to enable localized ICN DNSs.
- o N1cn: This extension shall support two functions: (i) control plane programmability to enable ICN PDU sessions applicable to 5GC to map to name based forwarding rules in ICN-GW; (ii) control plane extensions to enable ICN mobility anchoring at ICN-GW, in which case it also acts as POA for ICN flows. Features such as ICN mobility as a service can be supported with this extension [ICNMOB].
- o Naf++: This extension supports 5GC control functions such as naming, addressing, mobility, and tunnel management for ICN PDU sessions to interact with SMF++ and AMF++.

- o Npcf++/Nudm++: This extension creates an interface to push ICN service and PDU session requirements to PCF and UDM functions that interact with the ICN-AF function for ICN slice specific configuration. These requirements are enforced at various steps, for instance, during ICN application registration, authentication, slice mapping, and provisioning of resources for these PDU sessions in the UPF.

#### 4.2.2. User Plane Extensions

The interconnection of a UE to an ICN-DN comprises of two segments, one from RAN to UL-CL and the other from UL-CL to ICN-GW. These segments use IP tunneling constructs, where the service semantic check at UL-CL is performed using IP's five tuples to determine both UL and DL tunnel mappings. We summarize the relevant UPFs and the interfaces for handling ICN PDU sessions as follows.

- o ICN Gateway (ICN-GW): ICN-GW is where the 5GC PDU sessions terminate and ICN service network begins. Compared to the traditional anchor points as in PGW, the ICN-GW is also a service gateway as it can host services or cache content enabled through the ICN architecture. The ICN-GW also includes the UPF functions to manage multiple tunnel interfaces enabling the relay of ICN PDU flows to appropriate UL-CL instances in the DL. Note that there may be multiple ICN-GWs serving different ICN services or slices. ICN-GW also manages other ICN functions such as enforcing the dynamic name based forwarding state, mobility state, in-network service function management, resource management with respect to sharing caching, storage, and compute resources among multiple services[Ravindran].
- o ICN Packet Data Network (ICN-(P)DN): ICN-DN represents a set of ICN nodes used for ICN networking and with heterogeneous service resources such as storage and computing points. An ICN network enables both network and application services, with network services including caching, mobility, multicast, multi-path routing (and possibly network layer computing), and application services including network resources (such as cache, storage, network state resources) dedicated to the application.
- \* Considering multiple ICN architecture proposals and multiple ICN deployment models discussed in [I-D.irtf-icnrg-deployment-guidelines], an alternate backward compatible (IP-over-)ICN solution is proposed in [TROSSEN]. Such an ICN-(P)DN can simply consist of SDN forwarding nodes and a logically centralized path computation entity (PCE), where the PCE is used to determine suitable forwarding identifiers being used for the path-based forwarding in the

SDN-based transport network. In addition, the PCE is responsible for maintaining the appropriate forwarding rules in the SDN switches. For interconnection to IP-based peering networks, a packet gateway is being utilized that mirrors the convergence layer functionality to map incoming ICN traffic back in to outgoing IP traffic and vice versa. This form of deployment would require minimal changes to the 5GC's user and control plane procedures, as the applications on these IP end points are not exposed (or minimally exposed) to any ICN state or configuration.

- o Uplink Classifier (UL-CL): UL-CL enables classification of flows based on source or destination IP address and steers the traffic to an appropriate network or service function anchor point. If the ICN-GW is identified based on service IP address associated with the ICN-UE's flows, UL-CL checks the source or destination address to direct traffic to an appropriate ICN-GW. For native ICN UE, ICN shall be deployed over 5G-NR; here, there may not be any IP association. For such packet flows new classification schema shall be required, such as, using 5G-NR protocol extensions to determine the tunnel interface to forward the ICN payload on, towards the next ICN-GW.

#### 4.2.2.1. Normative Interface Extensions

- o N3: Though the current architecture supports heterogeneous service PDU handling, future extensions can include user plane interface extensions to offer explicit support to ICN PDU session traffic, for instance, an incremental caching and computing function in RAN or UL-CL to aid with content distribution.
- o N9: Extensions to this interface can consider UPFs to enable richer service functions, for instance to aid context processing. In addition extensions to enable ICN specific encapsulation to piggyback ICN specific attributes such as traffic or mobility data between the UPF branching point and the ICN-GW.
- o N6: This interface is established between the ICN-GW and the ICN-DN, whose networking elements in this segment can be deployed as an overlay or as a native Layer-3 network.

#### 4.2.2.2. ICN over non-IP PDU

5GC accommodates non-IP PDU support which is defined for Ethernet or any unstructured data[TS23.501]. This feature allows native support of ICN over 5G RAN, with the potential enablement of ICN-GW in the BS itself as shown in Figure 2. Formalizing this feature to recognize ICN PDUs has many considerations:



- o Attach Procedures for UE with Non-IP PDN: Assuming a 5GC can support both IP and non-IP PDN, this requires control plane support. In a typical scenario, when UE sends an attach message to 5GC, the type of PDU connection is indicated in the PCO-IE field, for e.g. in this case as being non-IP PDN to invoke related control plane session management tasks. ICN over non-IP PDU session will allow the UE to attach to 5GC without any IP configuration. 5GC attach procedures specified [TS23.501] can be used to support authentication of UE with PDN type set to non-IP, using existing AUSF/UDM functions in coordination with the ICN-AF function discussed earlier if required.
- o User Plane for UE with Non-IP PDN: Without any IP tunnel configuration and ICN's default consumer agnostic mode of operation requires ways to identify the ICN-UE in the user plane, this can be enabled by introducing network identifier in the lower layers such as in the PDCP or MAC layer, that can assist for functions such as policy and charging at the BS and related session management tasks. These identifiers can also be used to demultiplex the DL traffic from the ICN-GW in the BS to the respective ICN-UEs. Also, ICN extensions can be incorporated in control plane signaling to identify an ICN-UE device and these parameters can be used by SMF to conduct non-IP routing. The policing and charging functions can be enforced by the UPF function in the BS which obtains the traffic filtering rules from the SMF. To enable flat ICN network from the BS requires distributed policy, charging and legal intercept which requires further research. Further ICN slice multiplexing can be realized by also piggybacking slice-ID (NSSI) along with device ID to differentiate handover to multiple ICN slices at the base station. Inter-working function (IWF) is required if services based on non-IP UE has to transact or communicate with transport, applications functions or other UE based on IP services. This also has implications on how mobility is managed for such PDU sessions.
- o Mobility Handling: Considering mobility can be support by ICN, it is inefficient to traverse other intermediate IP networks between the BS and the next ICN hop. This requires ICN PDU to be handled by an ICN instance in the BS itself, in association with UL-CL function local to the BS as shown in Figure 2. Control plane extensions discussed in Section 4.2.1 can be used in tandem with distributed mobility protocols to handle ICN mobility, one such solution for producer mobility is proposed in [ICNMOB]
- o Routing Considerations: Flat ICN network realizations also offers the advantage of optimal routing, compared to anchor point based realization in LTE. This also leads to optimal realization of the data plane considering the absence of overhead due to tunneling

while forwarding ICN traffic. However, developing a routing control plane in to handle the ICN PDU sessions either leveraging SMF functions or a distributed realization requires more investigation. In the centralized approach the SMF could interact with ICN-SMF to set the forwarding rules in the ICN-GW in the BS and other ICN-UPFs, however this may also lead to scalability issues if a flat ICN network is to be realized. This also has implications to route the non-IP PDU sessions efficiently to the closest ICN-MEC instance of the service.

- o IP over ICN: Native support of ICN in the RAN raises the possibility of leveraging the mobility functions in ICN protocols as a replacement for GTP tunneling in the 5GC, as described in [I-D.white-icnrg-ipoc] and [TROSSEN].
- o Mobile Edge Computing: Another significant advantage is with respect to service-centric edge computing at the ICN-GW or other ICN points, either through explicit hosting of service functions[VSER] in ICN or in-network computing based on NFN proposal[NFN]. A certain level of orchestration is required to ensure service interconnection and its placement with appropriate compute resources and inter-connected with bandwidth resources so that the desired SLA is offered.

#### 4.2.3. Dual Stack ICN Deployment

##### 4.2.3.1. 5G User Plane Protocol Stack

It is important to understand that a User Equipment (UE) can be either consumer (receiving content) or publisher (pushing content for other clients). The protocol stack inside mobile device (UE) is complex as it has to support multiple radio connectivity access to gNB(s).

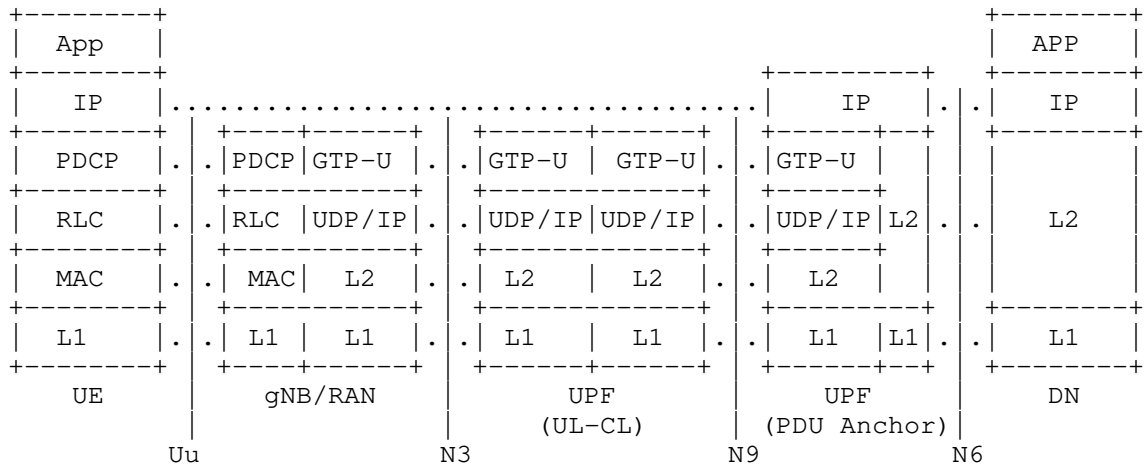


Figure 3: 5G User Plane Protocol Stack

Figure 3 provides high level description of a 5G user plane protocol stack, where: 1) the lower 4 layers (i.e. L1, MAC, RLC, PDCP) at UE is for radio access and air interface to gNB; 2) the IP layer (i.e. PDU layer) at UE is used for providing IP transport infrastructure to support PDU session between UE and UPF (PDU Anchor); 3) GTP-U provides tunneling between gNB and UPF, or between two UPFs. Although UDP/IP exists under GTP-U, IP mainly refers to "IP" between UE and UPF (PDU Anchor) for the rest of this document, unless explicitly clarified; 4) UL-CL is only for uplink traffic and UPF (UL-CL) shall not be needed for downlink traffic towards UE.

#### 4.2.3.2. Protocol Stack for ICN Deployment in 5G

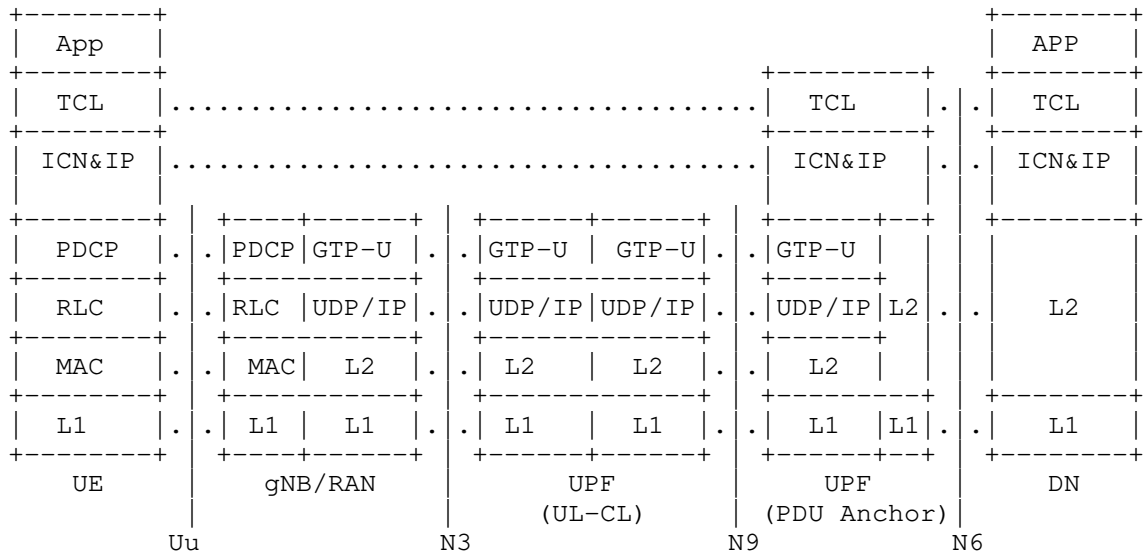


Figure 4: Dual Stack ICN Deployment

ICN can be deployed in dual stack model for 5G user plane as illustrated in Figure 4, where: 1) both ICN and IP (i.e. dual stack) can reside between TCL and PDCP to provide transport infrastructure from UE to UPF (PDU Anchor); 2) in order to support the dual ICN&IP transport layer, PDCP needs some enhancements; 3) a new Transport Convergence Layer (TCL) is introduced to coordinate between applications and ICN&IP transport layer; 4) Applications on top of TCL could be ICN applications or IP applications.

With this dual stack model, four different cases are possible for the deployment of ICN natively and/or with IP dependent on which types of applications (ICN or IP) uses which types of underline transport (ICN or IP), from the perspective of the applications either on UE (or content provider).

#### Case 1. IP over IP (IPoIP)

In this scenario UE uses applications tightly integrated with the existing IP transport infrastructure. In this option, the TCL has no additional function since the packets are directly forwarded using IP protocol stack, which in turn sends the packets over the IP transport.

#### Case 2. ICN over ICN (ICNoICN)

Similar to case 1 above, ICN applications tightly integrate with the ICN transport infrastructure. The TCL has no additional responsibility since the packets are directly forwarded using ICN protocol stack, which in turn sends the packets over the ICN transport.

#### Case 3. ICN over IP (ICNoIP)

In ICN over IP scenario, the underlying IP transport infrastructure is not impacted (i.e., ICN is implemented as an overlay over IP between UE and content provider). IP routing is used from Radio Access Network (gNB) to mobile backhaul, IP core and UPF. UE attaches to UPF (PDU Anchor) using IP address. Content provider in DN is capable of serving content either using IP or ICN, based on UE request.

An alternative approach to implement ICN over IP is provided in Hybrid ICN [I-D.muscariello-intarea-hicn], which implements ICN over IP by mapping of ICN names to the IPv4/IPv6 addresses.

#### Case 4. IP over ICN (IPoICN)

In IP over ICN scenario, IP application utilize an ICN-based routing while preserving the overall IP protocol semantics, as shown, e.g., in H2020 project [H2020]. Implementing IP services over ICN provides an opportunity leveraging benefit of ICN in the transport infrastructure.

Note that the IP over ICN case could be supported for pure IP (over IP) UEs through introducing a Network Attachment Point (NAP) to interface to an ICN network. Here, the UPF (PDU Anchor) interfaces to said NAP in the northbound; alternatively, the NAP can be integrated as a part of UPF (PDU Anchor). For this scheme, the NAP provides a standard IP network interface towards the IP-enabled UE via UPF (PDU Anchor), encapsulates any received IP service (e.g. HTTP) request into an appropriate ICN packet which is then published as an appropriately formed named information item. Conversely, the NAP subscribes to any appropriately formed named information items, where the information identifier represents any IP-exposed service that is exposed at any IP-level UE locally connected to the NAP. Any received ICN packet is then forwarded to the appropriate local IP-enabled UE after being appropriately decapsulated, recovering the original IP service (e.g. HTTP) request.

In a dual-stack UE that supports the above cases, the TCL helps determine what type of transport (e.g. ICN or IP), as well as type of radio interface (e.g. 5G or WiFi or both), is used to send and receive the traffic based on preference e.g. content location,

content type, content publisher, congestion, cost, quality of service etc. It helps to configure and decide the type of connection as well as the overlay mode (ICNoIP or IPoICN, explained above) between application and the protocol stack (IP or ICN) to be used.

TCL can use a number of mechanisms for the selection of transport (i.e. ICN or IP). It can use a per application configuration through a management interface, possibly even a user-facing setting realized through a user interface, similar to those used today that select cellular over WiFi being used for selected applications. In another option, it might use a software API, which an adapted IP application could use to specify e.g. an ICN transport for obtaining its benefits.

Another potential application of TCL is in implementation of network slicing, where it can have a slice management capability locally or it can interface to an external slice manager through an API [I-D.galis-anima-autonomic-slice-networking]. This solution can enable network slicing for IP and ICN transport selection from the UE itself. The TCL could apply slice settings to direct certain traffic (or applications) over one slice and others over another slice, determined by some form of 'slicing policy'. Slicing policy can be obtained externally from slice manager or configured locally on UE.

#### 4.2.3.3. Protocol Interactions and Impacts

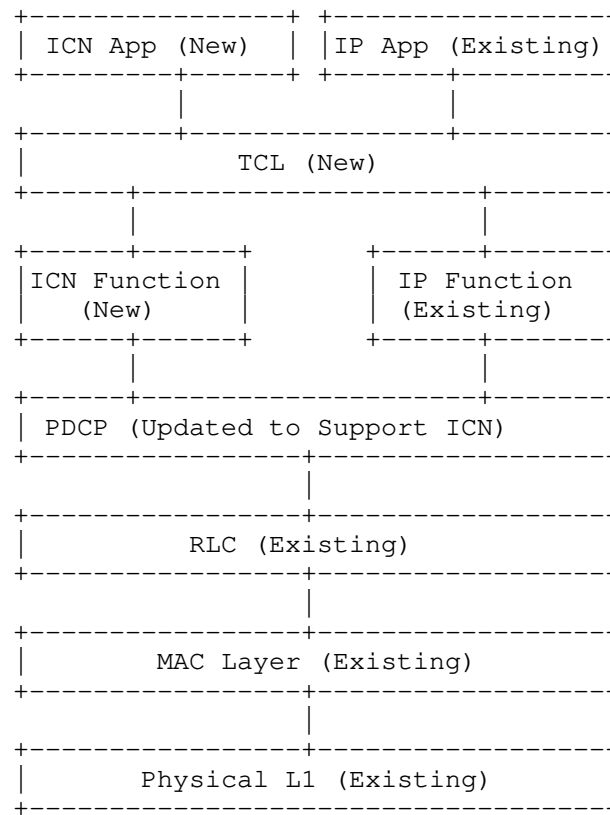


Figure 5: Dual Stack ICN Protocol Interactions at UE

The protocol interactions and impact of supporting tunneling of ICN packet into IP or to support ICN natively are described in Figure 5.

- o Existing application layer can be modified to provide options for new ICN based application and existing IP based applications. UE can continue to support existing IP based applications or host new applications developed either to support native ICN as transport, ICNoIP or IPoICN based transport. Application layer has the option of selecting either ICN or IP transport layer as well as radio interface to send and receive data traffic. Our proposal is to provide a common Application Programming Interface (API) to the application developers such that there is no impact on the application development when they choose either ICN or IP transport for exchanging the traffic with the network. TCL function handles the interaction of application with the multiple transport options.

- o The TCL helps determine what type of transport (e.g. ICN or IP) as well as type of radio interface (e.g. 5G NR or WiFi or both), is used to send and receive the traffic. Application layer can make the decision to select a specific transport based on preference e.g. content location, content type, content publisher, congestion, cost, quality of service etc. There can be an Application Programming Interface (API) to exchange parameters required for transport selection. The southbound interactions of TCL will be either to IP or ICN at the network layer. When selecting the IPoICN [TROSSEN] mode, the TCL is responsible for receiving an incoming IP or HTTP packet and publishing the packet under a suitable ICN name (i.e. the hash over the destination IP address for an IP packet or the hash over the FQDN of the HTTP request for an HTTP packet) to the ICN network. In the HTTP case, the TCL maintains a pending request mapping table to map returning responses to the originating HTTP request. The common API will provide a common 'connection' abstraction for this HTTP mode of operation, returning the response over said connection abstraction, akin to the TCP socket interface, while implementing a reliable transport connection semantic over the ICN from the UE to the receiving UE or the PGW. If the HTTP protocol stack remains unchanged, therefore utilizing the TCP protocol for transfer, the TCL operates in local TCP termination mode, retrieving the HTTP packet through said local termination. The southbound interactions of the Transport Convergence Layer (TCL) will be either to IP or ICN at the network layer.
- o ICN function (forwarder) is introduced in parallel to the existing IP layer. ICN forwarder contains functional capabilities to forward ICN packets, e.g. Interest packet to gNB or response "data packet" from gNB to the application.
- o For dual stack scenario, when UE is not supporting ICN at network layer, we use IP underlay to transport ICN packets. ICN function will use IP interface to send Interest and Data packets for fetching or sending data using ICN protocol function. This interface will use ICN overlay over IP using any overlay tunneling mechanism.
- o To support ICN at network layer in UE, PDCP layer has to be aware of ICN capabilities and parameters. PDCP is located in the Radio Protocol Stack in the 5G Air interface, between IP (Network layer) and Radio Link Control Layer (RLC). PDCP performs following functions [TS36.323]:
  - \* Data transport by listening to upper layer, formatting and pushing down to Radio Link Layer (RLC).



- \* Header compression and decompression using Robust Header Compression (ROHC).
- \* Security protections such as ciphering, deciphering and integrity protection.
- \* Radio layer messages associated with sequencing, packet drop detection and re-transmission etc.
- o No changes are required for lower layer such as RLC, MAC and Physical (L1) because they are not IP aware.

## 5. 5GLAN Architecture with ICN Support

### 5.1. 5GC Architecture Extensions for 5GLAN Support

In this section, we present an overview of ongoing work to provide cellular LAN connectivity over a 5GC compliant network for Release 16 and above deployments.

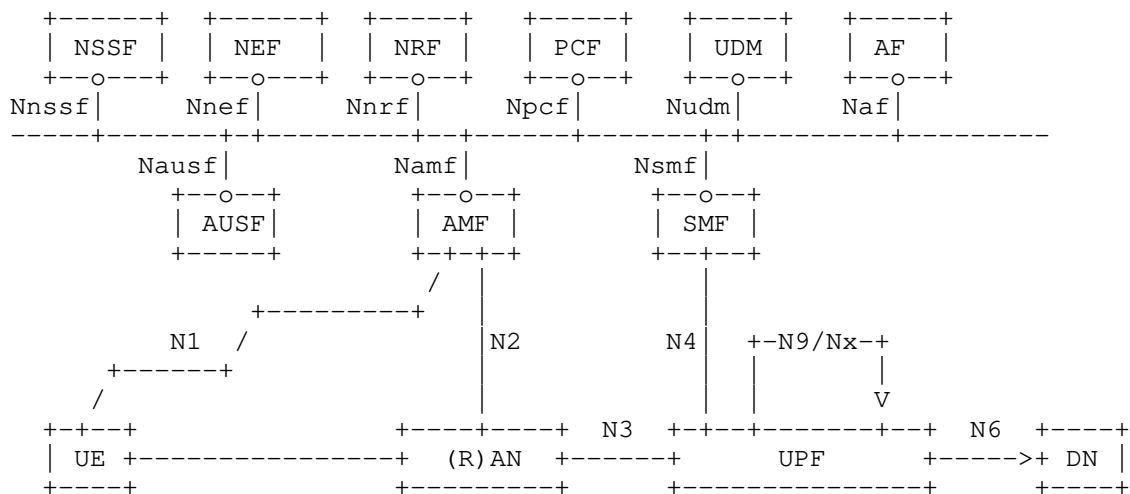


Figure 6: 5G Core Network with Vertical\_LAN (5GLAN) Extensions

Figure 6 shows the current 5G Core Network Architecture being discussed within the scope of the normative work addressing 5GLAN Type services in the 3GPP System Architecture Working Group 2 (3GPP SA2), referred formally as "5GS Enhanced support of Vertical and LAN Services" [SA2-5GLAN]. The goal of this work item is to provide distributed LAN-based connectivity between two or more terminals or

User Equipment entities (UEs) connected to the 5G network. The Session Management Function (SMF) provides a registration and discovery protocol that allows UEs wanting to communicate via a relevant 5GLAN group towards one or more UEs also members of this 5GLAN group, to determine the suitable forwarding information after each UE previously registered suitable identifier information with the SMF responsible to manage the paths across UEs in a 5GLAN group. UEs register and discover (obtain) suitable identifiers during the establishment of a Protocol Data Unit (PDU) Session or PDU Session Modification procedure. Suitable identifier information, according to [SA2-5GLAN], are Ethernet MAC addresses as well as IP addresses (the latter is usually assigned during the session setup through the SMF).

The SMF that manages the path across UEs in a 5GLAN group, then establishes the suitable procedures to ensure the forwarding between the required UPFs (user plane functions) to ensure the LAN connectivity between the UEs (user equipments) provided in the original request to the SMF. When using the N9 interface to the UPF, this forwarding will rely on a tunnel-based approach between the UPFs along the path, while the Nx interface uses path-based forwarding between UPFs, while LAN-based forwarding is utilized between the final UPF and the UE (utilizing the N3 interface towards the destination UE).

#### 5.1.1. Realization of Nx Interface

In the following, we discuss ongoing work to realize the Nx interface, i.e., path-based forwarding is assumed with the utilization of a path identifier for the end-to-end LAN communication. Here, the path between the source and destination UPFs is encoded through a bitfield, provided in the packet header. Each bitposition in said bitfield represents a unique link in the network. Upon receiving an incoming packet, each UPF inspects said bitfield for the presence of any local link that is being served by one of its output ports. Such presence check is implemented via a simple binary AND and CMP operation. If no link is being found, the packet is dropped. Such bitfield-based path representation also allows for creating multicast relations in an ad-hoc manner by combining two or more path identifiers through a binary OR operation. Note that due to the assignment of a bitposition to a link, path identifiers are bidirectional and can therefore be used for request/response communication without incurring any need for path computation on the return path.

For sending a packet from one Layer 2 device (UE) connected to one UPF (via a RAN) to a device connected to another UPF, we provide the MAC address of the destination and perform a header re-write by

providing the destination MAC address of the ingress UPF when sending from source device to ingress and placing the end destination MAC address in the payload. Upon arrival at the egress UPF, after having applied the path-based forwarding between ingress and egress UPF, the end destination address is restored while the end source MAC is placed in the payload with the egress L2 forwarder one being used as the L2 source MAC for the link-local transfer. At the receiving device, the end source MAC address is restored as the source MAC, creating the perception of a link-local L2 communication between the end source and destination devices.

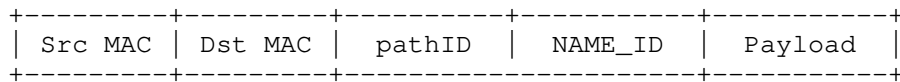


Figure 7: General Packet Structure

For this end-to-end transfer, the general packet structure of Figure 7 is used. The Name\_ID field is being used for the ICN operations, while the payload contains the information related to the transaction-based flow management and the PATH\_ID is the bitfield-based path identifier for the path-based forwarding.

#### 5.1.1.2. Bitfield-based Forwarding in Existing Transport Networks

An emerging technology for Layer 2 forwarding that suits the 5GLAN architecture in Figure 6 is that of Software-defined networking (SDN) [SDNDef], which allows for programmatically forwarding packets at Layer 2. Switch-based rules are being executed with such rules being populated by the SDN controller. Rules can act upon so-called matching fields, as defined by the OpenFlow protocol specification [OpenFlowSwitch]. Those fields include Ethernet MAC addresses, IPv4/6 source and destination addresses and other well-known Layer 3 and even 4 transport fields.

As shown in [Reed], efficient path-based forwarding can be realized in SDN networks by placing the aforementioned path identifiers into the IPv6 source/destination fields of a forwarded packet. Utilizing the IPv6 source/destination fields allows for natively supporting 256 links in a transport network. Larger topologies can be supported by extension schemes but are left out of this paper for brevity of the presentation. During network bootstrapping, each link at each switch is assigned a unique bitnumber in the bitfield (through the SMF function of the 5GC). In order to forward based on such bitfield path information, the NR instructs the SDN controller to insert a suitable wildcard matching rule into the SDN switch. This wildcard

at a given switch is defined by the bitnumber that has been assigned to a particular link at that switch during bootstrapping. Wildcard matching as a generalization of longest prefix matching is natively supported since the OpenFlow v1.3 specification, efficiently implemented through TCAM based operations. With that, SDN forwarding actions only depend on the switch-local number of output ports, while being able to transport any number of higher-layer flows over the same transport network without specific flow rules being necessary. This results in a constant forwarding table size while no controller-switch interaction is necessary for any flow setup; only changes in forwarding topology (resulting in a change of port to bitnumber assignment) will require suitable changes of forwarding rules in switches.

Although we focus the methods in this draft on Layer 2 forwarding approaches, path-based transport networks can also be established as an overlay over otherwise Layer 2 networks. For instance, the BIER (Bit Indexed Explicit Replication) [RFC8279] efforts within the Internet Engineer Task Force (IETF) establish such path-based forwarding transport as an overlay over existing, e.g., MPLS networks. The path-based forwarding identification is similar to the aforementioned SDN realization although the bitfield represents ingress/egress information rather than links along the path.

Yet another transport network example is presented in [Khalili], utilizing flow aggregation over SDN networks. The flow aggregation again results in a path representation that is independent from the specific flows traversing the network.

## 5.2. ICN over 5GLAN

ICN aims at replacing the routing functionality of the Internet Protocol (IP). It is therefore natively supported over a Layer 2 transport network, such as Ethernet-based networks. Deployments exists over WiFi and local LAN networks, while usually overlaying (over IP) is being used for connectivity beyond localized edge networks.

With the emergence of the 5GLAN capability in (future) Release 16 based 5G networks, such cellular LAN connectivity to provide pure ICN could be utilized for pure ICN-based deployments, i.e. without the dual stack capability outlined in Section 4.2.3.2. With this, the entire 5G network would be interpreted as a local LAN, providing the necessary Layer 2 connectivity between the ICN network components. With the support of roaming in 5GLAN, such '5G network' can span several operators and therefore large geographies.

Such deployment, however, comes without any core network integration, similar to the one outlined in Section 4.1, and therefore does not utilize ICN capabilities within the overall 5G core and access network. Benefits such as those outlined in the introduction, e.g., caching, would only exist at the endpoint level (from a 5GLAN perspective).

However, ICN components could be provided as SW components in a network slice at the endpoints of such 5GLAN connectivity, utilizing in-network compute facilities, e.g., for caching, CCN routing capabilities and others. Such endpoint-driven realization of a specific ICN deployment scenario is described in more detail in [I-D.trossen-icnrg-IP-over-icn], focusing on the provisioning of IP-based services over an ICN, which in turn is provided over a LAN (and therefore also 5GLAN) based transport network.

## 6. Deployment Considerations

The work in [I-D.irtf-icnrg-deployment-guidelines] outlines a comprehensive set of considerations related to the deployment of ICN. We now relate the solutions proposed in this draft to the two main aspects covered in the deployment considerations draft, namely the 'deployment configuration' (covered in Section 4 of [I-D.irtf-icnrg-deployment-guidelines]) that is being realized and the 'deployment migration paths' (covered in Section 5 of [I-D.irtf-icnrg-deployment-guidelines]) that are being provided.

The solutions proposed in this draft relate to those 'deployment configuration' as follows:

- o The integration with the 5GC, as proposed in Section 4.2, follows the 'Clean-slate ICN' deployment configuration, i.e., integrating the ICN capabilities natively into the 5GC through appropriate extensions at the control and user plane level.
- o The utilization of the 5GLAN capabilities, as proposed in Section 5.2, follows the 'ICN-as-an-Overlay', interpreting the 5GLAN as an overlay capability with no 5GC integration being considered (as in the 'Clean-slate ICN' configuration).
- o The deployment of 5GLAN based ICN capabilities can be realized following the 'ICN-as-a-Slice' deployment configuration, i.e., the 5GLAN connectivity is provided to a 'vertical 5G customer' which in turn provides the ICN capability over 5GLAN within said network (and compute) slice at the endpoints of the 5GLAN connectivity, as proposed in Section 5.2.

In relation of the 'deployment migration paths', the solutions in this draft relate as follows:

- o The integration with the 5GC, as proposed in Section 4.2, facilitates 'edge network migration' (interpreting the cellular sub-system here as an edge network albeit a possibly geographically large one).
- o The dual-stack deployment, as proposed in Section 4.2.3, facilitates 'application and services migration' through not only supporting ICN applications but also IP-based applications through the proposed IP-over-ICN mapping in the terminal.
- o The ICN over 5GLAN deployment, possibly combined with an ICN-as-a-Slice deployment, facilitates the 'content delivery networks migration' through a deployment of ICN-based 5GLAN connected CDN elements in (virtualized) edge network nodes or POP locations in the customer (5G) network.

## 7. Conclusion

In this draft, we explore the feasibility of realizing future networking architectures like ICN within the proposed 3GPP's 5GC architecture. Towards this, we summarized the design principles that offer 5GC the flexibility to enable new network architectures. We then discuss 5GC architecture aspects along with the user/control plane extensions required to handle ICN PDU sessions formally to realize ICN with 5GC integration as well as ICN over a pure 5GLAN connectivity.

## 8. IANA Considerations

This document requests no IANA actions.

## 9. Security Considerations

This draft proposes extensions to support ICN in 5G's next generation core architecture. ICN being name based networking opens up new security and privacy considerations which have to be studied in the context of 5GC. This is in addition to other security considerations of 5GC for IP or non-IP based services considered in [TS33.899].

## 10. Acknowledgments

...

## 11. Informative References

- [H2020] H2020, "The POINT Project", <https://www.point-h2020.eu/> .
- [I-D.galis-anima-autonomic-slice-networking]  
Galis, A., Makhijani, K., Yu, D., and B. Liu, "Autonomic Slice Networking", draft-galis-anima-autonomic-slice-networking-05 (work in progress), September 2018.
- [I-D.irtf-icnrg-deployment-guidelines]  
Rahman, A., Trossen, D., Kutscher, D., and R. Ravindran, "Deployment Considerations for Information-Centric Networking (ICN)", draft-irtf-icnrg-deployment-guidelines-06 (work in progress), May 2019.
- [I-D.irtf-icnrg-icn-lte-4g]  
suthar, P., Stolic, M., Jangam, A., Trossen, D., and R. Ravindran, "Native Deployment of ICN in LTE, 4G Mobile Networks", draft-irtf-icnrg-icn-lte-4g-03 (work in progress), March 2019.
- [I-D.muscariello-intarea-hicn]  
Muscariello, L., Carofiglio, G., Auge, J., and M. Papalini, "Hybrid Information-Centric Networking", draft-muscariello-intarea-hicn-01 (work in progress), December 2018.
- [I-D.white-icnrg-ipoc]  
White, G., Shannigrahi, S., and C. Fan, "Internet Protocol Tunneling over Content Centric Mobile Networks", draft-white-icnrg-ipoc-01 (work in progress), June 2018.
- [ICNMOB] Azgin, A., Ravindran, R., Chakraborti, A., and G. Wang, "Seamless Producer Mobility as a Service in Information Centric Networks.", 5G/ICN Workshop, ACM ICN Sigcomm 2016, 2016.
- [Jacobson]  
Jacobson, V. and et al., "Networking Named Content", Proceedings of ACM Context, , 2009.
- [Khalili] Khalili, R., Poe, W., Despotovic, Z., and A. Hecker, "Reducing State of SDN Switches in Mobile Core Networks by Flow Rule Aggregation", IEEE ICCCN 2016, Hawaii, USA, August 2016.

- [lteversus5g] Kim, J., Kim, D., and S. Choi, "3GPP SA2 architecture and functions for 5G mobile communication system.", ICT Express 2017, 2017.
- [NFN] Sifalakis, M., Kohler, B., Christopher, C., and C. Tschudin, "An information centric network for computing the distribution of computations", ACM, ICN Sigcomm, 2014.
- [OpenFlowSwitch] Open Networking Foundation, available at <https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf>, "OpenFlow Switch Specification V1.5.1", 2018.
- [Ravindran] Ravindran, R., Chakraborti, A., Amin, S., Azgin, A., and G. Wang, "5G-ICN : Delivering ICN Services over 5G using Network Slicing", IEEE Communication Magazine, May, 2016.
- [Reed] Reed, M., AI-Naday, M., Thomos, N., Trossen, D., Petropoulos, G., and S. Spirou, "Stateless Multicast Switching in Software Defined Networks", IEEE ICC 2016, Kuala Lumpur, Malaysia, 2016.
- [RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <<https://www.rfc-editor.org/info/rfc8279>>.
- [SA2-5GLAN] 3gpp-5glan, "SP-181129, Work Item Description, Vertical\_LAN(SA2), 5GS Enhanced Support of Vertical and LAN Services", 3GPP , [http://www.3gpp.org/ftp/tsg\\_sa/TSG\\_SA/TSGS\\_82/Docs/SP-181120.zip](http://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_82/Docs/SP-181120.zip).
- [SDNDef] Open Networking Foundation, available at <https://www.opennetworking.org/sdn-definition/>, "Software-Defined Networking (SDN) Definition", 2018.
- [TROSSEN] Trossen, D., Reed, M., Riihijarvi, J., Georgiades, M., and G. Xylomenos, "IP Over ICN - The Better IP ?", EuCNC, European Conference on Networks and Communications , July, 2015.



- [TS23.501] 3gpp-23.501, "Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Rel.15)", 3GPP , December 2018.
- [TS23.502] 3gpp-23.502, "Technical Specification Group Services and System Aspects; Procedures for the 5G System; Stage 2 (Rel. 15)", 3GPP , January 2019.
- [TS23.799] 3gpp-23.799, "Technical Specification Group Services and System Aspects; Study on Architecture for Next Generation System (Rel. 14)", 3GPP , 2017.
- [TS33.899] 3gpp-33.899, "Study on the security aspects of the next generation system", 3GPP , 2017.
- [TS36.323] 3gpp-36.323, "Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification (Rel. 15)", 3GPP , January 2019.
- [TS38.300] 3gpp-38-300, "Technical Specification Group Radio Access Network; NR; NR and NG-RAN Overall Description; Stage 2 (Rel.15)", 3GPP , January 2019.
- [VSER] Ravindran, R., Liu, X., Chakraborti, A., Zhang, X., and G. Wang, "Towards software defined ICN based edge-cloud services", CloudNetworking(CloudNet), IEEE International Conference on, IEEE International Conference on CloudNetworking(CloudNet), 2013.

#### Authors' Addresses

Ravi Ravindran  
Futurewei Technologies  
2330 Central Expressway  
Santa Clara 95050  
USA

Email: ravi.ravindran@futurewei.com

Prakash Suthar  
Cisco Systems  
9501 Technology Blvd.  
Rosemont 50618  
USA

Email: [psuthar@cisco.com](mailto:psuthar@cisco.com)  
URI: <http://www.cisco.com/>

Dirk Trossen  
InterDigital Inc.  
64 Great Eastern Street, 1st Floor  
London EC2A 3QR  
United Kingdom

Email: [Dirk.Trossen@InterDigital.com](mailto:Dirk.Trossen@InterDigital.com)  
URI: <http://www.InterDigital.com/>

Chonggang Wang  
InterDigital Inc.  
1001 E Hector St, Suite 300  
Conshohocken PA 19428  
United States

Email: [Chonggang.Wang@InterDigital.com](mailto:Chonggang.Wang@InterDigital.com)  
URI: <http://www.InterDigital.com/>

Greg White  
CableLabs  
858 Coal Creek Circle  
Louisville CO 80027  
USA

Email: [g.white@cablelabs.com](mailto:g.white@cablelabs.com)