                          CDNI Request Routing
                draft-lelouedec-cdni-request-routing-00

Abstract

   The present document proposes to clarify the CDNI Request Routing
   interface introduced in [I-D.ietf-cdni-framework] and [I-D.ietf-cdni-
   problem-statement], as well as related terminology.

   In particular the present document proposes to split the CDNI Request
   Routing interface into two separate interfaces with clearer roles,
   named respectively CDNI Routing interface and CDNI Downstream
   Resource Identifier Signaling interface (CDNI DRIS interface).

   This part of the CDN interconnection framework the IETF has been
   referring to so far with the term "CDNI Request Routing" is just
   another routing, signaling and forwarding problem in a long series of
   the telecommunication history. For example, one can draw a direct
   analogy between the IP/MPLS-TE framework and the CDN interconnection
   framework.

   In addition, this document recommends that the specification of ALL
   CDN interconnection interfaces in the scope of the CDNI IETF WG
   relies on the equivalent concept to IP prefix for CDN
   interconnection, named 'contentRequestScope'. This highly useful and
   powerful concept SHALL be used to simplify the specification of ALL
   CDN interconnection interfaces, as well as to ensure performance and
   scalability in CDN interconnection.

   All these proposals can be smoothly integrated in the WG drafts,
   especially [I-D.ietf-cdni-framework] and
   [I-D.ietf-cdni-requirements], as they essentially propose (useful)
   clarifications of the existing framework.

Status of this Memo

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

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

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

This Internet-Draft will expire on January 10, 2013.

Copyright Notice

Table of Contents

1. Introduction

   The present document proposes to clarify the CDNI Request Routing
   interface introduced in [I-D.ietf-cdni-framework] and [I-D.ietf-cdni-
   problem-statement], as well as the related terminology.

   In particular the present document proposes to split the CDNI Request
   Routing interface into two separate interfaces with clearer roles,
   named respectively CDNI Routing interface and CDNI Downstream
   Resource Identifier Signaling interface (CDNI DRIS interface).

   Section 2 presents the motivations for splitting the CDNI Request
   Routing interface. It also proposes to review and to expand the
   terminology about the terms iterative request routing and recursive
   request routing.

   Section 3 and Section 4 provide a short overview respectively of the
   CDNI Routing interface and of the CDNI DRIS interface. The detailed
   specifications of each of these interfaces will be made public
   shortly by the authors in dedicated IETF drafts.

   Section 5 provides an analogy with IP/MPLS-TE framework. This just
   aims at helping readers to understand the proposals and
   recommendations in this document.

   Section 6 concludes this draft with recommendations for the IETF CDNI
   WG. This includes recommendations about the CDNI WG charter, as well
   as about [I-D.ietf-cdni-framework] and [I-D.ietf-cdni-requirements].

   Appendix A proposes a text to replace Section 4.2 of
   [I-D.ietf-cdni-framework].

2. Motivations and Terminology Clarifications

   Section 2.1 and Section 2.2 present the motivations for splitting and
   clarifying the CDNI Request Routing interface.

   This document adopts the terminology described in [I-D.ietf-cdni-
   problem-statement] and [I-D.ietf-cdni-framework], except for the
   terms 'iterative request routing' and 'recursive request routing'.
   Section 2.3 proposes to review and to expand the terminology about
   these two terms.

   In addition this document extends this terminology with the terms
   'CDNI routing interface', 'CDNI DRIS interface' and
   'contentRequestScope' introduced and illustrated respectively in
   Sections 3, 4, and 5. It also introduces the term 'delivery

resource'. From the perspective of the uCDN, a delivery resource is a delivery node or a request routing system towards which a content request may be redirected: either a delivery node inside the uCDN, or the request routing system of a CDN different from the uCDN, or a delivery node within a CDN different from the upstream CDN.

2.1. The CDNI Request Routing interface should be split

One key message of the present document is that the term "CDNI Request Routing" is confusing as it makes a mix-up of two clearly different sets of processes and interfaces.

As described in [I-D.ietf-cdni-problem-statement], the CDNI Request Routing interface's role is twofold:

1. "To enable a downstream CDN to provide to the upstream CDN (static or dynamic) information (e.g., resources, footprint, load) to facilitate selection of the downstream CDN by the upstream CDN request routing system when processing subsequent content requests from User Agents."

2. "To allow the downstream CDN to control what the upstream Request Routing function should return to the User Agent in the redirection message."

These two functions are fundamentally different. Besides, depending on the considered use cases, described in [I-D.ietf-cdni-use-cases], they could be implemented differently (e.g., the first function could be implemented manually while the second function would be implemented with a network protocol, or vice versa, or they could rely on different network protocols, etc.).

This idea that the request routing interface comprises two parts has started to be visible in the latest version of the Section 4.2 in [I-D.ietf-cdni-framework]. We think that it is a good first step that requires clarifications. Indeed, [I-D.ietf-cdni-framework] mentions a 'synchronous part' that makes again a mix-up between operations involving communications with the user agent and operations only between the CDNs. And the latter ones can be purely asynchronous operations in some cases, as illustrated in Section 4 of the present memo.

Therefore, in order to simplify and accelerate the specification of the CDNI interfaces, the present document proposes to split the current CDNI Request Routing interface into two separate interfaces with clearer roles, further detailed in Section 3 and Section 4 respectively:

1. The CDNI Routing Interface

2. The CDNI Downstream Resource Identifier Signaling (DRIS)
    Interface.

2.2. The routing function of the CDNI Request Routing interface should
    be clarified

   Quoting [I-D.ietf-cdni-problem-statement], "The CDNI Request Routing
   interface enables a Request Routing function in an upstream CDN to
   query a Request Routing function in a downstream CDN to determine if
   the downstream CDN is able (and willing) to accept the delegated
   content request".

   The "ability" and the "willingness" of the downstream CDN to accept
   the delegated content request match two fully different processes in
   CDN interconnection. And the description of both these processes
   should be reviewed and clarified for the following reasons:

   o Regarding the former process, i.e. related to the "ability"
      ("enables a Request Routing function in an upstream CDN to query a
      Request Routing function in a downstream CDN to determine if the
      downstream CDN is able (...) to accept the delegated content
      request"), a routing protocol is used to achieve such a process,
      called routing process, in many other technologies and networks
      (IP, ATM, etc.). In all these technologies, it is the downstream
      entity that provides routing information to the upstream entity.
      The same approach should be applied to CDN interconnection. The
      reverse approach ("uCDN queries it downstream CDNs dCDN 1, dCDN 2,
      dCDN 3, and then it selects one of them based on their responses")
      would suffer from latency, signaling overhead and/or lack of
      reactivity to events that impact the ability of the downstream
      CDNs to accept delegated content requests.

   o Regarding the latter process, i.e. related to the "willingness"
      ("enables a Request Routing function in an upstream CDN to query a
      Request Routing function in a downstream CDN to determine if the
      downstream CDN (...) is willing (...)to accept the delegated
      content request"), it is to be noted that the relationship between
      a uCDN and a dCDN is always in the frame of a contractual
      agreement between the administrative entity owning the uCDN,
      acting as the customer, and the administrative entity owning the
      dCDN, acting as the service provider. Therefore, consider a case
      where

      * the uCDN has a content request to redirect which is in full
         conformance with the terms of this contractual agreement, and

   * the uCDN has selected this dCDN.

   As the uCDN knows, thanks to the routing information exchanged via
   the aforementioned routing process, that the dCDN is able to
   accept this content request, the uCDN MAY redirect the content
   request to the dCDN and the dCDN MUST accept the content request.
   Exchanging over the CDNI Request Routing interface information
   about the "willingness" of the dCDN to accept the content request
   is not relevant.

2.3. CDNI request redirection strategies

   The terms "Iterative CDNI request routing" and "Recursive CDNI
   request routing" are defined in [I-D.ietf-cdni-framework]. As
   explained above the term "CDNI request routing" is confusing,
   therefore, the present document proposes to use the terms "Iterative
   CDNI request redirection" and "Recursive CDNI request redirection"
   instead. We consider that these new terms are more appropriate than
   the definitions provided in [I-D.ietf-cdni-framework].

   The definition of "Recursive CDNI request routing" given in
   [I-D.ietf-cdni-framework] indicates "...that the Downstream CDN may
   elect to have the request redirected directly to a delivery node
   inside the Downstream CDN, to the Request-Routing System of the
   Downstream CDN, to another CDN, or to any other system that the
   Downstream CDN sees as fit for handling the redirected request."

   In order to clarify this point, and for the purpose of Section 3 to
   Section 6, the present document introduces the terms "full-recursive
   CDNI request redirection" and "semi-recursive CDNI request
   redirection".

   Full-recursive CDNI request redirection and semi-recursive CDNI
   request redirection are two subtypes of recursive CDNI request
   redirection. A recursive CDNI request redirection is either a full-
   recursive CDNI request redirection or a semi-recursive request
   redirection.

   "Full-recursive CDNI request redirection" refers to the case where
   the considered CDN redirects the content request directly to a
   delivery node of another CDN, as illustrated in Figure 4. It means
   the content request is redirected:

   o either directly to a delivery node inside the Downstream CDN,

   o or directly to a delivery node inside a downstream CDN of the
      downstream CDN,

o   or directly to a delivery node inside a downstream CDN of a
    downstream CDN of the downstream CDN,

o   etc. (anyway directly to a delivery node).

In contrast, "semi-recursive CDNI request redirection" refers to the
case, illustrated on Figure 5, where the considered CDN does not
directly redirect the content request to a delivery node. It means
the request is redirected:

o   either to the request routing system of the Downstream CDN that
    will deliver the content

o   or to the request routing system of another CDN that will deliver
    the content and that is a downstream CDN of the downstream CDN,

o   or to the request routing system of another CDN that will deliver
    the content and that is a downstream CDN of a downstream CDN of
    the downstream CDN,

o   etc. (anyway not directly to a delivery node).

Full-recursive redirection has the advantage over semi-recursive
redirection of being more transparent from the user agent's
perspective, but the disadvantage of the downstream CDN exposing more
of its internal structure (in particular, the addresses of delivery
nodes) to the upstream CDN (and possibly to the upstream CDN(s) of
its upstream CDN, etc.). By contrast, semi-recursive redirection
does not require dCDN to expose the addresses of its delivery nodes
to uCDN. The specifications for the CDNI interfaces elaborated
within the IETF CDNI WG MUST allow to support both these recursive
CDNI request redirection strategies.

3. Overview of the CDNI Routing interface

   The CDNI routing interface is dedicated to the CDNI routing process.
   The CDNI routing process consists in the advertisement of CDNI
   routing information from the downstream CDN to the upstream CDN.

   Note: The upstream CDN never provides the downstream CDN with any
   CDNI routing information.

   CDNI routing information details "capabilities" of the downstream CDN
   that the upstream CDN must take into account in its CDN selection
   process. The CDN selection process consists for the upstream CDN to
   select the CDN (either the upstream CDN or one of its downstream
   CDN(s)) that should handle the content request the upstream CDN

receives from the user agent.

Note: The term "capabilities" refer to the capacity of the downstream CDN to handle correctly content requests from user agents. It does not necessarily mean that the downstream CDN will satisfy (alone) all these content requests by delivering the requested content to the user agent. Indeed the downstream CDN may have its own downstream CDN(s) and may decide that some or all of these content requests will be handled by its own downstream CDN(s).

Figure 1 provides a short illustration of the CDNI routing interface. The upstream CDN on Figure 1 has two downstream CDNs: CDN1 and CDN2.

Via their respective CDNI routing interfaces established with the upstream CDN, CDN1 and CDN2 provide the upstream CDN with CDNI routing information about the capabilities that they offer to the upstream CDN in the frame of their respective CDNI agreements contracted with the upstream CDN.

The inner architecture of the upstream CDN is beyond the scope of the current charter of the IETF CDNI WG. Figure 1 provides a high level representation of a possible implementation just in order to ease understanding of the global picture.

Basically and typically, the CDNI routing information received from CDN1 and CDN2 will feed a CDNI routing table controlled by a routing module in the upstream CDN. And every time the upstream CDN receives a content request from a user agent (either a DNS request in case of DNS based request redirection, or an HTTP request in case of HTTP based request redirection, etc.), its CDN selection module (in charge of the CDN selection process) will consult this CDNI routing module to decide about which CDN will handle this request (the upstream CDN, CDN1, or CDN2). In the example on Figure 1, the upstream CDN selects CDN 2 for the considered content request.

```
+-----+   +-----------------------------------------------+
|     |   |                 Upstream CDN                  |
|     |   |                                               |
|     |   |   (1) Content request  +---------------------+ |
|     |   |--------------------------->|                 | |
|     |   |                        |   CDN selection     | |
|     |   |                  (2)   |   Module            | |
|     |   |           +--------->|                       | |
|     |   |           |          |   (Selection of CDN2) | |
|     |   |           |          +---------------------+ |
|User |   |           |                                 |
|Agent|   |           |                                 |
|     |   |           |                                 |
|     |   |           V                                 |
|     |   |   +-----------------------------------------+ |
|     |   |   |                                         | |
|     |   |   |       CDNI Routing Module               | |
|     |   |   |                                         | |
|     |   |   +-----------------------------------------+ |
|     |   |           ^                    ^            |
|     |   |           |                    |            |
|     |   +-----------|--------------------|------------+
|     |   |           |(A)  CDNI Routing  (A)|
|     |   |           |         Interfaces   |
|     |   +-------+-------+    +-------+-------+
|     |   |Downstream CDN1|    |Downstream CDN2|
+-----+   +---------------+    +---------------+
```
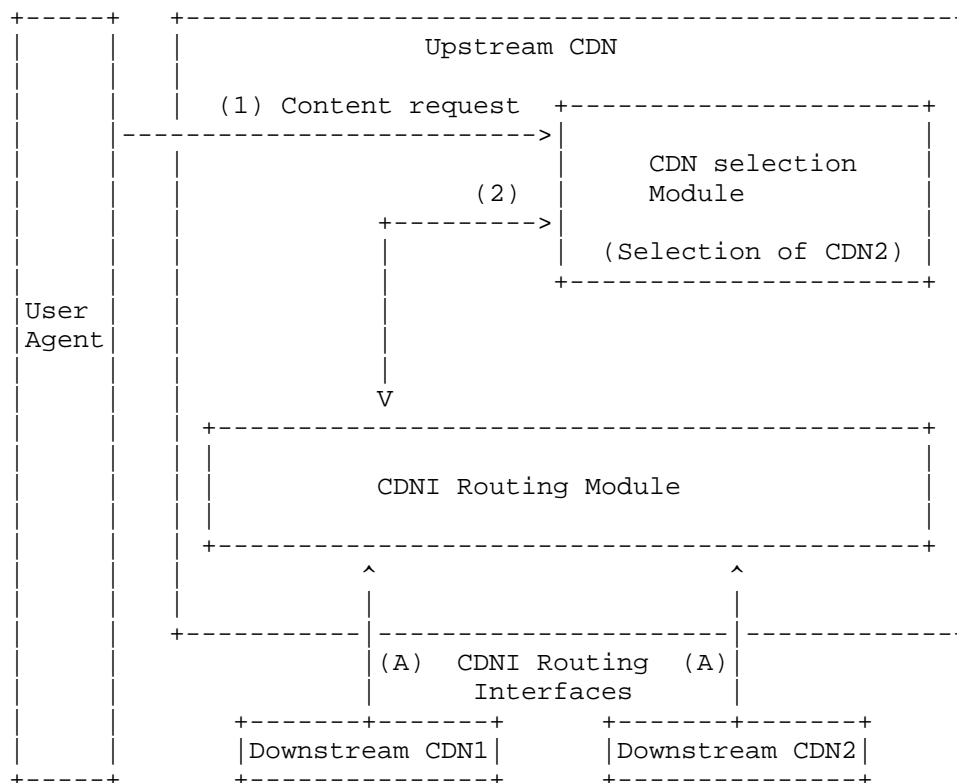
                     Figure 1: CDNI routing interface

   The operations shown on Figure 1 are as follows:

   1. A content request from a user agent (and/or from an intermediate
       local DNS in case it is a DNS request) arrives at the upstream
       CDN.

   2. Within the upstream CDN, the CDN selection module consults the
       routing module to decide which CDN will handle this content
       request (i.e. whether it will be the upstream CDN or one of its
       downstream CDNs). The interface between these internal modules
       is out of standardization scope.

   A. The operations achieved over the CDNI Routing interfaces aim at
   providing all information about the capabilities offered by the
   downstream CDNs to the upstream CDN in the frame of their respective
   CDNI agreements contracted with the upstream CDN.

The operations referenced with numeric indexes on Figure 1 ("1", "2")
show a time dependency: every time the upstream CDN receives a
content request ("1") its CDN selection process consults the routing
table ("2").

In contrast, Routing information is to be exchanged continuously,
with keep alive and update messages. There is no time dependency
between the content requests received by the upstream CDN and the
operations over the CDNI routing interfaces. By reference to the
terminology used in [I-D.ietf-cdni-framework], CDNI Routing
operations are asynchronous. The CDNI routing interfaces are
referenced with an alphabetic index ("A") on Figure 1 to reflect this
point.

4. The CDNI Downstream Resource Identifier Signaling (DRIS) Interface

4.1. Overview of the CDNI DRIS interface

   AFTER the CDN selection process is achieved by the upstream CDN for a
   given content request (see Section 3), the upstream CDN MUST generate
   a response redirecting that content request towards the selected
   delivery resource by using in this response an identifier of that
   selected delivery resource.

   The selected delivery resource can be:

   1. Either a delivery node within the upstream CDN

   This corresponds to the case where the CDN selection process of the
   upstream CDN selects the upstream CDN for that content request. In
   this case, the upstream CDN delivers the requested content to the
   client through one of its delivery nodes. The selected delivery
   resource is the upstream CDN's delivery node.

   2. Or the request routing system of a CDN different from the
   upstream CDN

   This corresponds to the case where the upstream CDN did select a
   downstream CDN for that content request, and where the CDNI request
   redirection strategy between the upstream CDN and this downstream CDN
   is based either on the iterative mode or on the semi-recursive mode.
   If the CDNI request redirection strategy between the uCDN and the
   dCDN is based on the iterative mode, the CDN to which the request
   routing system belongs is the downstream CDN selected by the upstream
   CDN for that content request (see Figure 3: Iterative CDNI request
   redirection). If the CDNI request redirection strategy between the
   uCDN and the dCDN is based on the semi-recursive mode, the CDN to

which that request routing system belongs can possibly be

o the downstream CDN selected by the upstream CDN for that content
   request, or

o a downstream CDN of that downstream CDN, or

o a downstream CDN of a downstream CDN of that downstream CDN,

o etc.,

due to the recursive nature of this mode (see Figure 5).

3. Or a delivery node within a CDN different from the upstream CDN

This corresponds to the case where the upstream CDN did select a
downstream CDN for that content request, and where the CDNI request
redirection strategy between the upstream CDN and this downstream CDN
is based on the full-recursive mode.

The CDN to which the selected delivery node belongs can possibly be a
delivery node within the downstream CDN selected by the upstream CDN
for that content request, or a delivery node within a downstream CDN
of that downstream CDN, or a delivery node within a downstream CDN of
a downstream CDN of that downstream CDN, etc., due to the recursive
nature of this mode (see Figure 4: Full-recursive CDNI request
redirection).

In the cases "2." and "3.", the downstream CDN MUST provide the
identifier of the selected delivery resource to the upstream CDN.
The downstream CDN uses the CDNI DRIS interface to provide the
upstream CDN with this identifier called "Downstream Resource
Identifier" (DRI).

Note. In the recursive modes (full-recursive and semi-recursive) the
upstream CDN does not need to know if the selected delivery resource
belongs to the downstream CDN that the uCDN selected for that content
request or to another CDN (i.e. to one downstream CDN of this
downstream CDN, or to one downstream CDN of one downstream CDN of
this downstream CDN, etc.). The upstream CDN just needs to get from
the selected downstream CDN the "Downstream Resource Identifier"
(DRI) of the selected delivery resource, so as to generate with this
identifier a response redirecting that content request towards that
selected delivery resource.

```
  +-----+              +-------------------------------------------------+
  |     |  |(1) Content |                                                 |
  |     |  |  request   |              Upstream CDN                       |
  |     |  |------------------+                                           |
  |     |  |            |  |                                              |
  |     |  |            |  |  (5) Content request response                |
  |     |  |<--------------|-------------------------------+               |
  |     |  |            |  |                               |              |
  |     |  |            |  +-V-------------+    +--------+---------+ |
  |     |  |            |  | CDN selection |    | Content request  | |
  |     |  |            |  | Module        |(3) | response          | |
  |     |  |            |  |               |------->| generation   | |
  |     |  |            |  | (Selection    |Internal | Module      | |
  |User |  |            |  |  of CDN2)     |Interface|              | |
  |Agent|  |            |  +---------------+    +------------------+ |
  |     |  |            |         ^                      ^          |
  |     |  |            |      (2)|Internal       (4)|Internal       |
  |     |  |            |         |interface         |interface      |
  |     |  |            |         V                      V          |
  |     |  |            |  +-------------+     +-----------+        |
  |     |  |            |  |CDNI Routing |     |DRI        |        |
  |     |  |            |  |Module       |     |Module     |        |
  |     |  |            |  +-------------+     +-----------+        |
  |     |  |            |        ^   ^             ^   ^            |
  |     |  |  +----------|---|-----------------|---|-----------+
  |     |  |  CDNI Routing|(A)+--------------+  |(B)|CDNI DRIS
  |     |  |  Interfaces  |    +------------|--+    |Interfaces
  |     |  |              |    |            |       |
  |     |  |  +------+-----v--+    +-+------v------+
  |     |  |  |Downstream CDN1|    |Downstream CDN2|
  +-----+  |  +--------------+    +--------------+
```
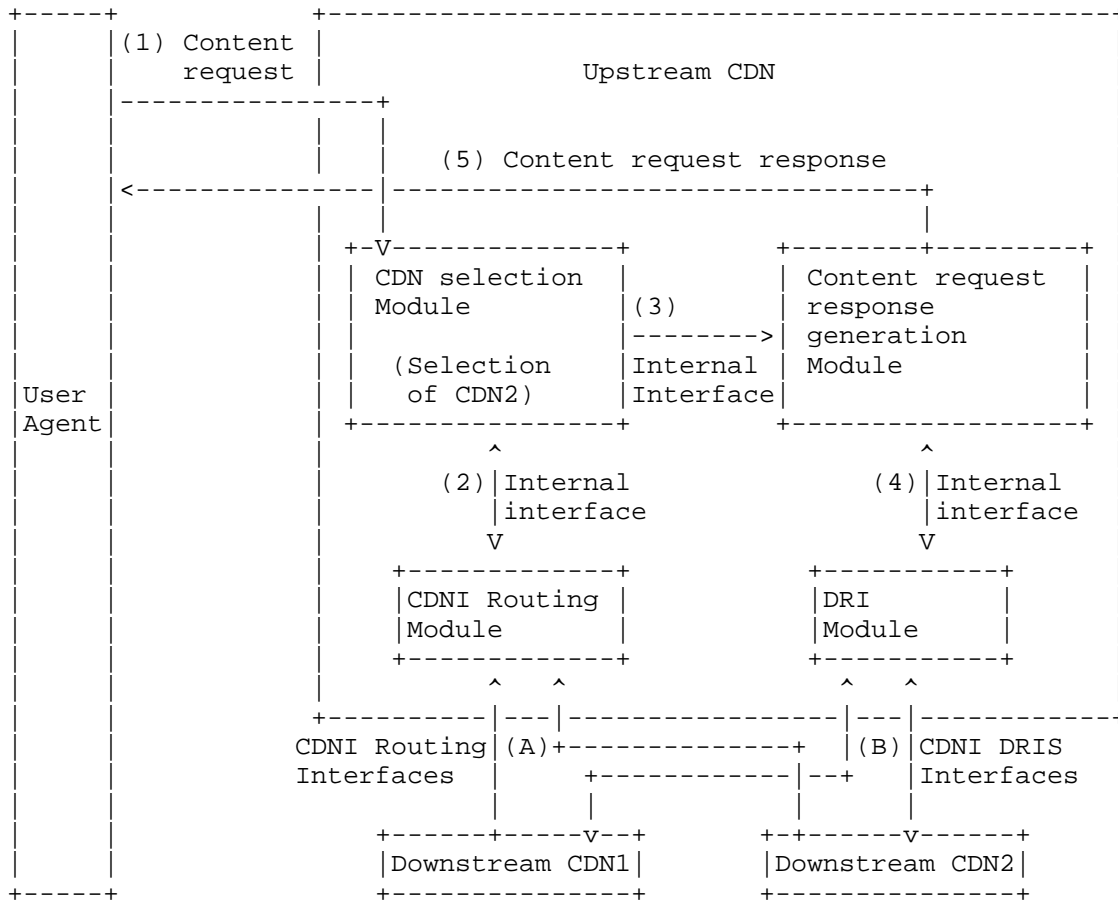
               Figure 2: CDNI DRIS interface

   Figure 2 is an extension of Figure 1. Again, the inner architecture
   of the CDN is beyond the scope of the IETF CDNI WG charter; Figure 2
   provides a high level representation of a possible implementation
   just in order to ease the understanding of the global picture.

   The operations shown on Figure 2 are as follows:

   1. A content request from a user agent (and/or from an intermediate
       local DNS in case it is a DNS request) arrives at the upstream
       CDN.

   2. Within the upstream CDN, the CDN selection Module consults the
       routing module to decide which CDN will handle this content

request (i.e. whether it will be the upstream CDN or one of its
downstream CDNs). The interface between these internal modules
is out of standardization scope. Let us consider that the CDN
selection module selected the downstream CDN2 for that content
request (so as to continue with the illustrating example
introduced in Section 3).

3. Once this CDN selection is achieved, the module in charge of
    generating the response to the content request ('Content request
    response generation Module' on Figure 2) is called. The
    interface between these internal modules is out of
    standardization scope. In order to generate the response, the
    Content request response generation module needs to get the
    identifier of the selected delivery resource (DRI) to where the
    content request must be redirected.

4. The Content request response generation module consults the DRI
    module to get this adequate DRI. The interface between these
    internal modules is out of standardization scope. The DRI module
    is in charge of getting and storing the DRI(s) provided by each
    downstream CDN over their CDNI DRIS interfaces with the upstream
    CDN.

5. Once the Content request response generation module has this
    adequate DRI, it generates the response to the content request so
    as to the request be redirected towards the selected delivery
    resource.

A. The operations achieved over the CDNI Routing interfaces aim at
providing all information about the capabilities offered by the
downstream CDNs to the upstream CDN in the frame of their respective
CDNI agreements contracted with the upstream CDN.

B. The operations achieved over the CDNI DRIS Interface with the
downstream CDN2 aim at providing the upstream CDN with the DRI, i.e.
the identifier of the selected delivery resource.

The operations referenced with numeric indexes on Figure 2 ("1", "2",
"3", "4", "5") show a strict time dependency. In particular, it is
to be highlighted that the CDN selection process of the upstream CDN
does not take the DRI information into account. This identifier is
used by the content request response generation module only after the
CDN selection process is achieved.

As explained in Section 3, the CDNI routing interfaces are referenced
with an alphabetic index ("A") to indicate that the CDNI routing
information are exchanged asynchronously and continuously, with keep
alive and update messages.

The CDNI DRIS interfaces are also referenced with an alphabetic index
("B") to indicate there is not always a systematic time dependency
between the content requests received by the upstream CDN and the
operations occurring over the CDNI DRIS interfaces. This depends,
among others, on the CDNI request redirection strategy enforced
between the CDNs (recursive or iterative). This means that the
specification of the CDNI DRIS interface MUST enable synchronous and
asynchronous operations, as illustrated in the next Section 4.2.

4.2. Overview of CDNI DRIS operations

   This section gives an overview of operations over the CDNI DRIS
   interfaces for each CDNI request redirection strategy (iterative,
   full-recursive and semi-recursive).

   All the figures present the same network topology with three cascaded
   CDNs so as to show very clearly the difference between the full-
   recursive and semi-recursive modes. The case where only two CDNs are
   involved can be straightforwardly deduced. In addition, the CDNI
   request redirection strategy applied between the first CDN and the
   second CDN (e.g., iterative) could differ from the strategy applied
   between the second CDN and the third CDN (e.g., semi- recursive).
   The three figures are for illustration purpose only; they do not aim
   at reflecting exhaustively the full flexibility that the CDN service
   providers have when choosing their CDNI request redirection
   strategies.

   The same situation and content request are considered in all the
   three figures:

   o There is one CDNI agreement and one CDN interconnection between
      CDN-a and CDN-b for the considered request. CDN-b is a downstream
      CDN of CDN-a. And there is one CDNI DRIS interface between CDN-a
      and CDN-b, bootstrapped with configuration information exchanged
      over the CDNI control interface during the initial CDN
      interconnection activation process (see
      [I-D.ietf-cdni-framework]).

   o In the same way there is one CDNI agreement, one CDN
      interconnection and one CDNI DRIS interface between CDN-b and
      CDN-c. CDN-c is a downstream CDN of CDN-b for the considered
      request.

   o CDN-a receives a content request and decides CDN-b will handle it.
      CDN-b decides the content request will be handled by CDN-c. CDN-c
      delivers the content to the user agent from one of its delivery
      nodes (cache hit).

4.2.1. Case of iterative CDNI request redirection

   Figure 3 presents the case where the iterative CDNI request
   redirection mode is applied between CDN-a and CDN-b, and between
   CDN-b and CDN-c.

```
+-----+                                          +-----------------+
|     |                                          |CDN-a            |
|     |  1                                       | (uCDN for CDN-b) |
|     |  ------------------------------------->|                 |
|     |  2                                       |                 |
|     |  <-------------------------------------|                 |
|     |      +-----------------+                 |  +--------+     |
|     |      |CDN-b            |                 |  |        |     |
|     |  3   | (uCDN for CDN-c) |                 |  |DRI     |     |
|     |  ----->|                 |   +------------->|Module  |     |
|     |  4   |     +--------+    | 0 | |            |  |        |     |
|     |  <-----|     |        |    |-------+ |            |  +--------+     |
|     |      |     |        |    |       |            +-----------------+
|User |      |     |DRI     |    |       |
|Agent|      |     |Module  |    |       |
|     |      |     |        |    |   | 0
|     |      |     |        |    |<-------+
|     |      |     +--------+    |   | |
|     |      +-----------------+  |   | |           +-----------------+
|     |                           |   |           |CDN-c            |
|     |                           |   |           |  +--------+     |
|     |                           |   |           |  |        |     |
|     |                      +-------------| |DRI     |     |
|     |                           |           |  |Module  |     |
|     |  5                        |           |  |        |     |
|     |  ------------------------------------->|  +--------+     |
|     |  6                        |           |                 |
|     |  <-------------------------------------|                 |
|     |  7                        |           |  +--------+     |
|     |  ------------------------------------->|  |        |     |
|     |  8                        |           |  |Delivery|     |
|     |  <-------------------------------------|Node   |     |
|     |                           |           |  |        |     |
|     |                           |           |  +--------+     |
+-----+                                          +-----------------+
```
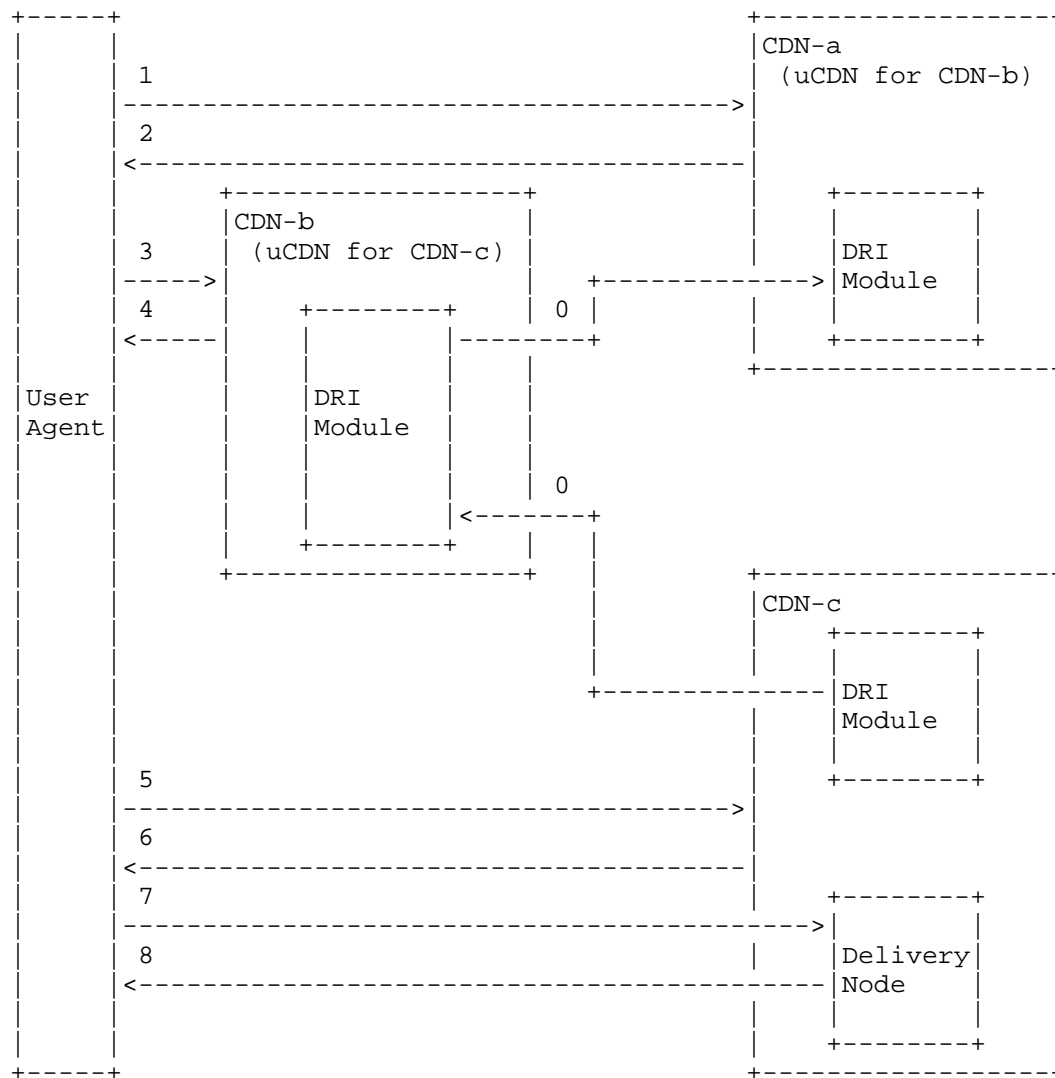
             Figure 3: Iterative CDNI request redirection

The operations shown in Figure 3 are as follows:

0. CDN-a gets from CDN-b, over the CDNI DRIS interface between CDN-a
and CDN-b, a DRI valid for any content request within their mutual
CDNI agreement, and possibly before any user agent content request is
received by CDN-a.

In the iterative case, the CDNI DRIS interface may indeed provide the
upstream CDN with a unique DRI valid in the frame of their mutual
CDNI agreement for any content request received by the upstream CDN.

Basically, it means that the downstream CDN informs the upstream CDN
that, when the upstream CDN decides to redirect a content request to
this downstream CDN in the frame of their mutual CDNI agreement, the
upstream CDN just has to use systematically that DRI to forge the
response sent by the upstream CDN to redirect the content request to
that downstream CDN.

For example, in case the CDNI request redirection mechanism is based
on HTTP 302 redirect messages, this DRI includes a distinguished CDN-
domain, which is to be "stacked" in front of the original URL to
construct the new URL to redirect the content request to the request
routing system of the downstream CDN, as detailed in
[I-D.ietf-cdni-framework].

This static and unique DRI for a CDNI agreement set between CDN-a and
CDN-b can be exchanged even before the first content request received
by CDN-a. This is an example of asynchronous operation over the CDNI
DRIS interface. And this is one case where it may be relevant to
implement the CDNI DRIS interface "manually"; for example the DRI can
be exchanged on the phone or by email simply and only once for the
whole duration of the CDNI agreement between CDN-a and CDN-b.

We describe the other operations of Figure 3 below.

0. The same way, CDN-b gets from CDN-c, over the CDNI DRIS interface
between CDN-b and CDN-c, a DRI valid for any content request within
their mutualCDNI agreement, and possibly before any user agent
content request is received by CDN-b.

1. The considered content request, sent by a user agent (and/or by
an intermediate local DNS in case it is a DNS request), arrives at
CDN-a.

2. CDN-a decides this content request is to be best handled by
CDN-b. Therefore, CDN-a sends a response to the content request,
forged with the DRI provided by CDN-b.

3. A request is thus sent by the user agent (and/or by an
intermediate local DNS in case it is a DNS request) to CDN-b.

4. CDN-b decides this content request is to be best handled by
CDN-c. Thus, CDN-b sends a response to the content request forged
with the DRI provided by CDN-c.

5. A request is sent by the user agent (and/or by an intermediate
local DNS in case it is a DNS request) to CDN-c.

6. CDN-c decides it will handle the request, i.e. it will be the one
to deliver the content to the user agent. CDN-c selects one of its
delivery nodes to handle that content request. CDN-c generates a
response redirecting the content request to this delivery node.

7. The user agent emits a content request to the selected delivery
node.

8. The selected delivery node from CDN-c delivers the requested
content to the user agent.

4.2.2. Case of recursive CDNI request redirection

Figure 4 presents the case where the full-recursive CDNI request
redirection mode is applied between CDN-a and CDN-b, and between
CDN-b and CDN-c.

```
   +-----+                                   +-----------------+
   |     |                                   |CDN-a            |
   |     |    1                              | (uCDN for CDN-b) |
   |     |  |--------------------------------->|               |
   |     |    6                              |                 |
   |     |  |<--------------------------------|               |
   |     |      +-----------------+           |  +--------+    |
   |     |      |CDN-b            | +----------|  |        |    |
   |     |      | (uCDN for CDN-c)| |          |  |DRI     |    |
   |     |      |                 | | +-------->|Module  |    |
   |     |      |     +--------+  | 2 |        |  |        |    |
   |     |      |     |        |  |<-------+   |  +--------+    |
   |     |      |     |        |  | 5     |    +-----------------+
   |User |      |     |DRI     |  |-------------+
   |Agent|      |     |Module  |  | 3         |
   |     |      |     |        |  |-------------+
   |     |      |     |        |  | 4         |
   |     |      |     |        |  |<-------+  |
   |     |      |     +--------+  |       |   |
   |     |      +-----------------+       |   |   +-----------------+
   |     |                                |   |   |CDN-c            |
   |     |                                |   |   |  +--------+    |
   |     |                                |   +-------->|        |    |
   |     |                                |       |DRI     |    |
   |     |                                +-------------|Module  |    |
   |     |                                        |  |        |    |
   |     |                                        |  +--------+    |
   |     |    7                                   |  +--------+    |
   |     |  |----------------------------------------->|        |    |
   |     |    8                                   |  |Delivery|    |
   |     |  |<----------------------------------------|Node    |    |
   |     |                                        |  |        |    |
   |     |                                        |  +--------+    |
   +-----+                                        +-----------------+
```
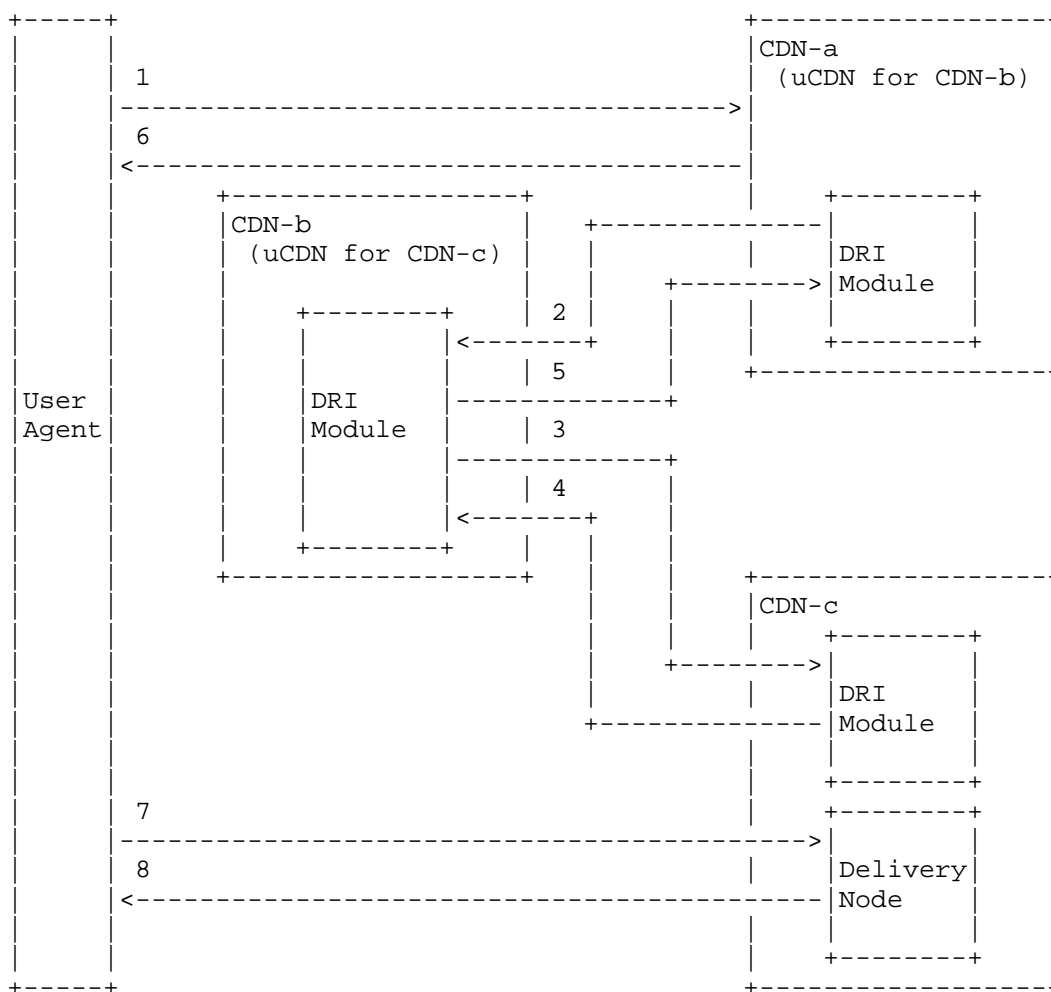
              Figure 4: Full-recursive CDNI request redirection

     The operations shown in Figure 4 are as follows:

     1. The considered content request, sent by a user agent (and/or by
     an intermediate local DNS in case it is a DNS request), arrives at
     CDN-a.

     2. CDN-a decides this content request is to be best handled by
     CDN-b. CDN-a sends a request to CDN-b over the CDNI DRIS interface
     set up between CDN-a and CDN-b. The request includes all information
     about the content request CDN-b needs in order to select the most

appropriate delivery resource and to response to CDN-a with the corresponding adequate DRI.

3. CDN-b decides this content request is to be best handled by CDN-c. CDN-b sends a request to CDN-c over the CDNI DRIS interface set up between CDN-b and CDN-c. The request includes all information about the content request CDN-c needs in order to select the most appropriate delivery resource and to response to CDN-b with the corresponding adequate DRI.

4. CDN-c decides it will handle the content request, i.e. it will be the one to deliver the content to the user agent. CDN-c selects one of its delivery nodes to handle that content request. CDN-c sends a response to CDN-b, over the CDNI DRIS interface between CDN-b and CDN-c, with a DRI corresponding to that delivery node.

5. CDN-b sends a response to CDN-a, over the CDNI DRIS interface between CDN-a and CDN-b, with a DRI corresponding to the selected delivery node from CDN-c. This DRI is based on the DRI provided by CDN-c. Details about the manipulations achieved by CDN-b over the DRI received from CDN-c to generate the DRI sent to CDN-a are to be provided in the dedicated specification of the CDNI DRIS interface.

6. CDN-a sends a response to the content request, forged with the DRI provided by CDN-b.

7. The user agent emits a content request directly to the selected delivery node from CDN-c.

8. The selected delivery node from CDN-c delivers the requested content to the user agent.

The Figure 5 below presents the case where the semi-recursive CDNI request redirection mode is applied between CDN-a and CDN-b, and between CDN-b and CDN-c.
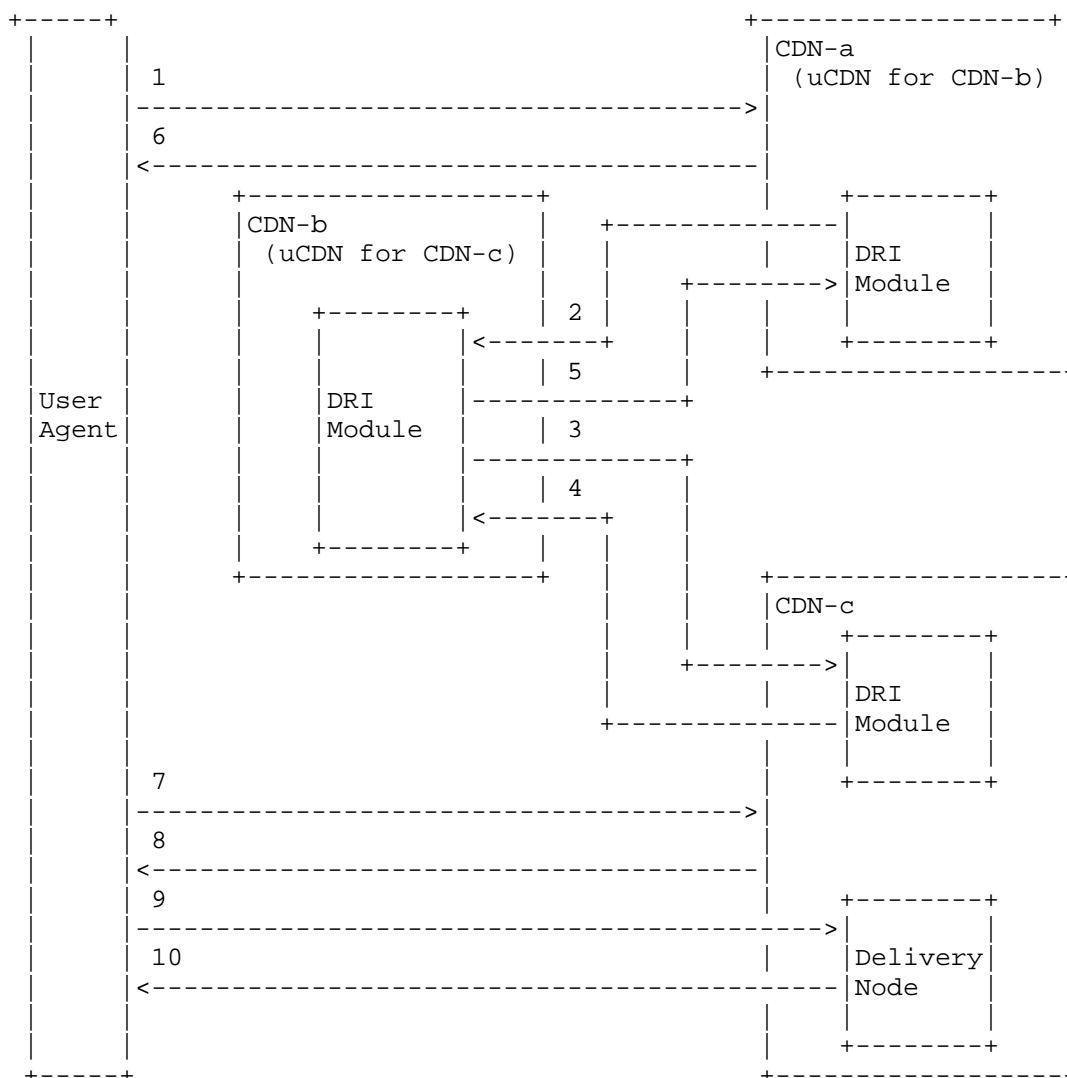
```
 +-----+                                   +-----------------+
 |     |                                   |CDN-a            |
 |     |  1                                | (uCDN for CDN-b) |
 |     |  |--------------------------------------->|         |
 |     |  6                                |         |
 |     |  |<---------------------------------------|         |
 |     |     +-----------------+           |    +--------+   |
 |     |     |CDN-b            |    +--------------|        |   |
 |     |     | (uCDN for CDN-c) |   |     |  |DRI    |   |
 |     |     |                 |   |     +------->|Module |   |
 |     |        +--------+  |  2 |    |    |        |   |
 |     |        |        |  |<-------+  |    |    +--------+   |
 |     |        |        |  | 5  |    |    +-----------------+
 |User |        |DRI     |  |------------+    |
 |Agent|        |Module  |  | 3         |
 |     |        |        |  |------------+    |
 |     |        |        |  | 4  |    |
 |     |        |        |  |<-------+  |
 |     |        +--------+  |    |  |
 |     |     +-----------------+    |  |
 |     |                          |  |    +-----------------+
 |     |                          |  |    |CDN-c            |
 |     |                          |  |    |   +--------+   |
 |     |                        +-------->|         |   |
 |     |                          |  |    |DRI     |   |
 |     |                        +--------------|Module  |   |
 |     |                               |    |        |   |
 |     |  7                            |    +--------+   |
 |     |  |--------------------------------------->|         |
 |     |  8                            |         |
 |     |  |<---------------------------------------|         |
 |     |  9                            |    +--------+   |
 |     |  |--------------------------------------->|         |
 |     |  10                           |    |Delivery|   |
 |     |  |<--------------------------------------- |Node    |   |
 |     |                               |    |        |   |
 |     |                               |    +--------+   |
 +-----+                               +-----------------+
```

        Figure 5: Semi-recursive CDNI request redirection

   The operations shown in Figure 5 are as follows:

   1. The considered content request, sent by a user agent (and/or by
   an intermediate local DNS in case it is a DNS request), arrives at
   CDN-a.

   2. CDN-a decides this content request is to be best handled by

CDN-b. CDN-a sends a request to CDN-b over the CDNI DRIS interface
set up between CDN-a and CDN-b. The request includes all information
about the content request CDN-b needs in order to select the most
appropriate delivery resource and to response to CDN-a with the
corresponding adequate DRI.

3. CDN-b decides this content request is to be best handled by
CDN-c. CDN-b sends a request to CDN-c over the CDNI DRIS interface
set up between CDN-b and CDN-c. The request includes all information
about the content request CDN-c needs in order to select the most
appropriate delivery resource and to response to CDN-b with the
corresponding adequate DRI.

4. CDN-c decides it will handle the content request, i.e. it will be
the one to deliver the content to the user agent. CDN-c sends a
response to CDN-b, over the CDNI DRIS interface between CDN-b and
CDN-c, with a DRI corresponding to its request routing system.

5. CDN-b sends a response to CDN-a, over the CDNI DRIS interface
between CDN-a and CDN-b, with a DRI corresponding to the selected
delivery resource (i.e. to the request routing system of CDN-c).
This DRI is based on the DRI provided by CDN-c. Details about the
manipulations achieved by CDN-b over the DRI received from CDN-c to
generate the DRI sent to CDN-a are to be provided in the dedicated
specification of the CDNI DRIS interface.

6. CDN-a sends a response to the content request, forged with the
DRI provided by CDN-b.

7. The content request is redirected by the user agent (and/or by an
intermediate local DNS in case it is a DNS request) to the selected
delivery resource, i.e. to CDN-c.

8. CDN-c decides it will handle the content request, i.e. it will be
the one to deliver the content to the user agent. CDN-c selects one
of its delivery nodes to handle that content request. CDN-c
generates a response redirecting the content request to this delivery
node.

9. The user agent emits a content request to the selected delivery
node.

10. The selected delivery node from CDN-c delivers the requested
content to the user agent.

As shown on Figure 4 and 5, when the recursive mode is applied, the
upstream CDN may have to send a request over a CDNI DRIS interface
every time it receives a content request. This is an example of

synchronous operations over the CDNI DRIS interface. Having one CDNI
DRIS request per content request may lead to scalability issues, even
if the CDNI DRIS interface is implemented with a network protocol
rather than "manually". Yet these issues can be solved. This is to
be documented in the dedicated specification of the CDNI DRIS
interface.

4.3. Key points to note about the CDNI DRIS interface

   To conclude this section, the key points to note about the CDNI DRIS
   interface are the following:

   o The CDNI DRIS interface MUST always exist between two
      interconnected CDNs. Whatever the CDNI request redirection
      strategy applied (recursive or iterative), the CDNI DRIS interface
      exists, even if it is implemented "manually" in some cases, as
      suggested in the example of Figure 3 above.

   o Moreover the CDNI DRIS interface MUST be unique from a functional
      perspective. Another way to say this is that, from a functional
      perspective, this is the exact same CDNI DRIS interface that is
      used, and the exact same type of information that are exchanged
      over this interface, whatever the CDNI request redirection
      strategy applied (recursive or iterative).

   o The specification of the CDNI DRIS interface MUST enable
      synchronous and asynchronous CDNI DRIS operations: operations
      possibly triggered by the CDN selection process for a given
      content request (as shown on Figure 4 and 5), as well as
      operations achieved independently of any content request (as shown
      on Figure 3).

   o The specification of the CDNI DRIS interface MUST enable
      operations initiated by uCDN (as shown on Figure 4 and 5) or by
      dCDN (as shown on Figure 3).

   o The specification of the CDNI DRIS interface MUST be compatible
      with "automatic" (i.e. "with a specific network protocol") and
      "manual" implementations, as discussed in Section 4.2. The
      specification of the CDNI DRIS interface SHALL include a
      description for each of these implementation options.

   o The CDNI routing and CDNI DRIS interfaces MUST be independent.

5. CDNI request routing is just another routing and signaling problem

   This part of the CDN interconnection framework the IETF has been
   referring to so far with the term "CDNI Request Routing" is just
   another routing, signaling and forwarding problem in a long series of
   the telecommunication history.

   For example one can draw an analogy with the IP/MPLS-TE framework,
   notably, as shown on Figure 6 and below, between:

   o the CDNI routing interface and any IP routing protocol/interface
      (such as BGP/IGP IP routing protocols),

   o the CDNI DRIS interface and a signaling protocol such as RSVP-TE.

   It is to be noted that such an analogy must be considered with care
   because the context and objectives of CDN interconnection are
   different from IP routing, signaling and forwarding. In particular,
   the type of information exchanged via the CDNI DRIS interface is
   totally different from the information exchanged by protocols like
   RSVP-TE in IP/MPLS networks. Anyway this analogy may ease to
   understand the role of the CDNI DRIS interface in the CDNI framework.

   Another key point to note in this analogy is that IP prefix is one of
   the most important concepts in IP networks. It is used in most IP
   processes (IP routing, packet filtering, MPLS-TE signaling, etc.).
   It has been a highly useful and powerful concept to simplify the
   specification of TCP/IP protocols as well as to ensure performance
   and scalability of IP networks. For example exchanging IP routing
   information per IP prefix has been a MUST to ensure IP routing
   performance and scalability.

   The equivalent concept to IP prefix has the same importance in the
   CDN interconnection framework. The concept is named
   "contentRequestScope" in Figure 6. A contentRequestScope designates
   a class of content requests sharing common properties (e.g., "all the
   HTTP requests", or "all HTTP requests issued by French users", or
   "all RTSP and HTTP requests with signed URL", etc.).

   In the same way as IP prefix is a key concept for specifying TCP/IP
   protocols, "contentRequestScope" is THE main concept for the CDN
   interconnection framework. This highly useful and powerful concept
   SHALL be used to simplify the specification of ALL CDN
   interconnection interfaces, as well as to ensure performance and
   scalability in CDN interconnection.

IP/MPLS-TE concepts	CDN Interconnection concepts
IP packet Forwarding	Content request redirection
IP prefix (correspond to a class of IP addresses sharing the same prefix)	ContentRequestScope (correspond to a class of content requests sharing common properties)
Ingress Provider Edge Router (ILSR)	upstream CDN (uCDN)
Egress Provider Edge Router (ELSR)	Delivery resource (delivery node or downstream CDN)
IP routing interfaces (running IGP/BGP IP routing protocols)	CDNI routing interface
IP routing information (IP prefix <-> Next Hop)	CDNI routing information (ContentRequestScope <-> dCDN)
IP FEC table construction (IP prefix <-> ELSR)	Redirection table construction (ContentRequestScope <-> delivery resource)
Signaling information provided by the RSVP-TE protocol (MPLS label <-> ELSR)	Signaling information provided by the CDNI DRIS interface (DRI <-> delivery resource)
To encapsulate an IP packet in an MPLS frame (using an MPLS Label)	To generate a redirected content request (using a DRI)
To forward a packet out of the nominal route based on IGP and BGP routing information, thanks to MPLS Traffic Engineering technologies	To redirect a content request out of the authoritative CDN, to another CDN, thanks to CDN interconnection technologies
To forward an IP packet encapsulated in an MPLS frame via - or to - the destination ELSR thanks to the MPLS header of the MPLS frame	To redirect a content request to to the selected downstream CDN - or to the selected delivery node - thanks to the Downtream Resource Identifier (DRI) used to forge the response to the initial content request

Figure 6: Analogy between IP/MPLS-TE and CDNI concepts

CDNI Routing interfaces exchanges CDNI routing information per
contentRequestScope, i.e. per class of content requests sharing
common properties.

Analogically: IP routing protocols like IGP/BGP exchange routing
information per IP prefix, i.e. per class of IP addresses sharing
common properties.

The upstream CDN builds its redirection table with information
provided by the CDNI Routing interface and the CDNI DRIS interface.

Analogically: The Ingress Provider Edge Router (ILSR) builds its
forwarding table with information provided by the IP routing and
MPLS-TE signaling protocols.

When receiving a content request, which has possibly been already
redirected by some upstream CDNs, a CDN consults its forwarding table
to generate a response using the adequate DRI, provided by the CDNI
DRIS interface, which redirects the content request to the selected
delivery resource (which is a CDN or a node downstream).

Analogically: When receiving an IP packet, possibly encapsulated in
an MPLS frame, an Ingress Provider Edge Router (ILSR) consults its
forwarding table to manipulate it adequately. This manipulation may
include to encapsulate it in an MPLS frame, based on the signaling
information provided by the RSVP-TE protocol, which makes it being
then forwarded to the right Egress Provider Edge router (ELSR).

6. Conclusion and recommendations

A first key message of this document is that the term "CDNI request
routing" is confusing as it makes a mix-up of clearly different set
of processes and interfaces.

A second key message of this document is that this part of the CDN
interconnection framework the IETF has been referring to so far with
the term "CDNI Request Routing" is just another routing, signaling
and forwarding problem in a long series of the telecommunication
history. For example, one can draw a direct analogy between the IP/
MPLS-TE framework and the CDN interconnection framework.

Thus one can directly be inspired by the best practices and results
of the efforts invested over years, even decades, to develop
technologies in the past, such as IP/MPLS-TE, to specify adequately
CDN interconnection interfaces.

The proposals from the current document can be smoothly integrated in the WG drafts, especially [I-D.ietf-cdni-framework] and [I-D.ietf-cdni-requirements], as they essentially propose (useful) clarifications of the existing framework.

We recommend to review [I-D.ietf-cdni-framework], at least its section 4.2. Appendix A provides a proposal to replace the current Section 4.2 of [I-D.ietf-cdni-framework].

We recommend to review [I-D.ietf-cdni-requirements], to clearly distinguish the requirements specific to the CDNI routing interface from the requirements specific to the CDNI DRIS interface.

We also recommend to review the charter of the CDNI WG, in particular to replace the following objective:

o "WG Creation + 18 months: Submit specification of the CDNI Request Routing protocol to IESG as Proposed Standard"

by the two following objectives:

o "WG Creation + 18 months: Submit specification of the CDNI Routing interface to IESG as Proposed Standard"

o "WG Creation + 18 months: Submit specification of the CDNI Downstream Resource Identifier Signaling (DRIS) Interface to IESG as Proposed Standard"

Last but not least, we recommend that the specification of ALL CDN interconnection interfaces, including the CDNI routing interface and the CDNI DRIS interface, rely on the "contentRequestScope" concept, i.e. the equivalent concept to IP prefix for CDN interconnection. In the same way as IP prefix is a key concept for specifying TCP/IP protocols, "contentRequestScope" is THE main concept for the CDN interconnection framework. This highly useful and powerful concept SHALL be used to simplify the specification of ALL CDN interconnection interfaces, as well as to ensure performance and scalability in CDN interconnection.

7. Acknowledgments

The authors would like to thank Chris Hawinkel, Larry Peterson, Yvan Massot, Ali Gouta, Emile Stephan, and Patrick Fleming for their inputs and comments.

They also thank the contributors of the EU FP7 OCEAN project in the frame of which these proposals have been elaborated.

The work leading to these results has received funding from the
European Union's Seventh Framework Programme (FP7/2007-2013) in OCEAN
project (FP7-ICT-248775; http://www.ict-ocean.eu/).

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

Those are discussed in [I-D.ietf-cdni-problem-statement].

10. Informative References

[I-D.ietf-cdni-framework]
          Peterson, L. and B. Davie, "Framework for CDN
          Interconnection", draft-ietf-cdni-framework-00 (work in
          progress), April 2012.

[I-D.ietf-cdni-problem-statement]
          Niven-Jenkins, B., Faucheur, F., and N. Bitar, "Content
          Distribution Network Interconnection (CDNI) Problem
          Statement", draft-ietf-cdni-problem-statement-08 (work in
          progress), June 2012.

[I-D.ietf-cdni-requirements]
          Leung, K. and Y. Lee, "Content Distribution Network
          Interconnection (CDNI) Requirements",
          draft-ietf-cdni-requirements-03 (work in progress),
          June 2012.

[I-D.ietf-cdni-use-cases]
          Bertrand, G., Emile, S., Burbridge, T., Eardley, P., Ma,
          K., and G. Watson, "Use Cases for Content Delivery Network
          Interconnection", draft-ietf-cdni-use-cases-08 (work in
          progress), June 2012.

Appendix A. Proposal for Section 4.2 of CDNI framework draft

   "4.2. Request Routing Interface

The request routing interface is not a single interface, it
corresponds to two separate interfaces with clear roles:

1) The CDNI Routing Interface

The CDNI routing interface is dedicated to the CDNI routing process.
The CDNI routing process consists in the advertisement of CDNI
routing information from the downstream CDN to the upstream CDN (the
upstream CDN never provides the downstream CDN with any CDNI routing
information). CDNI routing information details capabilities of the
downstream CDN that the upstream CDN must take into account in its
dCDN selection process.

2) The CDNI Downstream resource Identifier Signaling (DRIS) Interface

AFTER the CDN selection process is achieved by the upstream CDN for a
given content request, the upstream CDN MUST generate a response
redirecting that request towards the selected delivery resource by
inserting in this response an identifier of that selected delivery
resource.

This selected delivery resource can be:

1. a delivery node within the upstream CDN,

2. a downstream CDN (i.e. the request routing system of that
    downstream CDN), or

3. a delivery node within a downstream CDN.

In the cases "2." and "3." , the identifier of that selected delivery
resource MUST be provided by the downstream CDN to the upstream CDN.
The downstream CDN uses the CDNI DRIS interface to provide the
upstream CDN with this identifier called "Downstream resource
Identifier" (DRI).

Authors' Addresses

   Yannick Le Louedec
   France Telecom - Orange
   2 avenue Pierre Marzin
   Lannion,   22307
   France

   Phone: +33 2 96 05 17 64
   Email: yannick.lelouedec@orange.com

Anne Marrec
France Telecom - Orange
2 avenue Pierre Marzin
Lannion,   22307
France

Phone: +33 2 96 05 18 71
Email: anne.marrec@orange.com

Gilles Bertrand
France Telecom - Orange
38-40 rue du General Leclerc
Issy les Moulineaux,   92130
France

Phone: +33 1 45 29 89 46
Email: gilles.bertrand@orange.com

Marcin Pilarski
Orange Polska / WUT
ul. Obrzezna 7 / ul. Koszykowa 75
Warsaw, Mazowieckie 02-691 / 00-662
Poland

Email: marcin.pilarski@orange.com / marcin.pilarski@mini.pw.edu.pl