                SDCP: Streaming Distributed Control Protocol
                     draft-perez-dispatch-sdcp-01

Abstract

   This memorandum describes SDCP, a protocol to control multimedia
   streaming in cases where streaming generation should be distributed
   to improve performance. This is especially useful for Human-Things
   streams. Usually, real time applications such as virtual reality
   generate a user-controlled multimedia streaming. This is a time
   continuous data flux that could be divided spatially or temporally to
   distribute processing, memory or network resources. This protocol
   does not describe streaming communication, but the control of each
   single streaming generation in a best-effort by many nodes or things.

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 July 6, 2017.

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

The amount of information transmitted from human-to-computer (H2C) is
usually very small. This is the case of information generated by
input devices, for example, keyboards, mouses or touch screens.
Conversely, the amount of information transmitted from computer-to-
human (C2H) is huge which is increasing over time. This is the case
of information generated for output devices, such as computer
monitors, mobile phone screens or virtual reality headsets.
Furthermore, the hardware resources such as data processing, network
bandwidth or storage are also considerable. H2C control data is
required to generate C2H data, such as virtual reality and other
applications. In this way, H2C control data may be sending to many
nodes in multicast method by best-effort delivery and processing.
The protocol has been implemented by [Perez-Monte14] with good
results and its has been descripted in detail by [Perez-Monte16].

Streaming Distributed Control Protocol (SDCP) is an application-level
protocol for control of streaming distributed generation. SDCP is
built over the User Datagram Protocol (UDP) [RFC0768] or the
Lightweight User Datagram Protocol (UDP-Lite) [RFC3828], which
provides a connection less deterministic transport mechanism. SDCP
provides the complete information for suitable streaming distributed
generation. Other mechanism have been specified to transmit
multimedia streaming, including the Real Time Streaming Protocol
(RTSP) [RFC2326]. The SDCP is not meant to displace this mechanism
but rather complement it.

1.1. Requirements Language

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].

1.2. Terminology

Some clarifications and additional definitions follow:

o Multimedia Streaming: It is a group of successive multimedia real-
   time data blocks over time. A real-time data block can be an
   audio level for multimedia audio streaming or a frame for
   multimedia video streaming. Successive blocks of multimedia
   streaming must be ordered in time.

o Data Block (DB): Data portion of stream with the same shared time
   slot.

o Spatial Data Segment (SDS): Spatial Data segment is subdivision or
   partition of each Data block to distributed generation. These
   fragments could be a piece of a render image.

o Processor nodes: These nodes generate the multimedia streaming
   under a distributed scheme.

o Administrator Node: This node controls multimedia streaming
   generation by periodically sending streaming control to the
   processor nodes.

o Integrator node: This node receives multimedia streaming from
   Processor nodes to display this to a human receptor.

Integrator and Administrator nodes are the human-side and Processor
nodes are the things-side of the communication system.

2. Distributed Scheme

   Figure 1 shows scheme of a distributed stream generation system.
   Each processor node has processing, bandwidth or storing resources
   for partial stream generation.

```
+--------------------------------------------------------------------+
|Remote Administrator Node                                           |
+--------------------------------------------------------------------+
                                  | Multicast SDCP data communication
                                  V
+---------------++---------------++---------------++---------------+
|Local Proc Node||Local Proc Node||Local Proc Node||Local Proc Node|
+---------------++---------------++---------------++---------------+
                                  ||Uncompressed stream communication
                                  \/
+-------------------------------++-------------------------------+
| Local Integrator Node         || Local Integrator Node         |
+-------------------------------++-------------------------------+
                                  ||Compressed stream communication
                                  \/
+--------------------------------------------------------------------+
| Remote Human Receptors                                             |
+--------------------------------------------------------------------+
```
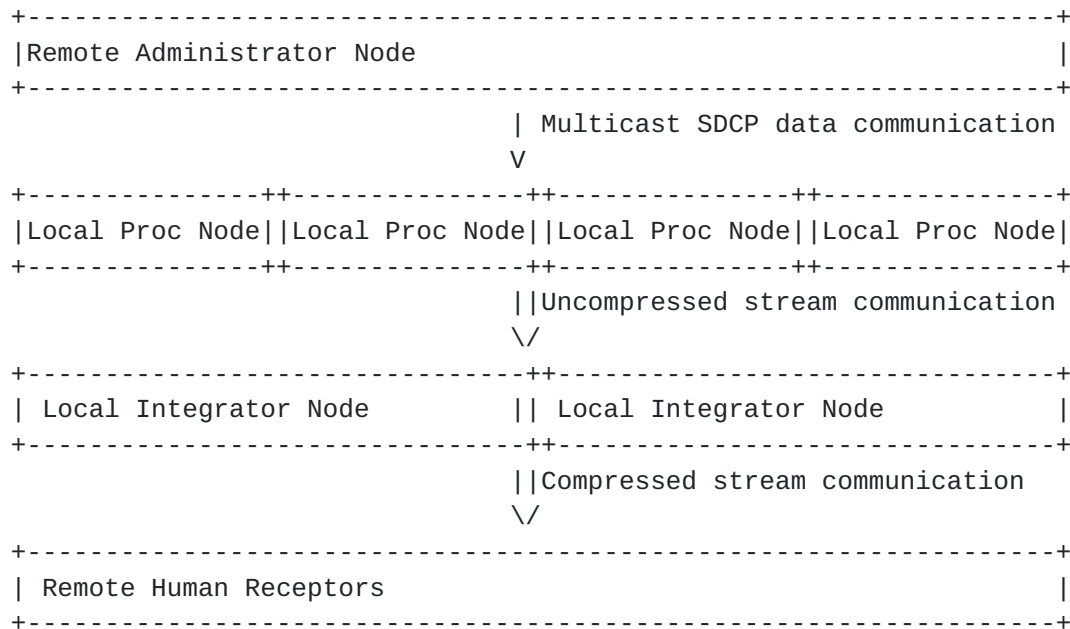
                       Distributed Scheme.

                          Figure 1

   Administrator Node sends periodically SDCP multicast control
   datagrams to Processor Nodes. The use of multicast is mandatory to
   select processor group ID. The amount of SDCP datagrams should be
   sufficient to compensate losses and to allow real-time operation.
   These losses may occur by delivery problems or it could be ignored
   datagrams by processor nodes. Administrator Node MAY assign
   different Processor Node for processing each SDS.

   Each unoccupied Processor Node receive SDCP datagrams. Occupied
   Processor Node SHOULD ignore SDCP datagrams. Each Processor Node
   generates stream portion through the use of more current SDCP control
   data. This generated stream is sent to an appropriate Integrator
   Node.

   Integrator Node receives stream portion unicast communication. All
   the stream portion received are integrated in a single stream that is
   sent to remote human receptors or locally visualized.

Administrator Node MAY assign different destination Integrator Node
for each SDS. Each integrator node MAY receive multiple streams, a
same DB or multiple/single SDS of multiple Processor Node. However,
each SDS is assigned to only one Integrator node. While that
different SDS of same stream MAY be assigned to send these to
different integrator nodes, each SDS of same stream MUST NOT be sent
to more than only one Integrator node.

3. SDCP Constant

TO DO

3.1. Multicast Addressing

TO DO

3.2. UDP Ports

TO DO

4. SDCP Format

Main SDCP format is shown in figure 2.

```
+-------------------+---------------------+--------+
| General DB Header | Specific SDS Header | Payload|
+-------------------+---------------------+--------+
```

                        SDCP Format.

                        Figure 2

o General DB Header: 256-bits length field. This header is required
   and identifies fields from all the DB.

o Specific SDS Header: Multiple of 128-bits, variable-length field.
   This header is optional and identifies fields from specific SDS.
   If this header is not present, all SDS of same DB SHOULD be
   treated equally.

o Payload: Variable-length field. Stream Control Data.

4.1. General DB Header

DB Header is required.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|control data type|M| RD  |    Stream Generation Source ID      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                     Timestamp (64 bits)                       |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        SDCP Counter                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Var DB Counter                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          DB size                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          SDS size                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          DB Type          |       Next Header Counter         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
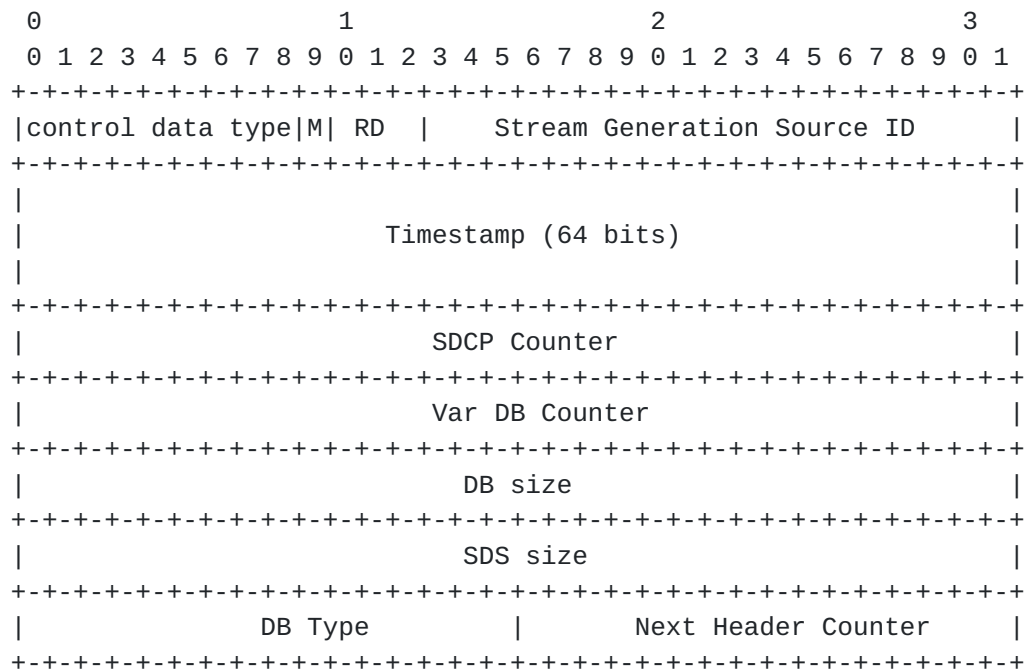
                      DB Header Format.

                           Figure 3

   Processor Node or Processor Node Group 64 bit ID is determined by
   multicast destination address of IP stack.

   Control data type: 8-bit selector. Type of control streaming
   generation data. Types are defined in accordance with specific
   requirement of application. E.g. virtual reality, game or video
   streaming, drone controller application, etc.

   Control data mode: 1-bit selector. Instant or Historical Mode.

   0 - Instant Mode: The payload has the last control data configuration
   for the Processor Nodes, which means that the Administrator Node
   sends control data in a deterministic way with the last setup.

   1 - Historical Mode: Administrator Node sends previous and actual
   control data to the processor nodes, in order to help them to
   generate the next streaming sequence.

   RD: 3-bit selector. Reserved for future use.

   Streaming Generation Source ID: 20-bit unsigned integer. Multimedia
   generation data source identification. It identifies the data source
   generating multimedia stream.

   Timestamp: 64-bits unsigned fixed-point. It includes a 32-bit
   unsigned seconds field spanning 136 years and a 32-bit fraction field
   resolving 232 picoseconds such as RFC 5905 [RFC5905]. This 64-bit
   timestamp format is used in General DB header and payload.

   SDCP Counter: 32-bit unsigned integer. Total number of SDCP
   datagrams sent.

   Var DB Counter: 32-bit unsigned integer. Total number of SDCP
   datagrams sent with control data changes.

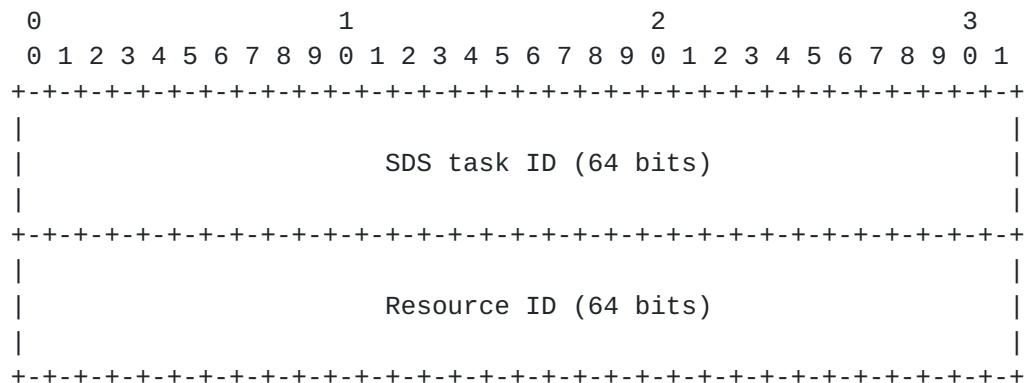   DB type: 16-bit unsigned integer.

   DB size: 32-bit unsigned integer.

   SDS size: 32-bit unsigned integer.

   Next Header Counter: 16-bit unsigned integer. Number of Optional SDS
   Headers. Length of optional headers in 16-octet units.

4.2. Specific SDS Header

   SDS header is optional. This header specifies SDS allocation to
   nodes. Two functions are defined. On the one hand, this header MAY
   determine which SDS data are assigned to generate by processor node.
   On the other hand, this header MAY determine which SDS data are
   assigned to send from processor node to integrator node. Each unique
   64 bit id can identify a node, node group and node role or SDS data
   task or SDS data task group. The node roles are processor,
   integrator and administrator but others roles can be defined.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                    SDS task ID (64 bits)                      |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                    Resource ID (64 bits)                      |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

                       SDS Header Format.

                            Figure 4

SDS task ID: 64-bit selector. It identifies individual SDS task or
SDS group tasks for allocation to nodes. The tasks already assigned
to a node can also assigned to other node by setting SDS task ID with
its node ID.

Resource ID: 64-bit selector, identifies integrator or processor node
from its interface identifier from IPv6 unicast destination address
or identifies processor node group from its low-order 64 bits of an
IPv6 multicast destination address such as IP Version 6 Addressing
Architecture [RFC2373]. Allocated Processor Node MUST process all
SDS assigned in SDS group ID and MUST NOT process SDS not assigned.
Non-allocated Processor Node MAY process all SDS. SDS not assigned
to any Integrator Node MUST be sent to Default Integrator Node.
Similarly, SDS assigned more than one Integrator Node MUST be sent
only to Default Integrator Node.

4.3. Payload

Payload data format is specified in control data type field of
general header. This field determines in virtual reality
applications variables such as camera positions, light positions,
etc.

Two modes are supported.

Instant Mode: Last change control data is only sent.

Historical Mode: All changes control data are sent.

Types of control data: TO DO.

5. Identificators Format

TO DO

5.1. SDS index

TO DO

5.2. Node index

TO DO

6. Payload types

TO DO

7. Streaming considerations

   TO DO

7.1. Streaming protocols

   TO DO

8. Acknowledgements

   I would like to thank the resources and support of GRIDTICS and
   LICPaD of the Universidad Tecnologica Nacional Regional Mendoza (UTN
   FRM), LIDIC of the Universidad Nacional de San Luis (UNSL), the Joint
   Laboratory for System Evaluation (JLSE) at Argonne National
   Laboratory and Dept. of Bioengineering, Dept. of Biomedical and
   Health Information Sciences to the University of Illinois at Chicago
   (UIC). Especially, I am deeply grateful to Gustavo Mercado,
   Christian O'Flaherty, Ines Robles and Gabriel Montenegro for their
   support.

9. IANA Considerations

   This memo includes no request to IANA.

10. Security Considerations

   TO DO

11. References

11.1. Normative References

   [RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768,
              DOI 10.17487/RFC0768, August 1980,
              <http://www.rfc-editor.org/info/rfc768>.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC3828] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., Ed.,
              and G. Fairhurst, Ed., "The Lightweight User Datagram
              Protocol (UDP-Lite)", RFC 3828, DOI 10.17487/RFC3828, July
              2004, <http://www.rfc-editor.org/info/rfc3828>.

11.2. Informative References

[Perez-Monte14]
          Perez-Monte, C., Mercado, G., Taffernaberry, J., and M.
          Piccoli, "Protocolo de comunicaciones para renderizacion
          distribuida en tiempo real - I Workshop Pre-IETF", 2014.

[Perez-Monte16]
          Perez-Monte, C., Perez, M., Luciano, C., Rizzi, S., and M.
          Piccoli, "Protocolo de comunicaciones para control de la
          generacion distribuida de flujo multimedia - WPIETFIRTF -
          III Workshop Pre-IETF/IRTF", 2016.

[RFC2326] Schulzrinne, H., Rao, A., and R. Lanphier, "Real Time
          Streaming Protocol (RTSP)", RFC 2326,
          DOI 10.17487/RFC2326, April 1998,
          <http://www.rfc-editor.org/info/rfc2326>.

[RFC2373] Hinden, R. and S. Deering, "IP Version 6 Addressing
          Architecture", RFC 2373, DOI 10.17487/RFC2373, July 1998,
          <http://www.rfc-editor.org/info/rfc2373>.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch,
          "Network Time Protocol Version 4: Protocol and Algorithms
          Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010,
          <http://www.rfc-editor.org/info/rfc5905>.

Authors' Addresses

   Cristian Federico Perez-Monte (editor)
   GridTICs - UTN FRM
   Rodriguez 273 Cuarto Piso Bloque Dpto Electronica
   Ciudad de Mendoza, Mendoza M5502AJE
   AR

   Phone: +54 261 524 4563
   Email: cristian.perez@gridtics.frm.utn.edu.ar

   Ana Laura Diedrichs
   GridTICs - UTN FRM
   Rodriguez 273 Cuarto Piso Bloque Dpto Electronica
   Ciudad de Mendoza, Mendoza M5502AJE
   AR

   Phone: +54 261 524 4563
   Email: ana.diedrichs@gridtics.frm.utn.edu.ar