

Delay-Tolerant Networking
Internet-Draft
Intended status: Experimental
Expires: September 12, 2019

E. Birrane
E. DiPietro
D. Linko
Johns Hopkins Applied Physics Laboratory
March 11, 2019

Licklider Transmission Protocol Agent Application Data Model
draft-birrane-dtn-adm-ltp-01

Abstract

This document describes the Application Data Model (ADM) for a Licklider Transmission Protocol Agent (LTPA) in compliance with the template provided by [[I-D.birrane-dtn-adm](#)].

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 September 12, 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.

Internet-Draft

LTP ADM

March 2019

Table of Contents

1.	Introduction	2
1.1.	Technical Notes	2
1.2.	Scope	3
1.3.	Requirements Language	3
2.	Structure and Design of this ADM	3
3.	Naming and Identification	4
3.1.	Namespace and Nicknames	4
4.	LTP Agent ADM JSON Encoding	5
5.	IANA Considerations	21
6.	References	21
6.1.	Informative References	21
6.2.	Normative References	21
	Authors' Addresses	22

[1.](#) Introduction

An Application Data Model (ADM) provides a guaranteed interface for the management of an application or protocol in accordance with the Asynchronous Management Architecture (AMA) defined in [\[I-D.birrane-dtn-ama\]](#). The ADM described in this document complies with the ADM Template provided in [\[I-D.birrane-dtn-adm\]](#) as encoded using the JSON syntax.

Licklider Transmission Protocol Agents (LTPAs) are software instances that implement functionality required by the Licklider Transmission Protocol [\[RFC5326\]](#). LTP is a convergence layer protocol. There is no flow control or congestion control in LTP. LTP must run either over UDP or directly over a link layer protocol. Because of this, LTP cannot be used in every situation. This LTP Agent ADM provides the set of information necessary to provide retransmission based reliability on challenged networks, focusing on the information that an agent on a network implementing LTP would need in order to run correctly.

[1.1.](#) Technical Notes

- o This document describes Version 0.0 of the LTP Agent ADM.
- o The AMM Resource Identifier (ARI) for this ADM is NOT correctly set. A sample ARI is used in this version of the specification and MAY change in future versions of this ADM until an ARI

registry is established. This notice will be removed at that time.

- o Agent applications MAY choose to ignore the name, description, or other annotative information associated with the component

definitions within this ADM where such items are only used to provide human-readable information or are otherwise not necessary to manage a device.

[1.2.](#) Scope

This ADM specifies those components of the Asynchronous Management Model (AMM) common to the management of any instance of a LTP Agent.

Any Manager software implementing this ADM MUST perform the responsibilities of an AMA Manager as outlined in [\[I-D.birrane-dtn-adm\]](#) as they relate to the objects included in this document.

Any Agent software implementing this ADM MUST perform the responsibilities of an AMA Agent as outlined in [\[I-D.birrane-dtn-adm\]](#) as they relate to the objects included in this document.

[1.3.](#) 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](#)].

[2.](#) Structure and Design of this ADM

The LTP Agent ADM's structure is in accordance to [\[I-D.birrane-dtn-adm\]](#). This ADM contains metadata, edd, report templates, and controls. Externally Defined Data (EDD) are values that are calculated external to the ADM system. Controls are predefined and sometimes parameterized opcodes that can be run on an Agent. Report Templates are ordered sets of data descriptions that show how values will be represented in a corresponding report. There are no variables, table templates, macros, or operators in this ADM at this time. The contents of this ADM are derived from the main functions and data that an agent needs to run LTP [RFC 5326](#) [[RFC5326](#)].

The Metadata that is present in this document is common to all ADMs. All ADMs have metadata that includes the name, namespace, and version of the ADM as well as the name of the organization that is issuing that particular ADM. This is important for identification purposes of the ADMs and to ensure version control. The main idea identified in LTP that is present in this ADM is a span of potential LTP data interchange between engines(nodes on a network that use LTP).

3. Naming and Identification

This section outlines the namespaces used to uniquely identify ADM objects in this specification.

3.1. Namespace and Nicknames

In accordance with [[I-D.birrane-dtn-adm](#)], every ADM is assigned a moderated Namespace. In accordance with [[I-D.birrane-dtn-amp](#)], these namespaces may be enumerated for compactness. The namespace and ADM identification for these objects is defined as follows.

Identifier	Value
Namespace	DTN/ltp_agent
ADM Enumeration	3

Table 1: Namespace Information

Given the above ADM enumeration, in accordance with [[I-D.birrane-dtn-amp](#)], the following AMP nicknames are defined.

Internet-Draft

LTP ADM

March 2019

Nickname	Collection as an ARI
60	DTN/ltp_agent/Const
61	DTN/ltp_agent/Ctrl
62	DTN/ltp_agent/Edd
63	DTN/ltp_agent/Mac
64	DTN/ltp_agent/Oper
65	DTN/ltp_agent/Rptt
67	DTN/ltp_agent/Tblt
69	DTN/ltp_agent/Var
70	DTN/ltp_agent/Mdat
71-79	DTN/ltp_agent/Reserved

Table 2: LTP ADM Nicknames

4. LTP Agent ADM JSON Encoding

```
{
  "Mdat": [{
    "name": "name",
    "type": "STR",
    "value": "ltp_agent",
    "description": "The human-readable name of the ADM."
  },
  {
    "name": "namespace",
    "type": "STR",
    "value": "DTN/ltp_agent",
    "description": "The namespace of the ADM."
  },
  {
    "name": "version",
    "type": "STR",
    "value": "v0.0",
    "description": "The version of the ADM."
  }
],
```

```
{
  "name": "organization",
  "type": "STR",
  "value": "JHUAPL",
  "description": "The name of the issuing organization of the
                  ADM."
}
],
"Edd": [{
  "name": "span_remote_engine_nbr",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }
  ]
},
```

```

    "description": "The remote engine number of this span."
  },
  {
    "name": "span_cur_expt_sess",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "Expected sessions on this span."
  },
  {
    "name": "span_cur_out_seg",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The current number of outbound segments for
                    this span."
  },
  {
    "name": "span_cur_imp_sess",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The current number of import segments for
                    this span."
  },
  {
    "name": "span_cur_in_seg",

```

```

    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The current number of inbound segments for
                    this span."
  },

```

```

{
  "name": "span_reset_time",
  "type": "UFAST",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The last time the span counters were reset."
},
{
  "name": "span_out_seg_q_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The output segment queued count for the span."
},
{
  "name": "span_out_seg_q_bytes",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The output segment queued bytes for the span."
},
{
  "name": "span_out_seg_pop_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The output segment popped count for the span."
},
{
  "name": "span_out_seg_pop_bytes",
  "type": "UINT",
  "parmspec": [{

```

"type": "UINT",


```

    "name": "\ltp_span"
  }],
  "description": "The output segment popped bytes for the span."
},
{
  "name": "span_out_ckpt_xmit_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "\ltp_span"
  }],
  "description": "The output checkpoint transmit count for the
    span."
},
{
  "name": "span_out_pos_ack_rx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "\ltp_span"
  }],
  "description": "The output positive acknowledgment received
    count for the span."
},
{
  "name": "span_out_neg_ack_rx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "\ltp_span"
  }],
  "description": "The output negative acknowledgment received
    count for the span."
},
{
  "name": "span_out_cancel_rx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "\ltp_span"
  }],
  "description": "The output canceled received count for the
    span."
},
{
  "name": "span_out_ckpt_rexmit_cnt",
  "type": "UINT",

```

```
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The output checkpoint retransmit count for the
                    span."
  },
  {
    "name": "span_out_cancel_xmit_cnt",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The output cancel retransmit count for the
                    span."
  },
  {
    "name": "span_out_complete_cnt",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The output completed count for the span."
  },
  {
    "name": "span_in_seg_rx_red_cnt",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The input segment received red count for the
                    span."
  },
  {
    "name": "span_in_seg_rx_red_bytes",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The input segment received red bytes for the
                    span."
  },
}
```

```
{
  "name": "span_in_seg_rx_green_cnt",
```

```
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The input segment received green count for the
                    span."
  },
  {
    "name": "span_in_seg_rx_green_bytes",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The input segment received green bytes for the
                    span."
  },
  {
    "name": "span_in_seg_rx_redundant_cnt",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The input segment received redundant count for
                    the span."
  },
  {
    "name": "span_in_seg_rx_redundant_bytes",
    "type": "UINT",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "The input segment received redundant bytes for
                    the span."
  },
  {

```

```

    "name": "span_in_seg_rx_mal_cnt",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "ltp_span"
    }],
    "description": "The input segment malformed count for the span."
},
{

```

```

    "name": "span_in_seg_rx_mal_bytes",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "ltp_span"
    }],
    "description": "The input segment malformed bytes for the span."
},
{
    "name": "span_in_seg_rx_unk_sender_cnt",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "ltp_span"
    }],
    "description": "The input segment unknown sender count for the
                    span."
},
{
    "name": "span_in_seg_rx_unk_sender_bytes",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "ltp_span"
    }],
    "description": "The input segment unknown sender bytes for the
                    span."
},
{
    "name": "span_in_seg_rx_unk_client_cnt",
    "type": "UINT",
    "parmspec": [{

```

```

        "type": "UINT",
        "name": "\ltp_span"
    }],
    "description": "The input segment unknown client count for the
                    span."
},
{
    "name": "span_in_seg_rx_unk_client_bytes",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "\ltp_span"
    }],
    "description": "The input segment unknown client bytes for the
                    span."
},

```

```

{
    "name": "span_in_seg_stray_cnt",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "\ltp_span"
    }],
    "description": "The input segment stray count for the span."
},
{
    "name": "span_in_seg_stray_bytes",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "\ltp_span"
    }],
    "description": "The input segment stray bytes for the span."
},
{
    "name": "span_in_seg_miscolor_cnt",
    "type": "UINT",
    "parmspec": [{
        "type": "UINT",
        "name": "\ltp_span"
    }],

```

```

        "description": "The input segment miscolored count for the
                        span."
    },
    {
        "name": "span_in_seg_miscolor_bytes",
        "type": "UINT",
        "parmspec": [{
            "type": "UINT",
            "name": "\ltp_span"
        }],
        "description": "The input segment miscolored bytes for the
                        span."
    },
    {
        "name": "span_in_seg_closed_cnt",
        "type": "UINT",
        "parmspec": [{
            "type": "UINT",
            "name": "\ltp_span"
        }],
        "description": "The input segment closed count for the span."
    },
    {

```

```

        "name": "span_in_seg_closed_bytes",
        "type": "UINT",
        "parmspec": [{
            "type": "UINT",
            "name": "\ltp_span"
        }],
        "description": "The input segment closed bytes for the span."
    },
    {
        "name": "span_in_ckpt_rx_cnt",
        "type": "UINT",
        "parmspec": [{
            "type": "UINT",
            "name": "\ltp_span"
        }],
        "description": "The input checkpoint receive count for the
                        span."
    },
    {

```

```

{
  "name": "span_in_pos_ack_tx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The input positive acknowledgeable transmitted
                  count for the span."
},
{
  "name": "span_in_neg_ack_tx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The input negative acknowledgeable transmitted
                  count for the span."
},
{
  "name": "span_in_cancel_tx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The input cancel transmitted count for the span."
},
{

```

```

  "name": "span_in_ack_retx_cnt",
  "type": "UINT",
  "parmspec": [{
    "type": "UINT",
    "name": "ltp_span"
  }],
  "description": "The input acknowledgeable retransmit count for
                  the span."
},
{
  "name": "span_in_cancel_rx_cnt",

```

```

        "type": "UINT",
        "parmspec": [{
            "type": "UINT",
            "name": "ltp_span"
        }],
        "description": "The input cancel receive count for the span."
    },
    {
        "name": "span_in_complete_cnt",
        "type": "UINT",
        "parmspec": [{
            "type": "UINT",
            "name": "ltp_span"
        }],
        "description": "The input completed count for the span."
    }
],
"Tblt": [{
    "name": "engines",
    "columns": [{
        "type": "UVASt",
        "name": "peer_engine_nbr"
    }],
    "description": "This table lists all known remote engine ids."
}],
"Rptt": [{
    "name": "endpointReport",
    "parmspec": [{
        "type": "UINT",
        "name": "ltp_span"
    }],
    "definition": [
        {"ns": "DTN/ltp_agent",
         "nm": "edd.span_remote_engine_nbr",
         "ap": [{
             "type": "ParmName",
             "value": "ltp_span"
         ]
        }
    ]
}],

```

```

    }],
    {
        "ns": "DTN/ltp_agent",
    }

```



```

        "nm": "edd.span_cur_expt_sess",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    },
    {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_cur_out_seg",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    },
    {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_cur_imp_sess",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    },
    {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_cur_in_seg",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    },
    {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_reset_time",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    },
    {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_out_seg_q_cnt",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    }

```

```
    }],
  },
  {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_seg_q_bytes",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },
  {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_seg_pop_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },
  {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_seg_pop_bytes",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },
  {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_ckpt_xmit_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },
  {, {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_pos_ack_rx_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  }, {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_neg_ack_rx_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  }
```

},{

Internet-Draft

LTP ADM

March 2019

```
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_cancel_rx_cnt",
    "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
    }]
},{
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_ckpt_rexmit_cnt",
    "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
    }]
},{
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_cancel_xmit_cnt",
    "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
    }]
},{
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_out_complete_cnt",
    "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
    }]
},{
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_in_seg_rx_red_cnt",
    "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
    }]
},{
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_in_seg_rx_red_bytes",
    "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
```

```

    ]]
  },{
    "ns":"DTN/ltp_agent",
    "nm":"edd.span_in_seg_rx_green_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  }

```

```

  },{
    "ns":"DTN/ltp_agent",
    "nm":"edd.span_in_seg_rx_green_bytes",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },{
    "ns":"DTN/ltp_agent",
    "nm":"edd.span_in_seg_rx_redundant_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },{
    "ns":"DTN/ltp_agent",
    "nm":"edd.span_in_seg_rx_redundant_bytes",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },{
    "ns":"DTN/ltp_agent",
    "nm":"edd.span_in_seg_rx_mal_cnt",
    "ap": [{
      "type": "ParmName",
      "value": "ltp_span"
    }]
  },{
    "ns":"DTN/ltp_agent",
    "nm":"edd.span_in_seg_rx_mal_bytes",
    "ap": [{
      "type": "ParmName",

```



```

    "type": "ParmName",
    "value": "ltp_span"
  }]
}, {
  "ns": "DTN/ltp_agent",
  "nm": "edd.span_in_seg_miscolor_cnt",
  "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
  }]
}, {
  "ns": "DTN/ltp_agent",
  "nm": "edd.span_in_seg_miscolor_bytes",
  "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
  }]
}, {
  "ns": "DTN/ltp_agent",
  "nm": "edd.span_in_seg_closed_cnt",
  "ap": [{
    "type": "ParmName",

```

```

    "value": "ltp_span"
  }]
}, {
  "ns": "DTN/ltp_agent",
  "nm": "edd.span_in_seg_closed_bytes",
  "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
  }]
}, {
  "ns": "DTN/ltp_agent",
  "nm": "edd.span_in_ckpt_rx_cnt",
  "ap": [{
    "type": "ParmName",
    "value": "ltp_span"
  }]
}, {
  "ns": "DTN/ltp_agent",
  "nm": "edd.span_in_pos_ack_tx_cnt",

```

```

        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    }, {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_in_neg_ack_tx_cnt",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    }, {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_in_cancel_tx_cnt",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    }, {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_in_ack_retx_cnt",
        "ap": [{
            "type": "ParmName",
            "value": "ltp_span"
        }]
    }, {
        "ns": "DTN/ltp_agent",
        "nm": "edd.span_in_cancel_rx_cnt",
        "ap": [{

```

```

        "type": "ParmName",
        "value": "ltp_span"
    }]
}, {
    "ns": "DTN/ltp_agent",
    "nm": "edd.span_in_complete_cnt",
    "ap": [{
        "type": "ParmName",
        "value": "ltp_span"
    }]
}
],

```

```

    "description": "This is all known endpoint information"
  }],
  "Ctrl": [{
    "name": "reset",
    "parmspec": [{
      "type": "UINT",
      "name": "ltp_span"
    }],
    "description": "Resets the counters associated with the engine
                    and updates the last reset time for the span
                    to be the time when this control was run."
  ]
}
]
}

```

5. IANA Considerations

At this time, this protocol has no fields registered by IANA.

6. References

6.1. Informative References

[I-D.birrane-dtn-ama]
 Birrane, E., "Asynchronous Management Architecture",
[draft-birrane-dtn-ama-07](#) (work in progress), June 2018.

6.2. Normative References

[I-D.birrane-dtn-adm]
 Birrane, E., DiPietro, E., and D. Linko, "AMA Application
 Data Model", [draft-birrane-dtn-adm-02](#) (work in progress),
 June 2018.

[I-D.birrane-dtn-amp]
 Birrane, E., "Asynchronous Management Protocol", [draft-birrane-dtn-amp-04](#) (work in progress), June 2018.

[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>>.

[RFC5326] Ramadas, M., Burleigh, S., and S. Farrell, "Licklider
Transmission Protocol - Specification", [RFC 5326](#),
DOI 10.17487/RFC5326, September 2008,
<<https://www.rfc-editor.org/info/rfc5326>>.

Authors' Addresses

Edward J. Birrane
Johns Hopkins Applied Physics Laboratory

Email: Edward.Birrane@jhuapl.edu

Evana DiPietro
Johns Hopkins Applied Physics Laboratory

Email: Evana.DiPietro@jhuapl.edu

David Linko
Johns Hopkins Applied Physics Laboratory

Email: David.Linko@jhuapl.edu