Internet Working Group
Internet Draft
Intended status: Standards Track
Expires: April 2015

Y. Jiang
X. Liu
J. Xu
Huawei
R. Cummings
National Instruments
October 16, 2015

YANG Data Model for IEEE 1588v2
draft-jlx-tictoc-1588v2-yang-02.txt

Status of this Memo

Copyright Notice

Abstract

This document defines a YANG data model for the configuration of IEEE 1588v2 devices and clocks, and also retrieval of the configuration information, data set and running states of IEEE 1588v2 clocks.

Table of Contents

1. Introduction

As a synchronization protocol, IEEE 1588v2 [IEEE1588] is widely supported in the carrier networks. It can provide high precision time synchronization as high as nano-seconds. The protocol depends on a Precision Time Protocol (PTP) engine to automatically decide its state, and a PTP transportation layer to carry the PTP timing and various quality messages. The configuration parameters and state data sets of IEEE 1588v2 are numerous.

Some work on IEEE 1588v2 MIB [PTP-MIB] is in progress in the IETF TICTOC WG. But the work is only scoped with retrieval of the state data of IEEE 1588v2 by Simple Network Management Protocol (SNMP) and configuration is not considered, thus its use is limited.

Some service providers require the management of the IEEE 1588v2 synchronization network can be more flexible and more Internet-based (typically overlaid on their transport networks). Software Defined Network (SDN) is another driving factor which demands a greater control over synchronization networks.

YANG [RFC6020] is a data modeling language used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF) [RFC6241]. A small set of built-in data types are defined in [RFC6020], and a collection of common data types are further defined in [RFC6991]. Advantages of YANG include Internet based configuration capability, validation, roll-back and etc., all these characteristics make it attractive to become a modeling language for IEEE 1588v2.

This document defines a YANG [RFC6020] data model for the configuration of IEEE 1588v2 devices and clocks, and also retrieval of the state data of IEEE 1588v2 clocks.

In order to fulfill the need of a lightweight implementation, the core module is designed to be generic and minimal, but be extensible with capability negotiation. That is, if a node is verified with a capability of more functions, then more modules can be loaded on demand, otherwise, only a basic module is loaded on the node.

This document defines PTP system information, PTP data sets and running states following the structure and definitions in IEEE 1588v2, and compatible with [PTP-MIB]. The router specific 1588v2 information is out of scope of this document.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Terminology

Terminologies used in this document are extracted from [IEEE1588] and [PTP-MIB].

ARB Arbitrary Timescale

BC Boundary Clock

DS Data Set

E2E End-to-End

EUI Extended Unique Identifier.

GPS Global Positioning System

IANA	Internet Assigned Numbers Authority
IP	Internet Protocol
NIST	National Institute of Standards and Technology
NTP	Network Time Protocol
OC	Ordinary Clock
P2P	Peer-to-Peer
PTP	Precision Time Protocol
TAI	International Atomic Time
TC	Transparent Clock
UDP	User Datagram Protocol
UTC	Coordinated Universal Time

4. IEEE 1588V2 YANG Model hierarchy

This section describes the hierarchy of IEEE 1588v2 YANG module. Query and retrieval of device wide or port specific configuration information and clock data set is described for this version.

Query and retrieval of clock information include:

- Clock data set attributes in a clock node, including: current-DS, clock-parent-DS, default-DS, time-properties-DS, and transparentClock-default-DS.

- Port specific data set attributes, including: port-DS and transparentClock-port-DS.

```
   module: ietf-yang-ptp-dataset
      +--rw  ptp-datasets* [domain-number]
       +--rw domain-number   uint8
       +--rw default-DS
       |  +--rw two-step-flag?    boolean
       |  +--rw clock-identity?   binary
       |  +--rw number-ports?     uint16
       |  +--rw clock-quality
       |  |  +--rw clock-class?                 uint8
       |  |  +--rw clock-accuracy?              uint8
       |  |  +--rw offset-scaled-log-variance?  uint16
       |  +--rw priority1?        uint8
       |  +--rw priority2?        uint8
       |  +--rw slave-only?       boolean
       +--rw current-DS
       |  +--rw steps-removed?        uint16
       |  +--rw offset-from-master?   binary
       |  +--rw mean-path-delay?      binary
       +--rw parent-DS
       |  +--rw parent-port-identity
       |  |  +--rw clock-identity?   binary
       |  |  +--rw port-number?      uint32
       |  +--rw parent-stats?                              boolean
         |  +--rw observed-parent-offset-scaled-log-variance?   uint16
         |  +--rw observed-parent-clock-phase-change-rate?      int32
         |  +--rw grandmaster-identity?                         binary
         |  +--rw grandmaster-clock-quality
         |  |  +--rw grandmaster-clock-class?                  uint8
         |  |  +--rw grandmaster-clock-accuracy?               uint8
         |  |  +--rw grandmaster-offset-scaled-log-variance?   uint16
         |  +--rw grandmaster-priority1?                        uint8
         |  +--rw grandmaster-priority2?                        uint8
         +--rw time-properties-DS
         |  +--rw current-UTC-offset-valid?   boolean
         |  +--rw current-UTC-offset?         uint16
         |  +--rw leap59?                     boolean
         |  +--rw leap61?                     boolean
         |  +--rw time-traceable?             boolean
         |  +--rw frequency-traceable?        boolean
         |  +--rw PTP-timescale?              boolean
         |  +--rw time-source?                uint8
         +--rw port-DS-list* [port-number]
         |  +--rw port-number                    uint32
         |  +--rw port-identity
         |  |  +--rw clock-identity?   binary
         |  |  +--rw port-number?      uint32
         |  +--rw port-state?                    uint8
```

```
|  +--rw log-min-delay-req-interval?    int8
|  +--rw peer-mean-path-delay?          int64
|  +--rw log-announce-interval?         int8
|  +--rw announce-receipt-timeout?      uint8
|  +--rw log-sync-interval?             int8
|  +--rw delay-mechanism?               enumeration
|  +--rw log-min-Pdelay-req-interval?   int8
|  +--rw version-number?                uint8
+--rw transparent-clock-default-DS
|  +--rw clock-identity?    binary
|  +--rw number-ports?      uint16
|  +--rw delay-mechanism?   enumeration
|  +--rw primary-domain?    uint32
+--rw transparent-clock-port-DS-list* [port-number]
   +--rw port-number                    uint32
   +--rw port-identity
   |  +--rw clock-identity?   binary
   |  +--rw port-number?      uint32
   +--rw log-min-Pdelay-req-interval?   int8
   +--rw faulty-flag?                   boolean
   +--rw peer-mean-path-delay?          int64
```

5. IEEE 1588v2 YANG Module

```
module ietf-yang-ptp-dataset {
  namespace "urn:ietf:params:xml:ns:yang:1588v2";
  prefix "ptp-dataset";
  organization "IETF TICTOC WG";
  contact "jiangyuanlong@huawei.com";
  description
       "This YANG module defines a data model for the configuration
       of IEEE 1588v2 devices and clocks, and also retrieval of the
       state data of IEEE 1588v2 clocks.";
  revision "2015-10-15"{
    description "Initial revision.";
     reference "draft-jxl-tictoc-1588v2-yang";
   }

  grouping default-DS-entry {
    description
          "This group bundles together all information about the
          PTP clock Default Datasets for a single device.";

    leaf two-step-flag {
      description
            "This object specifies whether the Two Step process is
            used.";
      type boolean;
    }
    leaf clock-identity {
      description
            "This object specifies the clockIdentity of the local
            clock";
      config true;
      type binary {
        length "8";
      }
    }

    leaf number-ports {
      description
            "This object specifies the number of PTP ports on the
            device.";
      type uint16;
    }

    container clock-quality {
      description
```

```
            "This object specifies the default clockQuality of the
            device, which contains clockClass, clockAccuracy and
            offsetScaledLogVariance.";

        leaf clock-class {
          description
                "This object specifies the Quality Class in the
                defaultDS.";
          type uint8;
        }
        leaf clock-accuracy {
          description
                "This object specifies the Quality Accuracy in the
                defaultDS.";
          type uint8;
        }
        leaf offset-scaled-log-variance {
          description
                "This object specifies the Quality offset in the
                defaultDS.";
          type uint16;
        }
      }

      leaf priority1 {
        description
              "This object specifies the clock Priority1 in the
              defaultDS ";
        type uint8;
      }
      leaf priority2{
        description
              "This object specifies the clock Priority2 in the
              defaultDS ";
        type uint8;
      }


      leaf slave-only {
        description
              "This object indicates whether the SlaveOnly flag is
              set";
        type boolean;
      }
    }

    grouping current-DS-entry {
```

```
       description
             "This group bundles together all information about the
             PTP clock Current Datasets for a single device.";

       leaf steps-removed {
            description
               "This object specifies the distance measured by the
               number of Boundary clocks between the local clock and
               the Foreign master as indicated in the stepsRemoved
               field of Announce messages.";
          type uint16;
          default 0;
       }
       leaf offset-from-master {
          description
               "This object specifies the current clock dataset
               ClockOffset value. The value of the computation of the
               offset in time between a slave and a master clock.";
          type binary {
            length "1..255";
          }
       }
       leaf mean-path-delay {
          description
               "The mean path delay between a pair of ports as measured
               by the delay request-response mechanism.";
          type binary {
            length "1..255";
          }
       }
     }

     grouping parent-DS-entry {
       description
             "This group bundles together all information about the
             PTP clock Parent Datasets for a single device.";

       leaf parent-port-identity {
          description
               "This object specifies the value of portIdentity of the
               port on the master that issues the Sync messages used in
               synchronizing this clock.";

         leaf clock-identity {
           description
                 "This object identifies the clockIdentity of the
                 master clock.";
```

```
          type binary {
            length "8";
          }
        }

        leaf port-number {
          description
                "This object identifies the PortNumber for the port
                on the specific master.";
          type uint32{
            range "0..65535";
          }
        }
             }
      leaf parent-stats {
         description
              "This object indicates whether the values of
              parentDS.observedParentOffsetScaledLogVariance and
              parentDS.observedParentClockPhaseChangeRate have been
              measured and are valid.";
         type boolean;
         default false;
      }
      leaf observed-parent-offset-scaled-log-variance {
         description
              "This object specifies an estimate of the parent clock's
              phase change rate as measured by the slave clock.";

        type uint16;
         default 0xFFFF;
      }
      leaf observed-parent-clock-phase-change-rate {
         description
              "This object specifies the clock's parent dataset
              ParentClockPhaseChangeRate value. This value is an
              estimate of the parent clock's phase change rate as
              measured by the slave clock.";
         type int32;
      }
      leaf grandmaster-identity {
         description
              "This object specifies the clockIdentity of the
              Grandmaster clock.";
         type binary{
           length "8";
         }
      }
```

```
       container grandmaster-clock-quality {
         description
               "This object specifies the clockQuality of the
               grandmaster clock.";

         leaf grandmaster-clock-class {
           description
                 "This object specifies the Quality Class of
                 grandmaster clock.";
           type uint8;
         }
         leaf grandmaster-clock-accuracy {
           description
                 "This object specifies the Quality Accuracy of the
                 grandmaster clock.";
           type uint8;
         }
         leaf grandmaster-offset-scaled-log-variance {
           description
                 "This object specifies the Quality offset of the
                 grandmaster clock.";
           type uint16;
         }
       }
       leaf grandmaster-priority1 {
          description
               "This object specifies the priority1 attribute of the
               grandmaster clock.";
          type uint8;
       }
       leaf  grandmaster-priority2 {
          description
               "This object specifies the priority2 attribute of the
               grandmaster clock.";
          type uint8;
       }

     }



     grouping time-properties-DS-entry {
       description
               "This group bundles together all information about the
               PTP clock time properties datasets for a single device.";

       leaf current-UTC-offset-valid {
```

```
          description
               "This object indicates whether current UTC offset is
               valid.";
          type boolean;
       }
       leaf current-UTC-offset {
          description
               "This object specifies the offset between TAI and UTC
               when the epoch of the PTP system is the PTP epoch,
               otherwise the value has no meaning. The value shall be
               in units of seconds.";
          type uint16;
       }
       leaf leap59 {
          description
               "This object indicates whether the last minute of the
               current UTC day contains 59 seconds.";
          type boolean;
       }
       leaf leap61 {
          description
               "This object indicates whether the last minute of the
               current UTC day contains 61 seconds.";
          type boolean;
       }
       leaf time-traceable {
          description
               "This object indicates whether the timescale and the
               currentUtcOffset are traceable to a primary reference.";
          type boolean;
       }
       leaf frequency-traceable {
          description
               "This object indicates whether the frequency determining
               the timescale is traceable to a primary reference.";
          type boolean;
       }
       leaf PTP-timescale {
          description
               "This object indicates whether the clock timescale of
               the grandmaster clock is PTP.";
          type boolean;
       }
       leaf time-source {
          description
               "This object specifies the source of time used by the
               grandmaster clock.";
```

```
        type uint8;
      }
    }

     grouping port-DS-entry {
      description
              "This group bundles together all information about the
              clock ports dataset for a single clock port.";

      container port-identity {
         description
              "This object specifies the PTP clock port Identity,
              composed of clock-identity and portNumber.";
         leaf clock-identity {
           description
             "This object identify a specific PTP node.";
           config true;
           type binary {
             length "8";
           }
         }

         leaf port-number {
           description
                "This object specifies the PTP Portnumber for this
                port.";
           type uint32 {
             range "0..65535";
           }
         }
      }

      leaf port-state {
        description
              "This object specifies the current state of the protocol
              engine associated with the port.";
        type uint8;
        default 1;
      }

      leaf log-min-delay-req-interval {
        description
              "This object specifies the Delay_Req message
              transmission interval.";
        type int8;
      }
```

```
      leaf peer-mean-path-delay {
        description
              "This object specifies an estimate of the current one-
              way propagation delay on the link when the
              delayMechanism is P2P, otherwise it shall be zero.";
        type int64;
        default 0;
      }

      leaf log-announce-interval {
         description
              "This object specifies the Announce message transmission
              interval associated with this clock port.";
         type int8;
      }

      leaf announce-receipt-timeout {
         description
              "This object specifies the number of announceInterval
              that have to pass without receipt of an Announce message
              before the occurrence of the event
              ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES.";
         type uint8;
      }

      leaf log-sync-interval {
         description
              "This object specifies the mean time interval between
              successive Sync messages.";
         type int8;
      }

      leaf delay-mechanism {
         description
              "This object specifies the delay measuring mechanism
              used by the port. If the clock is an end-to-end clock,
              the value is e2e, else if the clock is a peer-to-peer
              clock, the value shall be p2p.";
         type enumeration {
           enum E2E {
             value 01;
             description
                   "The port is configured to use the delay request-
                   response mechanism.";
           }
           enum P2P {
```

```
              value 02;
              description
                    "The port is configured to use the peer delay
                    mechanism.";
            }
            enum DISABLED {
              value 254;
              description
                    "The port does not implement the delay mechanism.";
            }
          }
        }

        leaf log-min-Pdelay-req-interval {
          description
                "This object specifies the minimum permitted mean time
                interval between successive Pdelay_Req messages.";
          type int8;
        }

        leaf version-number {
           description
                "This object specifies the version of this standard
                implemented on the port.";
           type uint8;
        }
      }

      grouping transparent-clock-default-DS-entry {
        description
                "This group bundles together the default data sets of a
                transparent clock.";
        leaf clock-identity {
          description
                "This object specifies the value of the clockIdentity
                attribute of the local clock.";
          type binary {
            length "8";
          }
        }
        leaf number-ports {
          description
                "This object specifies the number of PTP ports of the
                device.";
          type uint16;
        }
        leaf delay-mechanism {
```

```
        description
              "This object specifies the delayMechanism of the
              transparent clock.";
        type enumeration {
          enum E2E {
            value 1;
            description
                  "The port is configured to use the delay request-
                  response mechanism.";
          }
          enum P2P {
            value 2;
            description
                  "The port is configured to use the peer delay
                  mechanism.";
          }
          enum DISABLED {
            value 254;
            description
                  "The port does not implement the delay mechanism.";
          }
        }
      }
      leaf primary-domain {
        description
              "This object specifies the value of the primary
              syntonization domain.";
        type uint32 {
          range "0..255";
        }
        default 0;
      }
    }

    grouping transparent-clock-port-DS-entry {
      description
              "This group bundles together the port data sets of a
              transparent clock.";

      container port-identity {
        description
              "This object specifies the portIdentity of the local
              port.";
        leaf clock-identity {
          config true;
          type binary {
            length "8";
```

```
          }
        }

        leaf port-number {
          type uint32 {
            range "0..65535";
          }
        }
      }
      leaf log-min-Pdelay-req-interval {
         description
              "This object specifies the minimum permitted mean time
              interval between successive Pdelay_Req messages.";
        type int8;
      }
      leaf faulty-flag {
         description
              "This object indicates whether the port is faulty.";
         type boolean;
         default false;
      }
      leaf peer-mean-path-delay {
         description
              "This object specifies an estimate of the current one-
              way propagation delay on the link when the
              delayMechanism is P2P, otherwise it shall be zero.";

        type int64;
         default 0;
      }
    }

    list ptp-datasets {

      key "domain-number";
      min-elements "1";

      description
            "List of one or more PTP datasets in the device, one for
            each domain-number (see IEEE 1588-2008 subclause 6.3)";

      leaf domain-number {
        type uint8;
      }

       container default-DS {
         uses default-DS-entry;
```

```
        }

        container current-DS {
          uses current-DS-entry;
        }

        container parent-DS {
          uses parent-DS-entry;
        }

        container time-properties-DS {
          uses time-properties-DS-entry;
        }


         list port-DS-list {
          key "port-number";
          min-elements "1";
          leaf port-number {
            type uint32 {
              range "0..65535";
            }
          }
          uses port-DS-entry;
        }

        container transparent-clock-default-DS {
          uses transparent-clock-default-DS-entry;
        }

        list transparent-clock-port-DS-list {
          key "port-number";
          min-elements "1";
          leaf port-number {
            type uint32 {
              range "0..65535";
            }
          }
          uses transparent-clock-port-DS-entry;
        }
      }
    }
```

6. Security Considerations

YANG modules are designed to be accessed via the NETCONF protocol [RFC6241], thus security considerations in [RFC6241] apply here. Security measures such as using the NETCONF over SSH [RFC6242] and restricting its use with access control [RFC6536] can further improve its security, avoid injection attacks and misuse of the protocol.

Some data nodes defined in this YANG module are writable, and any changes to them may adversely impact a synchronization network.

7. IANA Considerations

This document registers a URI in the IETF XML registry, and the following registration is requested to be made:

URI: urn:ietf:params:xml:ns:yang:1588v2

This document registers a YANG module in the YANG Module Names:

name: 1588v2 namespace: urn:ietf:params:xml:ns:yang:1588v2

8. References

8.1. Normative References

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

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF) ", RFC 6020, October 2010

[RFC6991] Schoenwaelder, J., "Common YANG Data Types", RFC 6991, July 2013

[IEEE1588] IEEE, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE Std 1588-2008, July 2008

8.2. Informative References

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, June 2011

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, March 2012

[PTP-MIB] Shankarkumar, V., Montini, L., Frost, T., and Dowd, G., "Precision Time Protocol Version 2 (PTPv2) Management Information Base", draft-ietf-tictoc-ptp-mib-07, Work in progress

9. Acknowledgments

TBD

Authors' Addresses

Yuanlong Jiang
Huawei Technologies Co., Ltd.
Bantian, Longgang district
Shenzhen 518129, China
Email: jiangyuanlong@huawei.com

Xian Liu
Huawei Technologies Co., Ltd.
Bantian, Longgang district
Shenzhen 518129, China

Jinchun Xu
Huawei Technologies Co., Ltd.
Bantian, Longgang district
Shenzhen 518129, China

Rodney Cummings
National Instruments
Email: Rodney.Cummings@ni.com