Internet Working Group Y. Jiang, Ed.
X. Liu
Internet Draft J. Xu
Huawei
Intended status: Standards Track R. Cummings, Ed.
National Instruments
Expires: September 2015 March 14, 2016

YANG Data Model for IEEE 1588v2
draft-jlx-tictoc-1588v2-yang-04.txt

Status of this Memo

Copyright Notice

Abstract

This document defines a YANG data model for the configuration of IEEE 1588-2008 devices and clocks, and also retrieval of the configuration information, data set and running states of IEEE 1588-2008 clocks.

Table of Contents

1. Introduction

As a synchronization protocol, IEEE 1588-2008 (also known as IEEE 1588v2) [IEEE1588] is widely supported in the carrier networks, industrial networks, automotive networks, and many other applications. 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 1588-2008 are numerous.

According to the concepts described in [RFC3444], IEEE 1588-2008 itself provides an information model in its normative specifications for the data sets (in IEEE 1588-2008 clause 8). Some

standardization organizations including the IETF have specified data models in MIBs (Management Information Bases) for IEEE 1588-2008 data sets (e.g. [PTP-MIB], [IEEE8021AS]). Since these MIBs are typically focused on retrieval of state data using the Simple Network Management Protocol (SNMP), configuration is not considered.

Some service providers and applications require that the management of the IEEE 1588-2008 synchronization network be flexible and more Internet-based (typically overlaid on their transport networks). Software Defined Network (SDN) is another driving factor which demands an improved configuration capability of synchronization networks.

YANG [RFC6020] is a data modeling language used to model configuration and state data manipulated by network management protocols like 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 so on. All of these characteristics make it attractive to become another candidate modeling language for IEEE 1588-2008.

This document defines a YANG [RFC6020] data model for the configuration of IEEE 1588-2008 devices and clocks, and also retrieval of the state data of IEEE 1588-2008 clocks. It defines PTP system information, PTP data sets and running states following the structure and definitions in IEEE 1588-2008, and compatible with [PTP-MIB]. The router specific 1588-2008 information is out of scope of this document.

When used in practice, network products in support of synchronization typically conform to one or more IEEE 1588-2008 profiles. Each profile specifies how IEEE 1588-2008 is used in a given industry (e.g. telecom, automotive) and application. A profile can require features that are optional in IEEE 1588-2008, and it can specify new features that use IEEE 1588-2008 as a foundation.

It is expected that the IEEE 1588-2008 YANG module will be used as follows:

o The IEEE 1588-2008 YANG module can be used as-is for products that conform to one of the default profiles specified in IEEE 1588-2008.

o When the IEEE 1588 standard is revised (e.g. the IEEE 1588 revision in progress scheduled to be published in 2017), it will add some new optional features to its data sets. The YANG module of this document can be revised and extended to add the new features (e.g. of IEEE 1588-2017). The YANG "revision" can be used to indicate changes to the YANG module.

o A profile standard based on IEEE 1588-2008 may create a dedicated YANG module for its profile. The profile's YANG module may use YANG "import" to import the IEEE 1588-2008 YANG module as its foundation. Then the profile's YANG module can use YANG "augment" to add any profile-specific enhancements.

o A product that conforms to a profile standard can also create its own YANG module. The product's YANG module can "import" the profile's module, and then use YANG "augment" to add any product-specific enhancements.

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].

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

UTC Coordinated Universal Time

4. IEEE 1588-2008 YANG Model hierarchy

This section describes the hierarchy of IEEE 1588-2008 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, parent-ds, default-ds, time-properties-ds, and transparentClock-default-ds.

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

A simplified graphical representation of the data model is typically used by YANG modules as described in [REST-CONF]. This document uses the same representation and the meaning of the symbols in these diagrams is as follows:

o Brackets "[" and "]" enclose list keys.

o Abbreviations before data node names: "rw" means configuration data (read-write) and "ro" state data (read-only).

o Symbols after data node names: "?" means an optional node, "!" means a presence container, and "*" denotes a list and leaf-list.

o Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").

o Ellipsis ("...") stands for contents of subtrees that are not shown.

```
module: ietf-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?       uint16
      |  +--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         -> ../port-identity/port-number
|  +--rw port-identity
|  |  +--rw clock-identity?   binary
|  |  +--rw port-number?      uint16
|  +--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?    uint8
+--rw transparent-clock-port-ds-list* [port-number]
   +--rw port-number         -> ../port-identity/port-number
   +--rw port-identity
   |  +--rw clock-identity?   binary
   |  +--rw port-number?      uint16
   +--rw log-min-pdelay-req-interval?   int8
   +--rw faulty-flag?                   boolean
   +--rw peer-mean-path-delay?          int64
```

5. IEEE 1588-2008 YANG Module

```
<CODE BEGINS> file "ietf-ptp-dataset@2015-11-10.yang"

module ietf-ptp-dataset{
  namespace "urn:ietf:params:xml:ns:yang:ietf-ptp-dataset";
  prefix "ptp-dataset";
  organization "IETF TICTOC WG";
  contact
      "WG Web:   http://tools.ietf.org/wg/tictoc/
       WG List:  <mailto:tictoc@ietf.org>
       WG Chair: Karen O'Donoghue
```

```
                  <mailto:odonoghue@isoc.org>
          WG Chair: Yaakov Stein
                  <mailto: Yaakov_s@rad.com>
          Editor:   Yuanlong Jiang
                  <mailto:jiangyuanlong@huawei.com>
          Editor:   Rodney Cummings
                  <mailto:rodney.cummings@ni.com>";
      description
        "This YANG module defines a data model for the configuration
        of IEEE 1588-2008 clocks, and also retrieval of the state
        data of IEEE 1588-2008 clocks.";
      revision "2015-11-10" {
        description "Latest revision.";
        reference "draft-jxl-tictoc-1588v2-yang";
       }

      grouping default-ds-entry {
        description
          "Collection of members of the default data set.";

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

        leaf number-ports {
```



```
          type uint16;
          description
            "The number of PTP ports on the device.";
        }

        container clock-quality {
          description
```

```
          "The clockQuality of the local clock. It contains
           clockClass, clockAccuracy and offsetScaledLogVariance.";

          leaf clock-class {
            type uint8;
            default 248;
            description
              "The clockClass denotes the traceability of the time
              or frequency distributed by the grandmaster clock.";
          }
          leaf clock-accuracy {
            type uint8;
            description
              "The clockAccuracy indicates the expected accuracy
               of a clock when it is the grandmaster.";
          }
          leaf offset-scaled-log-variance {
            type uint16;
            description
              "An estimate of the variations of the local clock
              from a linear timescale when it is not synchronized
              to another clock using the protocol.";
          }
        }

        leaf priority1 {
          type uint8;
          description
            "The priority1 attribute of the local clock.";
        }
        leaf priority2{
          type uint8;
          description
            "The priority2 attribute of the local clock. ";
        }


        leaf slave-only {
          type boolean;
          description
```



```
            "Indicates whether the clock is a slave-only clock.";
```

```
    }
  }

  grouping current-ds-entry {
    description
      "Collection of members of current data set.";

    leaf steps-removed {
      type uint16;
      default 0;
      description
        "The number of communication paths traversed
        between the local clock and the grandmaster clock.";

    }
    leaf offset-from-master {
      type binary {
        length "1..255";
      }
      description
        "An implementation-specific representation of the
        current value of the time difference between a master
        and a slave clock as computed by the slave.";
    }
    leaf mean-path-delay {
      type binary {
        length "1..255";
      }
      description
        "An implementation-specific representation of the
         current value of the mean propagation time between a
         master and slave clock as computed by the slave.";

    }
  }

  grouping parent-ds-entry {
    description
      "Collection of members of the parent data set.";

    container parent-port-identity {
      description
        "The portIdentity of the port on the master.
        It contains two members: clockIdentity and portNumer.";
```

```
          leaf clock-identity {
            type binary {
              length "8";
            }
            description
              "The clockIdentity of the master clock.";
          }

          leaf port-number {
            type uint16;
            description
              "The portNumber for the port on the specific
               master.";
          }
        }
        leaf parent-stats {
          type boolean;
          default false;
          description
            "Indicates whether the values of
             observedParentOffsetScaledLogVariance and
             observedParentClockPhaseChangeRate of parentDS
             have been measured and are valid.";
        }
        leaf observed-parent-offset-scaled-log-variance {
          type uint16;
          default 0xFFFF;
          description
            "An estimate of the parent clock's PTP variance
             as observed by the slave clock.";
        }
        leaf observed-parent-clock-phase-change-rate {
          type int32;
          description
            "An estimate of the parent clock's phase change rate
             as observed by the slave clock.";
        }
        leaf grandmaster-identity {
          type binary{
            length "8";
          }
          description
           "The clockIdentity attribute of the grandmaster clock.";

        }
        container grandmaster-clock-quality {
          description
```

```
          "The clockQuality of the grandmaster clock. It contains
         clockClass, clockAccuracy and offsetScaledLogVariance.";

        leaf grandmaster-clock-class {
          type uint8;
          default 248;
          description
           "The clockClass attribute of the grandmaster clock.";

        }
        leaf grandmaster-clock-accuracy {
          type uint8;
          description
            "The clockAccuracy attribute of the grandmaster
             clock.";
        }
        leaf grandmaster-offset-scaled-log-variance {
          type uint16;
          description
            "The offsetScaledLogVariance of the grandmaster
             clock.";
        }
      }
      leaf grandmaster-priority1 {
        type uint8;
        description
          "The priority1 attribute of the grandmaster clock.";

      }
      leaf grandmaster-priority2 {
        type uint8;
        description
          "The priority2 attribute of the grandmaster clock.";

      }

    }

    grouping time-properties-ds-entry {
      description
        "Collection of members of the timeProperties data set.";
```

```
        leaf current-utc-offset-valid {
          type boolean;
          description
            "Indicates whether current UTC offset is valid.";
        }
```



```
        leaf current-utc-offset {
          type uint16;
          description
            "The offset between TAI and UTC when the epoch of the
             PTP system is the PTP epoch, otherwise the value has
             no meaning.";
        }
        leaf leap59 {
          type boolean;
          description
            "Indicates whether the last minute of the current UTC
             day contains 59 seconds.";
        }
        leaf leap61 {
          type boolean;
          description
            "Indicates whether the last minute of the current UTC
             day contains 61 seconds.";
        }
        leaf time-traceable {
          type boolean;
          description
            "Indicates whether the timescale and the
             currentUtcOffset are traceable to a primary
             reference.";
        }
        leaf frequency-traceable {
          type boolean;
          description
            "Indicates whether the frequency determining the
             timescale is traceable to a primary reference.";
        }
        leaf PTP-timescale {
          type boolean;
          description
            "Indicates whether the clock timescale
             of the grandmaster clock is PTP.";
```

```
      }
      leaf time-source {
        type uint8;
        description
          "The source of time used by the grandmaster clock.";

      }
    }

    grouping port-ds-entry {
```



```
      description
        "Collection of members of the port data set.";

      container port-identity {
        description
          "The PortIdentity attribute of the local port.
           It contains two members: clockIdentity and
           portNumber.";

        leaf clock-identity {
          type binary {
            length "8";
          }
          description
            "The clockIdentity of the local clock.";
        }

        leaf port-number {
          type uint16;
          description
            "The portNumber for a port on the local clock.";

        }
      }

      leaf port-state {
        type uint8;
        default 1;
        description
          "Current state associated with the port.";
      }
```

```
        leaf log-min-delay-req-interval {
          type int8;
          description
            "The logarithm to the base 2 of the minDelayReqInterval
             (the minimum permitted mean time interval between
             successive Delay_Req messages).";
        }

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



```
        }

        leaf log-announce-interval {
          type int8;
          description
            "The logarithm to the base 2 of the of the mean
             announceInterval (mean time interval between
             successive Announce messages).";
        }

        leaf announce-receipt-timeout {
          type uint8;
          description
            "The number of announceInterval that have to pass
             without receipt of an announce message before the
             occurrence of the event ANNOUNCE_RECEIPT_TIMEOUT_
             EXPIRES.";
        }

        leaf log-sync-interval {
          type int8;
          description
            "The logarithm to the base 2 of the mean SyncInterval
             for multicast messages.  The rates for unicast
             transmissions are negotiated separately on a per port
             basis.";
        }
```

```
      leaf delay-mechanism {
        type enumeration {
          enum E2E {
            value 01;
            description
              "The port uses the delay request-response
               mechanism.";
          }
          enum P2P {
            value 02;
            description
              "The port uses the peer delay mechanism.";
          }
          enum DISABLED {
            value 254;
            description
              "The port does not implement the delay
               mechanism.";
          }
```



```
        }
        description
          "The propagation delay measuring option used by the
           port in computing meanPathDelay.";
      }

      leaf log-min-Pdelay-req-interval {
        type int8;
        description
          "The logarithm to the base 2 of the
           minPdelayReqInterval (minimum permitted mean time
           interval between successive Pdelay_Req messages).";

      }

      leaf version-number {
        type uint8;
        description
          "The PTP version in use on the port.";
      }
    }
```

```
grouping transparent-clock-default-ds-entry {
  description
    "Collection of members of the transparentClockDefault data
      set (default data set for a transparent clock).";

  leaf clock-identity {
    type binary {
      length "8";
    }
    description
      "The clockIdentity of the transparent clock.";
  }
  leaf number-ports {
    type uint16;
    description
      "The number of PTP ports on the device.";
  }
  leaf delay-mechanism {
    type enumeration {
      enum E2E {
        value 1;
        description
          "The port uses the delay request-response
           mechanism.";
      }
```



```
      enum P2P {
        value 2;
        description
          "The port uses the peer delay mechanism.";
      }
      enum DISABLED {
        value 254;
        description
          "The port does not implement the delay
           mechanism.";
      }
    }
    description
      "The propagation delay measuring option
       used by the transparent clock.";
  }
  leaf primary-domain {
```

```
        type uint8;
        default 0;
        description
         "The domainNumber of the primary syntonization domain.";

      }
    }

    grouping transparent-clock-port-ds-entry {
      description
        "Collection of members of the transparentClockPort data
         set (port data set for a transparent clock).";

      container port-identity {
        description
          "This object specifies the portIdentity of the local
           port.";

        leaf clock-identity {
          type binary {
            length "8";
          }
          description
            "The clockIdentity of the transparent clock.";
        }

        leaf port-number {
          type uint16;
          description
            "The portNumber for a port on the transparent
```



```
             clock.";
        }
      }
      leaf log-min-pdelay-req-interval {
        type int8;
        description
          "The logarithm to the base 2 of the
           minPdelayReqInterval (minimum permitted mean time
           interval between successive Pdelay_Req messages).";
      }
      leaf faulty-flag {
        type boolean;
```

```
        default false;
        description
          "Indicates whether the port is faulty.";
      }
      leaf peer-mean-path-delay {
        type int64;
        default 0;
        description
          "An estimate of the current one-way propagation delay
           on the link when the delayMechanism is P2P, otherwise
           it is zero.";
      }
    }

    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;
        description
         "The domainNumber of the current syntonization domain.";
      }

       container default-ds {
         description
           "The default data set of the clock.";
         uses default-ds-entry;
       }
```



```
       container current-ds {
         description
           "The current data set of the clock.";
         uses current-ds-entry;
       }
```

```
        container parent-ds {
          description
            "The parent data set of the clock.";
          uses parent-ds-entry;
        }

        container time-properties-ds {
          description
            "The timeProperties data set of the clock.";
          uses time-properties-ds-entry;
        }

        list port-ds-list {
          key "port-number";
          description
            "List of port data sets of the clock.";
          leaf port-number{
            type leafref{
              path "../port-identity/port-number";
            }
            description
              "Refers to the portNumber memer of
              portDS.portIdentity.";
          }
          uses port-ds-entry;
        }

        container transparent-clock-default-ds {
          description
            "The members of the transparentClockDefault Data Set";
          uses transparent-clock-default-ds-entry;
        }

        list transparent-clock-port-ds-list {
          key "port-number";
          description
            "List of transparentClockPort data sets
             of the transparent clock.";
          leaf port-number{
            type leafref{
```



```
              path "../port-identity/port-number";
            }
```

```
            description
              "Refers to the portNumber memer
               of transparentClockPortDS.portIdentity.";
          }
          uses transparent-clock-port-ds-entry;
        }
      }
    }
    <CODE ENDS>
```

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:ietf-ptp-dataset

This document registers a YANG module in the YANG Module Names: name: ietf-ptp-dataset namespace: urn:ietf:params:xml:ns:yang:ietf-ptp-dataset

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

[IEEE8021AS] IEEE, "Timing and Synchronizations for Time-Sensitive Applications in Bridged Local Area Networks", IEEE 802.1AS-2001, 2011

[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-08, Work in progress

[REST-CONF] Bierman, A., Bjorklund, M., and Watsen, K., "RESTCONF protocol", draft-ietf-netconf-restconf-09, Work in progress

[RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and Data Models", RFC 3444, January 2003,

[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

9. Acknowledgments

The authors would like to thank reviews and suggestions from Mahesh Jethanandani and Tal Mizrahi.

Authors' Addresses

Yuanlong Jiang (Editor)
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
lene.liuxian@huawei.com

Jinchun Xu
Huawei Technologies Co., Ltd.
Bantian, Longgang district
Shenzhen 518129, China
xujinchun@huawei.com

Rodney Cummings (Editor)
National Instruments
11500 N. Mopac Expwy
Bldg. C
Austin, TX 78759-3504
Email: Rodney.Cummings@ni.com