NAFEM-MONITOR-MIB DEFINITIONS ::= BEGIN

-- From file: "monitor.mib"

IMPORTS

-- enterprises FROM RFC1155-SMI
OBJECT-TYPE FROM RFC-1212
IpAddress FROM RFC1065-SMI
DisplayString FROM RFC1213-MIB
NafemDateTime,
NafemValue,
NafemChoice,
NafemInterval,
NafemAlarmStatus FROM NAFEM-TC-MIB
monitorGroup,
monConfig,
monData,
monAlarm,
monLog,
monNotify FROM NAFEM-REG-MIB;

-- nafemMonitorModule MODULE-IDENTITY
-- LAST-UPDATED " 2003010100000000000"
-- ORGANIZATION "NAFEM"
-- ORGANIZATION "NAFEM"
-- CONTACT-INFO
-- " Contact: NAFEM Technical Liaison Committee
--
-- Postal: North American Association of Food Equipment
-- Manufacturers (NAFEM)
-- 161 North Clark Street, Suite 2020
-- Chicago IL 60611
-- Phone: +1 312.821.0201
-- Fax: +1 312.821.2020
-- Web: http://www.nafem.org/
--"
-- DESCRIPTION
-- " The Monitor Object Group module.
--
-- Copyright (c) 2003 North American Association of Food Equipment
-- Manufacturers (NAFEM). All rights reserved. Reproduction of this document
-- is authorized on the condition that the foregoing copyright notice is
-- included.
--
-- This SNMP MIB module (Specification) embodies NAFEM's proprietary
-- intellectual property. NAFEM retains all title and ownership in the
-- Specification, including any revisions.
--
-- NAFEM grants all interested parties a non-exclusive license to use and
-- distribute an unmodified copy of this Specification in connection with
-- management of products that incorporate the NAFEM Communications
-- Protocol, and without fee, provided this copyright notice and license
-- appear on all copies.
--
-- This Specification is supplied 'AS IS' and NAFEM makes no warranty,
-- either expressed or implied, as to the use, operation, condition, or
-- performance of the Specification. "
--
Monitor CONFIGURATION TABLE

monConfigTable OBJECT-TYPE
SYNTAX  SEQUENCE OF MonConfigEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION " Table of information describing data objects for monitoring the device."
::= { monConfig 1 }

monConfigEntry OBJECT-TYPE
SYNTAX  MonConfigEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION " An entry in the table of information describing data objects for monitoring the device."
INDEX { monConfigEntryIndex }
::= { monConfigTable 1 }

MonConfigEntry ::= SEQUENCE {
  monConfigEntryIndex  INTEGER,
  monConfigName        DisplayString,
  monConfigUnits       INTEGER,
  monConfigDescription DisplayString,
  monConfigLocation    DisplayString,
  monConfigLotId       DisplayString
}

monConfigEntryIndex OBJECT-TYPE
SYNTAX  INTEGER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION " An index into the table of information describing data objects for monitoring the device."
::= { monConfigEntry 1 }

monConfigName OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..32))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION " A text string for the name of the data to be monitored."
::= { monConfigEntry 2 }

monConfigUnits OBJECT-TYPE
SYNTAX  INTEGER NafemUnits
ACCESS  read-write
STATUS  mandatory
DESCRIPTION " The units of the data to be monitored."
::= { monConfigEntry 3 }
monConfigDescription OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..32))
ACCESS read-write
STATUS mandatory
DESCRIPTION "A textual description of the object to be monitored."
::= { monConfigEntry 4 }

monConfigLocation OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..32))
ACCESS read-write
STATUS mandatory
DESCRIPTION "A textual description of the physical location or zone of the object to be monitored."
::= { monConfigEntry 5 }

monConfigLotId OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..32))
ACCESS read-write
STATUS mandatory
DESCRIPTION "A textual description to identify the lot."
::= { monConfigEntry 6 }

-- Monitor DATA TABLE

monDataTable OBJECT-TYPE
SYNTAX SEQUENCE OF MonDataEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "A table of collected information for the data objects described in the monConfigTable."
::= { monData 1 }

monDataEntry OBJECT-TYPE
SYNTAX MonDataEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "An entry in the table of collected information for the data objects described in the monConfigTable."
INDEX { monDataEntryIndex }
::= { monDataTable 1 }

MonDataEntry ::= SEQUENCE {
  MonDataEntryIndex INTEGER,
  monDataValue NafemValue,
  monDataStatus NafemAlarmStatus,
  monDataTimestamp NafemDateTime
}

monDataEntryIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "An index into the table of collected information for the data objects described in the monConfigTable."
::= { monDataEntry 1 }
monDataValue OBJECT-TYPE
SYNTAX NafemValue
ACCESS read-only
STATUS mandatory
DESCRIPTION "The value of a data object."
::= {monDataEntry 2 }

monDataStatus OBJECT-TYPE
SYNTAX NafemAlarmStatus
ACCESS read-only
STATUS mandatory
DESCRIPTION "The status of a data object."
::= {monDataEntry 3 }

monDataTimestamp OBJECT-TYPE
SYNTAX NafemDateTime
ACCESS read-only
STATUS mandatory
DESCRIPTION "Time the data value was last updated."
::= {monDataEntry 4 }

-- Monitor ALARM TABLE

monAlarmTable OBJECT-TYPE
SYNTAX SEQUENCE OF MonAlarmEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "Table of information to configure alarms."
::= {monAlarmTable 1 }

monAlarmEntry OBJECT-TYPE
SYNTAX MonAlarmEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "An entry in the table of information to configure alarms."
INDEX {monAlarmEntryIndex }
::= {monAlarmEntry 1 }

MonAlarmEntry ::= SEQUENCE {
  monAlarmEntryIndex INTEGER,
  monAlarmCritLimitHigh NafemValue,
  monAlarmCritLimitLow NafemValue,
  monAlarmWarnLimitHigh NafemValue,
  monAlarmWarnLimitLow NafemValue,
  monAlarmDeviation INTEGER,
  monAlarmClear NafemChoice,
  monAlarmBypass NafemChoice
}

monAlarmEntryIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "The index into the table of information to configure alarms."
::= {monAlarmEntry 1 }


monAlarmCritLimitHigh OBJECT-TYPE
SYNTAX NafemValue
ACCESS read-write
STATUS mandatory
DESCRIPTION "The upper limit which, if exceeded, will generate a critical alarm. A critical alarm indicates immediate attention is necessary."
::= { monAlarmEntry 2 }

monAlarmCritLimitLow OBJECT-TYPE
SYNTAX NafemValue
ACCESS read-write
STATUS mandatory
DESCRIPTION "The lower limit which, if exceeded, will generate a critical alarm. A critical alarm indicates immediate attention is necessary."
::= { monAlarmEntry 3 }

monAlarmWarnLimitHigh OBJECT-TYPE
SYNTAX NafemValue
ACCESS read-write
STATUS mandatory
DESCRIPTION "An upper warning limit which if exceeded may require a scheduled or planned action."
::= { monAlarmEntry 4 }

monAlarmWarnLimitLow OBJECT-TYPE
SYNTAX NafemValue
ACCESS read-write
STATUS mandatory
DESCRIPTION "An lower warning limit which, if exceeded, may require a scheduled or planned action."
::= { monAlarmEntry 5 }

monAlarmDeviation OBJECT-TYPE
SYNTAX NAFEM Value
ACCESS read-write
STATUS mandatory
DESCRIPTION "Consecutive data readings whose absolute difference exceeds this value will trigger an Alarm event."
::= { monAlarmEntry 6 }

monAlarmClear OBJECT-TYPE
SYNTAX NafemChoice
ACCESS read-write
STATUS mandatory
DESCRIPTION "Set to 'true' to clear an active alarm condition which is returned to its normal state."
::= { monAlarmEntry 7 }

monAlarmBypass OBJECT-TYPE
SYNTAX NafemChoice
ACCESS read-write
STATUS mandatory
DESCRIPTION "Set to 'true' to disable an active alarm condition."
::= { monAlarmEntry 8 }
monLogTable OBJECT-TYPE
  SYNTAX        SEQUENCE OF MonLogEntry
  ACCESS        not-accessible
  STATUS        mandatory
  DESCRIPTION   " Table of items logging parameters. "
  ::= { monLog 1 }

monLogEntry OBJECT-TYPE
  SYNTAX        MonLogEntry
  ACCESS        not-accessible
  STATUS        mandatory
  DESCRIPTION   " Entry into table of logging parameters. "
  INDEX         { monLogEntryIndex }
  ::= { monLogTable 1 }

MonLogEntry ::= SEQUENCE {
    monLogEntryIndex         INTEGER,
    monDataEntryIndex        INTEGER,
    monLogEntryEnable        NafemChoice,
    monLogInterval           INTEGER,
    monLogDeviation          INTEGER,
    monLogForcedInterval     INTEGER,
    monLogMaxHoldDays        INTEGER,
}

monLogEntryIndex OBJECT-TYPE
  SYNTAX        INTEGER
  ACCESS        read-only
  STATUS        mandatory
  DESCRIPTION   " Index of logging parameter table item. "
  ::= { monLogEntry 1 }

monLogEntryEnable OBJECT-TYPE
  SYNTAX        NafemChoice
  ACCESS        read-write
  STATUS        mandatory
  DESCRIPTION   " 'true' Enables history table activation. "
  ::= { monLogEntryEnable 2 }

monLogInterval OBJECT-TYPE
  SYNTAX        INTEGER
  ACCESS        read-write
  STATUS        mandatory
  DESCRIPTION   " The period in milliseconds that expires between logging of value to history table. Also known as sample rate. "
  ::= { monLogEntry 3 }

monLogDeviation OBJECT-TYPE
  SYNTAX        INTEGER
  ACCESS        read-write
  STATUS        mandatory
  DESCRIPTION   " Instead of interval, the user can specify a range change in current value to trigger a history record to be stored. Example: current reading 125 and deviation of 2 is entered. Once the reading goes above 127 or below 123 a history record is stored and the new current reading is either 127 or 123. A zero represents no deviation action. "
  ::= { monLogEntry 4 }
monLogForcedInterval OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION " Allows the device to assure that a reading is recorded to history regardless of deviation at this rate. Example: current reading is 125 and stays there for three hours, if the user sets Min rate to 60,000 milliseconds than there will be a history record every hour at a minimum. "
::= { monLogEntry 5 }

monLogMaxHoldDays OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION " Maximum time to hold logged items in history table before overwriting entries with newer data."
::= { monLogEntry 6 }

monLogHistory OBJECT IDENTIFIER ::= { monLog 2 }

monLogHistoryTable OBJECT-TYPE
SYNTAX SEQUENCE OF MonLogHistoryEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION " Table of logged items. "
::= { monLogHistoryTable 1 }

MonLogHistoryEntry OBJECT-TYPE
SYNTAX MonLogHistoryEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION " Entry into table of logged items. "
INDEX { monLogHistoryEntryIndex }
::= { monLogHistoryTable 1 }

MonLogHistoryEntry ::= SEQUENCE {
    MonLogEntry Index INTEGER,
    monLogHistoryEntryIndex INTEGER,
    monLogHistoryValue INTEGER,
    monLogHistoryTimeStamp NafemDateTime
}

monLogEntryIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION " The log item index of the log item in the monLogTable to which this history is a part. "
::= { monLogHistoryEntryIndex 1 }

monLogHistoryEntryIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION " Index of history items that have been logged. "
::= { monLogHistoryEntry 2 }
monLogHistoryValue OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION " Data value of logged item. "
::= { monLogHistoryEntry 3 }

monLogHistoryTimeStamp OBJECT-TYPE
SYNTAX NaFeMDateTime
ACCESS read-only
STATUS mandatory
DESCRIPTION " Time items where logged to history table. "
::= { monLogHistoryEntry 4 }

-- Monitor NOTIFY Table

monitorNotifyTable OBJECT-TYPE
SYNTAX SEQUENCE OF MonitorNotifyEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION " Table of notify message parameters. "
::= { monitorNotify 1 }

monitorNotifyEntry OBJECT-TYPE
SYNTAX MonitorNotifyEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION " Entry into notify message table."
INDEX { monitorNotifyTableIndex }
::= {monitorNotifyTable 1 }

MonitorNotifyEntry ::= SEQUENCE {
  monitorNotifyTableIndex INTEGER,
  monitorNotifyMessage DisplayString ,
  monitorNotifyCode INTEGER,
  monitorNotifyTimeStamp NaFeMDateTime,
  monitorNotifyEnable NaFeMChoice
  monitorNotifyAck NaFeMChoice,
  monitorNotifyInterval NaFeMInterval,
  monitorNotifyRetries INTEGER,
  monitorNotifyResponse IpAddress
}

monitorNotifyTableIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION " Index for the rows of the monitorNotifyTable. "
::= { monitorNotifyEntry 1 }

monitorNotifyMessage OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..32))
ACCESS read-only
STATUS mandatory
DESCRIPTION " A textual description of the notify message. "
::= { monitorNotifyEntry 2 }


monitorNotifyCode OBJECT-TYPE
SYNTAX INTEGER NafemNotifyCode
ACCESS read-only
STATUS mandatory
DESCRIPTION "A numerical description of the notify message."
::= {monitorNotifyEntry 3 }

monitorNotifyTimeStamp OBJECT-TYPE
SYNTAX NafemDateTime
ACCESS read-only
STATUS mandatory
DESCRIPTION "The date and time at which an event occurred."
::= {monitorNotifyEntry 4 }

monitorNotifyEnable OBJECT-TYPE
SYNTAX NafemChoice
ACCESS read-write
STATUS mandatory
DESCRIPTION "1 = Enable Trap generation,
2 = No trap generation,
Reference current NAFEM textual convention."
::= {monitorNotifyEntry 5 }

monitorNotifyAckTable OBJECT-TYPE
SYNTAX SEQUENCE OF MonitorNotifyAckEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "Table of associated Acknowledgement feature attribute to the notification message. This table is used when the system requires a response from a manager when a trap is sent."
::= {monitorNotify 2 }

monitorNotifyAckEntry OBJECT-TYPE
SYNTAX MonitorNotifyAckEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "Entry in acknowledgement table."
INDEX {monitorNotifyAckTableIndex}
::= {monitorNotifyAckTable 1 }

MonitorNotifyAckEntry ::= SEQUENCE {
  monitorNotifyAckTableIndex INTEGER,
  monitorNotifyAck NafemChoice,
  monitorNotifyInterval NafemInterval,
  monitorNotifyRetries INTEGER,
  monitorNotifyResponse IpAddress
}

monitorNotifyAckTableIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "Index for the rows of the monitorNotifyAckTable."
::= {monitorNotifyAckEntry 1 }
monitorNotifyAck OBJECT-TYPE
   SYNTAX     NafemChoice
   ACCESS     read-write
   STATUS     mandatory
   DESCRIPTION " 1 = No acknowledgement required from manager,
                 2 = Acknowledgement required,
                 Reference current NAFEM textual convention."
   ::= { monitorNotifyAckEntry 2 }

monitorNotifyInterval OBJECT-TYPE
   SYNTAX     NafemInterval
   ACCESS     read-write
   STATUS     mandatory
   DESCRIPTION " Delay interval between non-acknowledged trap events. Applicable
                 only when an acknowledgement is required. Reference current NAFEM textual convention."
   ::= { monitorNotifyAckEntry 3 }

monitorNotifyRetries OBJECT-TYPE
   SYNTAX     INTEGER
   ACCESS     read-write
   STATUS     mandatory
   DESCRIPTION " Number of attempts of non-acknowledged trap events. Applicable
                 only when an acknowledgement is required."
   ::= { monitorNotifyAckEntry 4 }

monitorNotifyResponse OBJECT-TYPE
   SYNTAX     IpAddress
   ACCESS     read-write
   STATUS     mandatory
   DESCRIPTION " Response word from SNMP Manager to acknowledge to. Manager writes
                 his/her Ipaddress to confirm receipt."
   ::= { monitorNotifyAckEntry 5 }

END