

Network Working Group	M. Scott	
Internet-Draft	Ericsson	
Intended status: Standards Track	M. Bjorklund	
Expires: December 24, 2010	Tail-f Systems	
	Jun 22, 2010	

[TOC](#)

## **YANG Module for NETCONF Monitoring draft-ietf-netconf-monitoring-15**

### **Abstract**

This document defines a NETCONF data model to be used to monitor the NETCONF protocol. The monitoring data model includes information about NETCONF datastores, sessions, locks and statistics. This data facilitates the management of a NETCONF server. This document also defines methods for NETCONF clients to discover data models supported by a NETCONF server and defines a new NETCONF <get-schema> operation to retrieve them.

### **Status of this Memo**

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 24, 2010.

### **Copyright Notice**

Copyright (c) 2010 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 (<http://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.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

---

## Table of Contents

- [1.](#) Introduction
  - [1.1.](#) Definition of Terms
- [2.](#) Data Model to Monitor NETCONF
  - [2.1.](#) The /netconf-state Subtree
    - [2.1.1.](#) The /netconf-state/capabilities Subtree
    - [2.1.2.](#) The /netconf-state/datastores Subtree
    - [2.1.3.](#) The /netconf-state/schemas Subtree
    - [2.1.4.](#) The /netconf-state/sessions Subtree
    - [2.1.5.](#) The /netconf-state/statistics Subtree
- [3.](#) Schema Specific Operations
  - [3.1.](#) The <get-schema> Operation
- [4.](#) Examples
  - [4.1.](#) Retrieving Schema List via <get> Operation
  - [4.2.](#) Retrieving Schema Instances
- [5.](#) NETCONF Monitoring Data Model
- [6.](#) Security Considerations
- [7.](#) Acknowledgements
- [8.](#) IANA Considerations
- [9.](#) Appendix A - Change Log
  - [9.1.](#) draft-ietf-netconf-monitoring-14
  - [9.2.](#) draft-ietf-netconf-monitoring-13
  - [9.3.](#) draft-ietf-netconf-monitoring-12
  - [9.4.](#) draft-ietf-netconf-monitoring-11
  - [9.5.](#) draft-ietf-netconf-monitoring-10
  - [9.6.](#) draft-ietf-netconf-monitoring-09
  - [9.7.](#) draft-ietf-netconf-monitoring-08
  - [9.8.](#) draft-ietf-netconf-monitoring-07
  - [9.9.](#) draft-ietf-netconf-monitoring-06
  - [9.10.](#) draft-ietf-netconf-monitoring-05
  - [9.11.](#) draft-ietf-netconf-monitoring-04
  - [9.12.](#) draft-ietf-netconf-monitoring-03
  - [9.13.](#) draft-ietf-netconf-monitoring-02

<a href="#">9.14.</a>	<a href="#">draft-ietf-netconf-monitoring-01</a>
<a href="#">9.15.</a>	<a href="#">draft-scott-netconf-monitoring-00</a>
<a href="#">10.</a>	References
<a href="#">10.1.</a>	Normative References
<a href="#">10.2.</a>	Informative References
<a href="#">§</a>	Authors' Addresses

---

## 1. Introduction

[TOC](#)

This document defines a [\[YANG\]](#) (Bjorklund, M., "YANG - A data modeling language for NETCONF," June 2010.) model to be used to monitor the NETCONF protocol. It provides information about NETCONF sessions and supported schema as defined in [\[RFC4741\]](#) (Enns, R., "NETCONF Configuration Protocol," February 2006.).

Considerations such as different schema formats, feature optionality and access controls can all impact the applicability and level of detail the NETCONF server sends to a client during session setup. The methods defined in this document address the need for further means to query and retrieve schema and NETCONF state information from a NETCONF server. These are provided to complement existing base NETCONF capabilities and operations and in no way affect existing behaviour. A new <get-schema> operation is also defined to support explicit schema retrieval via NETCONF.

---

### 1.1. Definition of Terms

[TOC](#)

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, [\[RFC2119\]](#) (Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, March 1997.).

---

## 2. Data Model to Monitor NETCONF

[TOC](#)

The NETCONF monitoring data model defined in this document provides operational information on the NETCONF server. This includes details specific to the NETCONF protocol (e.g., protocol specific counters such as 'in-sessions') as well as data related to schema retrieval (e.g., schema list).

A server that implements the data model defined in this document MUST advertise the capability URI "urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring", as described in [\[YANG\] \(Bjorklund, M., "YANG - A data modeling language for NETCONF," June 2010.\)](#).

This section presents an overview of the monitoring data model. For detailed descriptions refer to the normative YANG module provided in this document (see [Section 5 \(NETCONF Monitoring Data Model\)](#)).

---

## 2.1. The /netconf-state Subtree

[TOC](#)

The netconf-state container is the root of the monitoring data model.

```
netconf-state
  /capabilities
  /datastores
  /schemas
  /sessions
  /statistics
```

**capabilities** List of NETCONF capabilities supported by the server.

**datastores** List of NETCONF configuration datastores (e.g., running, startup, candidate) supported on this device and related information.

**schemas** List of schemas supported on the server. Includes all the information required to identify the schemas and to support their retrieval.

**sessions** List of all active NETCONF sessions on the device. Includes per session counters for all NETCONF sessions.

**statistics** Includes global counters for the NETCONF server.

---

### 2.1.1. The /netconf-state/capabilities Subtree

[TOC](#)

The /netconf-state/capabilities subtree contains the capabilities supported by the NETCONF server. The list MUST include all capabilities exchanged during session setup still applicable at the time of the request.

---

### 2.1.2. The /netconf-state/datastores Subtree

[TOC](#)

The /netconf-state/datastores subtree contains the list of available datastores for the NETCONF server and includes information on their lock state.

```
configuration
  /name
  /locks
```

**name (leaf, netconf-datastore-type)** Enumeration of supported datastores; candidate, running, startup.

**locks (grouping, lock-info)** List of locks for the datastore. Information is provided for both global and partial locks [\[RFC5717\] \(Lengyel, B. and M. Bjorklund, "Partial Lock Remote Procedure Call \(RPC\) for NETCONF," December 2009.\)](#). For partial locks the list of locked nodes and the select expressions originally used to request the lock are returned.

---

### 2.1.3. The /netconf-state/schemas Subtree

[TOC](#)

The list of supported schema for the NETCONF server.

```
schema
  /identifier (key)
  /version (key)
  /format (key)
  /namespace
  /location
```

The elements identifier, version, and format are used as a key in the schema list. These are used in the <get-schema> operation.

**identifier (string)** Identifier for the schema list entry. The identifier is used in the <get-schema> operation and may be used for other means such as file retrieval.

**version (string)** Version of the schema supported. Multiple versions MAY be supported simultaneously by a NETCONF server. Each version MUST be reported individually in the schema list, i.e., with same identifier, possibly different location, but different version.

For YANG data models, version is the value of the most recent

YANG 'revision' statement in the module or submodule, or the empty string if no revision statement is present.

**format (identityref, schema-format)** The data modeling language the schema is written in. The data modeling language is represented as a YANG identity. This document defines the identities "xsd", "yang", "yin", "rng" and "rnc" (see [Section 5 \(NETCONF Monitoring Data Model\)](#)).

**namespace (inet:uri)** The Extensible Markup Language (XML) namespace [\[XML-NAMES\] \(Hollander, D., Tobin, R., Thompson, H., Bray, T., and A. Layman, "Namespaces in XML 1.0 \(Third Edition\)," December 2009.\)](#) defined by the schema.

**location (union: enum, inet:uri)** One or more locations from which this specific schema can be retrieved. The list SHOULD contain at least one entry per schema.

---

#### 2.1.4. The /netconf-state/sessions Subtree

[TOC](#)

Includes session specific data for NETCONF management sessions. The session list MUST include all currently active NETCONF sessions.

```
session
  /session-id (key)
  /transport
  /username
  /source-host
  /login-time
  /in-rpcs
  /in-bad-rpcs
  /out-rpc-errors
  /out-notifications
```

**session-id (uint32, 1..max)** Unique identifier for the session. This value is the NETCONF session identifier, as defined in [\[RFC4741\] \(Enns, R., "NETCONF Configuration Protocol," February 2006.\)](#).

**transport (identityref, transport)** Identifies transport for each session. The transport is represented as a YANG identity. This document defines the identities "netconf-ssh", "netconf-soap-

over-beep", "netconf-soap-over-https", "netconf-beep", and "netconf-tls" (see [Section 5 \(NETCONF Monitoring Data Model\)](#)).

**username (string)** The username is the client identity that was authenticated by the NETCONF transport protocol. The algorithm used to derive the username is NETCONF transport protocol specific and in addition specific to the authentication mechanism used by the NETCONF transport protocol.

**source-host (inet:host)** Host identifier (IP address or name) of the NETCONF client.

**login-time (yang:date-and-time)** Time at the server at which the session was established.

**in-rpcs (yang:zero-based-counter32)** Number of correct <rpc> requests received.

**in-bad-rpcs (yang:zero-based-counter32)** Number of messages received when a <rpc> message was expected, that were not correct <rpc> messages. This includes XML parse errors and errors on the rpc layer.

**out-rpc-errors (yang:zero-based-counter32)** Number of <rpc-reply> messages sent which contained an <rpc-error> element.

**out-notifications (yang:zero-based-counter32)** Number of <notification> messages sent.

---

#### 2.1.5. The /netconf-state/statistics Subtree

[TOC](#)

Statistical data pertaining to the NETCONF server.

```
statistics
  /netconf-start-time
  /in-bad-hellos
  /in-sessions
  /dropped-sessions
  /in-rpcs
  /in-bad-rpcs
  /out-rpc-errors
  /out-notifications
```

**statistics:** Contains management session related performance data for the NETCONF server.

**netconf-start-time (yang:date-and-time)**

Date and time at which the management subsystem was started.

**in-bad-hellos (yang:zero-based-counter32)** Number of sessions silently dropped because an invalid <hello> message was received.

**in-sessions (yang:zero-based-counter32)** Number of sessions started.

**dropped-sessions (yang:zero-based-counter32)** Number of sessions that were abnormally terminated, e.g., due to idle timeout or transport close.

**in-rpcs (yang:zero-based-counter32)** Number of correct <rpc> requests received.

**in-bad-rpcs (yang:zero-based-counter32)** Number of messages received when a <rpc> message was expected, that were not correct <rpc> messages.

**out-rpc-errors (yang:zero-based-counter32)** Number of <rpc-reply> messages sent which contained an <rpc-error> element.

**out-notifications (yang:zero-based-counter32)** Number of <notification> messages sent.

---

### 3. Schema Specific Operations

[TOC](#)

---

#### 3.1. The <get-schema> Operation

[TOC](#)

**Description:**

This operation is used to retrieve a schema from the NETCONF server.

**Parameters:**



**identifier (string):** Identifier for the schema list entry.  
Mandatory parameter.

**version (string):** Version of the schema requested.  
Optional parameter.

**format (identityref, schema-format):** The data modeling language of the schema.  
Default value is 'yang' when not specified.  
Optional parameter.

**Positive Response:**

The NETCONF server returns the requested schema.

**Negative Response:**

If the requested schema does not exist, the <error-tag> is 'invalid-value'.

If more than one schema matches the requested parameters, the <error-tag> is 'operation-failed', and <error-app-tag> is 'data-not-unique'.

---

## 4. Examples

[TOC](#)

---

### 4.1. Retrieving Schema List via <get> Operation

[TOC](#)

A NETCONF client retrieves the list of supported schema from a NETCONF server by retrieving the /netconf-state/schemas subtree via a <get> operation.

Available schema for the requesting session are returned in the reply containing the <identifier>, <version>, <format>, and <location> elements.

The response data can be used to determine the available schema and their versions. The schema itself (i.e., schema content) is not returned in the response. The optional <location> element contains a URI, which can be used to retrieve the schema by another protocol such as ftp [\[RFC0959\]](#) (Postel, J. and J. Reynolds, "File Transfer Protocol," October 1985.) or http(s) [\[RFC2616\]](#) (Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1," June 1999.) [\[RFC2818\]](#) (Rescorla, E., "HTTP Over TLS," May 2000.), or the special value

'NETCONF', which means that the schema can be retrieved from the device via the <get-schema> operation.  
Example:

```

<rpc message-id="101"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter type="subtree">
      <netconf-state xmlns=
        "urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
        <schemas/>
      </netconf-state>
    </filter>
  </get>
</rpc>

```

The NETCONF server returns a list of schema available for retrieval.

```

<rpc-reply message-id="101"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <netconf-state
      xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
      <schemas>
        <schema>
          <identifier>foo</identifier>
          <version>1.0</version>
          <format>xsd</format>
          <namespace>http://example.com/foo</namespace>
          <location>ftp://ftp.example.com/schemas/foo_1.0.xsd</location>
          <location>http://www.example.com/schema/foo_1.0.xsd</location>
          <location>NETCONF</location>
        </schema>
        <schema>
          <identifier>foo</identifier>
          <version>1.1</version>
          <format>xsd</format>
          <namespace>http://example.com/foo</namespace>
          <location>ftp://ftp.example.com/schemas/foo_1.1.xsd</location>
          <location>http://www.example.com/schema/foo_1.1.xsd</location>
          <location>NETCONF</location>
        </schema>
        <schema>
          <identifier>bar</identifier>
          <version>2008-06-01</version>
          <format>yang</format>
          <namespace>http://example.com/bar</namespace>
          <location>

```

```
        http://example.com/schema/bar@2008-06-01.yang
    </location>
    <location>NETCONF</location>
</schema>
<schema>
  <identifier>bar-types</identifier>
  <version>2008-06-01</version>
  <format>yang</format>
  <namespace>http://example.com/bar</namespace>
  <location>
    http://example.com/schema/bar-types@2008-06-01.yang
  </location>
  <location>NETCONF</location>
</schema>
</schemas>
</netconf-state>
</data>
</rpc-reply>
```

---

## 4.2. Retrieving Schema Instances

[TOC](#)

Given the reply in the previous section, the following examples illustrate the retrieval of 'foo', 'bar', and 'bar-types' schema at multiple locations, with multiple formats, and in multiple locations.

1. foo, version 1.0 in xsd format:

- a. Via FTP using location  
ftp://ftp.example.com/schemas/foo\_1.0.xsd
- b. Via HTTP using location  
http://www.example.com/schema/foo\_1.0.xsd
- c. Via <get-schema> using identifier, version, and format parameters.

```
<rpc message-id="101"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-schema
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
    <identifier>foo</identifier>
    <version>1.0</version>
    <format>xsd</format>
  </get-schema>
</rpc>
```

```
<rpc-reply message-id="101"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <!-- foo 1.0 xsd schema contents here -->
    </xs:schema>
  </data>
</rpc-reply>
```

2. bar, version 2008-06-01 in YANG format:

- a. Via HTTP using location  
http://example.com/schema/bar@2008-06-01.yang
- b. Via <get-schema> using identifier and version parameters:

```
<rpc message-id="102"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-schema
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
    <identifier>bar</identifier>
    <version>2008-06-01</version>
  </get-schema>
</rpc>
```

```

<rpc-reply message-id="102"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
    module bar {
      //default format (yang) returned
      //bar version 2008-06-01 yang module
      //contents here ...
    }
  </data>
</rpc-reply>

```

3. bar-types, version 2008-06-01 in default YANG format:

a. Via <get-schema> using identifier parameter:

```

<rpc message-id="103"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-schema
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
    <identifier>bar-types</identifier>
  </get-schema>
</rpc>

<rpc-reply message-id="103"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">
    module bar-types {
      //default format (yang) returned
      //latest revision returned
      //is version 2008-06-01 yang module
      //contents here ...
    }
  </data>
</rpc-reply>

```

---

## 5. NETCONF Monitoring Data Model

[TOC](#)

The data model described in this memo is defined in the following YANG module.

This YANG module imports typedefs from [\[YANG-TYPES\] \(Schoenwaelder, J., "Common YANG Data Types," April 2010.\)](#) and references [\[RFC4741\] \(Enns, R., "NETCONF Configuration Protocol," February 2006.\)](#), [\[RFC4742\]](#)

[\(Wasserman, M. and T. Goddard, "Using the NETCONF Configuration Protocol over Secure SHell \(SSH\)," December 2006.\), \[RFC4743\] \(Goddard, T., "Using NETCONF over the Simple Object Access Protocol \(SOAP\)," December 2006.\), \[RFC4744\] \(Lear, E. and K. Crozier, "Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol \(BEEP\)," December 2008.\), \[RFC5539\] \(Badra, M., "NETCONF over Transport Layer Security \(TLS\)," May 2009.\), \[xmlschema-1\] \(Biron, Paul V. and Ashok. Malhotra, "XML Schema Part 1: Structures Second Edition W3C Recommendation 28 October 2004," October 2004.\), \[YANG\] \(Bjorklund, M., "YANG - A data modeling language for NETCONF," June 2010.\), \[ISO/IEC 19757-2:2008\] \(ISO/IEC, "Document Schema Definition Language \(DSDL\) -- Part 2: Regular-grammar-based validation -- RELAX NG," December 2008.\), and \[RFC5717\] \(Lengyel, B. and M. Bjorklund, "Partial Lock Remote Procedure Call \(RPC\) for NETCONF," December 2009.\).](#)

<CODE BEGINS>

```
module ietf-netconf-monitoring {

    namespace "urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring";
    prefix "ncm";

    import ietf-yang-types { prefix yang; }
    import ietf-inet-types { prefix inet; }

    organization
        "IETF NETCONF (Network Configuration) Working Group";

    contact
        "WG Web:  <http://tools.ietf.org/wg/netconf/>
        WG List:  <mailto:netconf@ietf.org>

        WG Chair: Mehmet Ersue
                  <mailto:mehmet.ersue@nsn.com>

        WG Chair: Bert Wijnen
                  <mailto:bertietf@bwijnen.net>

        Editor:   Mark Scott
                  <mailto:mark.scott@ericsson.com>

        Editor:   Martin Bjorklund
                  <mailto:mbj@tail-f.com>";

    description
        "NETCONF Monitoring Module.
        All elements in this module are read-only.

        Copyright (c) 2010 IETF Trust and the persons identified as
        the document authors.  All rights reserved.

        Redistribution and use in source and binary forms, with or
        without modification, is permitted pursuant to, and subject
        to the license terms contained in, the Simplified BSD License
        set forth in Section 4.c of the IETF Trust's Legal Provisions
        Relating to IETF Documents
        (http://trustee.ietf.org/license-info).

        This version of this YANG module is part of RFC XXXX; see
        the RFC itself for full legal notices.";
    // RFC Ed.: replace XXXX with actual RFC number and remove this note

    revision 2010-06-22 {
```



```

    description
        "Initial revision.";
    reference
        "RFC XXXX: YANG Module for NETCONF Monitoring";
// RFC Ed.: replace XXXX with actual RFC number and remove this note
}

typedef netconf-datastore-type {
    type enumeration {
        enum running;
        enum candidate;
        enum startup;
    }
    description
        "Enumeration of possible NETCONF datastore types.";
    reference
        "RFC 4741: NETCONF Configuration Protocol";
}

identity transport {
    description
        "Base identity for NETCONF transport types.";
}

identity netconf-ssh {
    base transport;
    description
        "NETCONF over Secure Shell (SSH).";
    reference
        "RFC 4742: Using the NETCONF Configuration Protocol
            over Secure Shell (SSH)";
}

identity netconf-soap-over-beep {
    base transport;
    description
        "NETCONF over Simple Object Access Protocol (SOAP) over
            Blocks Extensible Exchange Protocol (BEEP).";
    reference
        "RFC 4743: Using NETCONF over the Simple Object
            Access Protocol (SOAP)";
}

identity netconf-soap-over-https {
    base transport;
    description
        "NETCONF over Simple Object Access Protocol (SOAP)
            over Hypertext Transfer Protocol Secure (HTTPS).";
    reference

```

```

        "RFC 4743: Using NETCONF over the Simple Object
            Access Protocol (SOAP)";
    }

    identity netconf-beep {
        base transport;
        description
            "NETCONF over Blocks Extensible Exchange Protocol (BEEP).";
        reference
            "RFC 4744: Using the NETCONF Protocol over the
                Blocks Extensible Exchange Protocol (BEEP)";
    }

    identity netconf-tls {
        base transport;
        description
            "NETCONF over Transport Layer Security (TLS).";
        reference
            "RFC 5539: NETCONF over Transport Layer Security (TLS)";
    }

    identity schema-format {
        description
            "Base identity for data model schema languages.";
    }

    identity xsd {
        base schema-format;
        description
            "W3C XML Schema Definition.";
        reference
            "W3C REC REC-xmlschema-1-20041028:
                XML Schema Part 1: Structures";
    }

    identity yang {
        base schema-format;
        description
            "The YANG data modeling language for NETCONF.";
        reference
            "RFC YYYY: YANG - A data modeling language for NETCONF";
        // RFC Ed.: replace YYYY with actual RFC number and remove this note
    }

    identity yin {
        base schema-format;
        description
            "The YIN syntax for YANG.";
        reference

```

```

    "RFC YYYY: YANG - A data modeling language for NETCONF";
    // RFC Ed.: replace YYYY with actual RFC number and remove this note
}

identity rng {
    base schema-format;
    description
        "Regular Language for XML Next Generation (RELAX NG).";
    reference
        "ISO/IEC 19757-2:2008: RELAX NG";
}

identity rnc {
    base schema-format;
    description
        "Relax NG Compact Syntax";
    reference
        "ISO/IEC 19757-2:2008: RELAX NG";
}

grouping common-counters {
    description
        "Counters that exist both per session, and also globally,
        accumulated from all sessions.";

    leaf in-rpcs {
        type yang:zero-based-counter32;
        description
            "Number of correct <rpc> messages received.";
    }
    leaf in-bad-rpcs {
        type yang:zero-based-counter32;
        description
            "Number of messages received when a <rpc> message was expected,
            that were not correct <rpc> messages. This includes XML parse
            errors and errors on the rpc layer.";
    }
    leaf out-rpc-errors {
        type yang:zero-based-counter32;
        description
            "Number of <rpc-reply> messages sent which contained an
            <rpc-error> element.";
    }
    leaf out-notifications {
        type yang:zero-based-counter32;
        description
            "Number of <notification> messages sent.";
    }
}
}

```

```

container netconf-state {
  config false;
  description
    "The netconf-state container is the root of the monitoring
    data model.";

  container capabilities {
    description
      "Contains the list of NETCONF capabilities supported by the
      server.";

    leaf-list capability {
      type inet:uri;
      description
        "List of NETCONF capabilities supported by the server.";
    }
  }

  container datastores {
    description
      "Contains the list of NETCONF configuration datastores.";

    list datastore {
      key name;
      description
        "List of NETCONF configuration datastores supported by
        the NETCONF server and related information.";

      leaf name {
        type netconf-datastore-type;
        description
          "Name of the datastore associated with this list entry.";
      }

      container locks {
        presence
          "This container is present only if the datastore
          is locked.";
        description
          "The NETCONF <lock> and <partial-lock> operations allow
          a client to lock specific resources in a datastore. The
          NETCONF server will prevent changes to the locked
          resources by all sessions except the one which acquired
          the lock(s).

          Monitoring information is provided for each datastore
          entry including details such as the session which acquired
          the lock, the type of lock (global or partial) and the
          list of locked resources. Multiple locks per datastore

```

```

        are supported.";

grouping lock-info {
    description
        "Lock related parameters, common to both global and
        partial locks.";

    leaf locked-by-session {
        type uint32;
        mandatory true;
        description
            "The session ID of the session that has locked
            this resource. Both a global lock and a partial
            lock MUST contain the NETCONF session-id.

            If the lock is held by a session which is not managed
            by the NETCONF server (e.g., a CLI session), a session
            id of 0 (zero) is reported.";
        reference
            "RFC 4741: NETCONF Configuration Protocol";
    }
    leaf locked-time {
        type yang:date-and-time;
        mandatory true;
        description
            "The date and time of when the resource was
            locked.";
    }
}

choice lock-type {
    description
        "Indicates if a global lock or a set of partial locks
        are set.";

    container global-lock {
        description
            "Present if the global lock is set.";
        uses lock-info;
    }

    list partial-locks {
        key lock-id;
        description
            "List of partial locks.";
        reference
            "RFC 5717: Partial Lock Remote Procedure Call (RPC) for
            NETCONF";
    }
}

```

```

    leaf lock-id {
      type uint32;
      description
        "This is the lock id returned in the <partial-lock>
        response.";
    }
    uses lock-info;
    leaf-list select {
      type string;
      min-elements 1;
      description
        "The xpath expression which was used to request
        the lock. The select expression indicates the
        original intended scope of the lock.";
    }
    leaf-list locked-nodes {
      type instance-identifier;
      description
        "The list of instance-identifiers (i.e., the
        locked nodes).

        The scope of the partial lock is defined by the list
        of locked nodes.";
    }
  }
}
}
}
}
}
}
}

```

```

container schemas {
  description
    "Contains the list of data model schemas supported by the
    server.";

  list schema {
    key "identifier version format";

    description
      "List of data model schemas supported by the server.";

    leaf identifier {
      type string;
      description
        "Identifier to uniquely reference the schema. The
        identifier is used in the <get-schema> operation and may
        be used for other purposes such as file retrieval.

        For modeling languages which support or require a data

```

```

        model name (e.g., YANG module name) the identifier MUST
        match that name. For YANG data models, the identifier is
        the name of the module or submodule. In other cases an
        identifier such as a filename MAY be used instead.";
    }
    leaf version {
        type string;
        description
            "Version of the schema supported. Multiple versions MAY be
            supported simultaneously by a NETCONF server. Each
            version MUST be reported individually in the schema list,
            i.e., with same identifier, possibly different location,
            but different version.

            For YANG data models, version is the value of the most
            recent YANG 'revision' statement in the module or
            submodule, or the empty string if no revision statement
            is present.";
    }
    leaf format {
        type identityref {
            base schema-format;
        }
        description
            "The data modeling language the schema is written
            in (currently xsd, yang, yin, rng or rnc).

            For YANG data models, 'yang' format MUST be supported and
            'yin' format MAY also be provided.";
    }
    leaf namespace {
        type inet:uri;
        mandatory true;
        description
            "The XML namespace defined by the data model.

            For YANG data models, this is the module's namespace.
            If the list entry describes a submodule, this field
            contains the namespace of the module to which the
            submodule belongs.";
    }
    leaf-list location {
        type union {
            type enumeration {
                enum "NETCONF";
            }
            type inet:uri;
        }
        description

```

"One or more locations from which the schema can be retrieved. This list SHOULD contain at least one entry per schema.

A schema entry may be located on a remote file system (e.g., reference to file system for ftp retrieval) or retrieved directly from a server supporting the <get-schema> operation (denoted by the value 'NETCONF').";

```
    }  
  }  
}
```

```
container sessions {  
  description  
    "The sessions container includes session specific data for  
    NETCONF management sessions. The session list MUST include  
    all currently active NETCONF sessions.";
```

```
  list session {  
    key session-id;  
    description  
      "All NETCONF sessions managed by the NETCONF server  
      MUST be reported in this list.";
```

```
    leaf session-id {  
      type uint32 {  
        range "1..max";  
      }  
      description  
        "Unique identifier for the session. This value is the  
        NETCONF session identifier, as defined in RFC4741.";  
      reference  
        "RFC 4741: NETCONF Configuration Protocol";  
    }  
    leaf transport {  
      type identityref {  
        base transport;  
      }  
      mandatory true;  
      description  
        "Identifies the transport for each session, e.g.,  
        'netconf-ssh', 'netconf-soap', etc.";
```

```
    }  
    leaf username {  
      type string;  
      mandatory true;  
      description  
        "The username is the client identity that was authenticated  
        by the NETCONF transport protocol. The algorithm used to
```



```

        derive the username is NETCONF transport protocol specific
        and in addition specific to the authentication mechanism
        used by the NETCONF transport protocol.";
    }
    leaf source-host {
        type inet:host;
        description
            "Host identifier of the NETCONF client. The value
            returned is implementation specific (e.g., hostname,
            IPv4 address, IPv6 address)";
    }
    leaf login-time {
        type yang:date-and-time;
        mandatory true;
        description
            "Time at the server at which the session was established.";
    }
    uses common-counters {
        description
            "Per-session counters. Zero based with following reset
            behaviour:
            - at start of a session
            - when max value is reached";
    }
}

container statistics {
    description
        "Statistical data pertaining to the NETCONF server.";

    leaf netconf-start-time {
        type yang:date-and-time;
        description
            "Date and time at which the management subsystem was
            started.";
    }
    leaf in-bad-hellos {
        type yang:zero-based-counter32;
        description
            "Number of sessions silently dropped because an
            invalid <hello> message was received. This includes hello
            messages with a 'session-id' attribute, bad namespace, and
            bad capability declarations.";
    }
    leaf in-sessions {
        type yang:zero-based-counter32;
        description
            "Number of sessions started. This counter is incremented

```

when a <hello> message with a <session-id> is sent.

```
'in-sessions' - 'in-bad-hellos' =
    'number of correctly started netconf sessions';
}
leaf dropped-sessions {
    type yang:zero-based-counter32;
    description
        "Number of sessions that were abnormally terminated, e.g.,
        due to idle timeout or transport close. This counter is not
        incremented when a session is properly closed by a
        <close-session> operation, or killed by a <kill-session>
        operation.";
}
uses common-counters {
    description
        "Global counters, accumulated from all sessions.
        Zero based with following reset behaviour:
        - re-initialization of NETCONF server
        - when max value is reached";
}
}
}

rpc get-schema {
    description
        "This operation is used to retrieve a schema from the
        NETCONF server.

        Positive Response:
            The NETCONF server returns the requested schema.

        Negative Response:
            If requested schema does not exist, the <error-tag> is
            'invalid-value'.

            If more than one schema matches the requested parameters, the
            <error-tag> is 'operation-failed', and <error-app-tag> is
            'data-not-unique'.";

    input {
        leaf identifier {
            type string;
            mandatory true;
            description
                "Identifier for the schema list entry.";
        }
        leaf version {
            type string;
```



Contains identity information which could be used in an attempt to authenticate with the server.

This username is only meant for monitoring, and SHOULD NOT be used for other purposes, such as access control, without a detailed discussion of the limitations of this reported username. For example, it is possible that server A and server B might report the same username, but these might be for different persons.

---

## 7. Acknowledgements

[TOC](#)

The authors would like to thank Andy Bierman, Mehmet Ersue, Washam Fan, David Harrington, Balazs Lengyel, Hideki Okita, Juergen Schoenwaelder, Bert Wijnen and many other members of the NETCONF WG for providing important input to this document. The authors would also like to specifically acknowledge Sharon Chisholm's work on 'draft-netconf-chisholm-monitoring-02' and contribution to this document.

---

## 8. IANA Considerations

[TOC](#)

-- Editor note to IANA/RFC-Editor: we request that you make these assignments, in which case it is to be documented as below replacing 'RFCXXXX' as appropriate

This document registers one URI in "The IETF XML Registry". Following the format in [\[RFC3688\] \(Mealling, M., "The IETF XML Registry," January 2004.\)](#) the following is requested.

URI: urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring

Registrant Contact: The IESG.

XML: N/A, the requested URI is an XML namespace.

This document registers one module in the "YANG Module Names" registry. Following the format in [\[YANG\] \(Bjorklund, M., "YANG - A data modeling language for NETCONF," June 2010.\)](#) the following is requested.

name: ietf-netconf-monitoring

namespace: urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring

prefix: ncm

reference: RFCXXXX

---

## 9. Appendix A - Change Log

[TOC](#)

[-- Editor note to IANA/RFC-Editor: Please remove this section upon publication.]

---

### 9.1. draft-ietf-netconf-monitoring-14

[TOC](#)

- removed the session-id and session-id-or-zero typedefs from the YANG module
- 

### 9.2. draft-ietf-netconf-monitoring-13

[TOC](#)

- updated after AD and Gen-Art reviews
- Sec 6: new security text template (rev4) used
- updated IANA considerations to reference Yang Module Names registry
- updated references to latest versions
- removed unused informative reference to rfc5277
- updated username description to specify SSH algorithm
- updated <get-schema> negative response
- Yang Module changes:
  - added Martin to list of contacts
  - prefix changed to 'ncm' (netconf monitoring)
  - grouping netconf-datastore-type is now an enumeration
  - netconf-over-soap-over-beep description update
  - netconf-over-soap-over-https description update
  - yang identity description update
  - yin identity description update
  - fixed reference statement in yin identity
  - capabilities leaf-list description update
  - added presence statement to locks container
  - name changed from container to leaf node
  - name key added for datastores list
  - added typedef sessionid-or-zero
  - locked-time now mandatory
  - lock-type description added
  - schemas description added
  - schema description added
  - namespace now mandatory

---

### 9.3. draft-ietf-netconf-monitoring-12

[TOC](#)

- updated for AD review

---

### 9.4. draft-ietf-netconf-monitoring-11

[TOC](#)

- updated URI and YANG module revision
- updated sec 5 title to align with other text
- updated Acknowledgements
- added Description statements to identityrefs in the YANG module
- updated document expiry
- updated document title to v11
- sec 1.1: removed 'schema' definition; it will be placed in one of the base NETCONF documents instead (i.e. 4741bis)
- sec 2: reworted 'The NETCONF monitoring data defined' to 'The NETCONF monitoring data model defined'
- sec 2.1: reworted 'The /netcong-state subtree...' to 'The netconf-state container...'
- sec 2.1: reworted 'List of NETCONF datastores...' to 'Includes all supported datastore types...'
- sec 2.1: removed '(Container)' for consistency
- sec 2.1.3: reworted format to remove file/module with schema
- sec 2.1.3 reworted 'The data modeling language of the file/module...' to 'The data modeling language the schema is written in...' removing references to files/modules.
- sec 2.1.3: reworted 'defined by the data model' to 'defined by the schema'
- sec 2.1.3: reworted namespace from 'by the data model' to 'by the schema'
- sec 2.1.4 changed '[NETCONF Configuration Protocol]' to '[RFC4741]'
- sec 4.1 simplified wording in this section. Specifically removed the SHOULD and MUST statements which are already qualified in sec 2.1.3 and in YANG the module
- sec 4.1 to address comment about lack of standardized sftp definition changed 'sftp' to 'ftp' and updated examples to use standardized ftp URI
- sec 4.1 reworted 'data models' to 'schema'
- sec 4.1 updated YANG example URI to use '@' instead of '-'
- sec 4.2 updated YANG example URI to use '@' instead of '-'
- YANG module, revision statement: changed RFC title to match draft name
- YANG module, container netconf-state: updated description to match sec 2.1
- YANG module, container datastores: updated description
- YANG module, list partial-locks: added reference to RFC5717
- YANG module, leaf-list locked-nodes: removed text which was inconsistent with partial lock RFC
- YANG module, leaf format: updated description to match sec 2.1.3
- YANG module, container sessions: updated description to match sec 4.1
- YANG module, leaf source-host: updated description to clarify that identifier is likely to be implementation specific per sec 2
- References: moved YANG, Common Yang Data Types and XML from informative to normative

- References: updated partial lock details to RFC5717

---

## 9.5. draft-ietf-netconf-monitoring-10

[TOC](#)

- sec 1.1, updated schema definition, removed YANG definition (it is in references instead)
- reworded text in Intro about NETCONF clients adjusting 'capabilities', per agreement that capabilities can not change in a session (per 4741bis)
- introduction, removed description of NETCONF behaviour for schema changes on active sessions. Has been moved to 4741bis
- updated namespace from '\*:ns:netconf::state' to '\*:yang:ns:ietf-netconf-monitoring:DRAFT-10'  
For details see ML archive:  
'<http://www.ietf.org/mail-archive/web/netconf/current/msg05296.html>'
- moved majority of sec 2 into the yang module (making module self describing), leaving only overview text in the RFC body.  
For details see ML archive:  
'<http://www.ietf.org/mail-archive/web/netconf/current/msg05209.html>'
- reworded definition of 'schema'
- removed residual references to XSD types (i.e. xs:string)
- reworded text that schema may not always be retrievable
- will add additional text that yang format is both the default and mandatory; yin is optional
- updated text per Juergen's WGLC comments on v09.  
For details see ML archive:  
'<http://www.ietf.org/mail-archive/web/netconf/current/msg05306.html>'
- updated reference in sec 1.1 for RFC2119
- title change to 'YANG Module for NETCONF Monitoring'

---

## 9.6. draft-ietf-netconf-monitoring-09

[TOC](#)



- reversion of 'session-type' to 'transport'
- element naming consistency. All lowerCamelCase names have been converted to 'hyphen-delimited' or 'netconf-style-naming' as it is sometimes referred to on the ML. E.g. 'sessionType' -> 'session-type'. This change impacts both the draft text and yang module.
- <get-schema> operation updated:
  - now has only one mandatory parameter, 'identifier'
  - updated negative responses, including the rpc operation description in yang module
- comments added to the yang module indicating that the definition of 'session-id' is consistent with 4741bis 'session-id-type' and could be imported from that pending RFC. Similar for 'netconf-datastore-type'. This was in favour of adding further dependencies and delays by waiting for 4741bis to complete.

---

## 9.7. draft-ietf-netconf-monitoring-08

[TOC](#)

- Addresses action items from IETF75 and mailing list comments
- Most significant is the decision to adopt YANG as normative
  - As a result the publication of this draft depends on:
    - o WGLC updates (if required)
    - o updates to the normative model as result of netmod YANG changes (if required)
    - o data organization changes if WG adopts a new data hierarchy for netmod (this draft uses "/netconf-state/")
- XML Schema of model and IP host definition have been removed

---

## 9.8. draft-ietf-netconf-monitoring-07

[TOC](#)

- updates the examples in sec 4.2 per ML discussion

---

## 9.9. draft-ietf-netconf-monitoring-06

[TOC](#)

- replaced references to 'schema-retrieval capability' with '<get-schema> operation'
- upper/lowercase alignment on references to 'format'
- changed negative response for <get-schema> to 'data-missing'
- ML comments on v05 which required changes in the draft:
  - o 2.1.2: reworded to avoid suggestion that configurable data was contained in the sub-tree (per Balazs' request)
  - o 2.1.2: reworded paragraph partial lock and select statements (per Balazs' suggested text)
  - o Fixed examples in 3.1 and added negative response (per Balazs' comment)
  - o YANG: location changed to leaf-list (per Balazs' comment)
  - o added non-normative references to yang and yang-types (per Balazs' comment)
  - o 2.1.3: added YIN and RNC to format 'enums' text and models

---

## 9.10. draft-ietf-netconf-monitoring-05

[TOC](#)

This version addresses action items from IETF74 and comments on v4.

- updated statistics (and counter definitions) per offline discussion:
- updated counter definitions
- updated data types (addition of ZeroBasedCounter32)
- per session counters; see section 2.1.4
- global counters; see section 2.1.5
- unique sessionId handling and WG consensus:
  - o expected handling of NETCONF and non-NETCONF sessions defined
  - o explicit exclusion of sessionId=0 added
  - o see section 2.1.4
- removal of subscriptions subtree; was sec 2.1.5 in v4
- revised XSD and YANG models

---

## 9.11. draft-ietf-netconf-monitoring-04

[TOC](#)

- New RPC operation defined <get-schema>, replacing <list-schema>.
  - Updated Abstract and Introduction: new text, removed NETCONF protocol diagram.
  - Changed /netconfstate/ to /netconf-state
  - Added netconf-state/datastores
  - Changed /netconf-state/schema to /netconf-state/schemas
  - Added /netconf-state/statistics
  - Added /schema/format (and added to key)
  - Added /schema/namespace (and added to key)
  - Updated examples.
  - More detailed monitored data descriptions per ML comments.
- 

#### **9.12. draft-ietf-netconf-monitoring-03**

[TOC](#)

- incorporates ML list comments on v-02.
- 

#### **9.13. draft-ietf-netconf-monitoring-02**

[TOC](#)

- Merged content from 'draft-scott-netconf-schema-query' into this draft
  - this draft now covers both schema retrieval and monitoring
- 

#### **9.14. draft-ietf-netconf-monitoring-01**

[TOC](#)

- WG first draft, 'draft-ietf-netconf-monitoring-01'.
  - Related draft 'draft-scott-netconf-schema-query-00' also presented covering schema retrieval. Query draft proposes new NETCONF operations to retrieve schema defined in this draft.
-

## 9.15. draft-scott-netconf-monitoring-00

[TOC](#)

- Initial draft, 'draft-scott-monitoring-00'
- based on 'draft-netconf-chisholm-monitoring-02'
- adopted as working group document at IETF69.

---

## 10. References

[TOC](#)

---

### 10.1. Normative References

[TOC](#)

[ISO/IEC 19757-2:2008]	ISO/IEC, " <a href="#">Document Schema Definition Language (DSDL) -- Part 2: Regular-grammar-based validation -- RELAX NG</a> ," December 2008.
[RFC2119]	Bradner, s., " <a href="#">Key words for use in RFCs to Indicate Requirement Levels</a> ", BCP 14," RFC 2119, March 1997.
[RFC4741]	Enns, R., " <a href="#">NETCONF Configuration Protocol</a> ," RFC 4741, February 2006.
[RFC4742]	Wasserman, M. and T. Goddard, " <a href="#">Using the NETCONF Configuration Protocol over Secure SHell (SSH)</a> ," December 2006.
[RFC4743]	Goddard, T., " <a href="#">Using NETCONF over the Simple Object Access Protocol (SOAP)</a> ," December 2006.
[RFC4744]	Lear, E. and K. Crozier, " <a href="#">Using the NETCONF Protocol over the Blocks Extensible Exchange Protocol (BEEP)</a> ," December 2008.
[RFC5539]	Badra, M., " <a href="#">NETCONF over Transport Layer Security (TLS)</a> ," May 2009.
[RFC5717]	Lengyel, B. and M. Bjorklund, " <a href="#">Partial Lock Remote Procedure Call (RPC) for NETCONF</a> ," December 2009.
[XML-NAMES]	Hollander, D., Tobin, R., Thompson, H., Bray, T., and A. Layman, " <a href="#">Namespaces in XML 1.0 (Third Edition)</a> ," World Wide Web Consortium Recommendation REC-xml-names-20091208, December 2009 ( <a href="#">HTML</a> ).
[YANG]	Bjorklund, M., " <a href="#">YANG - A data modeling language for NETCONF</a> ," June 2010.
[YANG-TYPES]	Schoenwaelder, J., " <a href="#">Common YANG Data Types</a> ," April 2010.
[xmlschema-1]	

Biron, Paul V. and Ashok. Malhotra, " <a href="#">XML Schema Part 1: Structures Second Edition W3C Recommendation 28 October 2004</a> ," October 2004.
--

---

## 10.2. Informative References

[TOC](#)

[RFC0959]	Postel, J. and J. Reynolds, " <a href="#">File Transfer Protocol</a> ," STD 9, RFC 959, October 1985 ( <a href="#">TXT</a> ).
[RFC2616]	<a href="#">Fielding, R.</a> , <a href="#">Gettys, J.</a> , <a href="#">Mogul, J.</a> , <a href="#">Frystyk, H.</a> , <a href="#">Masinter, L.</a> , <a href="#">Leach, P.</a> , and <a href="#">T. Berners-Lee</a> , " <a href="#">Hypertext Transfer Protocol -- HTTP/1.1</a> ," RFC 2616, June 1999.
[RFC2818]	Rescorla, E., " <a href="#">HTTP Over TLS</a> ," RFC 2818, May 2000 ( <a href="#">TXT</a> ).
[RFC3688]	Mealling, M., " <a href="#">The IETF XML Registry</a> ," January 2004.

---

## Authors' Addresses

[TOC](#)

	Mark Scott
	Ericsson
	3500 Carling Ave
	Nepean, Ontario K2H 8E9
	Canada
Email:	<a href="mailto:mark.scott@ericsson.com">mark.scott@ericsson.com</a>
	Martin Bjorklund
	Tail-f Systems
	Klara Norra Kyrkogata 31
	SE-111 22 Stockholm,
	Sweden
Email:	<a href="mailto:mbj@tail-f.com">mbj@tail-f.com</a>