NETWORK MANAGEMENT
INMF: Advanced Mechanisms

Distributed Management

- Initial efforts were created on standard INMF working groups:
  - ✓ **RMON MIB** – for monitoring the Internet traffic (all layers) interfaces on a LAN (Remote Monitoring MIB).
  - ✓ **M2M MIB** – for added communication between managers (Manager to Manager MIB).
- Later, a special IETF working group was created, the DISMAN Group (IETF Distributed Management Working Group).

NETWORK MANAGEMENT
INMF: Advanced Mechanisms

DISMAN: Alarms

- This feature permits the remote configuration of management agents to implement alarms.
- It is a local process in the agent that is continuously monitoring the value of the instances of the MIB objects. This process compares these values and sets/triggers the associated alarms when set of pre-defined threshold values are attained.
- So, this process of monitoring and alarm setting/triggering is a management activity delegated by the manager to the agent.
- This concept was firstly introduced, in an implied form, on the RMON MIB.

NETWORK MANAGEMENT
INMF: Advanced Mechanisms

DISMAN: Events

- Events permit agents to inform, in an unsolicited way, managers about occurrences of pre-defined events.
- Events and alarms are complementing but also overlapping features as they are based, more or less, on the same concept. Both approaches could be united on a common feature. Although events convey a more generic concept there has been more developments on the alarm mechanism.
- Alarms and events generate notifications (or traps) that can be logged through a special MIB – the **Notifications Log MIB**.

NETWORK MANAGEMENT
INMF: Advanced Mechanisms

DISMAN: Delegation of Scripts

- This mechanism can be used to delegate management procedures from an SNMP manager to another SNMP entity (generally, an agent) that implements the referred code.
- The standard does not define or recommends any standard language or encoding technique for the script code.
- The standard also does not define a security model/system to be applied when dealing with the remote execution of management scripts.
- On the other hand, access control should be implemented using the VACM approach.

NETWORK MANAGEMENT
INMF: Advanced Mechanisms

DISMAN: Delegation of Scripts

- Mechanisms for identification of scripts (from a script repository) and their parameterization are implementation dependent and parameterization should be made offline.
- The most used methodologies either use a push method for transfer of the script code from the manager to the agent using normal MIB table manipulation techniques or a poll method from the agent which normally involves an HTTP or FTP access to an URL (passed by the manager to the agent) where the script code resides.
- It is not possible to re-use scripts.
- This type of distributed mechanism is complex to implement and the implementations are not universal.

NETWORK MANAGEMENT
INMF: Advanced Mechanisms

DISMAN: Other mechanisms...

- **Remote calculation of expressions** – this feature, defined on the **Expression MIB**, permits the computation of Boolean expressions for definition of alarm thresholds and event triggering conditions.
- **Scheduling of management procedures** – this feature can be used for scheduling of simple management procedures that can be abstracted by integer values; it is also possible to schedule management scripts if the management entity implementing the scheduler also implements the Script MIB. Scheduling of complex management procedures or conditional scheduling is not practical due to its implementation complexity so it is not conceptually supported.

NETWORK MANAGEMENT
INMF: Advanced Mechanisms

Policy Management

- **Configuration Management with SNMP, IETF Working Group.** It recommends methodologies/strategies to tackle the problem of configuration management using the regular SNMP architecture.
- **Network Management Research Group, IRTF Working Group.** This research group defined a variant of the INMF architecture that serves to create and manipulate management policies. This variant uses a higher-level modeling language, the **SMI for the Next Generation** (SMIng), to define policies and the **Structure of Policy Provisioning Information** (SPPI) is used to map SMIng specifications into **Policy Information Bases** (PIBs), which are the special MIBs used for definition of policy management objects. The SPPI standard is a sub-set of the SMIv2 specification. Due to the added complexity this variant is not commonly implemented.

NETWORK MANAGEMENT
INMF: Remote Monitoring MIB

- The main goals of the RMON MIB is **to obtain traffic statistics from all network layers in a LAN domain segment.**
- The SNMP agents that implement the RMON MIBs should be special hosts with special access and security permissions to scan/collect all needed network traffic from all logical protocol layers from all network cards on the LAN. These are known as **RMON probes**.
- The RMON probes should use **passive monitoring** of network traffic.
- There should be, at least, one RMON probe for each LAN on the network where traffic monitoring is desired.
- Each RMON probe gathers/scans network traffic at a configurable rate and calculates statistics. Normally these are long term that are then polled by the managers for further analysis. In case of special conditions (like errors) the probe can trigger alarms/notifications.

NETWORK MANAGEMENT
INMF: Remote Monitoring MIB

- A RMON probe will scan traffic data at the data link layer but, if version 2 is supported (RMONv2), it can calculate statistics about all network layers (and even about application communication protocols).
- RMONv2 supports, when compared to RMONv1, additional matrix data statistics. RMONv1 focuses on traffic statistics at the data link layer only.
- For example, an RMON probe can calculate the following statistics (for its LAN domain segment):
  - ✓ Number of packets, bytes, broadcast/multicast packets, etc.
  - ✓ CRC errors, length problem, collisions, etc.
  - ✓ Size histograms
    [<64, 65-127, 128-255, 256-511, 512-1023, 1024-1518]

NETWORK MANAGEMENT
INMF: Remote Monitoring MIB

Excerpt from RMON MIB (history/sampling configuration):

```
historyControlEntry OBJECT-TYPE
    SYNTAX     HistoryControlEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "A list of parameters that set up a periodic sampling of
        statistics.  As an example, an instance of the
        historyControlInterval object might be named
        historyControlInterval.2"
    INDEX { historyControlIndex }
    ::= { historyControlTable 1 }

HistoryControlEntry ::= SEQUENCE {
    historyControlIndex              Integer32,
    historyControlDataSource         OBJECT IDENTIFIER,
    historyControlBucketsRequested   Integer32,
    historyControlBucketsGranted     Integer32,
    historyControlInterval           Integer32,
    historyControlOwner              OwnerString,
    historyControlStatus             EntryStatus
}
```

NETWORK MANAGEMENT
INMF: Remote Monitoring MIB

Excerpt from RMON MIB (ethernet statistics):

```
EtherStatsEntry ::= SEQUENCE {
    etherStatsIndex                  Integer32,
    etherStatsDataSource             OBJECT IDENTIFIEI
    etherStatsDropEvents             Counter32,
    etherStatsOctets                 Counter32,
    etherStatsPkts                   Counter32,
    etherStatsBroadcastPkts          Counter32,
    etherStatsMulticastPkts          Counter32,
    etherStatsCRCAlignErrors         Counter32,
    etherStatsUndersizePkts          Counter32,
    etherStatsOversizePkts           Counter32,
    etherStatsFragments              Counter32,
    etherStatsJabbers                Counter32,
    etherStatsCollisions             Counter32,
    etherStatsPkts64Octets           Counter32,
    etherStatsPkts65to127Octets      Counter32,
    etherStatsPkts128to255Octets     Counter32,
    etherStatsPkts256to511Octets     Counter32,
    etherStatsPkts512to1023Octets    Counter32,
    etherStatsPkts1024to1518Octets   Counter32,
    etherStatsOwner                  OwnerString,
    etherStatsStatus                 EntryStatus
}
```

NETWORK MANAGEMENT
INMF: Remote Monitoring MIB

Excerpt from RMON MIB (alarm/event/log configuration):

```
AlarmEntry ::= SEQUENCE {
    alarmIndex                  Integer32,
    alarmInterval               Integer32,
    alarmVariable               OBJECT IDENTIFIE
    alarmSampleType             INTEGER,
    alarmValue                  Integer32,
    alarmStartupAlarm           INTEGER,
    alarmRisingThreshold        Integer32,
    alarmFallingThreshold       Integer32,
    alarmRisingEventIndex       Integer32,
    alarmFallingEventIndex      Integer32,
    alarmOwner                  OwnerString,
    alarmStatus                 EntryStatus
}
```

```
EventEntry ::= SEQUENCE {
    eventIndex          Integer32,
    eventDescription    DisplayString,
    eventType           INTEGER,
    eventCommunity      OCTET STRING,
    eventLastTimeSent   TimeTicks,
    eventOwner          OwnerString,
    eventStatus         EntryStatus
}
```

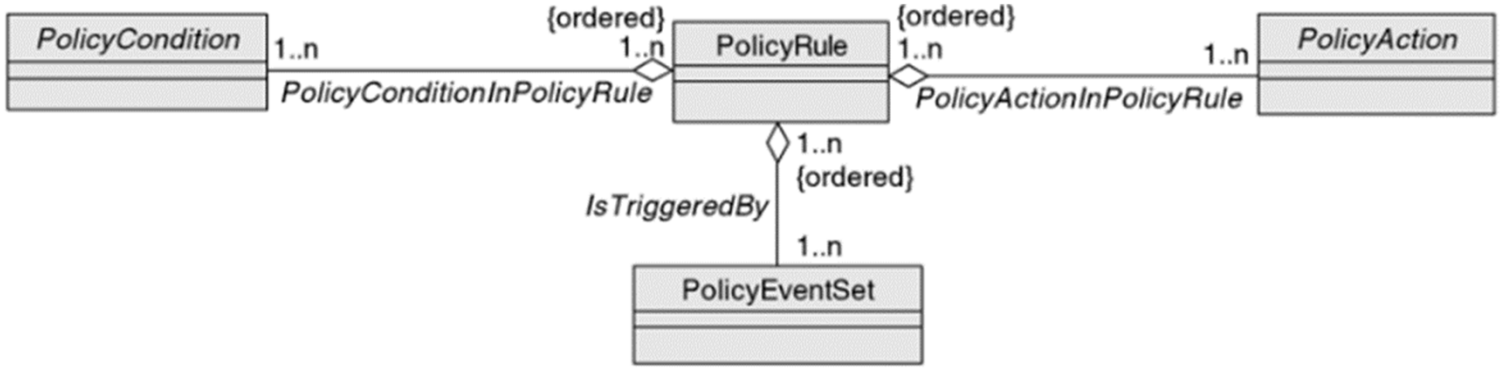
```
LogEntry ::= SEQUENCE {
    logEventIndex           Integer32,
    logIndex                Integer32,
    logTime                 TimeTicks,
    logDescription          DisplayString
}
```

NETWORK MANAGEMENT
Policy-Based Network Management

- Simplification of network management hiding low level management procedures using a standard/universal information model. It's a smart and more efficient alternative to overprovisioning of resources.
- Models management behavior, not policy implementation details.
- It's complex to implement efficiently just by using traditional architectures like the INMF/SNMP.
- Supports less well trained network administrators or system managers, permits higher-level management functionalities and directly supports automation.
- It allows higher security levels by limiting functionalities per user groups and by continuously analyzing management results.
- Supports real-time and time-critical management procedures.

NETWORK MANAGEMENT
Policy-Based Network Management

- Management policies are defined by a set of rules.
- Each rule includes conditions and actions.
- If a condition is met then a pre-determined action is executed.
- Events can be defined for automatic verification of conditions.



NETWORK MANAGEMENT
Policy-Based Network Management

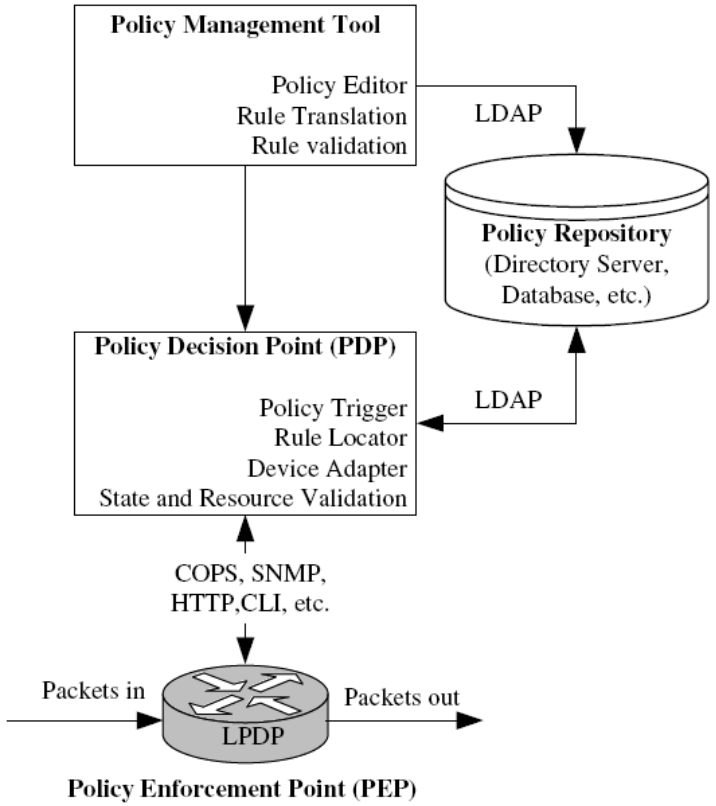
IETF RFC 3198

A standardization effort for policy management in the Internet. It integrates the following components:

- **Policy Decision Point** (PDP) – entity that manages/calculates policies for itself and to other entities in the management domain.
- **Policy Enforcement Point** (PEP) – entity that applies policies.
- **Policy Management Tool** – entity where policies can be edited, validated and converted.
- **Policy Information Repository** – entity that stores policy information (or PIB, like a MIB for policies).

NETWORK MANAGEMENT
Policy-Based Network Management

IETF RFC 3198



NETWORK MANAGEMENT
Policy-Based Network Management

IETF RFC 3198

A standardization effort for policy management in the Internet.
It includes the following standards:

- Common Open Policy Service
  (COPS, RFC 2748)
- Common Open Policy Service for Policy Provisioning
  (COPS-PR, RFC 3084 – used together with COPS)
- Structure of Policy Provisioning Information
  (SPPI, RFC 3159)
- Policy Information Bases
  (defined using the SPPI syntax)

NETWORK MANAGEMENT

Autonomic Network Management

A step further to automation…

- Supports building of automated systems to deal with higher management complexity.
- Reduces human/manual intervention (less personnel).
- Higher efficiency by constant feedback and faster readjustments of the management process.
- Behavior represented by management policies.
- It aims to implement a set of management self-functions: self-protecting, self-configuring, self-healing and self-optimizing.
- Supports Context-Awareness service management.

NETWORK MANAGEMENT

Autonomic Network Management

It has several key requirements:

- Policy-based management architectures and communication protocols that support automation, real-time and time-critical management procedures.
- Truly universal information models that can support the integration of any management service with any level of functionality.
- Management of network services are organized on layers of different abstraction, implementing different levels of management functionalities.

NETWORK MANAGEMENT
Network Services Management

Towards a full integration of management procedures…

FCAPS is no longer used to define management services and procedures, which can be divided into three main groups of services/procedures:

- **Operational Services Management**
  - Monitoring
  - Configuration
- **Administrative Services Management**
- **Strategic Services Management**

NETWORK MANAGEMENT
Network Services Management

Operational Management:

- Monitoring – Faults, Configuration, Quality of Services, Security & Accounting, etc.
- Configuration – Adaptation of strategic policies into administrative policies; Installation of all hardware devices and cabling; Installation and setup of all software needed; Configuration of devices and software services; etc.

NETWORK MANAGEMENT
Network Services Management

Administrative Management:

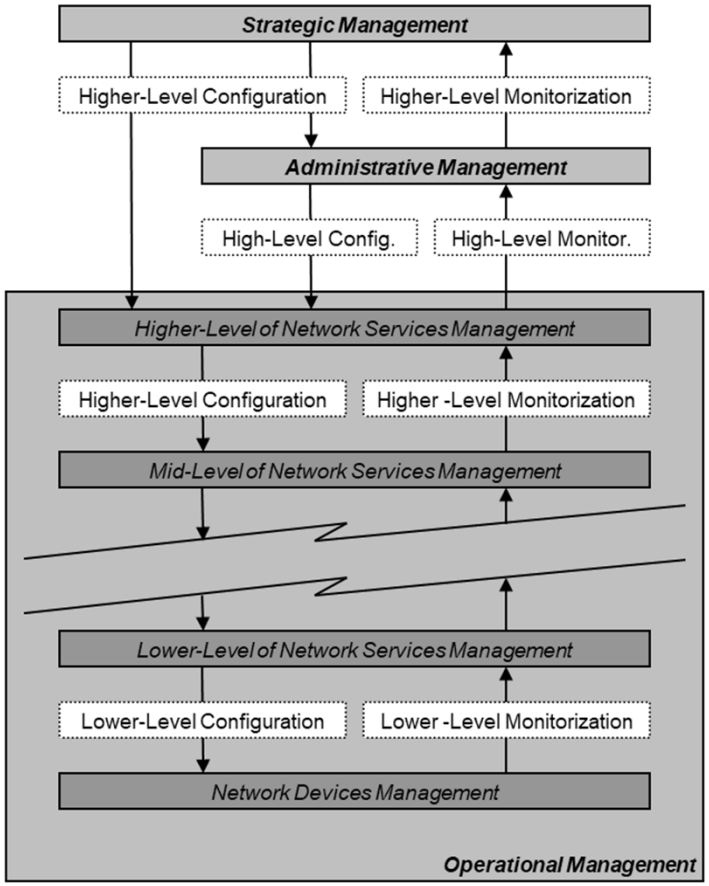
- Administrative assignment of addresses – Network addresses, domain names and interface addresses, etc.;
- Consumer and internal user support monitoring – Human live contacts, standardized consumer applications reports, snail-mail, internal communication documents, etc.;
- Consumer information data processing – Consumer reports or marketing documentation, etc.;
- Internal technical and informational reports – Input/output documents from/for operational and strategic management.

NETWORK MANAGEMENT
Network Services Management

Strategic Management:

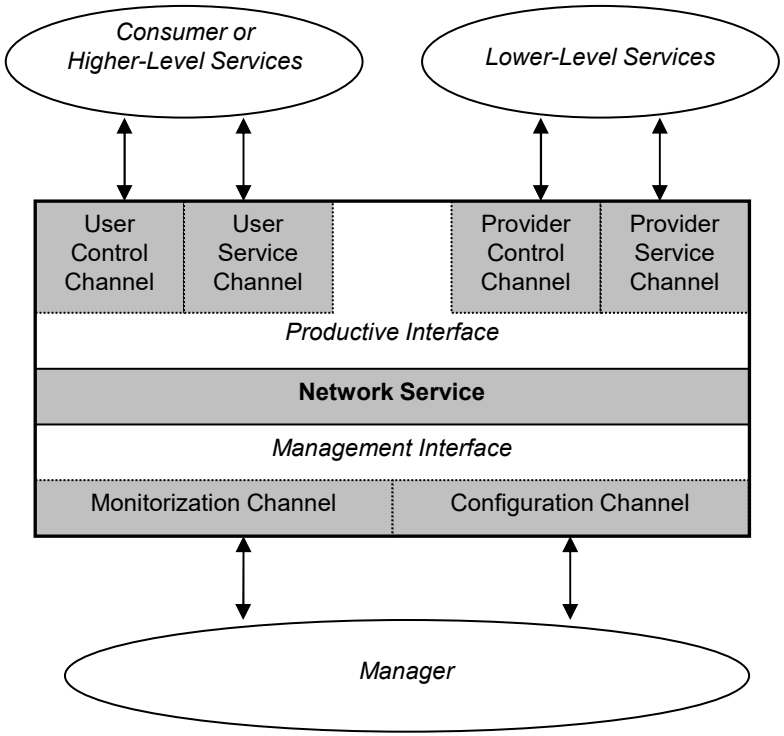
- Monitoring of administrative & operational reports – User satisfaction reports, operational management deployment and monitoring, etc.;
- Definition of services access policies – Security requirements, levels of quality of services and their pricing; naming & addressing strategies, etc.;
- Definition of a global consumer marketing and support policy;
- Coordination between all entities supporting the entire network operation, including network services management.

NETWORK MANAGEMENT
Network Services Management



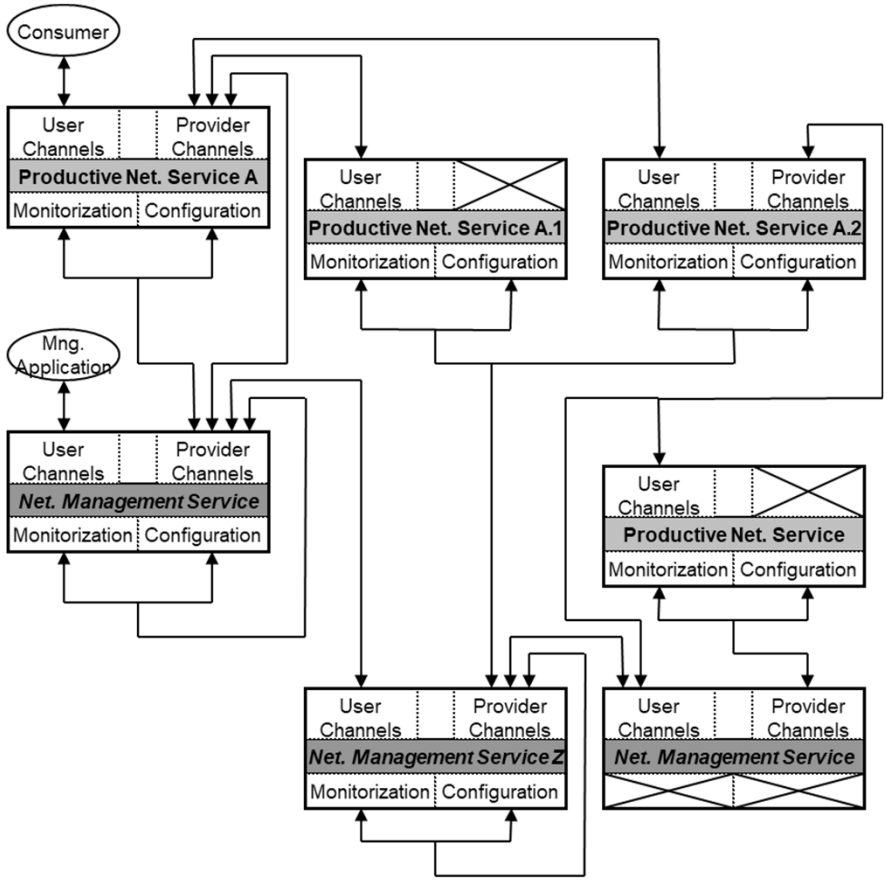
NETWORK MANAGEMENT
Network Services Management

Definition of a Network Service…



NETWORK MANAGEMENT
Network Services Management

An example…



NETWORK MANAGEMENT
Network Services Management

A framework for deployment of network services management should take into consideration the following components:

- **Architecture** – definition of the management entities, how they relate, the access and security policies, technologic constraints and administrative policies.
- **Communications Model** – definition of the services that compromise the entire management framework and the syntaxes and semantics of their user and management interfaces.
- **Functional Model*** – definition of the available Service Management Functions; in the case of management services, management procedures and their functionalities/parameters.

NETWORK MANAGEMENT
NETCONF

An alternative framework to SNMP created by Cisco Systems and that it was targeted for **configuration management** of routers. The key requirements (or selling points ;) were:

- Easy to use (at least easier than SNMP);
- Clear distinction between configuration data, operational state and statistics; support to multi-configuration data;
- Support for configuration of the entire network as a whole (master databases), rather than individual devices;
- Support configuration transactions across a number of devices;
- Support consistency checks of configurations;

NETWORK MANAGEMENT
NETCONF

An alternative framework to SNMP created by Cisco Systems and that it was targeted for **configuration management** of routers. The key requirements (or selling points ;) were:

- Support to role-based access control models and the principle of least privilege (à la Cisco Systems);
- Support for consistency and security checks of access control lists across devices in the network;
- Support to data-oriented but, more importantly, to task-oriented access control;
- Still maintain the same level of protocol simplicity and resource consumption of the SNMP framework.

NETWORK MANAGEMENT
NETCONF

The two essential components are:

- The NETCONF specification on RFCs 4741, 6241 and 8342 that defines a small set of operations, just like SNMP, but these are conveyed on a XML-based remote procedure call mechanism (encoded with UTF-8) that is task-oriented; it should be encapsulated over a connection-oriented transport service (usually SSH); all interactions between agents and managers should be carried out in NETCONF sessions; thus, security guarantees deployment is session-oriented.
- The **Yet Another Next Generation** (YANG) language for the specification of the configuration datastores (just like SMI for the SNMP framework), RFC 7950.

NETWORK MANAGEMENT
NETCONF

Operations defined on NETCONF RFC 4741 (later there was an update on RFC 6241 and another on RFC 8342):

Operation	Description
`<get>`	Retrieve running configuration and device state information
`<get-config>`	Retrieve all or part of specified configuration datastore
`<edit-config>`	Loads all or part of a configuration to the specified configuration datastore
`<copy-config>`	Replace an entire configuration datastore with another
`<delete-config>`	Delete a configuration datastore
`<commit>`	Copy candidate datastore to running datastore
`<lock> / <unlock>`	Lock or unlock the entire configuration datastore system
`<close-session>`	Graceful termination of NETCONF session
`<kill-session>`	Forced termination of NETCONF session