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

Internet-Drafts are working documents of the Internet Engineering
Task Force (IETF), its areas, and its working groups. Note that
other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html

This Internet-Draft will expire on July 18, 2014.

Copyright Notice

Copyright (c) 2014 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

Xu, et al Expires July 18, 2014 [Page 1]
1. Introduction

YANG [RFC6020] is a data modeling language used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF) [RFC6241]. A small set of built-in data types are defined in [RFC6020], a collection of common data types are further defined in [RFC6991].

This document defines a YANG data model for Network Functions Virtualization (NFV).

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

Virtual Machine (VM): A program and configuration of part of a host computer server. Note that the Virtual Machine inherits the properties of its host computer server e.g. location, network interfaces.

Virtualised Network Function (VNF): An implementation of an executable software program that constitutes the whole or a part of an NF and can be deployed on a virtualisation infrastructure.

VNF Descriptor (VNFD): A configuration template that describes a VNF in terms of its deployment and operational behavior, and is used in the process of on-boarding and instantiating a VNF. The deployment behavior describes the NFV Infrastructure resources that a VNF instance requires whereas the operational behavior describes the VNF instance topology and VNF instance lifecycle operations.

VNF Forwarding Graph: A graph specified by a Network Service Provider of bi-directional logical links connecting NF nodes where at least one node is a VNF through which network traffic is directed.

VNF Instance: a run-time instantiation of the VNF, resulting from completing the instantiation of VNF, using the VNF deployment and operational information captured in the VNFD, as well as additional run-time instance-specific information and constraints.

Virtualization Deployment Unit (VDU): a construct that can be used in an information model, supporting the description of the deployment and operational behaviour of a subset of a VNF, or the entire VNF if it was not componentized in subsets.

4. Data Model of Virtual Network Function Descriptor (VNFD)

The main objective of VNF modeling will include:

- Basic VNF attributes: VNF name, function description, sharing or non-sharing attribute.

- Deployment attributes: environment requirements of VNF deployment such as the number of VMs, virtual CPU, memory and disk requirements, image of each VM, and QoS requirements such as bandwidth and delay of VNF.
- Interface attributes: external interface, such as interface type, configuration parameters of these interfaces.

4.1. VNFD Module Structure

```plaintext
+--rw VNF-descriptor
    +--rw general-information
        | +--rw name? string
        | +--rw description? string
        | +--rw vendor? string
        | +--rw version? uint8
        | +--rw sharing? enumeration
    +--rw deploy-information
        | +--rw virtualization-deployment-unit* [index]
        |     | +--rw index uint16
        |     +--rw require-resource
        |     |     | +--rw CPU-unit? uint16
        |     |     | +--rw memory-size? uint64
        |     |     | +--rw disk-size? uint64
        |     | +--rw image-ref? string
    +--rw external-interface* [name]
        | +--rw name string
```
4.2. VNFD YANG Module

module ietf-vnfd {

    namespace "urn:ietf:params:xml:ns:yang:ietf-vnfd";
    prefix vnfd;

    organization "Huawei Technologies";
    contact "Yuanlong Jiang
        <mailto:jiangyuanlong@huawei.com>
        Weiping Xu
        <mailto:xuweiping@huawei.com>
        Cathy Zhou
        <mailto:cathy.zhou@huawei.com">

    description
    "This module contains a collection of YANG definitions for
    managing VNFD.
    Copyright (c) 2013 IETF Trust and the persons identified as
    authors of the code. 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)."

revision 2014-01-03 {
    description
    "Initial revision.";
}

container VNF-descriptor {
    description "A configuration template that describes a VNF.";
    container general-information {
        description "General information of a VNF.";
        leaf name {
            type string;
            description "the name of this VNF.";
        }
        leaf description {
            type string;
            description "description of this VNF.";
        }
        leaf vendor {
            type string;
            description "vendor of this VNF.";
        }
    }
}
type string;

description "the vendor generating this VNF.";

leaf version {
    type uint8;

description "the version number.";
}

leaf sharing {
    type enumeration {

        enum non-sharing {
            value "0";

            description "The VNF could not be shared by more than
            one consumer.";

        }

        enum sharing {
            value "1";

            description "The VNF could be shared, such as virtual
            STB is shared by more than one consumer.";

        }

    }

    description "The flag shows whether the VNF could be
    shared by more than one consumer.";
}
list virtualization-deployment-unit {
    key index;
    description "Used to store the deployment parameters of VNF.";

    leaf index {
        type uint16;
        description "the VDU index.";
    }
}

container require-resource {
    description "The required source for the VNF.";

    leaf CPU-unit {
        type uint16;
        description "The virtual CPU unit numbers";
    }

    leaf memory-size {

leaf disk-size {
  type uint64;
  description "The virtual disk size, unit:MByte.";
}

leaf image-ref {
  type string;
  description "the software image associated with the VNF.";
}

list external-interface {
  key name;
  description "The interface connected to other VNF.";

  leaf name {
    type string;
  }
}
5. VNF instance Data Model

The main objective of VNF modeling will include:

- Operational attributes: which defines the operational and management behavior, such as start, stop, pause, migration, etc.

5.1. VNF instance Module Structure

The data model for VNF instance has the following structure:

```
module: ietf-vnf-instance
  +--rw VNF-instance
    +--rw id? uint32
    +--rw VNFD-name? string
    +--rw operation
      +--rw action? enumeration
```
5.2. VNF instance YANG Module

module ietf-vnf-instance {


    prefix vnf;

    import ietf-inet-types { prefix inet; } 

    organization "Huawei Technologies";

    contact "Yuanlong Jiang

        <mailto:jiangyuanlong@huawei.com>

Expires July 18, 2014
description

"This module contains a collection of YANG definitions for managing VNF instance.

Copyright (c) 2013 IETF Trust and the persons identified as authors of the code. 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).";

revision 2014-01-03 {

description

"Initial revision.";

}

Expires July 18, 2014
container VNF-instance {
    description "VNF instance.";
}

leaf id {
    type uint32;
    description "the instance id.";
}

leaf VNFD-name {
    type string;
    description "the name of VNF descriptor.";
}

container operation {
    description "Performing an operation on VNF.";
}

leaf action {
    type enumeration {
        enum start {
            description "Start a VNF instance.";
        }
        enum stop {
            
        }
    }
}

Expires July 18, 2014
enum pause {
    description "Pause a VNF instance.";
}

description "Stop a VNF instance.";

enum migrate {
    description "Pause a VNF instance.";
}

description "Pause a VNF instance.";

enum scale-up {
    description "Add resource to a VNF instance.";
}

description "Add resource to a VNF instance.";

enum scale-down {
    description "Add resource to a VNF instance.";
}

description "The operation on VNF";

container parameter {
    description "The parameters that associated with scale procedure.";
}

choice action {

}
description "Different parameter with different action."

case scale {
    leaf CPU-unit {
        type uint16;
        description "The virtual CPU unit numbers";
    }

    leaf memory-size {
        type uint64;
        description "The virtual memory size, unit:KByte.";
    }

    leaf disk-size {
        type uint64;
        description "The virtual disk size, unit:MByte.";
    }
}

case migration {
    leaf destination-location {
        type inet:ip-address;
        description "The ip address of the destination location.";
    }
}
}
6. Security Considerations

TBD.

7. IANA Considerations

TBD.

8. References

8.1. Normative References


8.2. Informative References

ETSI GS NFV 003, Network Functions Virtualisation (NFV) – Terminology for Main Concepts in NFV, 2013

ETSI GS NFV 003, Network Functions Virtualisation (NFV) – Architectural Framework, 2013


9. Acknowledgments

TBD

Authors’ Addresses
Weiping Xu  
Huawei Technologies Co., Ltd.  
Bantian, Longgang district  
Shenzhen 518129, China  
Email: xuweiping@huawei.com

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

Cathy Zhou  
Huawei Technologies Co., Ltd.  
Bantian, Longgang district  
Shenzhen 518129, China  
Email: cathy.zhou@huawei.com