IBNemo BOF S. Hares
Internet-Draft Huawei
Intended status: Standards Track October 19, 2015
Expires: April 21, 2016

Intent-Based Nemo Overview
draft-hares-ibnemo-overview-01

Abstract

As IP networks grow more complicated, these networks require a new interaction mechanism between customers and their networks based on intent rather than detailed specifics. An intent-based language is need to enable customers to easily describe their diverse intent for network connectivity to the network management systems. This document describes the problem Intent-Based NEtwork Modelling (IB-NEMO) language is trying to solve, a summary of the use cases that demonstrate this problem, and a proposed scope of work. Part of the scope is the validation of the language as a minimal (or reduced) subset.

The IB-NEMO language consists of commands exchanged between an application and a network manager/controller. Some would call this boundary between the application and the network management system as northbound interface (NBI).

IB-NEMO focuses on creating minimal subset of the total possible Intent-Based commands to pass across this NBI. By creating a minimal subset (about 20% of the total possible) of all intent commands, the IB-NEMO can be a simple Intent interface for most applications (hopefully 80%). Part of validation of this command language is to to provide test cases where a set of commands are used to convey information for a use case which result in a particular data model in the network controller.

Status of This Memo

Table of Contents

1. Introduction 3
1.1. Where to start 3
1.2. FAQ 4
1.3. Definitions and Acronyms 7
2. Motivation for Intent Interfaces 8
2.1. Challenges in Intent-Based NEtwork MOdeling 8
2.2. Roles and User specific network information 9
2.3. What is a simple Intent-Based Protocol? 10
2.4. Intent-Based NBI Open Source is heading toward Products . 11
2.5. IB-NEMO Intent NBI is Synergistic to NETCONF and I2RS . . 12
2.6. Rest of Document 12
3. Scope 13
3.1. In-Scope 13
3.2. Out-of-Scope 13
4. Use cases for Intent-Based IB-NEMO 13
4.1. Virtual Wide-Area Network (WAN) 14
4.2. Virtual Data Center 15
4.3. Bandwidth on Demand 16
4.4. Service Chaining 18
5. Gap Analysis and where IB-NEMO Fits 19
5.1. IETF Working groups Gap Analysis 19
5.2. ODL Open-Source 20
5.3. Open Stack Policy initiatives 20
6. From Open Source and IRTF to IETF 21
7. IANA Considerations 22
8. Security Considerations 22

9. Informative References 22
Author's Address 22

1. Introduction

This document describes the problem Intent-Based Network MOdeling (IB-NEMO) language is trying to solve, a summary of the use cases and a proposed scope of work.

IB-NEMO language is a set of commands that allows an application to express what it wants (its intent) for a network to the network management system (or network controller). Some would describe the interface between the application and a network as the north bound interface (NBI) from the network manager. This paper will utilize that term to indicate the point the IB-NEMO commands are exchanged across. Intent simply means the user tells the network what the user wants, but not how to do it. Network provisioning can then creatively fulfil the user's desire. The key challenge is to provide the user with tools to express what the user wants.

Creating a Intent-Based language with a minimal set of commands requires boiling down the possible alternative to a minimal subset. These minimal set of commands will be adapted to different contexts by providing additional context. For networking, an example of this additional context may be the name to address mapping for the nodes an application desires to connect.

To test IB-NEMO language exchange, the working group must select use cases and develop prototypical data models that should occur when IB-NEMO commands are exchanged.

1.1. Where to start

In the spirit of minimalism, this introduction starts with a 5 question FAQ (frequently asked questions) for those who are familiar with the concepts of Intent-Based networking to answer "what is Intent-Based NEMO". If the FAQ answers your questions, jump off to the use cases in this document or the [I-D.xia-sdnrg-nemo-language] along with its management yang modules [I-D.zhou-netmod-intent-nemo].

If you are new to the Intent-Based networking, you'll want to read through the motivation section before looking at the rest of the document.

The purpose of this document is simple: to provide others outside the project with "what, when, where, how, and why" the IB-NEMO network language should be standardized in the IETF as part of the larger Intent-Based network effort.

1.2. FAQ

Q1: There are many industry forums working on an Intent-based policy interface for applications. Why should the IETF form a Working Group to examine an Intent-Based language?

Over the years industry forums have tried to create a mosaic of standards groups where each standards group focuses on it's key role. IETF has focused its efforts on protocols that communicate across the IP network, and management protocols to manage these efforts.

The Intent-Based Network Modelling (IB-NEMO) language is communicated between an application and a network management system that controls traffic through the network. Different forums may call this network management different names (E.g. SDN controller or centralized controller or others).

IB-NEMO seeks to provide a minimal set of commands to express the intent from an application to the network management system which is controlling the networks.

Q2: Can Intent North Bound Interfaces (NBIs) control more than networks?

A user may use Intent-based language commands to control storage or CPU cycles, but an intent-based networking language focuses on networks.

Telefonica and some of the cable operators supporting this work want to control virtual networks, service-based forwarding in networks or data center networks, home-networks, and mobile networks. If Intent based networking is successful, then the community may turn to controlling networks plus storage plus CPU. The group is starting with what they know.

The [I-D.xia-sdnrg-nemo-language] focuses on three basic components: logical node, logical link, and a logical data flow.

Q3: Why focus on creating a minimal set of commands? How will you control all of the network management devices that control the network?

IBNemo design goals are to create a simple language with a minimal set of commands so that most applications can easily use this interface to establish network connections. Often most application users (say 80%) using a language utilize only 20% of the operations. We'll call this within this paper as the 80/20 rule of communication.

The IB-NEMO commands [I-D.xia-sdnrg-nemo-language] allows groups of applications to simplify the interface by providing the capability to transfer a data model that can store common information (e.g. names or addresses) for nodes and links plus rate of data flow (e.g. 10Gbit). As an example, an application for a home-network on a cable network can simply load one set of data from a library and pass them to the network management system. Applications for virtual networks for a company could load a different set of data from a library and send it to the network management system.

The goal of this language is not to support all possible Intent language commands nor all network management systems. The intent is to work within the 80/20 rule.

Open-Daylight (ODL) has three Intent-Based Code projects:

o Network Intent Composition (NIC) (https://wiki.opendaylight.org/view/ Project_Proposals:Network_Intent_Composition) (ODL:NIC),

o Open Daylight NEMO (ODL NEMO) https://wiki.opendaylight.org/view/ NEMO:Main, and

o Group Based Policy (ODL-GBP) (https://wiki.opendaylight.org/view/ Group_Based_Policy_(GBP)).

The ODL-NIC project is creating a Intent based interface whose focus is to include all necessary intent commands in the interface between the application and the network management system. The ODL NEMO project is creating an interface with a minimal set of intent commands. The ODL-GBP sees Group-based policy as the automation of Intent by creating contracts between groups of endpoints.

Q4: Is it time for IETF standardization?

An Open Source release of the Open Daylight code for IB-NEMO (ODL NEMO)under the Open Daylight NEMO occurred July of 2015. Releases from July 2015 will contain versions of the IBNemo language interface.

The IB-NEMO project team is working with the OPNFV Movie project (https://wiki.opnfv.org/movie) to provide use cases that will allow matching the ODL code bases with the OPNFV deployments. Much of the open source code from ODL and OPNFV open source projects has moved into the product code bases of vendors.

Now is the time for the IETF to begin to standardize the interoperability of the IB-NEMO commands as the ODL Nemo code is being distributed in these open source code bases.

Telefonica, BT and the DOCSIS group see this as a key way to speed up provisioning by obtaining their users desires via the Intent Interface.

Q5: What data models will IB-NEMO focus on?

IB-NEMO language when passed to a network management system should fill in a set of data in a data model.

IB-NEMO work standardization effort is focused on providing a suite of test scenario for applications and network management systems. Each test scenario will provide the following:

- a description of the test's context,

- a set of Intent Based commands to be sent from the application,

- yang data model used by the network management system,

- a set of information that should loaded in the network management systems' data model.

The purpose is to indicate what service data models (such as the L3VPN service data model) input should be filled in by the IBNemo commands.

IB-NEMO work is not to create the service yang data models, but to describe how these data models might be filled in by IB-NEMO commands.

IB-NEMO work plan does not focus on being an automation architecture or protocol. ANIMA is working on this in the IETF.

```
                  context library
                       :
         +-----------:-----+
         |  application    |
         +-------||--------+
                 || http with IB-NEMO
                 || commands
    +-----------------------------------------+
    | network    .............   +=======+   |
    |management  : NEMO      :   | NEMO  |   |
    | system     : Intent    ===== Models|   |
    |            : Engine    :   | for   |   |
    |            ......|......   |content|   |
    |            :yang models:   +=======+   |
    |            :services   :               |
    +-----------------------------------------+
```

1.3. Definitions and Acronyms

ETSI: European Telecommunications Standards Institute

Intent-Based Interface: An interface which tells what what to do (go to store) rather than how to do it. (Travel 5 miles down this road to SAMS Club store)

Intent-Based interface: A intent-based command language interface consists of commands exchanged between an application and the network management system.

NETCONF: The Network Configuration Protocol

NFV: Network Function Virtualization

ODL: Open Daylight project

ODL NIC: ODL Network Intent Composition

ODL NEMO - Open Daylight NEMO

ODL GBP: Open Daylight Group Based Policy project

ONF: Open Network Forum

RESTCONF:REST-like protocol that provides a programmatic interface over HTTP for accessing data defined in YANG, using the datastores defined in NETCONF.

2. Motivation for Intent Interfaces

The IP networks within Carriers, Data Centers, Cloud provider, and Enterprises continue to grow in size and complexity. Simultaneously, the services that are demanded by customers, particularly the upper layer applications, are becoming more and more complicated. The users of these services demand that the services be available to mobile devices (E.g. iPADs, smart phones) as well as their desktops. New applications that demand these services have a short life span (months rather than years). The current rigid service models are lacking the flexibility to meet this combination of requirements and scenarios.

Recent efforts have looked to open source and open APIs for the IP devices and networks as a way to replace the rigid service models with fast-paced development. ETSI's NFV group, CableLabs DOCSIS (docsis.org), and ONF Intent-Based NBI (North-Bound interface) are industry forums looking at Intent based open APIs. OPNFV Movie project (https://wiki.opnfv.org/movie) is examining the intent-based use cases for OPNFV (https://www.opnfv.org/). The use cases in this document encapsulate many of the use cases discussed with operators and vendors individually or within these forums.

The idea of intent can be summed up in a simple phrase: "Do not tell me what to do, tell me what you want". Traditional networking configures devices, network protocols, and topologies within a network. It is network-device centric. Intent-based networking focuses on what an application needs from the network. It is application-centric. In Intent-based networking, the network provisioning or network automation is free to operate in any way it chooses as long as it provides the application the requested service.

Intent-based network models present the network as the application would see it. Intent-Based NEMO utilizes the application-centric view in its modelling of a network. These models may hide details the application does not need to know.

2.1. Challenges in Intent-Based NEtwork MOdeling

The challenges in Intent-Based NEMO are:

1. create a common definition of intent,

2. create a common architecture for an Interoperable Intent-Based Northbound API,

3. create a standard and interoperable command language the applications can use communicate with the network management systems, and

4. create a way to reduce the complexity that the context places on the intent engine.

The ODL projects, the Distributed Management Task Force (DMTF - www.dmtf.org), Open Networking Foundation (ONF) Intent-Based Northbound interface(NBI) working group (ONF Intent NBI WG) (https://www.opennetworking.org/technical-communities/areas/services/1916-northbound-interfaces), and OpenStack Congress (https://wiki.openstack.org/wiki/Congress) are working on definitions of Intent.

ONF Intent NBI WG (http://www.onfsdninterfaces.org/) and ODL-NIC project are working on common architecture principles for the Intent-Based Northbound API (https://wiki.opendaylight.org/view/Network_Intent_Composition:Main) with work to define application end points (https://wiki.opendaylight.org/view/Network_Intent_Composition:Dynamic_Attributes).

IB-NEMO seeks to simply apply this evolving work by creating an interoperable set of commands that application uses to communicate with the network management system (or network controller). The IB-NEMO language interface seeks to reduces the number of commands from a full-set of commands in order by supporting a portion of the commands most often used for key use cases.

The people on the ODL NEMO project (https://wiki.opendaylight.org/view/NEMO:Main) have selected a small set of commands and created an open-source prototype. The IETF work is to review and standardize the set of commands to make sure it provides an interoperable set for all applications.

2.2. Roles and User specific network information

Authentication, Authorization and Accounting (AAA) protocols such as Diameter and Radius pass information on the access permissions that certain users or user programs have to a network or virtual network. Group based policy suggests that a group of users may share a role which is associated with a set of policies that determines the access to the network or a virtual network. Role-based network access suggests that roles can better summarize what access the user or user programs have to the network. Since IB-NEMO is trying to use prototypical use cases to test the ability of the IB-NEMO command language to create the appropriate data models in the network

management system, it is natural to use the role-based concepts of summarize these data models.

The contextual information is the characteristics which make groups of applications unique when operating over the network. Logically most of this information may be associated with roles. For example, if you have a set of users in a home communicating over a home network the characteristics which are unique is a set names and address for devices, links, and policy within the home. If it is a virtual network for a company, the unique information is the names, addresses, links, and bandwidth expected on the links along with security issues. As these examples demonstrate, an intent language contains the intent plus contextual information.

2.3. What is a simple Intent-Based Protocol?

What is a simple interface? It is said that 80% of the applications only use 20% of the commands in any API. This paper calls this the 80/20 rule of networking. A simple Intent-based command language should only supports these 20% of commands that all applications will need to exchange information with the network management system. Of course, the challenge in any simply interface is to select the 20% of commands that are being repeated used by applications.

It is also important these commands be similar to a human being's natural language for easy debugging.

The challenge is that different industries may have a different 20% of commands that are commonly used. The NEMO Project teams in the ODL NEMO project and OPNFV Movie project are seeking uses cases to determine if there is common set of use cases that vary just by context. For example, a global L3VPN for a company with three sites may be similar to a three site L3VPN across a cable network.

After getting a set of uses cases, creating a simple interface is a four step repetitive process:

1. find use cases,

2. develop prototype code,

3. do early testing at proof of concept demonstrations and hack-a-thons

4. work with many vendors to clarify language to make the language small and interoperable, and

5. go back to step 1

Where is NEMO is this process?

IB-NEMO has gone through steps 1-3. Use cases are listed below, and the OPNFV project is working on use cases. IB-NEMO's ODL NEMO project is developing the code for the open source (ETA July release). IB-NEMO is at a stage where it needs to work in a standards body to create a small, efficient, interoperable set of commands.

The standardization through an IETF WG will help IB-NEMO to work on step 4.

2.4. Intent-Based NBI Open Source is heading toward Products

The following are Open Daylight Projects:

Open Daylight Group Based Policy (GBP)
https://wiki.opendaylight.org/view/Group_Based_Policy_(GBP)

OpenDaylight Network Intent Composition (ODL-NIC)
(https://wiki.opendaylight.org/view/
Project_Proposals:Network_Intent_Composition), and

Open Daylight Network Intent Composition: NEMO
https://wiki.opendaylight.org/view/NEMO:Main.

These are open-source coding efforts creating an intent-based northbound interface for intent-based networking.

The ODL Group Based Policy (GBP) views policy as a contract between two endpoints, and sees its work as the automation of Intent.

ODL-GBP was released in the ODL Lithium release in June of 2015.

The ODL-NIC project is creating a Northbound interface (NBI) for network orchestration systems, SDN applications, and Network operators. It may be defined as RESTCONF [I-D.ietf-netconf-restconf] protocol and/or Java APIs. This extensible interface will be designed to allow any and all new intent expressions to be exposed as part of a consistent and integrated single NBI to SDN applications. The singularity is necessary for the Composition Function to provide a comprehensive capability to manage network resources and resolve conflicts across application's intents. In a sense, the ODL-NIC project is suggesting a thin waist of a single API at the entrance to the networking layer, just as the IP protocol presents a thin waist of a single API at network layer.

ODL NIC project was released in June of 2015.

The ODL NEMO project has created a language with minimal operational commands. The ODL NEMO command language has 15 operational commands in three groups. Group 1 describes nodes, links, and flows between nodes. Group 2 deals with operational checks (query, notification, policy, connect, disconnect, session (start), and commit (end of commands). Group 3 defines the model that provides the context for nodes, links, flows and policy.

ODL NEMO project first release in ODL was in July, 2015.

ODL open source code is currently finding its way rapidly into other sources (E.g. OPNFV code base) and into products that are within 6 months to a year of release.

2.5. IB-NEMO Intent NBI is Synergistic to NETCONF and I2RS

The IETF NETCONF [RFC6541] and RESTCONF [I-D.ietf-netconf-restconf] protocols provide a network interface to the configuration and status information within IP network devices. The IETF I2RS (Interface to Routing System) WG is creating a highly dynamic network interface to the routing system which can inject or retrieve state regarding routing state, topologies, filters, and operational state. The PCE Working Group has protocols and methods to pass routing for calculation. Each of these interfaces and protocols have a purpose in managing and enhancing IP network infrastructures.

Intent Based NBI is synergistic to these IETF protocols to the devices. Synergistic means that sum of Intent Based NEMO commands + NETCONF + RESTCONF + I2RS + PCE is more than any of the parts alone. Intent Based NEMO command can signals the application's intent to a network management system which configures, manages, and monitors network devices through NETCONF, RESTCONF or I2RS protocol.

2.6. Rest of Document

Based on this motivation, the next sections discuss:

- o The Scope should the Intent-Based NBI work
- o Summary of Use cases for this scope
- o Gap Analysis and where IB-NEMO fits
- o Transition from IRTF to IETF

3. Scope

3.1. In-Scope

The initial scope of this IB-NEMO work has focused on:

1. creating a minimal set of language commands to express intent from the application to the network management device,

2. selecting use cases and associating them with prototype applications in order to determine the subset of commands that needs to be included in IB-NEMO language,

3. validating the IB-NEMO language by creating data models (which should exist in the network management system) for each application use case to determine if the language can help a network management system create the right data model

4. creating a yang data model to manage this Intent-Based Networking language, and

5. working with other forums to refine a definition of intent so that the minimal size language serves a wide range of use cases (target of 80% of known use cases) with an interoperable interface.

3.2. Out-of-Scope

The following things are outside the IB-NEMO scope:

- creating yang data models that describe service layers

- The creation of a language to communicate from a security network management system to the network security devices is outside this scope. (Work denoted as I2NSF)

4. Use cases for Intent-Based IB-NEMO

The following use cases are described in this section:

1. Virtual WAN

2. Virtual Data Center

3. Bandwidth on Demand

4. Service Chaining

4.1. Virtual Wide-Area Network (WAN)

Description: Enterprises want to set up their own virtual WAN for more control and optimization.

User Intent: Create virtual Wide-Area network between offices.

Network management systems do the following:

1. Deploy virtual routers and links for a customized topology.

2. Identify flows.

3. Steer the traffic flows though different paths. (E.g. real-time flow to go through a shortest path, and backup flow to go though a broadband path but may have more hops.)

The network management system should have a data model that captures this information. IB-NEMO commands are used by the application to pass the user's intent to set-up connections, the locations, and the type of flows.

Network operator: Creates web portal for business customers to request a WAN connecting offices. Interface request corporate ID, security ID, and a link to the payment system.

The sub-cases of this general use are the following.

Home LAN attached to Corporate Network

parental controls for child travelling outside the home

Details can be found in (draft-hares-nemo-usecases-00.txt)

```
          ==== real time (R1-)
          **** broadband
                  ....................
                  :   Virtual LAN    :
                  : (real time path) :
                  :                  :
        +-------+ : (real time path) : +---------+
        |      ===========================       |
        |       | :     e      f     : |         |
        |Beijing|-----R1- - - - - R2---| London  |
        |office | ***a|* \b  c / | d : | office  |
        +-------+ :   | * \   /  |****>+---------+
                  :   |  * \ /   |*  :
                  :   |    / \*  |*  :
                  :   |   /   \* |*  :
                  :   |  /     \*|*  :
                  :   R4- - - - R3   :
                  ....................
```

Figure 5-1:

4.2. Virtual Data Center

Description: User (corporate or home) creates a virtual data center with network. The virtual data center has a front-end network of router to exterior firewall to DMZ LAN to interior firewall to computing user.

User Intent: A Corporation wants to buy Cloud computing inside a virtual data center with secure computer cluster.

Network Service Provider: Sales person of the provider is given a reduced group of well-known building blocks (DMZ, protected area, unprotected area) and that he/she uses these blocks to compose different kinds of vDC infrastructures for the client. The sales person has an App on a PAD device that shows a cloud for the internet and the different building blocks (Exterior, DMZ, interior) and the user builds vDCs with these building blocks.

The application the sales person is running queries the network management system using IB-NEMO commands on existing model components with the potential vDC functions. The application expresses the user's desires/intent via IB-NEMO commands to the network management system.

Operator automation: Based on the context with Intent, corporate context, secure vDC context, the operator automation series will place the virtual cluster in a data center, and set-up the vDC and

the Cloud computer clusters. The Corporate customer IDs that are pushing data to this vDC will have the vDC defined in the Corporate culture.

Specific use cases from this prototypical use case are:

- o User gets clean mail services with firewall and spam mail cleaner

- o SMB Manufacturing network with Virtual DataCenter

- o SMB with Sales-Marketing accounting on Virtual Data Center

These are described in (draft-hares-nemo-usecases-00.txt)

```
   [The user simply builds this as building blocks on
    the application.]

               (internet )
              |
    ..........|...................
        +-----|------+
        |  router    |
        +-----|------+
              |
        +-----|------+
        |  firewall  |
        | exterior   |
        +-----|------+
              |
       ===============
        DMZ   |
        +-----|-------+
        |  firewall   |
        | interior    |
        +-----|-------+
              |
        ( protected  )
        (  cloud     )

          Figure 5-2
```

4.3. Bandwidth on Demand

Description: The corporate user wants to create a virtual link between remote offices and headquarters that has bandwidth that can be adjusted based on time of day.

User Intent:Corporation wants to connect branch office with corporate office with 10G of bandwidth for data flow 8am to 6pm, and 1G of bandwidth from 6pm to 8am.

User interface: A web portal allows him to login (corporate ID and security IDs) and indicate this intent via a graphic picture of his network that allows him to indicate on-demand bandwidth size and time of day.

Network Operator:Creates Web portal for business customers to put in request with corporate ID and level of security for entrance into the corporate intent site. The Web portal allows for prototypical use case (virtual WAN, Virtual DC, Bandwidth-on-demand Virtual Private Network (VPN), Service Function Chaining (SFC). The network operators store enough application-level topology that the the users intent is defined.

Operator automation: Based on the IB-NEMO queries and commands, the application will pass the User's Intent to the the provisioning software which will automatically allocate bandwidth between these two sites at the rate indicated. The access router/switch can optionally limit at a rate over this value.

Corporate Virtual Data Center information: includes the IP address, DNS names, and application addresses (Transport Ports, application identifiers) of subnet with application works on, and the applications transferring data. The corporate data also includes information on whether L2VPN or L3VPN is used by the customer.

```
==== 8am to 6pm 10GB
**** 6pm to 8am 1BG
        ....................
        :   VPN            :
        :                  :
  +-------+ :     daytime      : +---------+
  |      ===========================       |
  |       | :     e      f     : |         |
  |Branch |-----R1- - - - - R2---| HQ      |
  |office | ***a|* \b  c / | d : | site    |
  +-------+ :   | * \   /  |****>+---------+
        :   |  * \ /   |*  :
        :   |    / \*  |*  :
        :   |   /   \* |* night time
        :   |  /     \*|*  :
        :   R4- - - R3   :
        ....................
```

Figure 5-3:

The following use cases are specific examples of this prototype use case:

Home Network gaming system

Home Security system zoom-in

Application Big Data or SAP Transfers at night

Database applications contact other database applications

4.4. Service Chaining

Description: Apply several virtual network functions, such as firewall, load balancer, WAN optimization between virtual private cloud and the internet.

User Intent: User has a private cloud and wants to get a secure interface to the Internet.

Network Operator network management system defines the secure access ring of protection around the private cloud to be the following virtual network topology:

- o firewall
- o load balance
- o DPI inspection

Network Operator: Has Web portal or Phone App for business customers to put in request with corporate ID and level.

Corporate Information: Corporate context has the topology of private cloud, and the access points. The network operator will access service chaining to through a virtual access ring.

Operator automation: Based on the context of the network topology of the private cloud's link to the carrier network and the access points to service chains, the network automation sets up the traffic flow so that the traffic to and from the private cloud flows through a firewall, load balancer, and DPI inspection.

```
              (internet )
              |
    ..........|...................
        +-----|------+
        |  firewall  |
     |--|  function  |--------|
     |  +-----|------+        |
     |        |               |
     |  +-----|---------+     |
     |  | load balancer |     |
     |  | function      |     |
     |  +-----|-------|-+     |
     |        |       |       |
     |    +---|-+ +---|--+    |
     +----|DPI 1| |DPI 2 |----+
          +---|-+ +---|--+
              |       |
            ( private Cloud   )
            ( for corporation )

            Figure 5-4
```

The specific use cases for this prototype are:

Providers access edge box replaced by service chaining for wired and wireless (LTE and Wifi)

Corporate access edge box replaced by service chaining for wired and wireless

Wifi offload of LTE does service chaining to replace mobile services

5. Gap Analysis and where IB-NEMO Fits

5.1. IETF Working groups Gap Analysis

No working group is working on an Intent-Based commands.

SUPA proposes to create an information model for event-condition-action based policy.

NETCONF and NETMOD are not creating an intent-based interface.

5.2. ODL Open-Source

ODL network intent composition (ODL-NIC) is creating a full intent-based North Bound Interface. ODL NEMO is creating a set of commands with a minimal set of operations as of the open source project. The IETF IB-NEMO work will leverage the lessons learned from the ODL NEMO open source work to create a minimal subset.

OPNFV Movie project (https://wiki.opnfv.org/movie) is defining the use cases for Intent-Based networking for OPNFV. IETF IB-NEMO will expand on these use cases for non-OPNFV scenarios in Cable Networks (MSO).

5.3. Open Stack Policy initiatives

None of the Open Stack Congress work focuses on intent-based command language.

Open Stacks policy includes network, compute, and storage. Its work combines automation (scheduling of resources, monitoring cloud services, Event-Condition-Action (ECA) policy, ECA based management), store-related policy, and meta-data policy storage. The projects are:

OpenStack has Congress (https://wiki.openstack.org/wiki/Congress) with its Congress initiative which aims to provide an extensible open-source framework for governance and regulatory compliance across any cloud services (e.g. application, network, compute and storage) within a dynamic infrastructure.

SolverScheduler (Nova blueprint): The SolverScheduler provides an interface for using different constraint solvers to solve placement problems for Nova. Depending on how it is applied, it could be used for initial provisioning, re-balancing loads, or both.

Gantt: A scheduler framework for use by different OpenStack components. Used to be a subgroup of Nova and focused on scheduling VMs based on resource utilization. Includes plugin framework for making arbitrary metrics available to the scheduler.

Neutron policy group: This group aims to add a policy API to Neutron, where tenants express policy between groups of networks and ports. Policy statements are of the form "for every network flow between groups A and B that satisfies these conditions, apply a constraint on that flow". Constraints are currently allow or deny, but this may expand.

Open Attestation: This project provides an SDK for verifying host integrity. It provides some policy-based management capabilities, though documentation is limited.

Policy-based Scheduling Module (Nova blueprint): This effort aims to schedule Nova resources per client, per cluster of resources, and per context (e.g. overload, time, etc.).

Tetris: This effort provides condition-action policies (Event-Condition-Action policy). It is intended to be a generic condition-action engine handling complex actions and optimization. This effort subsumes the Runtime Policies blueprint within Nova. It also appears to subsume the Neat effort. Tetris and Congress have recently decided to merge because of their highly aligned goals and approaches.

Convergence Engine (Heat): This effort separates the ideas of desired state and observed state for the objects Heat manages. The Convergence Engine will detect when the desired state and observed state differ and take action to eliminate those differences.

Swift Storage Policies: A Swift storage policy describes a virtual storage system that Swift implements with physical devices. Today each policy dictates how many partitions the storage system has, how many replicas of each object it should maintain, and the minimum amount of time before a partition can be moved to a different physical location since the last time it was moved.

Graffiti: Graffiti aims to store and query (hierarchical) meta-data about OpenStack objects, e.g. tagging a Glance image with the software installed on that image. Currently, the team is working within other OpenStack projects to add user interfaces for people to create and query meta-data and to store that meta-data within the project's database. This project is about creating meta-data, which could be useful for writing business policies, not about policies over that meta-data.

6. From Open Source and IRTF to IETF

As discussed above, the open-source work for ODL-NIC was first released in June, 2015, and ODL NEMO released its code in July of 2015. The movement of these code sources to OPNFV (https://www.opnfv.org/) will happen rapidly, aided by the OPNFV Movie project (https://wiki.opnfv.org/movie) use case work. In order to get a command language with minimal number of operations that application vendors and network management devices agree upon, it is important to standardize the command language in IETF.

Initial concepts for IB-NEMO have been presented in IRTF's NFVrg and SDNrg to obtain initial review.

7. IANA Considerations

This draft includes no request to IANA.

8. Security Considerations

The security in a Intent-Based interface may require that most Intent-Based Networking operate across a secure transport security with encryption. However, some use cases (in-home only) or some limited data may allow an unsecured transport.

9. Informative References

[I-D.ietf-netconf-restconf]
Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", draft-ietf-netconf-restconf-07 (work in progress), July 2015.

[I-D.xia-sdnrg-nemo-language]
Xia, Y., Jiang, S., Zhou, T., and S. Hares, "NEMO (NEtwork MOdeling) Language", draft-xia-sdnrg-nemo-language-03 (work in progress), October 2015.

[I-D.xia-sdnrg-service-description-language]
Xia, Y., Jiang, S., and S. Hares, "Requirements for a Service Description Language and Design Considerations", draft-xia-sdnrg-service-description-language-02 (work in progress), May 2015.

[I-D.zhou-netmod-intent-nemo]
Zhou, T., Liu, S., Xia, Y., and S. Jiang, "YANG Data Models for Intent-based NEtwork MOdel", draft-zhou-netmod-intent-nemo-00 (work in progress), February 2015.

[RFC6541] Kucherawy, M., "DomainKeys Identified Mail (DKIM) Authorized Third-Party Signatures", RFC 6541, DOI 10.17487/RFC6541, February 2012, <http://www.rfc-editor.org/info/rfc6541>.

Author's Address

Susan Hares
Huawei
7453 Hickory Hill
Saline, MI 48176
USA

Email: shares@ndzh.com