Y. Xia, Ed.
T. Zhou, Ed.
Y. Zhang, Ed.
S. Hares
Huawei
P. Aranda
D. Lopez
Telefornica
J. Crowcroft
Cambridge University
Y. Zhang
China Unicom
May 22, 2015

Intent Common Information Model
draft-xia-ibnemo-icim-00

Abstract

Intent Common Information Model (ICIM) generalizes a unified model for expressing different layers' intent whatever role, responsibility, knowledge, etc. This document provides an information model to be inherited and expanded to construct specific intent model in different areas. According to this information model, network intent model is put forward which can satisfy users' need in different layers, such as, end-users, business developers, and network administers.

Status of this Memo

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/1id-abstracts.html

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

Table of Contents

1 Introduction 4
1.1 Terminology 4
2. Intent Common Information Model Overview 4
2.1 Elements 5
2.1.1 User 5
2.1.2 Context 5
2.1.3 Object 5
2.1.4 Result 5
2.1.5 Operation 6
2.2 Relationships in ICIM 6
2.2.1 Relationship between Result and Operation 6
2.2.2 Relationship between Object and Operation 7
2.2.3 Relationship between Object and Result 7
2.3 Intent and Policy 7
2.4 Layers of Intent 8
3. Intent Modeling 8
3.1 Intent overview 8
3.2 Top level intent expression 8
3.3 Objects in the network 9
3.4 Type of result 10
3.5 Operation composition 10
4 Security Considerations 12
5 IANA Considerations 12
6 Acknowledgements 12
7 Informative References 12

Authors' Addresses 12

1 Introduction

Intent is a new philosophy to design open interface. Opposite to the 'bottom up' method which opens what system has, Intent interface uses 'top down' method which opens what user's requirement. With this Intent interface, users just need to express what their requirements are, rather than how to implement them, so Intent interface is user friendly. Intent interface is much closer to user, but different users have different intention manifestations, which have different granularity or different level. It depends on users' role, knowledge and their purpose. Intent can be some final results of objects and also can be some specific operations on objects in specific context. Although dictating operations is the manifestation of intent, a user just need to assign the operations he cares about, and has no need to plan complete and detailed operations list for the system to achieve the intent.

Intent Common Information Model (ICIM) generalizes a generic model for expressing key components of intent interface and the relationship between these components. This document provides a common model to be inherited and expanded to construct specific intent interface in specific areas. According to this information model, intent interface in network area is put forward which can satisfy user' intention in different layers or different roles, such as, end-users, business developers, network administers, etc

1.1 Terminology

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 RFC 2119 [RFC2119].

2. Intent Common Information Model Overview

Intent Common Information Model aims to generalize a unified information model which satisfied different areas, scenarios, and other constraints. So, it is a complete and detailed information model to define the constituent elements of intent, but some elements can be omitted or implied under some special situations when using this model to express specific data model.

From the overall perspective, construction elements of intent can be generalized into user of intent who author and own this intent, intent content which is a desired purpose and the specific context for implementing intent. Furthermore, in general, person's intent content is often visioning ultimate state of some objects or performing actions to these objects, so intent content can be abstracted into object which is the target for intent, result which

is a desired state and operation which is the specific actions to achieve a purpose.

2.1 Elements

2.1.1 User

User is an abstract class which refers the subject and owner to express the intent. For example, end-users, business designers, network administrators are all instances of User class. Intent has a strong relationship with the user. So intent is different for specific user when this information model is applied to specific scenario. So when ICIM is implemented, this class will involve a role mapping.

2.1.2 Context

Context refers to a set of specific background information such as, timer, price, and so on. Context has a huge influence on a person designing a detailed plan or selecting the best program to achieve a purpose. For example, when an enterprise plans to build a dedicated connection between two sites, price and distance will be the context in this scenario. While may not be part of how an entity expresses or executes some intent, it is a factor that must be considered with the expression of intent.

2.1.3 Object

Object refers an abstract class which defines some entities affected or managed by intent. For the management, users could manage life cycle of the objects through some concrete operations, such as, create, update, delete, etc. In addition, users could use other specific operations to affect the behavior of managed objects. For example, a business designer want all traffic be filtered by a special firewall. The object of this business designers intent could be the all traffic flowing on a specific network (e.g. L3VPN), and this intent impacts the forwarding behavior of the traffic network. Object is different in specific area. In network area, object is an aggregation class with node, connection and flow. For objects, users could construct some specific objects to achieve intent, and it is also allowed for users to assign intent to existing resources.

2.1.4 Result

Result is a type of intent which refers to an ultimate state or something an individual wants to achieve. This type of intent shields difference and diversity of an environment away from the users' intent. The person just envisions the ultimate state of objects

without worrying about how to achieve it. For example, a result could be that the company accesses any sites on the Internet safely. It just defines a result that ignores technology details, such as, firewall, ACL, and so on.

Though desired state is a general requirement, violation state is another special state which has an important status when achieving integral compliance. For example, a typical scenario is all virtual machines owned by different tenants should not be deployed in a same hypervisor. This type of result just shows the undesired state which is a type of violation state, and this kind of intent should be involved in this information model.

2.1.5 Operation

Operation is a type of intent which refers to some specific actions an individual desires to take for realizing the purpose. This type of intent formulates explicit plan to realize a purpose which may take a better control of the whole system. According to the diversity of system support capability, there are large sets of operations for users to take.

Generally, operations can be divided into two categories. One is action without condition which is called "command" usually. For example, create a virtual machine. This kind of operation defines a concrete action which is executed immediately without any trigger. The other is action with condition. For this kind of operations, condition is a trigger for the action. And actions will not be executed immediately until the condition clause is tested to be true. For example, "do load balancing when the utilization of a link exceeds 80%". In this example, "utilization of a link" is the trigger, and "do load balancing" is the action. Action will not be taken until the trigger is true. Actions are different by stages which depend on the layer of intent. Actions expressed in upper layer may lead to cascaded actions in other lower layers. For example, the action "do load balancing" will bring out many actions which are depend on technologies and devices.

2.2 Relationships in ICIM

2.2.1 Relationship between Result and Operation

Users are free to express their intent, no matter it is an ultimate result or specific operations in their mind, but there are some relationships between these two basic types of intent. For result, users just need to express the goal without worrying how to implement it in a specific system which facilitates users to focus on real requirement. When achieving the ultimate result, it needs some

reasoning mechanisms to transfer it to real executable operations which are supported by specific system. So in a specific scenario, a result can generate concrete operations. For example, for a geographical distributed enterprise, its intent is constructing a dedicated line between headquarters and branches at the least cost. This intent just expresses a result without defining how to construct this dedicated line. So business designers will design the best solution and concrete operations referring capability of devices, optional programs, prices, etc.

2.2.2 Relationship between Object and Operation

Operation refers to some specific actions on some objects, so object is the target of an action. In general, any action will include some objects to execute this action. When users want to execute some actions to achieve goals, they may construct the target objects and assign specific actions on them, and it is allowed for users to use existing resources to do some operations. Though object is the target of action, it offers the constraint for optional operations. For example, for a virtual machine, the optional operations are create, delete, immigrant, etc.

2.2.3 Relationship between Object and Result

Result refers to some ultimate state for some objects. This type of intent does not define which specific operations to take but express the desired state of objects. So it is independent on objects' capability. For example, intent is all virtual machines' CPU utilization could not exceed 80%. It does not assign specific operations. So reasoning mechanism will choose suitable operations to satisfy this intent, such as, immigrant virtual machine or expand it.

2.3 Intent and Policy

In industry, Policy already has a clear definition, such as in RFC3060. Policy rule consists of an event, a set of conditions and a set of actions. When an event occurs, actions will be taken until condition clauses are evaluated to be true.

As mentioned above, intent refers to a purpose in achieving ultimate result or performing operation. The intent has a larger scope compared with the policy since Intent can express both result and operation. On one hand if a result is described by intent, there may be no specific action given to show how to achieve this intent. On the other hand, if operation described by intent, conditions of action is optional. Policy is a specific form of operation in intent.

2.4 Layers of Intent

Intent reflects a person’s mental desire which depends on person’s role, knowledge, and other contextual factors. So users in different layers may have different intent. While different layer of intent has some differences in content and implementation, the expression of each layer of intent is same which defines an ultimate goal or dictates specific operations.

For example, in the network area the intent of end-users could be safe connectivity between two sites which a technology independent and device independent requirement. For business-based network designers, the network connectivity can be selected which is device-independent but technology specific. An example of the business-based technology is the L3VPN. For network administrators, intent can be specific operations on a set of devices such as configuring IP addresses on network servers in a data center.

3. Intent Modeling

This section defines the concept and hierarchy of intent, and describes the Intent Common Information Model.

3.1 Intent overview

In general, intent is one’s specific mental activity, so it strongly depends on the subject. Different users may have different manifestations and intent. In addition, context, omitted usually, is an important factor when achieving purpose, which offers necessary background information to impact the decision. Figure 1 illustrates the overview of the intent. Figure 1 indicates that the user has intent in some context. For example, an enterprise wants to block all http traffic in work time. In this intent, the user is the enterprise, the intent is to block all http traffic in the work hours, and the context includes the definition of the "enterprise" and the "work hours".

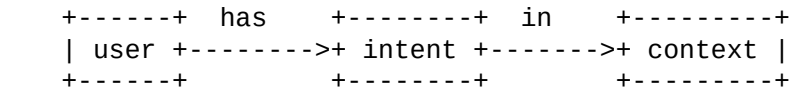

Figure 1 general prescription for intent

3.2 Top level intent expression

In Cambridge Dictionaries, the definition of "intent" is the fact that you want and plan to do something. So, in general, intent refers

to an agent's purpose in getting an ultimate result or performing some specific operation. In specific areas, these results or operations will relate to some objects. Figure 2 describes the general expression of intent.

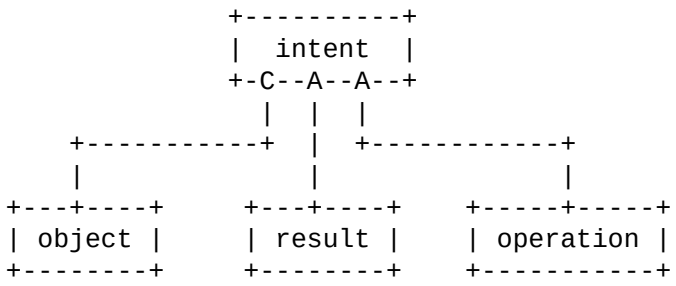

Figure 2 intent expression

One type of intent is to express key operations that a user wants to execute. The underlying intent system can generate a complete operation list from user's request. The other type of intent is to express an ultimate result or state without dictating any operations.

For example, intent of a user may be a result without defining how to realize it, such as, requiring security communication between two sites, or dictate some detailed operations in order to achieve a purpose, such as, filtering all traffics by firewall between these two sites.

3.3 Objects in the network

Object is an abstraction class which can be inherited and expanded in different area. In network area, the object, i.e. the target of intent, can be generalized into Node, Connection and Flow, as shown in Figure 3.

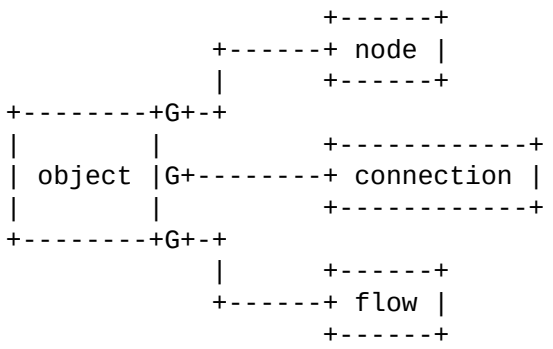

Figure 3 common objects in network area

The Node represents the functions a network node may provide in a network such as network services, forwarding functions (firewall, load balancer, virtual router, and others), or a group of network elements. A group of network elements can be a subnet, an autonomous system, or a confederation of autonomous systems.

The Connection describes logical connectivity between node entities. This connection is not limited to any physical link, but just expresses the communication capacity between nodes.

The Flow refers to the traffic in network which describes data packets have some certain common characters.

3.4 Type of result

Result refers an ultimate state which is expected or avoided. Figure 4 describes two types of result. For example, a user may express an intent is his computer's CPU utilization must less than 80%. This expression is a type of result which describes an expected state. Of course, this user can also express this intent as it's an error when his computer's CPU utilization exceeds 80%. This expression is another type of result which describe a avoid state. Users are free to describe either one.

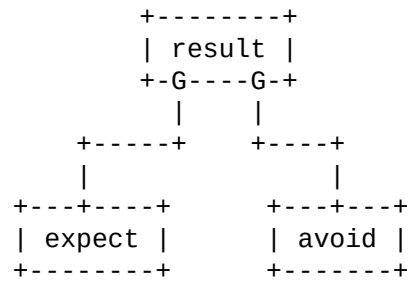

Figure 4 expression of Result

3.5 Operation composition

Operation refers to some specific actions in order to achieve some purposes. An operation must have some actions. However, if condition and constraint need to be defined in operations, it depends on specific scenario and users' require. Once a condition is involved in operation, actions will not be executed immediately until condition is true. In additional, constraint restricts action itself or the scope of action.

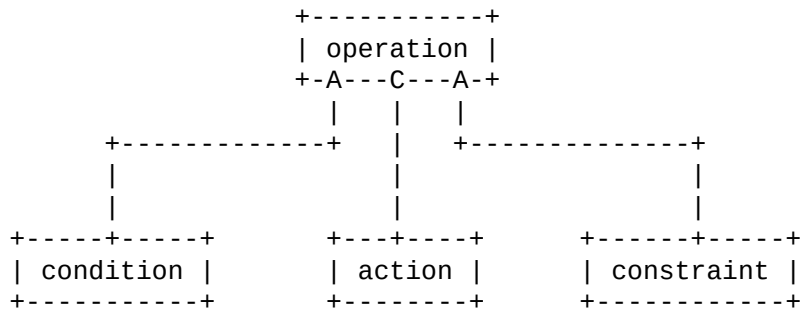

Figure 5 composition of operation

4 Security Considerations

TBD

5 IANA Considerations

This draft includes no request to IANA.

6 Acknowledgements

The authors would like to thanks the valuable comments made by Wei Cao, Sheng Jiang, Zhigang Ji, Xuewei Wang, Shixing Liu, Yan Zhang.

7 Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Authors' Addresses

Yinben Xia
Huawei Technologies Co., Ltd
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

EMail: xiayinben@huawei.com

Tianran Zhou
Huawei Technologies Co., Ltd
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

EMail: zhoutianran@huawei.com

Yali Zhang
Huawei Technologies Co., Ltd
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

EMail: zhangyali369@huawei.com

Susan Hares
Huawei Technologies Co., Ltd
7453 Hickory Hill
Saline, MI 48176
USA

Email: shares@ndzh.com

Pedro Andres Aranda
Telefornica I+D,
Don Ramon de la Cruz, 82 Street
Madrid, 28006, Spain

EMail: pedroa.aranda@telefonica.com

Diego R. Lopez
Telefornica I+D,
Don Ramon de la Cruz, 82 Street
Madrid, 28006, Spain

EMail: diego.r.lopez@telefonica.com

Jon Crowcroft
University of Cambridge Computer Laboratory
William Gates Building, 15 JJ Thomson Avenue
Cambridge, CB3 0FD UK

Email: jon.crowcroft@cl.cam.ac.uk

Yan Zhang
China Unicom P.R. China

Email: zhangy1036@chinaunicom.cn