Network Working Group
Internet Draft
Intended status: Informational
Expires: October 2020

C. Li
China Telecom
O. Havel
W. Liu
A. Olariu
Huawei Technologies
P. Martinez-Julia
NICT
J. Nobre
UFRGS
D. Lopez
Telefonica I+D

April 20, 2020

Intent Classification
draft-li-nmrg-intent-classification-03

Status of this Memo

Copyright Notice

Abstract

RFC7575 defines Intent as an abstract high-level policy used to operate the network. Intent management system includes an interface for users to input requests and an engine to translate the intents into the network configuration and manage their lifecycle. Up to now, there is no commonly agreed definition, interface or model of intent.

This document discusses what intent means to different stakeholders, describes different ways to classify intent, and an associated taxonomy of this classification. This is a foundation for discussion intent related topics.

Table of Contents

1. Introduction

The vision of intent-driven networks has attracted a lot of attention, as it promises to simplify the management of networks by human operators by simply specifying what should happen on the network, without giving any instructions on how to do it. This

promise led many telecom companies to begin adopting this new paradigm, and many SDOs to propose various intent variants.

All SDOs, such as IETF [ANIMA], ONF [ONF], ONOS [ONOS], have proposed intents as a declarative interface for defining a set of network operations to execute.

As such, IETF [ANIMA] defines intent as a declarative policy and focuses on providing a more complete definition of it, a tentative format, and a life-cycle. Within ONF [ONOS] intent is represented as a list of CLI commands that allows users to pass low-level details on the network, such as flows, or host addresses. ONF through its Boulder and Aspen projects focuses on NBI semantics and intent models.

As it can be observed, each of these SDOs came up with their own way of specifying an intent, and with their own understanding of what an Intent is in terms of the level of abstraction, intended users or scenarios.

However, all intent approaches proposed by SDOs share the same following features:

- It must be declarative in nature, meaning that a user specifies the goal on the network without specifying how to achieve that goal

- It must be vendor agnostic, in the sense that it abstracts the network capabilities, or the network infrastructure from the user, and it can be ported across different platforms

- It must provide an easy-to-use interface, which simplifies the users' interaction with the intent system through the usage of familiar terminology or concepts.

- It should be able to detect and resolve intent conflicts.

Currently, work is underway on unifying a common understanding of intent concepts and terminology. [CLEMM] is currently leading these efforts by defining intent as higher-level declarative policy that operates at the level of network and services it provides, and by capturing the differences between intent, policy and service.

However, even with proposed intent concepts and terminology, and agreement on common intent characteristics, an intent may still be viewed in different ways by different stakeholders for different use cases and solutions.

Thus, the goal of this document is to bring clarity to what an intent represents for different stakeholders, by means of classification on various dimensions, such as solutions, users and intent types. This classification would ensure a common understanding across all participants and it can be used to identify the scope and priorities of individual projects, PoCs, research or open-source projects.

This is achieved by proposing initial classification tables and the methodology used for generating them. This methodology can be used to update the tables by adding or removing different solutions, users or intent types in order to cater for future scenarios, applications or domains.

This draft together with [CLEMM] aims to become the foundation for future intent-related topic discussions where all participants have the same common understanding.

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. Acronyms

CFS: Customer Facing Service

CLI: Command Line Interface

DC: Data Center

ECA: Event-Condition-Action

RFS: Resource Facing Service

SDO: Standards Development Organization

SUPA: Simplified Use of Policy Abstractions

VPN: Virtual Private Network

3. Abstract intent requirements

In order to understand the different intent requirements that would drive intent classification, we first need to understand what intent means for different intent users.

3.1. What is Intent?

The term Intent has become very widely used in the industry for different purposes, sometimes it is not even in agreement with SDO shared principles mentioned in the Introduction.

Different stakeholders consider an intent to be an ECA policy, a GBP policy, a business policy, a network service, a customer service, a network configuration, application / application group policy, any operator/administrator task, network troubleshooting / diagnostics / test, a new app, a marketing term for existing management/orchestration capabilities, etc. Their intent is sometimes technical, non-technical, abstract or technology specific. For some stakeholders, intent is a subset of these and for other stakeholders intent is all of these. It has in some cases become a term to replace a very generic 'service' or 'policy' terminology.

Concerning this, [CLEMM] draft brings clarification with relation to what an intent is and how it differentiates from policies and services.

While it is easier for those familiar with different standards to understand what service, CFS, RFS, resource, policy continuum, ECA policy, declarative policy, abstract policy or intent policy is, it may be more difficult for the wider audience. Intent is very often just a synonym for policy. Those familiar with policies understand the difference between a business, intent, declarative, imperative and ECA policy. But maybe the wider audience does not understand the difference and sometimes equates the policy to an ECA policy.

Therefore, it is important to start a discussion in the industry about what intent is for different solutions and intent users. It is also imperative to try to propose some intent categories / classifications that could be understood by a wider audience. This would help us define intent interfaces, DSLs and models.

3.2. Intent Solutions & Intent Users

Different Solutions and Actors have different requirements, expectations and priorities for intent driven networking. They require different intent types and have different use cases. Some users are more technical and require intents that expose more technical information. Other users do not understand networks and require intents that shield them from different networking concepts and technologies. The following are the solutions and intent users that intent driven networking needs to support:

Solutions	Intent Users
Carrier Networks	Network Operator
	Service Designers
	Service Operators
	Customers/Subscribers
DC Networks	Cloud Administrator
	Underlay Network Administrator
	App Developers
	End Users
Enterprise Networks	Enterprise Administrator
	App Developers
	End Users

- For carrier networks scenario, for example, if the end users wants to watch high-definition video, then the intent is to convert the video image to 1080p rate for the users.

- For DC networks scenario, administrators have their own clear network intent such as load balancing. For all traffic flows that need NFV service chaining, restrict the maximum load of any VNF node/container below 50% and the maximum load of any network link below 70%.

- For Enterprise Networks scenario, enterprise administrators express their intent from an external client(application service provider).For example, when hosting a video conference, multiple remote access is required. The intent expressed to the network operator: For any user of this application, the arrival time of hologram objects of all the remote tele-presenters should be synchronised within 50ms to reach the destination viewer for each conversation session.

3.3. Current Problems & Requirements

Network APIs and CLIs are too complex due to the fact that they expose technologies & topologies. App developers and end-users do not want to set IP Addresses, VLANs, subnets, ports, etc. Operators and administrators would also benefit from the simpler interfaces, like:

- Allow Customer Site A to be connected to Internet via Network B
- Allow User A to access all internal resources, except the Server B
- Allow User B to access Internet via Corporate Network A
- Move all Users from Corporate Network A to the Corporate Network B
- Request Gold VPN service between my sites A, B and C
- Provide CE Redundancy for all Customer Sites
- Add Access Rules to my Service

Networks are complex, with many different protocols and encapsulations. Some basic questions are not easy to answer:

- Can User A talk to User B?
- Can Host A talk to Host B?
- Are there any loops in my network?
- Are Network A and Network B connected?
- Can User A listen to communications between Users B & C?

Operators and Administrators manually troubleshoot and fix their networks and services. They instead want:

- a reliable network that is self-configured and self-assured based on the intent
- to be notified about the problem before the user is aware
- automation of network/service recovery based on intent (self-healing, self-optimization)
- to get suggestions about correction/optimization steps based on experience (historical data & behaviour)

Therefore, Operators and Administrators want to:

- simplify and automate network operations

- o simplify definitions of network services
- o provide simple customer APIs for Value Added Services (operators)
- o be informed if the network or service is not behaving as requested
- o enable automatic optimization and correction for selected scenarios
- o have systems that learn from historic information and behaviour

End-Users cannot build their own services and policies without becoming technical experts and they must perform manual maintenance actions. Application developers and end-users/subscribers want to be able to:

- o build their own network services with their own policies via simple interfaces, without becoming networking experts
- o have their network services up and running based on intent and automation only, without any manual actions or maintenance

3.4. Intent Types that need to be supported

The following intent types need to be supported, in order to address the requirements from different solutions and intent users:

- o Customer network service intent
    - o for customer self-service
    - o for service operator orders
    - o for intent driven network configuration, verification, correction and optimization
- o Network resource management
    - o For network configuration
    - o For automated lifecycle management of network configurations
    - o For network resources (switches, routers, routing, policies, underlay)
- o Cloud and cloud resource management

- For DC configuration, VMs, DB Servers, APP Servers
- For communication between VMs
- For cloud resource lifecycle management (policy driven self-configuration & auto-scaling & recovery/optimization)

- Network Policy intent
  - For security, QoS, application policies, traffic steering, etc
  - For configuring & monitoring policies, alarms generation for non-compliance, auto-recovery

- Task based intents
  - For network migration
  - For server replacements
  - For device replacements
  - For network software upgrades
  - To automate any tasks that operators/administrator often perform

- System policies intents
  - For intent management system policies
  - For design models and policies for network service design
  - For design models and policies for network design
  - For design workflows, models and policies for task based intents

- Intents that affect other intents
  - It may be task based intent that modifies many other intents.
  - The task itself is short-lived, but the modification of other intents has an impact on their lifecycle, so those changes must continue to be continuously monitored and self-corrected/self-optimized.

4. Functional Characteristics and Behavior

Intent can be used to operate immediately on a target (much like issuing a command), or whenever it is appropriate (e.g., in response to an event). In either case, intent has a number of behaviors that serve to further organize its purpose, as described by the following subsections.

4.1. Abstracting Intent Operation

The modelling of Intents can be abstracted using the following three-tuple:

{Context, Capabilities, Constraints}

- Context grounds the intent, and determines if it is relevant or not for the current situation. Thus, context selects intents based on applicability.

- Capabilities describe the functionality that the intent can perform. Capabilities take different forms, depending on the expressivity of the intent as well as the programming paradigm(s) used.

- Constraints define any restrictions on the capabilities to be used for that particular context

Metadata can be attached via strategy templates to each of the elements of the three-tuple, and may be used to describe how the intent should be used and how it operates, as well as prescribe any operational dependencies that must be taken into account.

4.2. Intent User Types

Intent user types, or intent actors as they are known in the area of declarative policy, represent the users that define and issue the intent request. Depending on the Intent Solutions, there are specific intent actors. Examples of intent actors are customers, network operators, service operators, enterprise, cloud, and underlay network administrators, or application developers.

- Customers and end-users do not necessarily know the functional and operational details of the network that they are using. Furthermore, they lack skills to understand such details; in fact, such knowledge is typically not relevant to their job. In addition, the network may not expose these details to its users. This class of actor focuses on the applications that they run, and uses services offered by the network. Hence, they want to specify policies that provide consistent behaviour according to their business needs. They do not have to worry about how the intents are deployed onto the underlying network, and especially, whether the intents need to be translated to different forms to enable network elements to understand them.

- Application developers work in a set of abstractions defined by their application and programming environment(s). For example, many application developers think in terms of objects (e.g., a VPN). While this makes sense to the application developer, most network devices do not have a VPN object per se; rather, the VPN is formed through a set of configuration statements for that device in concert with configuration statements for the other devices that together make up the VPN. Hence, the view of application developers matches the services provided by the network, but may not directly correspond to other views of other actors.

- Management personnel, such as network operators, may have the knowledge of the underlying network. However, they may not understand the details of the applications and services of Customers and End-Users.

4.3. Intent Scope

Intent are used to manage the behaviour of the networks they are applied to and all intents are applied within a specific scope, such as:

- Connectivity scope, if the intent creates or modifies a connection.

- Security scope, if the intent specifies the security characteristics of the network or users.

- Application scope, when the intent specifies the applications to be affected by the intent request

- QoS Scope, when the intent specifies the QoS characteristics of the network

4.4. Intent Network Scope

Regardless on the intent user type, their intent request is affecting the network, or network components, which are representing the intent targets.

Thus, intent network scope, or policy target as known in the area of declarative policy, can represent VNFs or PNFs, Physical Network Elements, Campus networks, SD-WAN networks, radio access networks, cloud edge, cloud core, branch, etc.

4.5. Intent Abstraction

Intent can be classified by whether it is necessary to feedback technical network information or non-technical information to the intended proponent after the intent is executed. As well, intent abstraction covers the level of technical details in the intent itself.

- For ordinary users, they do not care how the intent is executed, or the details of the network. As a result, they do not need to know the configuration information of the underlying network. They only focus on whether the intent execution result achieves the goal, and the execution effect such as the quality of completion and the length of execution. In this scenario, we refer to an abstraction without technical feedback.

- For administrators, such as network administrators, they perform intents, such as allocating network resources, selecting transmission paths, handling network failures, etc. They require multiple feedback indicators for network resource conditions, congestion conditions, fault conditions, etc. after execution. In this case, we refer to an abstraction with technical feedback

4.6. Intent Lifecycle

Intents can be classified into transient and persistent intents:

- If intent is transient, it has no lifecycle management. As soon as the specified operation is successfully carried out, the intent is finished, and can no longer affect the target object.

- If the intent is persistent, it has lifecycle management. Once the intent is successfully activated and deployed, the system will keep all relevant intents active until they are deactivated or removed.

4.7. Hierarchy

In different phases of the autonomous driving network [TMF-auto], the intents are different. A typical example of autonomous driving network Level 0 to 5 are listed as below.

- Level 0 - Traditional manual network: O&M personnel manually control the network and obtain network alarms and logs. - No intent

- Level 1 - Partially automated network: Automated scripts are used to automate service provisioning, network deployment, and maintenance. Shallow perception of network status and decision making suggestions of machine; - No intent

- Level 2 - Automated network: Automation of most service provisioning, network deployment, and maintenance comprehensive perception of network status and local machine decision making; - simple intent on service provisioning

- Level 3 - Self-optimization network: Deep awareness of network status and automatic network control, meeting users' network intentions. - Intent based on network status cognition

- Level 4 - Partial autonomous network: In a limited environment, people do not need to participate in decision-making and adjust themselves. - Intent based on limited AI

- Level 5 - Autonomous network: In different network environments and network conditions, the network can automatically adapt to and adjust to meet people's intentions. - Intent based on AI

5. Intent Classification

This chapter proposes an intent classification approach that may help to classify mainstream intent related demos / tools.

The three classifications in this draft have been proposed from scratch, following the methodology presented, through three iterations: one for carrier Intent Solution, one for DC Intent Solution, and one for enterprise Intent Solution. For each Intent solution, we identified the specific Intent Users and Intent Types. Then, we further identified the Intent Scope, Network Scope, Abstractions, and Lifecycle requirements.

These classifications and the generated tables can be easily extended. For example, for the DC Intent Solution, a new category is

identified, i.e. Resource Scope, and the classification table has been extended accordingly.

In the future, as new scenarios, applications, and domain are emerging, new classifications and taxonomies can be identified, following the proposed methodology.

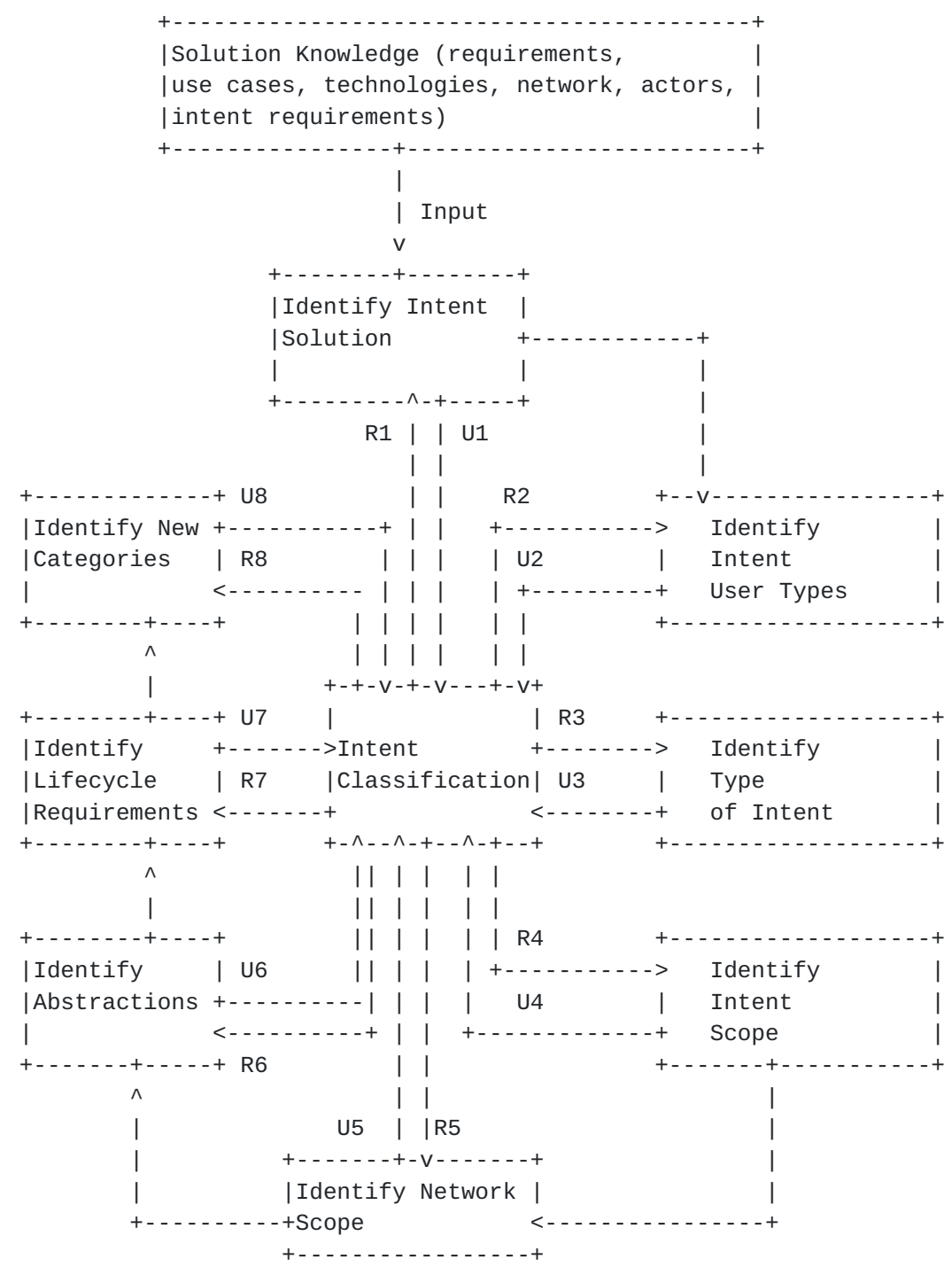
The output of the intent classification is the intent taxonomy introduced in the next sections.

Thus, this section first introduces the proposed intent classification methodology, followed by consolidated intent taxonomy for three intent solutions, and then by concrete examples of intent classifications for three different intent solutions (e.g. Carrier Network, Data Center, and Enterprise) that were derived using the proposed methodology and then can be filled in for PoCs, demos, research projects or future drafts.

5.1. Intent Classification Methodology

This section describes the methodology used to derive the initial classification proposed in the draft. The proposed methodology can be used to create new intent classifications from scratch, by analysing the solution knowledge. As well, the methodology can be used to update existing classification tables by adding or removing different solutions, users or intent types in order to cater for future scenarios, applications or domains.

We first classify intents into intent types and describe each type based on the solution it belongs to and what intent user it is for. We than present different categories that these intent type can belong to, based on intent scope, network scope, intent abstraction and lifecycle.


Solution Knowledge (requirements, use cases, technologies, network, actors, intent requirements)
Input
Identify Intent Solution
R1
U1
U8
R2
Identify New Categories
R8
U2
Identify Intent User Types
U7
R3
Identify Lifecycle Requirements
R7
Intent Classification
U3
Identify Type of Intent
U6
R4
Identify Abstractions
U4
Identify Intent Scope
R6
U5
R5
Identify Network Scope

In the above methodology, the arrows mean the following:

- Input represents the Solution Knowledge comprising of knowledge about solution requirements, targeted use cases, available technologies and networks, actors, intent requirements.

- R1-U1: Review existing classification and use/add/remove the intent solution

- R2-U2: Review existing classification and use/add/remove the intent user type

- R3-U3: Review existing classification and use/add/remove the intent types

- R4-U4: Review existing classification and use/add/remove the intent scopes

- R5-U5: Review existing classification and use/add/remove the network scopes

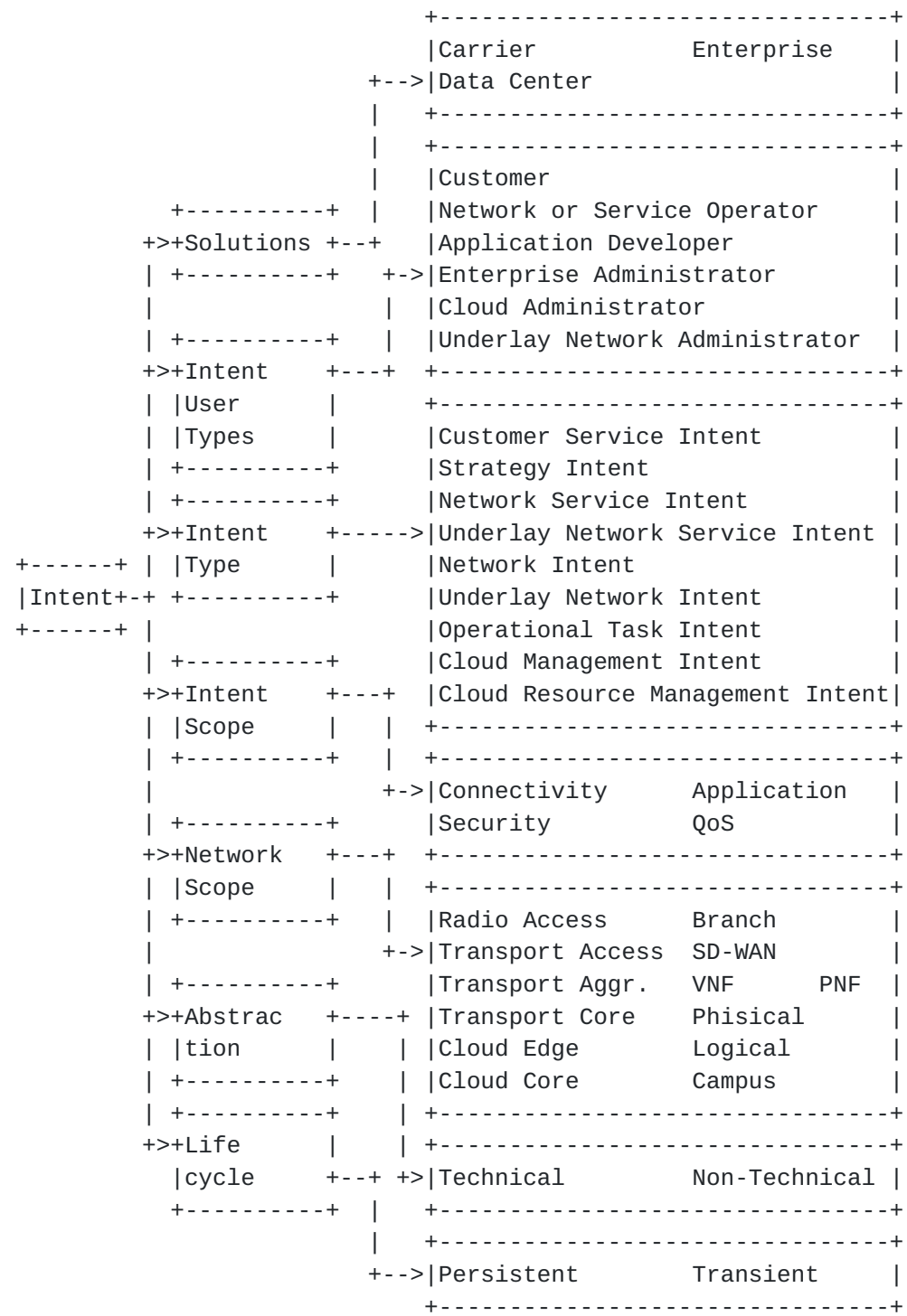
- R6-U6: Review existing classification and use/add/remove the abstractions

- R7-U7: Review existing classification and use/add/remove the lifecycle requirements

- R8-U8: Review existing classification and use/add the newly identified categories.

5.2. Intent Taxonomy

The following taxonomy describes the various intent solutions, intent user types, intent types, intent scopes, network scopes, abstractions and lifecycle and represents the output of the intent classification tables for each of the solutions addressed (i.e. Carrier Solution, Data Center, and Enterprise).


Carrier Enterprise
Data Center
Customer
Network or Service Operator
Solutions
Application Developer
Enterprise Administrator
Cloud Administrator
Underlay Network Administrator
Intent User Types
Customer Service Intent
Strategy Intent
Network Service Intent
Intent Type
Underlay Network Service Intent
Network Intent
Underlay Network Intent
Intent
Operational Task Intent
Cloud Management Intent
Cloud Resource Management Intent
Scope
Connectivity Application
Security QoS
Network
Scope
Radio Access Branch
Transport Access SD-WAN
Transport Aggr. VNF PNF
Abstrac tion
Transport Core Phisical
Cloud Edge Logical
Cloud Core Campus
Life cycle
Technical Non-Technical
Persistent Transient

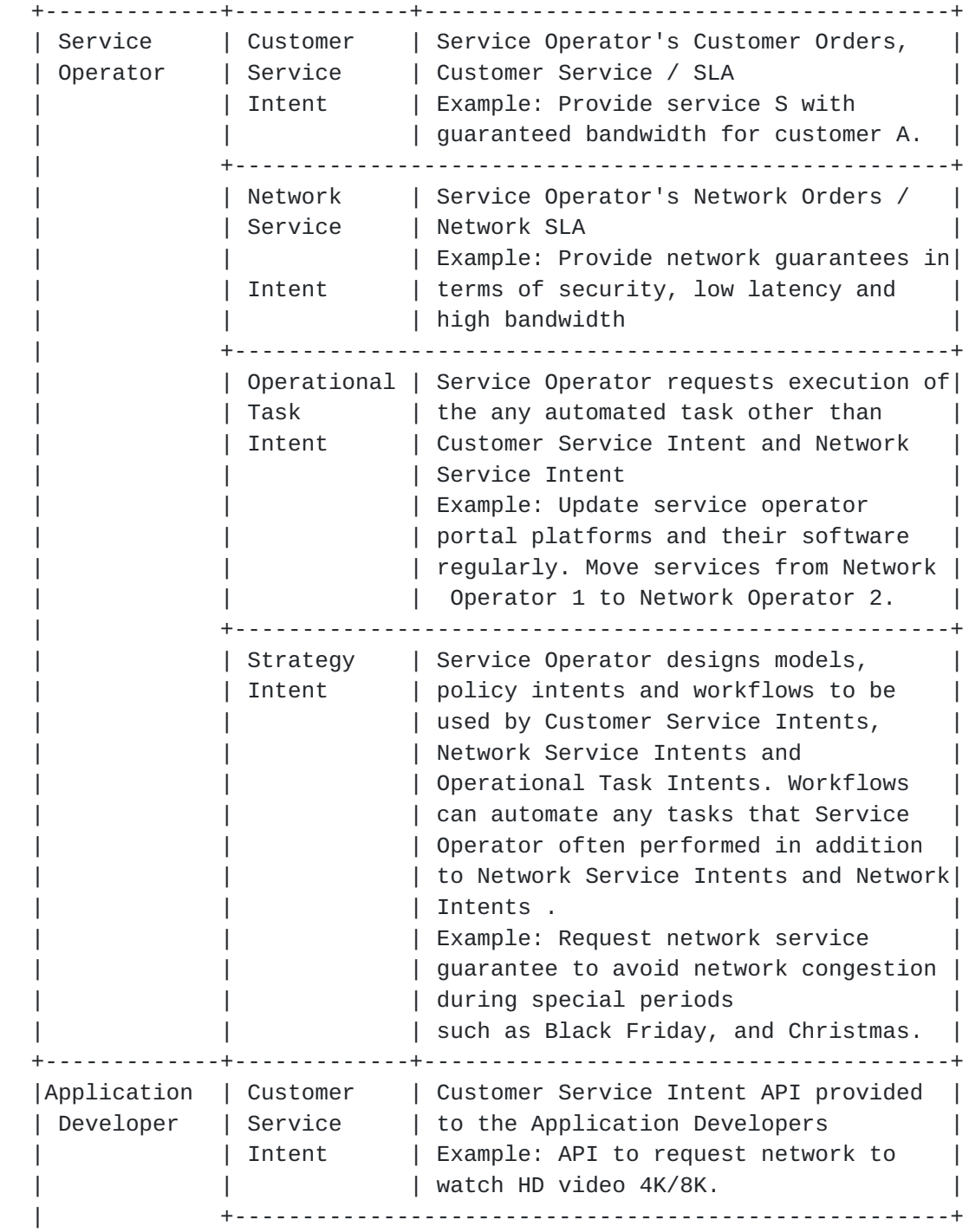
5.3. Intent Classification for Carrier Solution

5.3.1. Intent Users and Intent Types

The following table describes the Intent Users in Carrier Solutions and Intent Types with their descriptions for different intent users.

Intent User	Intent Type	Intent Type Description
Customer/ Subscriber	Customer Service Intent	Customer Self-Service with SLA and Value Added Service Example: Always maintain high quality of service and high bandwidth for gold level users.
	Strategy Intent	Customer designs models and policy intents to be used by Customer Service Intents. Example: Request reliable service during peak traffic periods for apps of type video.
Network Operator	Network Service Intent	Service provided by Network Service Operator to the Customer (e.g. the Service Operator) Example: Request network service with delay guarantee for access customer A.
Customer/ Subscriber	Customer Service Intent	Customer Self-Service with SLA and Value Added Service Example: Always maintain high quality of service and high bandwidth for gold level users.
	Strategy Intent	Customer designs models and policy intents to be used by Customer Service Intents. Example: Request reliable service during peak traffic periods for applications of type video.

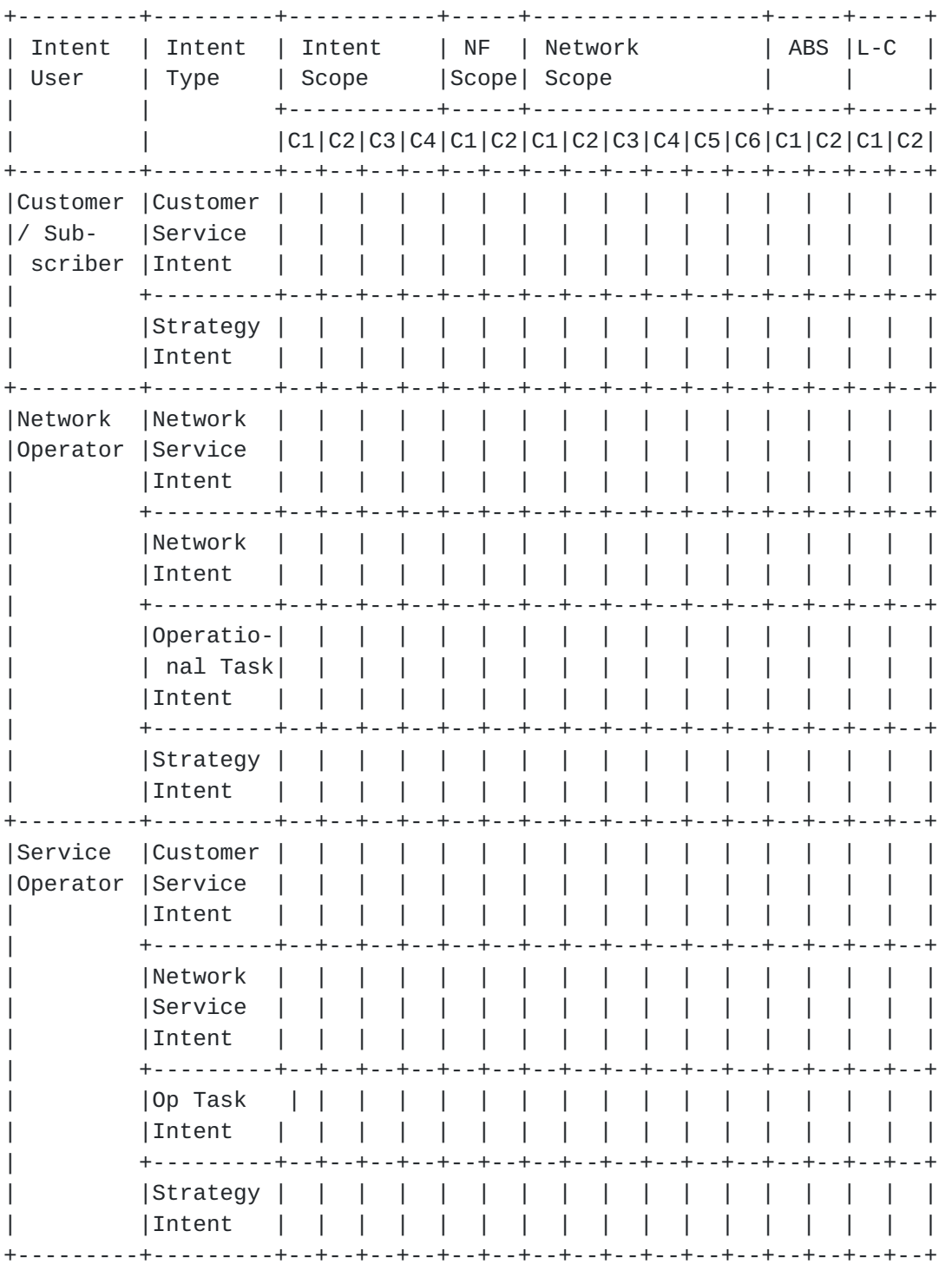
Network Operator	Network Service Intent	Service provided by the Network Service Operator to the Customer (e.g. the Service Operator) Example: Request network service with delay guarantee for access customer A.
	Network Intent	Network Operator requests network-wide (service underlay or other network-wide) configuration) or network resource configurations (switches, routers, routing, policies). Includes Connectivity, Routing, QoS, Security, Application Policies, Traffic Steering Policies, Configuration policies, Monitoring policies, alarm generation for non-compliance, auto-recovery, etc. Example: Request high priority queueing for traffic of class A.
	Operational Task Intent	Network Operator requests execution of any automated task other than Network Service Intent and Network Intent (e.g. Network Migration, Server Replacements, Device Replacements, Network Software Upgrades. Example: Request migration of all services in Network N to backup path P.
	Strategy Intent	Network Operator designs models, policy intents and workflows to be used by Network Service Intents, Network Intents and Operational Task Intents. Workflows can automate any tasks that Network Operator often performed in addition to Network Service Intents and Network Intents Example: Ensure the load on any link in the network is not higher than 50%.

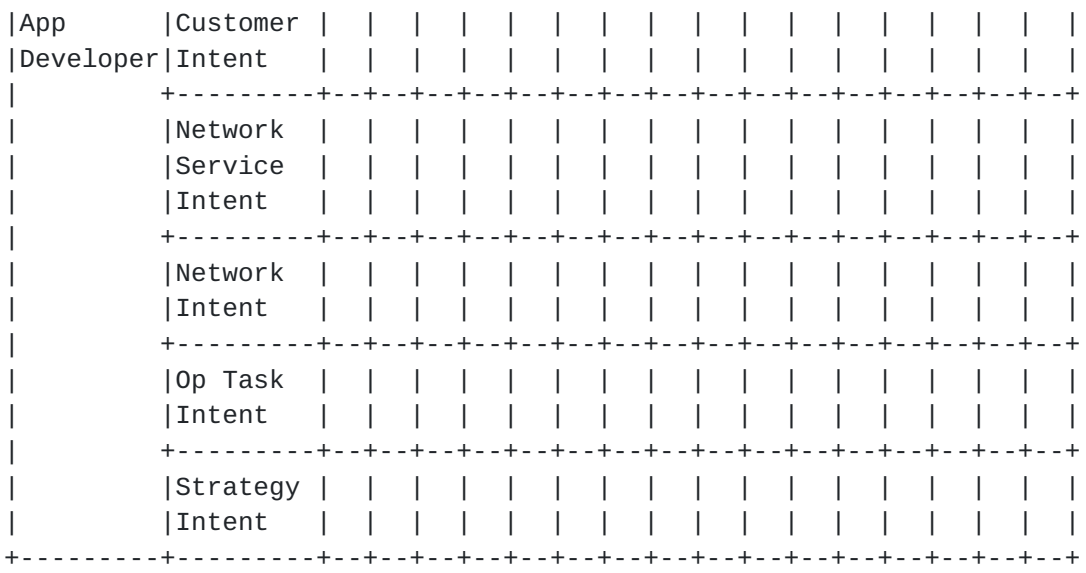

Service Operator
Customer Service Intent
Service Operator's Customer Orders, Customer Service / SLA
Example: Provide service S with guaranteed bandwidth for customer A.
Network Service Intent
Service Operator's Network Orders / Network SLA
Example: Provide network guarantees in terms of security, low latency and high bandwidth
Operational Task Intent
Service Operator requests execution of the any automated task other than Customer Service Intent and Network Service Intent
Example: Update service operator portal platforms and their software regularly. Move services from Network Operator 1 to Network Operator 2.
Strategy Intent
Service Operator designs models, policy intents and workflows to be used by Customer Service Intents, Network Service Intents and Operational Task Intents. Workflows can automate any tasks that Service Operator often performed in addition to Network Service Intents and Network Intents .
Example: Request network service guarantee to avoid network congestion during special periods such as Black Friday, and Christmas.
Application Developer
Customer Service Intent
Customer Service Intent API provided to the Application Developers
Example: API to request network to watch HD video 4K/8K.

	Network Service Intent	Network Service Intent API provided to the Application Developers Example: API to request network and monitoring an traffic grooming
	Network Intent	Network Intent API provided to the Application Developers Example: API to request network resources configuration.
	Operational Task Intent	Operational Task Intent API provided to the Application Developers. This is for the trusted internal Operator / Service Providers / Customer DevOps Example: API to request server migrations.
	Strategy Intent	Application Developer designs models, policy and workflows to be used by Customer Service Intents, Network Service Intents and Operational Task Intents. This is for the trusted internal Operator/Service Provider/ Customer DevOps Example: API to design network load balancing strategies during peak times

5.3.2. Intent Categories

The following arethe proposed categories:
Intent Scope: C1=Connectivity, C2=Security, C3=Application, C4=QoS
Network Function (NF) Scope: C1=VNFs, C2=PNFs
Network Scope: C1=Radio Access, C2=Transport Access, C3=Transport Aggregation, C4=Transport Core, C5=Cloud Edge, C6=Cloud Core)
Abstraction(ABS): C1=Technical(with technical feedback), C2=Non-technical (without technical feedback) , see Section 4.2
Life-cycle (L-C): C1=Persistent (Full life-cycle), C2=Transient (Short Lived)
The following is the Classification Table Example for Carrier.


Intent User
Intent Type
Intent Scope
NF Scope
Network Scope
ABS
L-C
C1 C2 C3 C4 C1 C2 C1 C2 C3 C4 C5 C6 C1 C2 C1 C2
Customer / Sub-scriber
Customer Service Intent
Strategy Intent
Network Operator
Network Service Intent
Network Intent
Operatio-nal Task Intent
Strategy Intent
Service Operator
Customer Service Intent
Network Service Intent
Op Task Intent
Strategy Intent

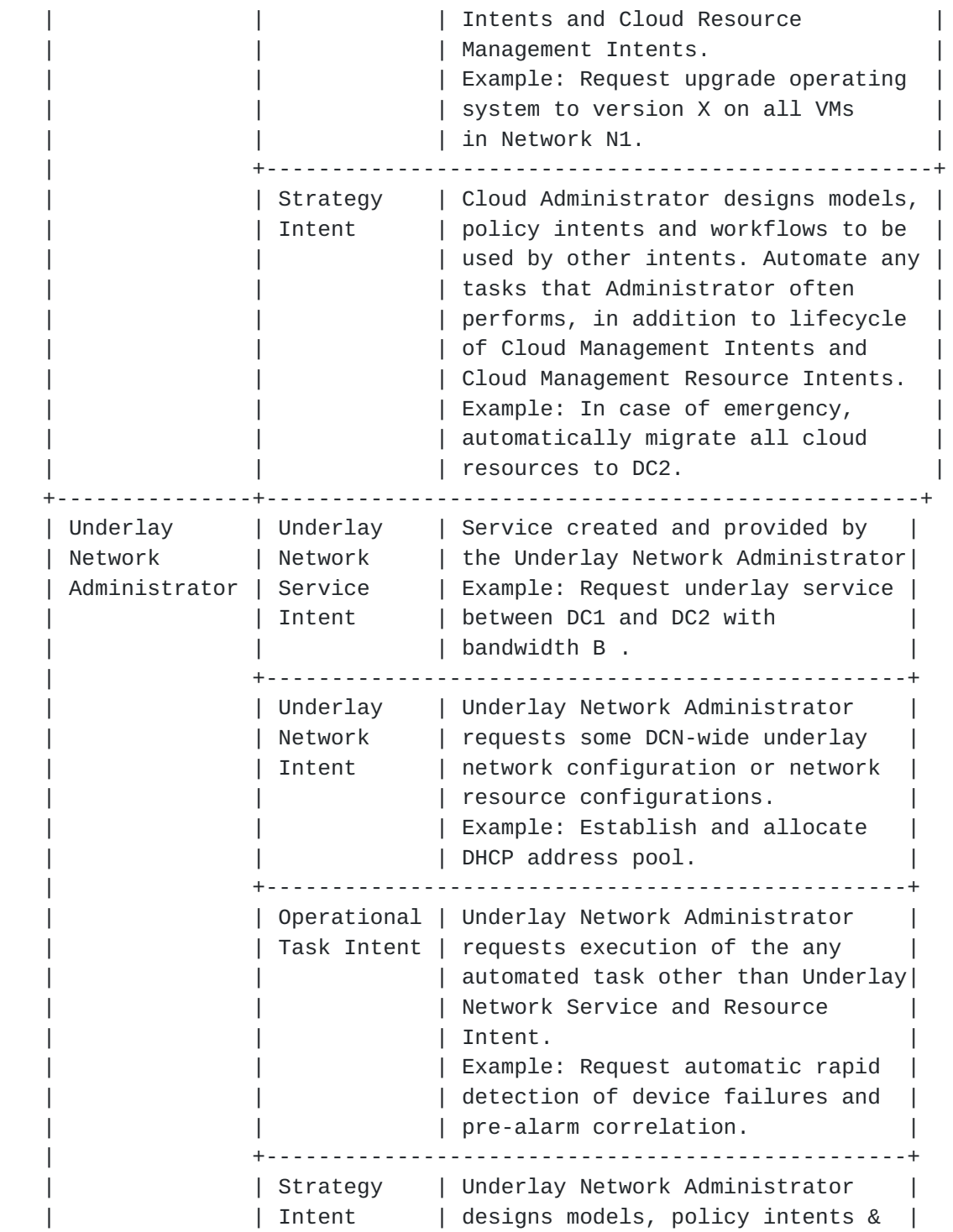

App Developer
Customer Intent
Network Service Intent
Network Intent
Op Task Intent
Strategy Intent

5.4. Intent Classification for Data Center Solutions

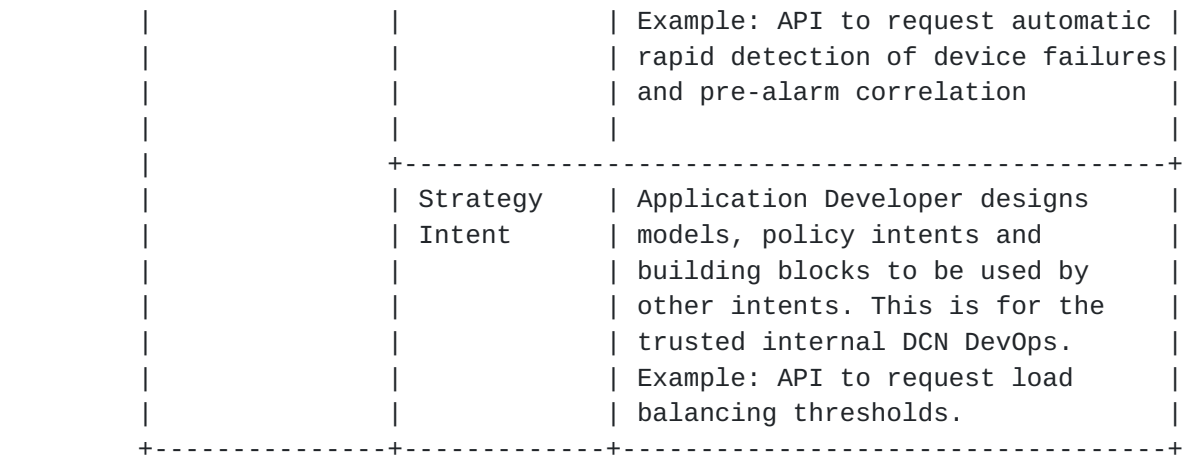
5.4.1. Intent Users and Intent Types

The following table describes the Intent Users in DCN Solutions and Intent Types with their descriptions for different intent users.

Intent User	Intent Type	Intent Type Description
Customer / Tenants	Customer Intent	Customer Self-Service via Tenant Portal, Customers may have multiple type of end users. Example: Request GPU computing and storage resources to meet 10k video surveillance services.
	Strategy Intent	This includes models and policy intents designed by Customers/ Tenants to be used by Customer and End-User Intents. Example: Request dynamic computing and storage resources of the service in special and daily times.
Cloud Administrator	Cloud Management Intent	Configuration of VMs, DB Servers, App Servers, Connectivity, Communication between VMs. Example: Request connectivity between VMs A,B,and C in Network N1.
	Cloud Resource Management Intent	Policy-driven self-configuration and and recovery / optimization Example: Request automatic life-cycle management of VM cloud resources.
	Operational Task Intent	Cloud Administrator requests execution of any automated task other than Cloud Management


		Intents and Cloud Resource
		Management Intents.
		Example: Request upgrade operating
		system to version X on all VMs
		in Network N1.
	Strategy Intent	Cloud Administrator designs models, policy intents and workflows to be used by other intents. Automate any tasks that Administrator often performs, in addition to lifecycle of Cloud Management Intents and Cloud Management Resource Intents. Example: In case of emergency, automatically migrate all cloud resources to DC2.
Underlay Network Administrator	Underlay Network Service Intent	Service created and provided by the Underlay Network Administrator Example: Request underlay service between DC1 and DC2 with bandwidth B .
	Underlay Network Intent	Underlay Network Administrator requests some DCN-wide underlay network configuration or network resource configurations. Example: Establish and allocate DHCP address pool.
	Operational Task Intent	Underlay Network Administrator requests execution of the any automated task other than Underlay Network Service and Resource Intent. Example: Request automatic rapid detection of device failures and pre-alarm correlation.
	Strategy Intent	Underlay Network Administrator designs models, policy intents &

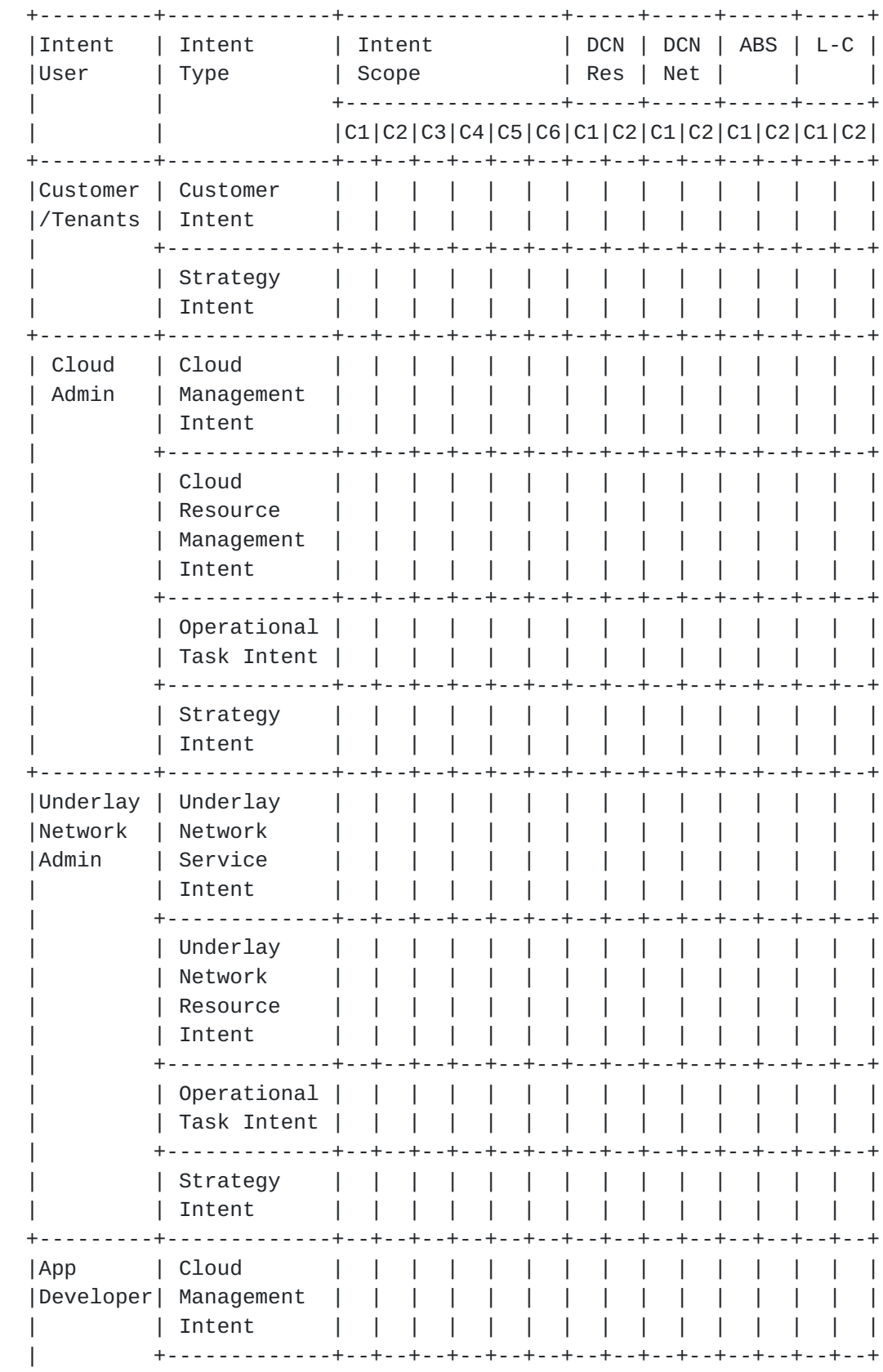
			workflows to be used by other intents. Automate any tasks that Administrator often performs Example: For all traffic flows that need NFV service chaining, restrict the maximum load of any VNF node/container below 50% and the maximum load of any network link below 70%.
Application Developer	Cloud Management Intent	Cloud Management Intent API provided to the Application Developers. Example: API to request configuration of VMs,or DB Servers	
	Cloud Resource Management Intent	Cloud Resource Management Intent API provided to the Application Developers. Example: API to request automatic lifecycle management of cloud resources.	
	Underlay Network Service Intent	Underlay Network Service API provided to the Application Developers. Example: API to request real-time monitoring of device condition.	
	Underlay Network Intent	Underlay Network Resource API provided to the Application Developers. Example: API to request dynamic management of IPv4 address pool resources.	
	Operational Task Intent	Operational Task Intent API provided to the trusted Application Developer (internal DevOps).	

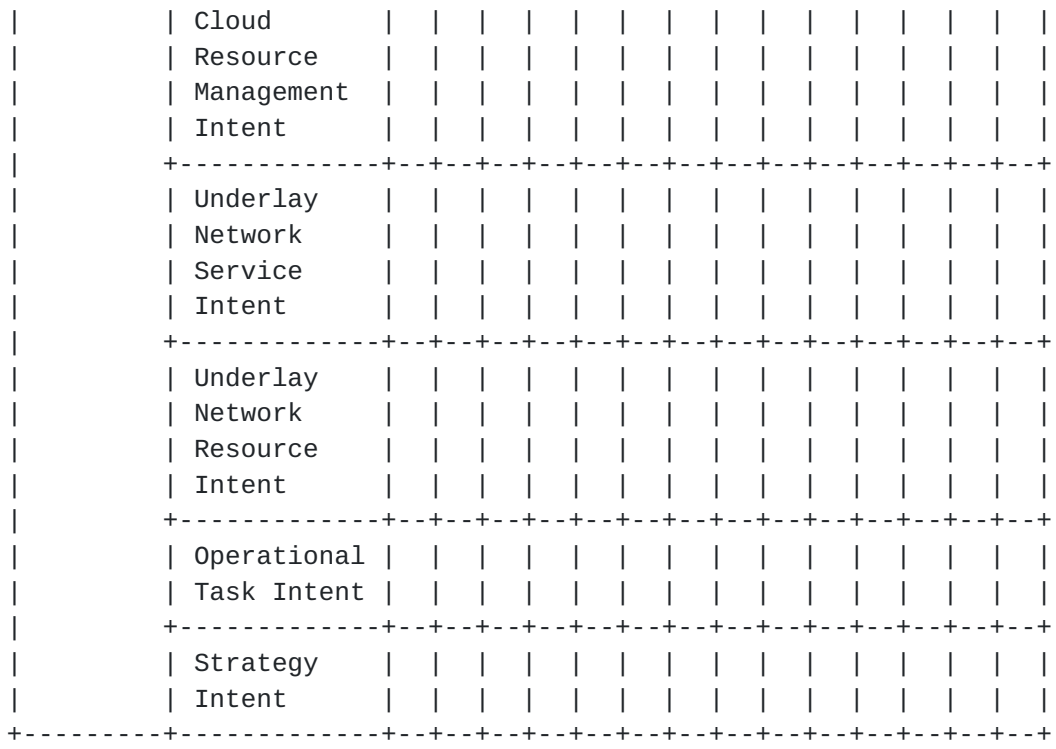

		Example: API to request automatic
		rapid detection of device failures
		and pre-alarm correlation
	Strategy Intent	Application Developer designs models, policy intents and building blocks to be used by other intents. This is for the trusted internal DCN DevOps. Example: API to request load balancing thresholds.

5.4.2. Intent Categories

The following are the proposed categories:
Intent Scope: C1=Connectivity, C2=Security, C3=Application,
C4=QoS C5=Storage C6=Compute
DCN Resource (DCN Res) Scope: C1=Virtual, C2=Physical
DCN Network (DCN Net) Scope: C1=Logical, C2=Physical
Abstraction(ABS): C1=Technical(with technical feedback),
C2=Non-technical (without technical feedback), see Section 4.2
Life-cycle (L-C): C1=Persistent (Full life-cycle), C2=Transient
(Short Lived)

The following is the Classification Table Example for DC Solutions.


Intent User
Intent Type
Intent Scope
DCN Res
DCN Net
ABS
L-C
C1 C2 C3 C4 C5 C6 C1 C2 C1 C2 C1 C2 C1 C2
Customer /Tenants
Customer Intent
Strategy Intent
Cloud Admin
Cloud Management Intent
Cloud Resource Management Intent
Operational Task Intent
Strategy Intent
Underlay Network Admin
Underlay Network Service Intent
Underlay Network Resource Intent
Operational Task Intent
Strategy Intent
App Developer
Cloud Management Intent



5.5. Intent Classification for Enterprise Solution

5.5.1. Intent Users and Intent Types

The following table describes the Intent Users in Enterprise Solutions and their Intent Types.

Intent User	Intent Type	Intent Type Description
End-User	End-User Intent	Enterprise End User Self-Service or Applications, Enterprise may have multiple types of End-Users. Example: Request access to VPN service. Request video conference between user A and B.
	Strategy Intent	This includes models and policy intents designed by End-Users to be used by End-User Intents and their Applications. Example: Create a video conference type for a weekly meeting.
Administrator (internal or MSP)	Network Service Intent	Service provided by the Administrator to the End-Users and their Applications. Example: For any user of application X, the arrival time of hologram objects of all the remote tele-presenters should be synchronised within 50ms to reach the destination viewer for each conversation session Create management VPN connectivity for type of service A.
	Network Intent	Administrator requires network wide configuration (e.g. underlay, campus) or resource configuration (switches, routers, policies). Example: Configure switches in campus network 1 to prioritise traffic of type A. Configure Youtube as business non-relevant.
	Operational	Administrator requests execution of

		Task Intent	any automated task other than Network Service Intents and Network Intents. Example: Request network security automated tasks such as Web filtering and DDOS cloud protection.
		Strategy Intent	Administrator designs models, policy intents and workflows to be used by other intents. Automate any tasks that Administrator often performs. Example: In case of emergency, automatically shift all traffic of type A through network N.
	Application Developer	End-User Intent	End-User Service / Application Intent API provided to the Application Developers. Example: API for request to open a VPN service.
		Network Service Intent	Network Service API Provided to Application Developers. Example: API for request network bandwidth and latency for hosting video conference.
		Network Intent	Network API Provided to Application Developers. Example: API for request of network devices configuration.
		Operational Task Intent	Operational Task Intent API provided to the trusted Application Developer (internal DevOps). Example: API for requesting automatic monitoring and interception for network security
		Strategy	Application Developer designs

```
|              | Intent      | models, policy intents and building |
|              |             | blocks to be used by other intents. |
|              |             | This is for the trusted internal    |
|              |             | DevOps.                             |
|              |             | Example: API for strategy intent in |
|              |             | case of emergencies.                |
|              |             |                                     |
+--------------+-------------+-------------------------------------+
```

5.5.2. Intent Categories

The following are the proposed categories:

Intent Scope: C1=Connectivity, C2=Security, C3=Application, C4=QoS

Enterpise Network (Net) Scope: C1=Campus, C2=Branch, C3=SD-WAN

Abstraction(ABS): C1=Technical(with technical feedback), C2=Non-technical (without technical feedback), see Section 4.2

Life-cycle (L-C): C1=Persistent (Full life-cycle), C2=Transient (Short Lived)

The following is the Intent Classification Table Example for Enterprise Solutions.

Intent User	Intent Type	Intent Scope				Net			ABS		L-C	
		C1	C2	C3	C4	C1	C2	C3	C1	C2	C1	C2
End-User	End-User Intent											
	Strategy Intent											
Enterprise Administrator	Network Intent											
	Strategy Intent											
Application Developer	End-User Intent											
	Network Service Intent											
	Network Intent											
	Operational Task Intent											
	Strategy Intent											

6. Involvement of intent in the application of AI to Network Management

In the application of AI to NM, an intent is expected to be, on the one hand, a formal definitions of a goal or policy instructed to the decision system and, on the other hand, a formal definition of the specific actions that some network controller must perform. Goal intents and policy intents have different meanings. The former will establish an objective for the automated management system to accomplish, such as "avoiding latency to be higher than 10 ms". Meanwhile, policy intents set the overall regulations and possible actions that the AI system can use to achieve those goals. Both goal and policy intents are expected to be provided by humans, although they must be in some very formal language that can be easily understood by computers. All those relations make the degree of formality an important dimension to classify intents so that users, which here are AI-based agents, can be able to choose the proper solution to consume them.

AI technology has played an important role in the different stages of the intent network implementation.

- Help identify and prevent security threats: Classification algorithms can attempt to identify malware or other undesirable web content or usage;

- Intentional translation: use AI algorithm to assist the translation module, split translation into the requirements contained in the semantics of the intention; automatic delivery and execution strategy;Automate tasks and appropriate network changes based on the existing network infrastructure configuration according to the policy model;

- Adaptive adjustment: perceive the quality of the user experience and perform predictive analysis to proactively optimize performance, such as excessive access time;

To enforce the resulting actions determined by AI-based control modules, action intents will have a format that avoids misconceptions as much as possible. This means that they will be closer to machine language structures than natural (human) language structures. This can sacrificing some degree of human understandability, so it forms another dimension in the classification of intents. This dimension allows automated systems to discern which format of intent to use in relation to the possibility and degree of humans to be involved in their exchanges.

Finally, as intents can use different words and languages to refer to the same concepts, all intents related to AI will be required to follow a specific ontology. This way, input intents will be easily semantically translated to formal structures. Output intents will also be composed by following the ontology, so receivers of those intents will be able to easily understand them.

For instance, in the intent classification, the machine learning algorithm can be utilized to extract the intent feature values and classify the intent according to the intent feature distribution. For example, using artificial intelligence clustering algorithm, a large number of intents proposed by different users are used as training data to extract multiple feature dimensions, such as vocabulary information intended to be used, related feature parameters, context proposed by the intent, and the like.Cluster analysis is performed in the same form as the coordinate system, and multiple categories are classified according to the characteristics of the sample point distribution. For the input intent later, the category of the intent is judged based on the similarity with all categories.

- For specific classification intents, such as safety or fault information, conditions can be preset in advance, and once a common error message occurs, it will automatically alarm.

- For the network resource information, set the corresponding threshold information. When there is a certain number of link users or the network traffic is too large, the adjustment intention is started.

- For users with higher priority, the resources can be configured preferentially.

7. Security Considerations

This document does not have any Security Considerations.

8. IANA Considerations

This document has no actions for IANA.

9. Contributors

The following people all contributed to creating this document, listed in alphabetical order:

Ying Chen, China Unicom
Richard Meade, Huawei
John Strassner, Huawei
Xueyuan Sun, China Telecom
Weiping Xu, Huawei

10. Acknowledgments

This document has benefited from reviews, suggestions, comments and proposed text provided by the following members, listed in alphabetical order: Brian E Carpenter, Juergen Schoenwaelder, Laurent Ciavaglia, Xiaolin Song.

11. References

11.1. Normative References

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

[RFC7575] Behringer, M., Pritikin, M., Bjarnason, S., Clemm, A., Carpenter, B., Jiang, S., and L. Ciavaglia, "Autonomic Networking: Definitions and Design Goals", RFC 7575, June 2015.

[RFC8328] Liu, W., Xie, C., Strassner, J., Karagiannis, G., Klyus, M., Bi, J., Cheng, Y., and D. Zhang, "Policy-Based Management Framework for the Simplified Use of Policy Abstractions (SUPA)", March 2018.

[RFC3198] Westerinen, A., Schnizlein, J., Strassner, J., Scherling, M., Quinn, B., Herzog, S., Huynh, A., Carlson, M., Perry, J., Waldbusser, S., "Terminology for Intent-driven Management", RFC 3198, November 2001.

11.2. Informative References

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC7285] R. Alimi, R. Penno, Y. Yang, S. Kiesel, S. Previdi, W. Roome, S. Shalunov, R. Woundy "Application-Layer Traffic Optimization (ALTO) Protocol", September 2014.

[ANIMA] Du, Z., "ANIMA Intent Policy and Format", 2017, <https://datatracker.ietf.org/doc/draft-du-anima-an-intent/>.

[ONF] ONF, "Intent Definition Principles", 2017, <https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-523_Intent_Definition_Principles.pdf>.

[ONOS] ONOS, "ONOS Intent Framework", 2017, <https://wiki.onosproject.org/display/ONOS/Intent+Framework/>.

[SUPA] Strassner, J., "Simplified Use of Policy Abstractions", 2017, <https://datatracker.ietf.org/doc/draft-ietf-supa-generic-policy-info-model/?include_text=1>.

[ANIMA-Prefix] Jiang, S., Du, Z., Carpenter, B., and Q. Sun, "Autonomic IPv6 Edge Prefix Management in Large-scale Networks", draft-ietf-anima-prefix-management-07 (work in progress), December 2017.

[TMF-auto] Aaron Richard Earl Boasman-Patel,et, A whitepaper of Autonomous Networks: Empowering Digital Transformation For the Telecoms Industry, inform.tmforum.org, 15 May, 2019.

[CLEMM] Alexander Clemm and Laurent Ciavaglia and Lisandro Zambenedetti Granville and Jeff Tantsura, Intent-Based Networking - Concepts and Overview, 4 Nov, 2019

Authors' Addresses

Chen Li
China Telecom
No.118 Xizhimennei street, Xicheng District
Beijing 100035
P.R. China
Email: lichen6@chinatelecom.cn

Olga Havel
Huawei Technologies
Email: olga.havel@huawei.com

Adriana Olariu
Huawei Technologies
Email: adriana.olariu@huawei.com

Will(Shucheng) Liu
Huawei Technologies
P.R. China
Email: liushucheng@huawei.com

Pedro Martinez-Julia
NICT
Japan
Email: pedro@nict.go.jp

Jeferson Campos Nobre
University of Vale do Rio dos Sinos
Porto Alegre
Brazil
Email: jcnobre@inf.ufrgs.br

Diego R. Lopez
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid 28006
Spain
Email: diego.r.lopez@telefonica.com