



# SUPA problem statement

**Georgios Karagiannis**, John Strassner, Huawei Technologies,  
Jun Bi, Tsinghua University,  
Dan Romascanu, Avaya  
Maxim Klyus, NetCracker,  
Q. Sun, China Telecom ,  
Luis M. Contreras, Telefonica

July, 2016

# Motivation for this ID

- Describes the problem that SUPA WG will solve

# Status and Changes from 00 version

- Status:
  - replaces Internet Drafts:
    - [karagiannis-suppa-problem-statement-07.txt](#)
    - [draft-klyus-suppa-value-proposition-00.txt](#)
  - incorporates:
    - SUPA Problem Statement,
    - Value of the SUPA Approach
    - brief description of the Application of Generic Policy-based Management
- Changes from 00 version:
  - new section on Terminology has been added
  - (old) Section 3: Related Work on gap analysis was deleted based on comments received during IETF'95

# Problem Statement

**Services**

**Network**

**Multiple  
Vendors**

**Multiple  
Technologies**

**Multiple  
Paradigms**

**Programmatic  
Control Tools**

**Syntax**

**Data Models**

**SDN, NFV, Legacy**

**CLI, TL1, scripts**

**Semantics**

**Management and Orchestration**

## **Challenges**

- **Complicated network infrastructure operation and management**
- **Hard to deploy new and manage existing network services**
- **Difficult to adapt new technologies to existing network operation and management ecosystem**

# Problem Statement

Services

Network

Multiple Vendors

Multiple Technologies

Multiple Paradigms

Programmatic Control Tools

Syntax

Data Models

SDN, NFV, Legacy

CLI, TL1, scripts

Semantics

Policy

Management and Orchestration

**Unified technology independent operation and management framework based on ECA (Event Condition Action) policies will help to solve the challenges and improve existing SP network infrastructure management; Focus on management policies interpreted externally of network elements; this interpretation typically results in configuration changes of collections of network elements**

# Value and Benefits of SUPA

## Vendor and Technology Independent Policy Framework

- Network Policy independence reduces complexity and vendor lockin. Helps unify network management.
- Simplifies deployment of new Network Function and Services.

## Unified Network Infrastructure Policy Management

- Increased abstraction enables simpler and effective network infrastructure management for operators;
- Define high-level, network-wide management policies that are interpreted outside network elements, to create interoperable network element configuration snippets

## Real-time and event-based Network Management

- Network infrastructure can automatically change based on context monitored by policy at the current moment of time

## New Independent Network Management (Policy) Layer

- Policy can help to build intermediate layer between SP and Subscribers for unified and shared management.
- Used by different actors with different skills and requirements: e.g., developers, network administrators, appl.
- Policy-holders can provide instruments to Policy-users for their network resource management.
- Creates management and operations interface to enable existing IETF data models (I2RS, L3SM) to be managed in unified way independent of application domain, technology and vendor

# ECA Policy Examples

## Network Service Management Example

Event: too many interface alarms received from an L3VPN service

Condition: alarms resolve to the same interface within a specified time period

Action: if error rate exceeds x% then put L3VPN service to Error State and migrate users to one or more new L3VPNs

## Security Management Example

Event: anomalous traffic detected in network

Condition: determine the severity of the traffic

Action: apply one or more actions to affected NEs based on the type of the traffic detected (along with other factors, such as the type of resource being attacked if the traffic is determined to be an attack)

# ECA Policy Examples

## Traffic Management Examples

Event: edge link close to being overloaded by incoming traffic

Condition: if link utilization exceeds Y% or if link utilization average is increasing over a specified time period

Action: change routing configuration to other peers that have better metrics

## Service Management Examples

Event: alarm received or periodic time period check

Condition: CPU utilization level comparison

Action: no violation: no action  
violation:

- 1) determine workload profile in time interval
- 2) determine complementary workloads (e.g., whose peaks are at different times in day)
- 3) combine workloads (e.g., using integer programming)

# Next steps

- Received comments on the mailing list to include the gap analysis (related work) in the appendix of this draft
  - Any objections on doing that?
- Propose to make this Problem Statement draft a SUPA working group draft