SUPA Declarative Policy

Jun Bi
Tsinghua Univ./CERNET
draft-bi-declarative-policy-00
Goal

- The goal, objective, **high level request**
- Express **what** should be done **without** telling how

From A to B, and I have $$ budget....
Declarative Policy

- Policy
  - Defines how policy rules are used to manage service behavior

- Policy Model
  - Defines rules for governing managed objects
  - Defines representation for rules
  - Can be used to govern service relationships

- Declarative Policy
  - More abstracted
  - More service level
  - Device independent
Types of Policy (1)

- **Event-Condition-Action (ECA)**
  - IF the Event is TRUE
  - IF the Condition is TRUE
    - THEN execute the Actions
  - Explicit programming of which condition and which action to be chosen (rationality is in the policy!)
Types of Policy (2)

- **Declarative (Goal)**
  - Express *what* should be done, *not how to do it*
  - Specifies criteria for choosing a set of states, any of which is acceptable
  - Rationality is generated by optimizer/planner

Diagram:
- **G**
- **Context manager**
  - User attribute
  - Flow info
  - Path info
- **Action manager**
- **Multi-constraint solver**
- **Multi-constraint solver**
- **To controller/device**
Declarative Description

- Only **describe the constraints** on possible state
- No information of how to do to achieve the state
The problem was to devise a walk through the city that would cross each bridge once and only once. Its negative resolution by Leonhard Euler in 1736 laid the foundations of graph theory and prefigured the idea of topology. The problem is too easy for me.

Programming: manually decompose logic an objects, permutation and combination. (complicate, rely on human knowledge and modeling)

Declarative: only describe objects and logic constraint

What is land
What is a bridge
How the 7 bridges are connected
Constraint on path: no path can be used twice
Output, no solution
The usage of policy rules to manage the behavior of managed entities

Policy is about governance, and can be expressed differently:
- Goal: No threshold will be violated in a set of link (link A, B and C)
- ECA: When this threshold of link A is violated, redirect the flow 1 to link B
SUPA Policy Engine Demo

External interface / declarative API

Goal Policy API (Constraint, Policy parameter)

declarative-policy
var {a:path; b:path;}
expr { a.delay<500 && b.delay<500 && a.links&b.links == EMPTY;}

Predefined Policy

User-Defined Policy
Define policy use standard policy interface. Express declarative without knowing network details.

Constraint Solving
The SUPA policy engine solves the multiple constraints and service requests.

Policy Enforcement
Output a solution and enforce the policy to network infrastructures with actions.
There will be a SUPA Policy Engine Demo at BnB on Thursday evening. Welcome to join us.

Questions?