



# Policy model for Control Loops

DMTF Redfish Forum  
Version 0.3 – October 2021



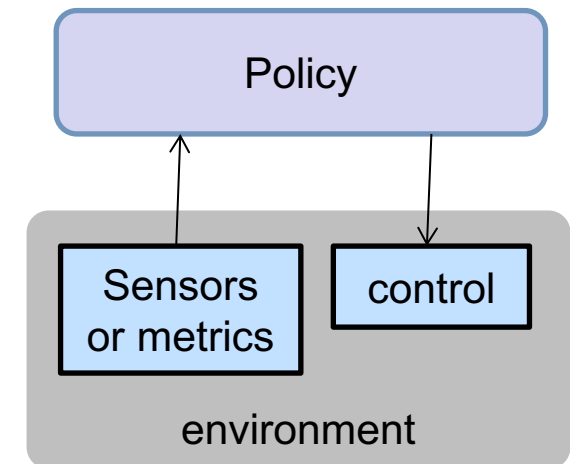
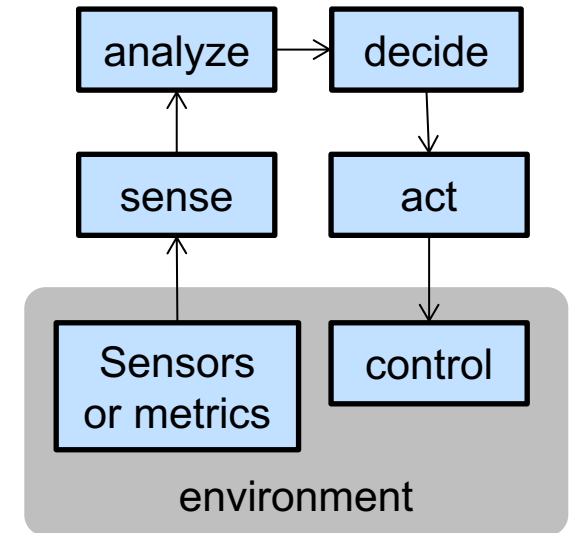
## Disclaimer

- The information in this presentation represents a snapshot of work in progress within the DMTF.
- This information is subject to change without notice. The standard specifications remain the normative reference for all information.
- For additional information, see the Distributed Management Task Force (DMTF) website.



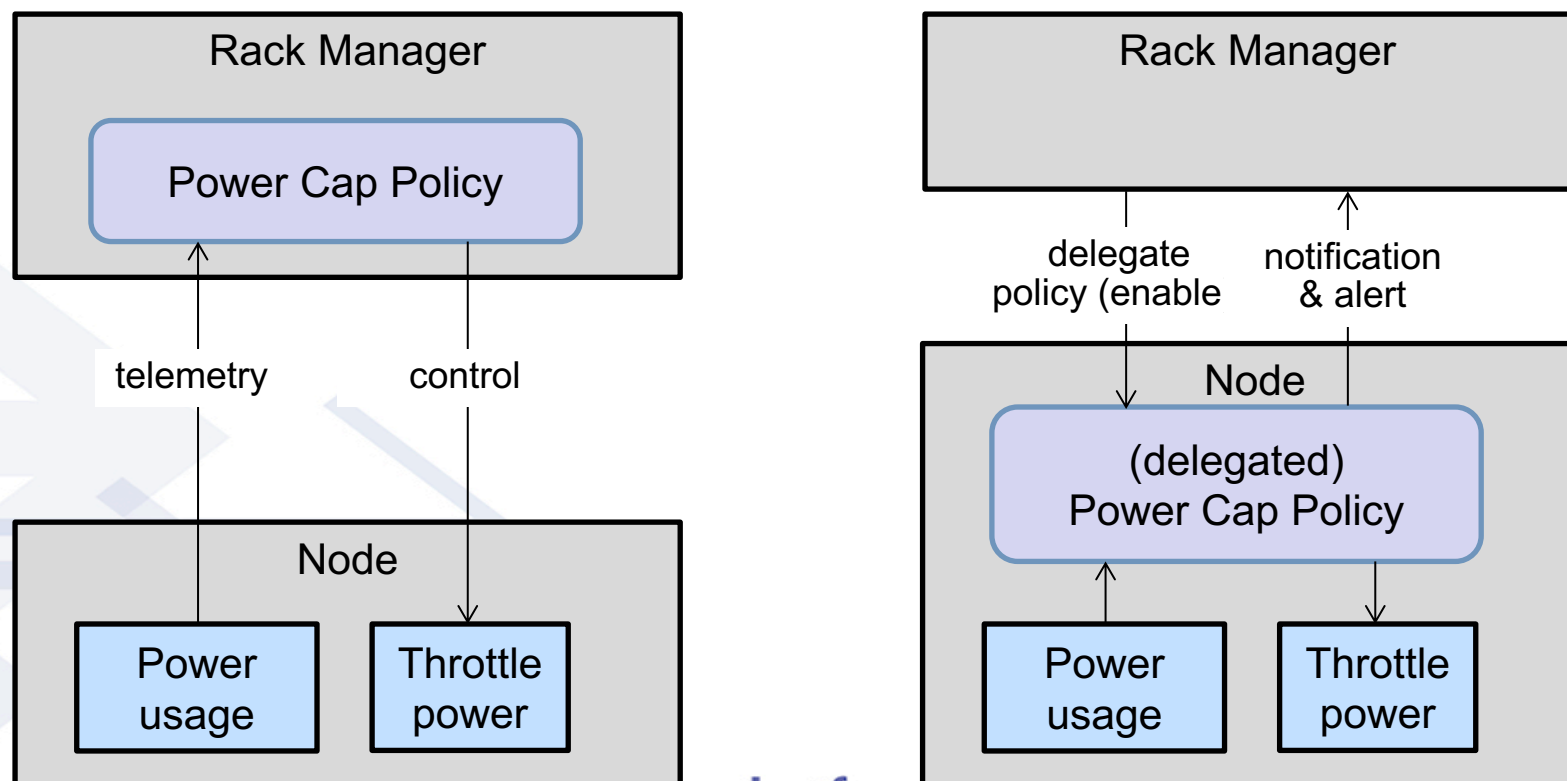
## Control Loops

- Models exist for metrics, sensors and controls
- A control loop can be constructed between sensors/metrics and controls
  - Sense input(s)
  - Analyzed the input(s)
  - Decide on action(s), if any
  - Perform action(s) via control(s)
- The control loop can be viewed as a policy construct
  - Analyze inputs and decide which controls to manipulate



## Polices can be Delegated (e.g. Power Capping)

- The policy construct can be delegated down a hierarchy
- A delegated policy authorizes a node to enforce a policy, locally





## Survey of Policy Management Models (details in backup)

- DEN/COPS Policy Statements
  - COPS = Common Open Policy Service (1996)
  - DEN = Directory Enabled Networking (circa 1998)
- DMTF/IETF Policy Framework (circa 2001)
  - PDP = Power Decision Points
  - PEP = Power Enforcement Points
- TM Forum - GB922 R18.0.2 "Shared Information/Data Model" (2018)
- ETSI - Context-Aware Policy Management Gap Analysis (2018)
- ONF - The Policy Framework for ONOS (2019)
- DSP1048 - Network Policy Management Profile (2021)
  - [https://www.dmtf.org/sites/default/files/standards/documents/DSP1048\\_1.0.0.pdf](https://www.dmtf.org/sites/default/files/standards/documents/DSP1048_1.0.0.pdf)
- ETSI - Draft - DGR/NFV-IFA042 v0.3.0 "Policy Model" (2021)



# DMTF Policy Mgmt Model

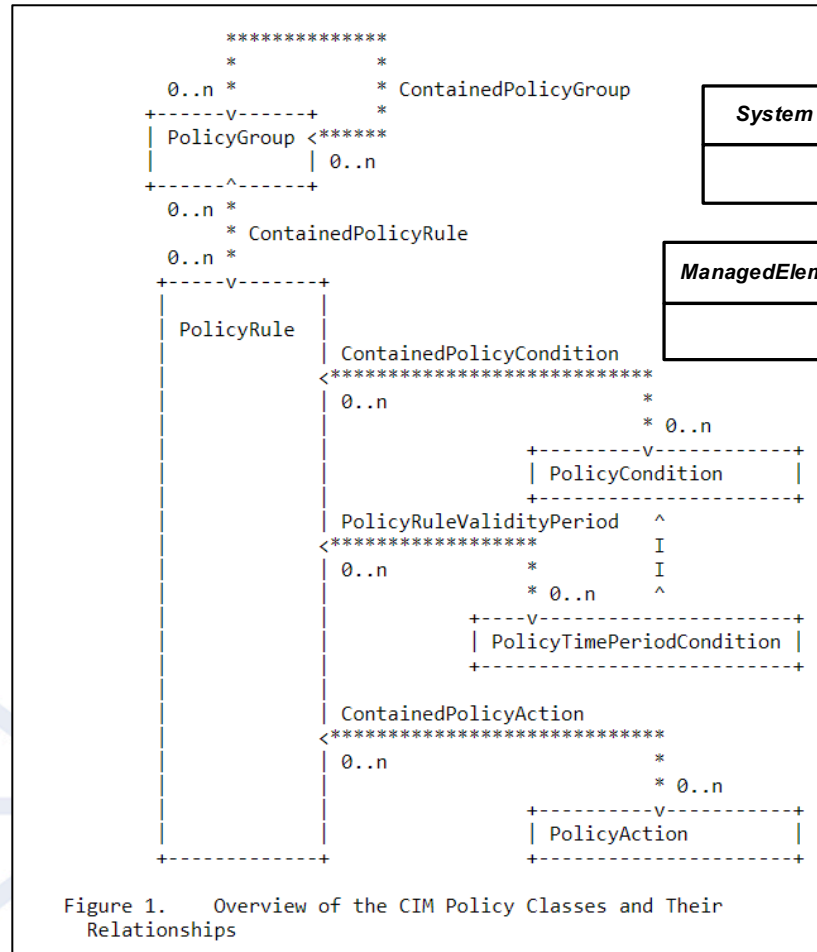
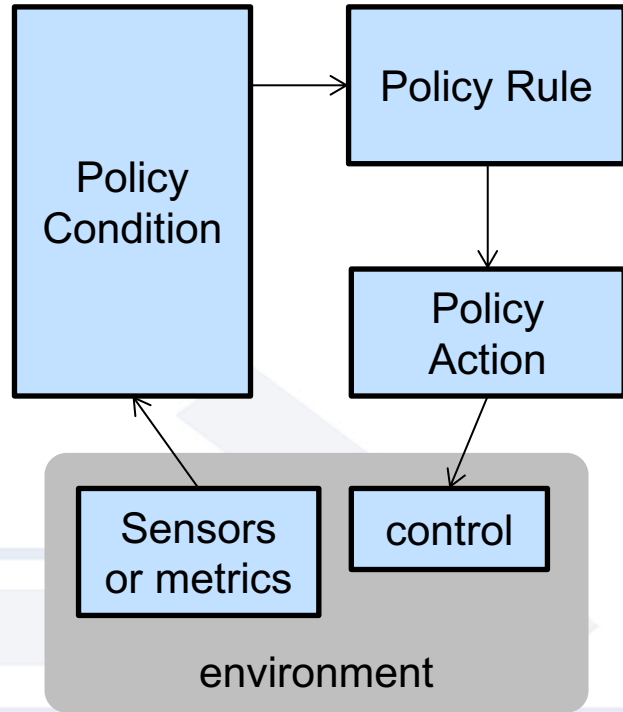
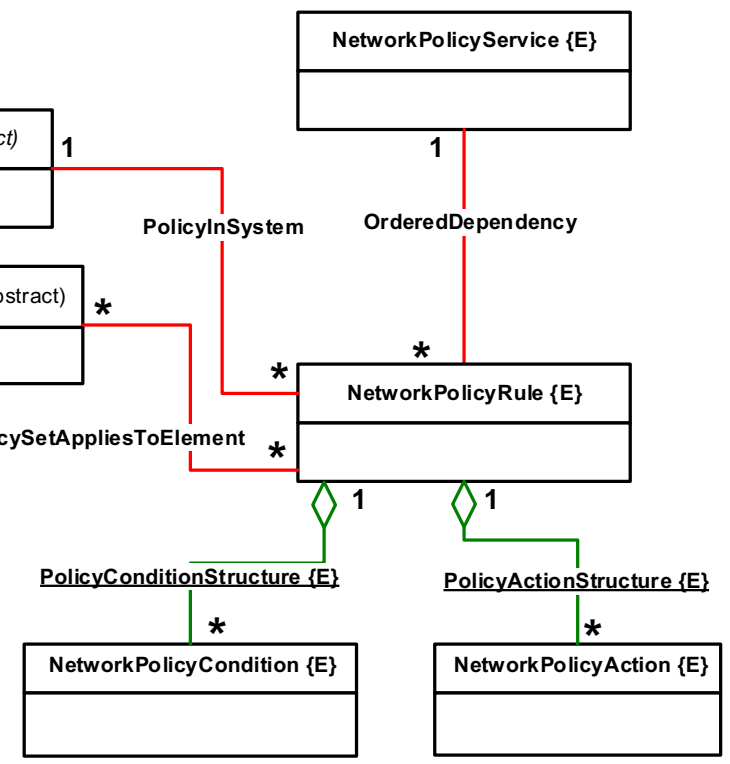


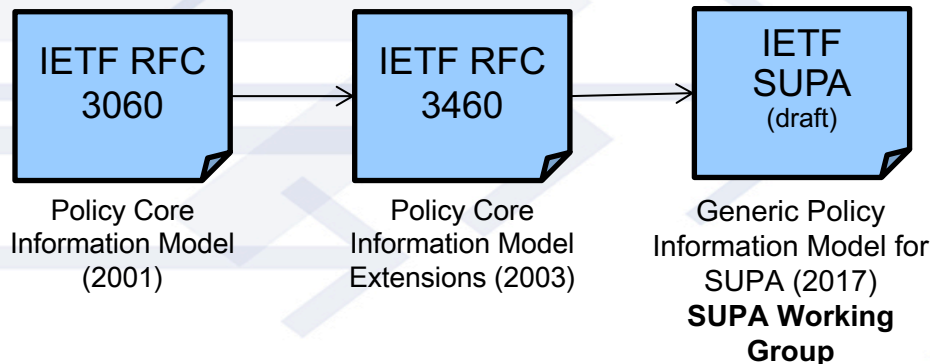
Figure 1. Overview of the CIM Policy Classes and Their Relationships



DSP1048 - Network Policy Management Profile (2021)

## TM Forum & ETSI References

- TM Forum - GB922 R18.0.2 "Shared Information/Data Model"
  - Policy - <https://www.tmforum.org/resources/standard/gb922-policy-r18-0-0>
  - R18.0.2 approved by TM Forum on 18-Jun-2018
- ETSI - Context-Aware Policy Management Gap Analysis (2018)
  - [https://www.etsi.org/deliver/etsi\\_gr/ENI/001\\_099/003/01.01.01\\_60/gr\\_ENI003v010101p.pdf](https://www.etsi.org/deliver/etsi_gr/ENI/001_099/003/01.01.01_60/gr_ENI003v010101p.pdf)
  - IETF's SUPA model - (Simplified Use of Policy Abstractions)
  - Draft DGR/NFV-IFA042 v0.3.0 Policy Model ,



### SUPA drafts

- "Generic Policy Information Model for Simplified Use of Policy Abstractions (SUPA)", May 30, 2017, draft-ietf-supa-generic-policy-info-model-03
- "SUPA Policy-based Management Framework", July 2017, draft-ietf-supa-policy-based-management-framework-03
- "Applicability of SUPA", March 2017, draft-cheng-sup-a-applicability-01



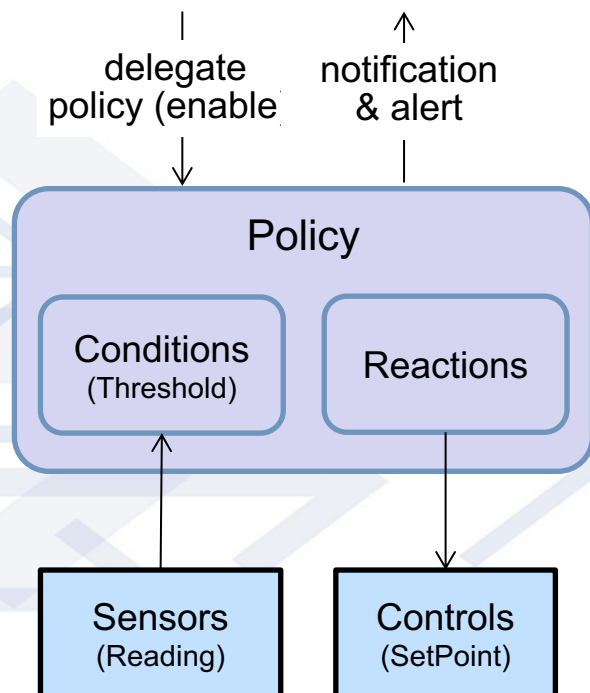
## Redfish Policy model

### Delegated Policy

- Policy schedule
- Conditions
  - dwell time
- Reactions
- Time to achieve
- Policy exception actions

### Notification & alerts

- Condition triggered (event?)
- Policy triggered (event?, prop?)
- Policy exception (event?)
  - Unable to achieve
- Invoking reaction (event?)
- Loss of sensor(s)



### Delegated Power Limit Policy example

- If power usage exceeds X, use delegated controls to reduce power usage below X within 50 ms (achieve goal within a timeframe)
- If unable to achieve, notify the delegator

### Some Variations

- Multiple conditions
  - Need to indicate AnyOf, AllOf, etc
- Multiple reactions
  - Policy prioritize of reaction?
  - Simultaneous

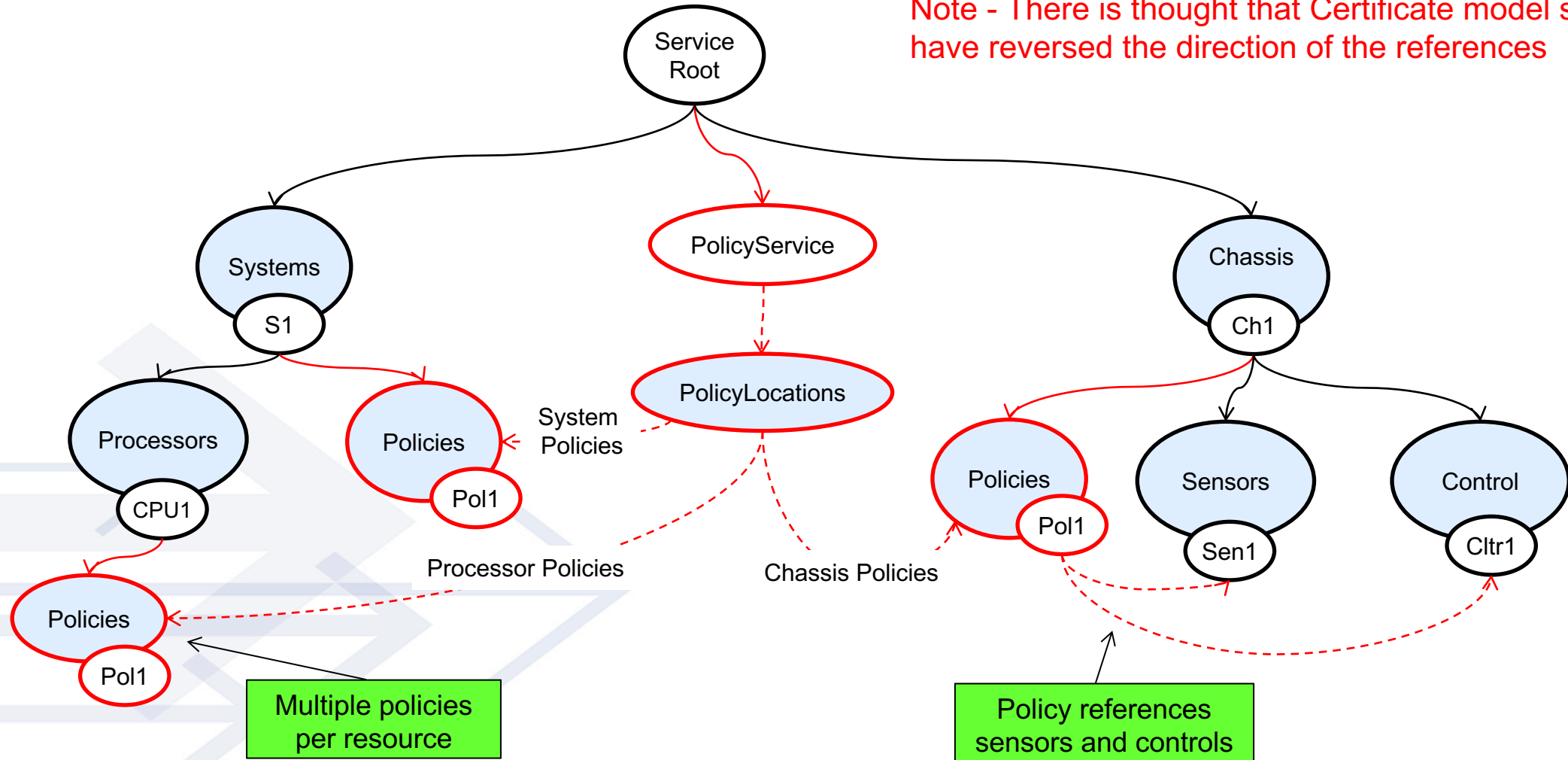




# Proposed Policy Model

Use the CertificateService modeling pattern

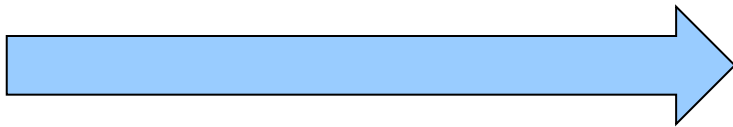
Note - There is thought that Certificate model should have reversed the direction of the references





## New resource schema

- PolicyService
- PolicyCollection
- PolicyLocations
- Policy



```
{
  "@odata.type": "#Policy.v1_0_0.Policy",
  "Name": "Policy0",
  "PolicyType": "PowerLimit",
  "PolicyEnabled": True,
  "PolicyTriggered": True,
  "State": { ... },
  "PolicySuspendPeriod": { ...},
  "Conditions": [
    {
      "Sensor": "/redfish/v1/Chassis/1/Sensors/TotalPower",
      "TriggerThreshold": 100
    }
  ],
  "Reactions": [
    {
      "Control": "/redfish/v1/Chassis/1/Controls/PowerUsage",
      "Setpoint": 120
    },
    {
      "Reaction": "SendEvent"
    }
  ],
  "PolicyExceptionActions": [
    {
      "Reaction": "SendEvent"
    }
  ]
}
```



## Questions

- Questions for the industry
  - What are use cases the model should comprehend
- Modeling questions:
  - Whether to model centralized policies (ref from PolicyLocations to Policies) vs a distributed policy model (inverse reference)?
  - Replace Reactions property with a Jobs construct for ordered and concurrent reactions
  - Disposition of existing "Triggers" resource and power controls
  - How to model a system with N processors, each with the same Policy (model scalability) - use wildcard mechanism?



# ***Backup Policy Management Models***



# DMTF/IETF models (circa 2001)

Document	Date	Status
<b>RFCs (6 hits)</b>		
RFC 3060 (was draft-ietf-policy-core-info-model) <b>Policy Core Information Model -- Version 1 Specification</b>	2001-02 100 pages	Proposed Standard RFC Updated by RFC3460
RFC 3198 (was draft-ietf-policy-terminology) <b>Terminology for Policy-Based Management</b>	2001-11 21 pages	Informational RFC
RFC 3460 (was draft-ietf-policy-pcim-ext) <b>Policy Core Information Model (PCIM) Extensions</b>	2003-01 93 pages	Proposed Standard RFC
RFC 3644 (was draft-ietf-policy-qos-info-model) <b>Policy Quality of Service (QoS) Information Model</b>	2003-11 73 pages	Proposed Standard RFC
RFC 3670 (was draft-ietf-policy-qos-device-info-model) <b>Information Model for Describing Network Device QoS Datapath Mechanisms</b>	2004-01 97 pages	Proposed Standard RFC
RFC 3703 (was draft-ietf-policy-core-schema) <b>Policy Core Lightweight Directory Access Protocol (LDAP) Schema</b>	2004-02 61 pages	Proposed Standard RFC Updated by RFC4104

<https://datatracker.ietf.org/wg/policy/documents>

<https://tools.ietf.org/html/draft-wang-netmod-yang-policy-dm-02>

<http://www.watersprings.org/pub/id/draft-wang-netmod-yang-policy-dm-01.html>

### 3. Overview of the Schema

The following diagram provides an overview of the five classes that comprise the CIM core schema, and their relationships to each other. Note that the two extension classes VendorPolicyCondition and VendorPolicyAction are not shown.

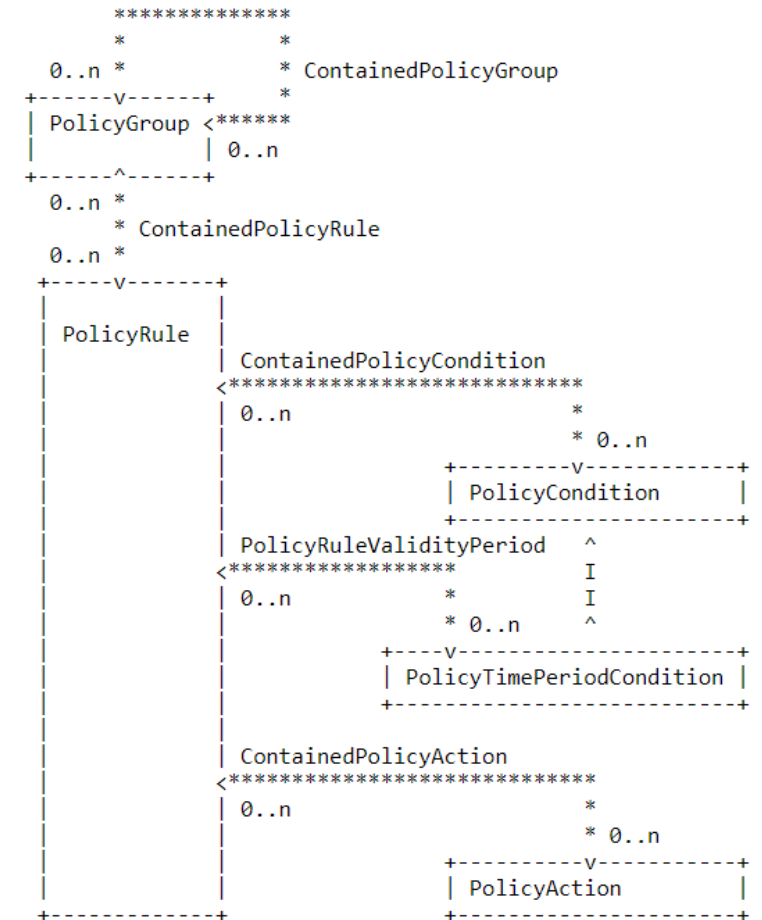


Figure 1. Overview of the CIM Policy Classes and Their Relationships

# "Policy-Based Network Management

2004 John Strasser, Morgan Kaufmann  
<https://books.google.com/books?id=lid7plhI1PQC>

COPS = Common Open Policy Service (1996)  
DEN = Directory Enabled Networking (circa 1998)  
DMTF/IETF (1999)  
PDP = Power Decision Points  
PEP = Power Enforcement Points

## 7.7 DEN-ng PolicyStatements

A *PolicyStatement* models the triplet (variable, operator, value) that is used by both the *PolicyCondition* and *PolicyAction* classes. Note that the semantics differ in how this triplet is used for a *PolicyCondition* compared with a *PolicyAction*; the difference is reflected in the types of operators that are allowed to be used in each case. For conditions, we want the semantics of "variable relates to value," where "relates to" is usually the match operator, but could also be other applicable operators (e.g., a comparison operator). For actions, we want the semantics of "set variable to value." Here, the only operator allowed is the set operator. These semantics are enforced using OCL.

Figure 7-7 shows a simplified representation of the *PolicyStatement* class and its three principal subclasses and relationships. This section will explore these classes and relationships in more detail.

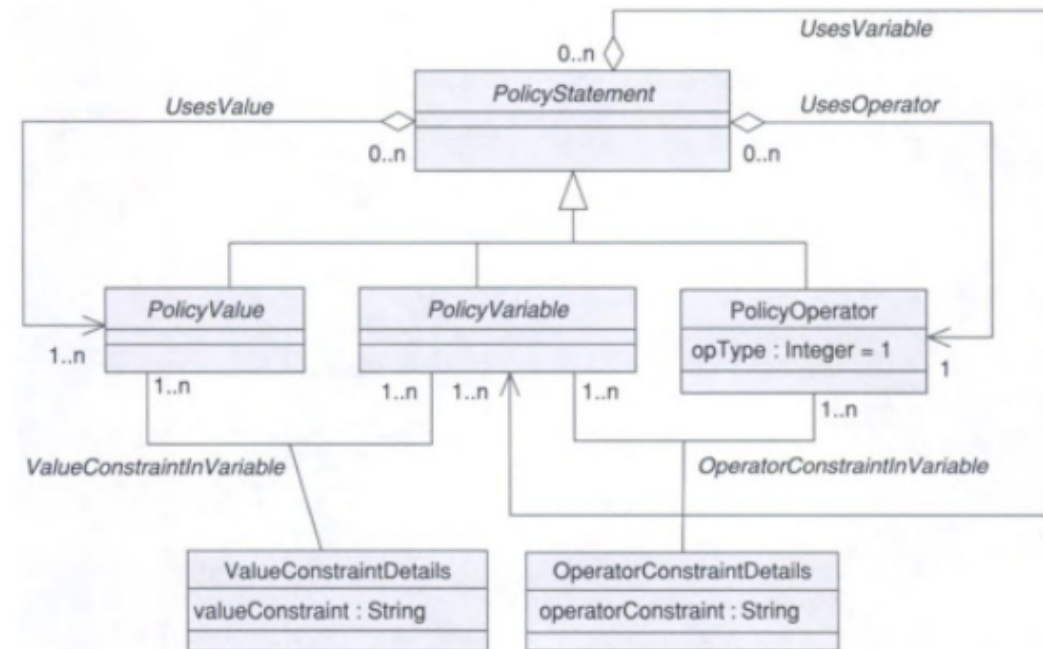


Figure 7-7 Simplified representation of a DEN-ng *PolicyStatement*.

### 7.7.1 Methods of a *PolicyStatement*



## ETSI NFV - RESTful protocols spec for the Policy Manager I/F

### Policy Resource

- Policy Version - Versions of the policy
- Selected Version - Selected version of the policy
- Activation Status - activated or deactivated
- Transfer Status - whether content of policy has been transferred
- Associations - identifiers of the entities to which the policy is associated

### Policy Actions

- CreatePolicy
- TransferPolicy
- DeletePolicy
- ModifyPolicy

[https://www.etsi.org/deliver/etsi\\_gs/NFV-SOL/001\\_099/012/03.04.01\\_60/gs\\_nfv-sol012v030401p.pdf](https://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/012/03.04.01_60/gs_nfv-sol012v030401p.pdf)



# ETSI NFV - RESTful protocols spec for the Policy Manager I/F

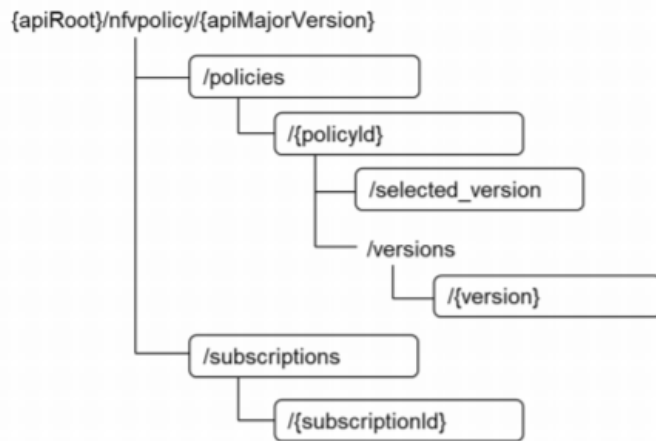


Figure 5.3-1: Resource URI structure of the policy management interface

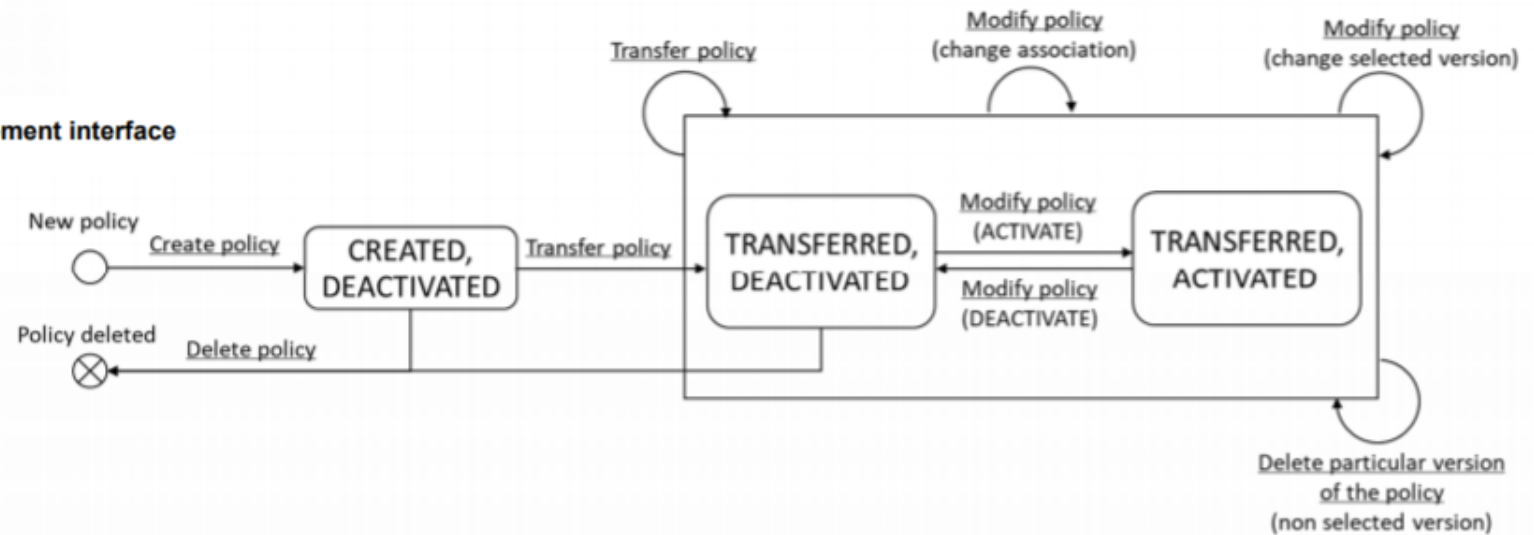


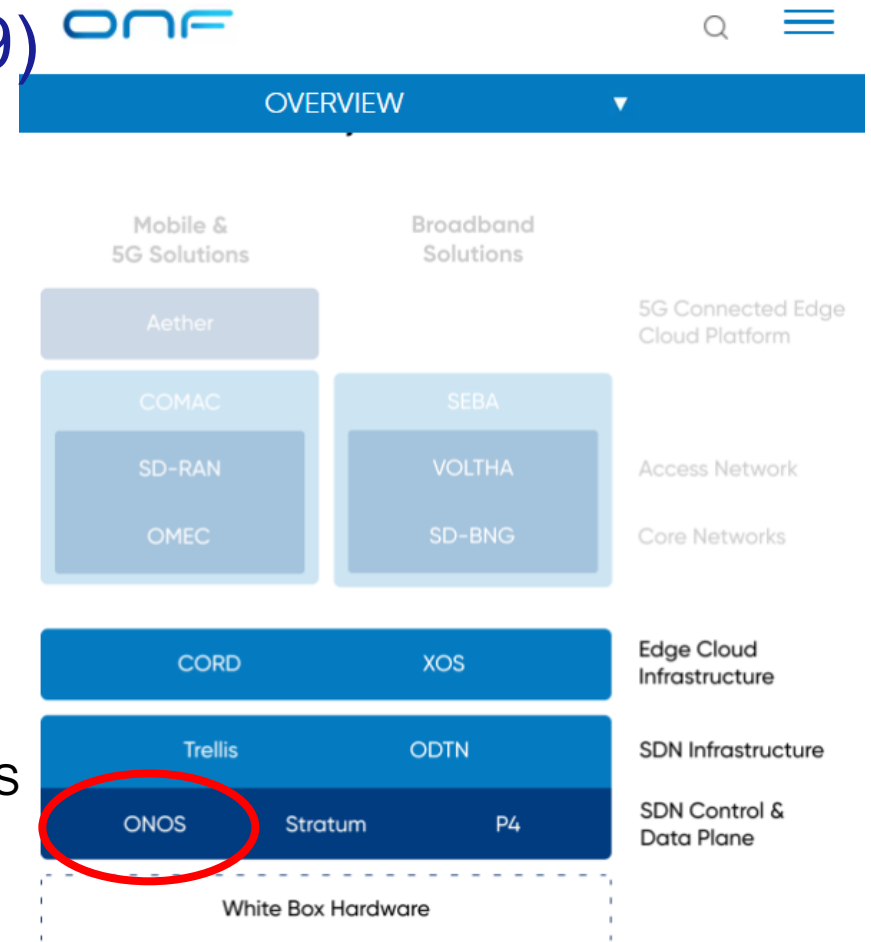
Figure 5.7.2-1: States and state transition of the policy management





# ONF Reference Implementation

- The Policy Framework for ONOS (Feb 2019)
  - ONOS = Open Network OS
  - The control plane for a software-defined network (SDN) - v1.0.0 (2014) to v2.5.1 (2021)



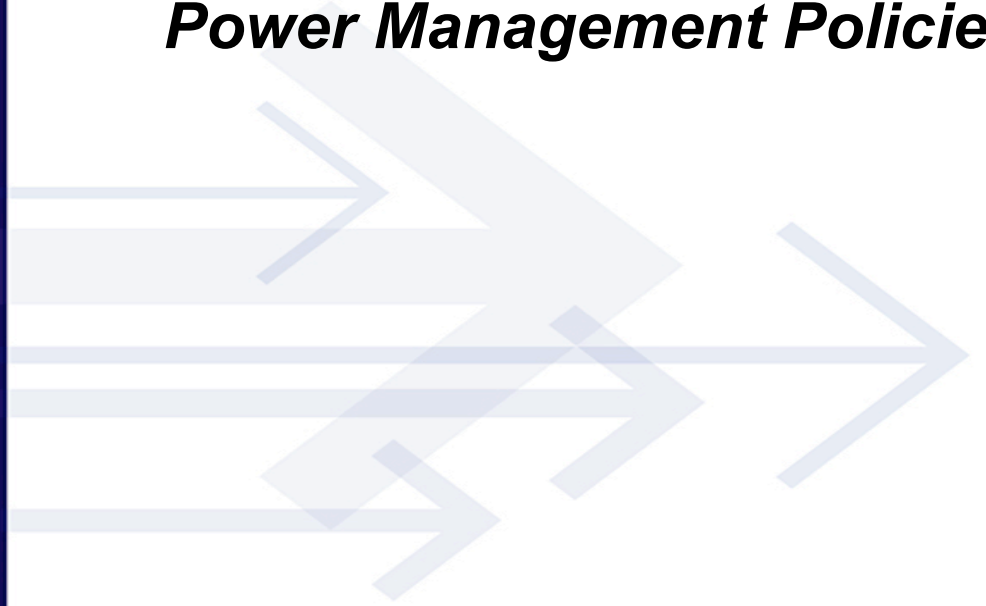
<https://wiki.onosproject.org/display/ONOS/POLICY+FRAMEWORK+FOR+ONOS>

<https://opennetworking.org/onos>



# ***Examples***

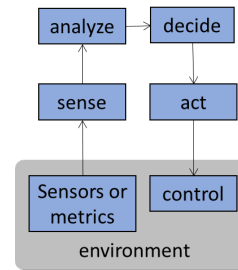
***Power Management Policies of Platform, Subsystem and Components***



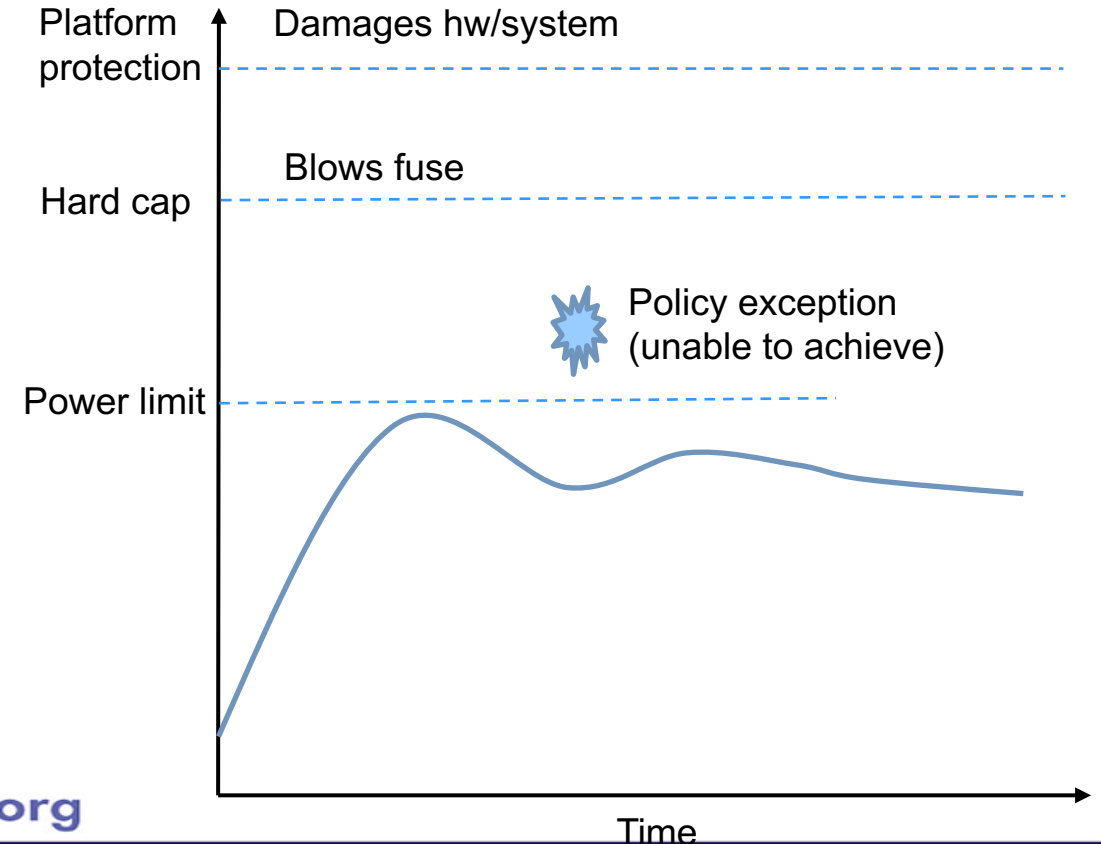
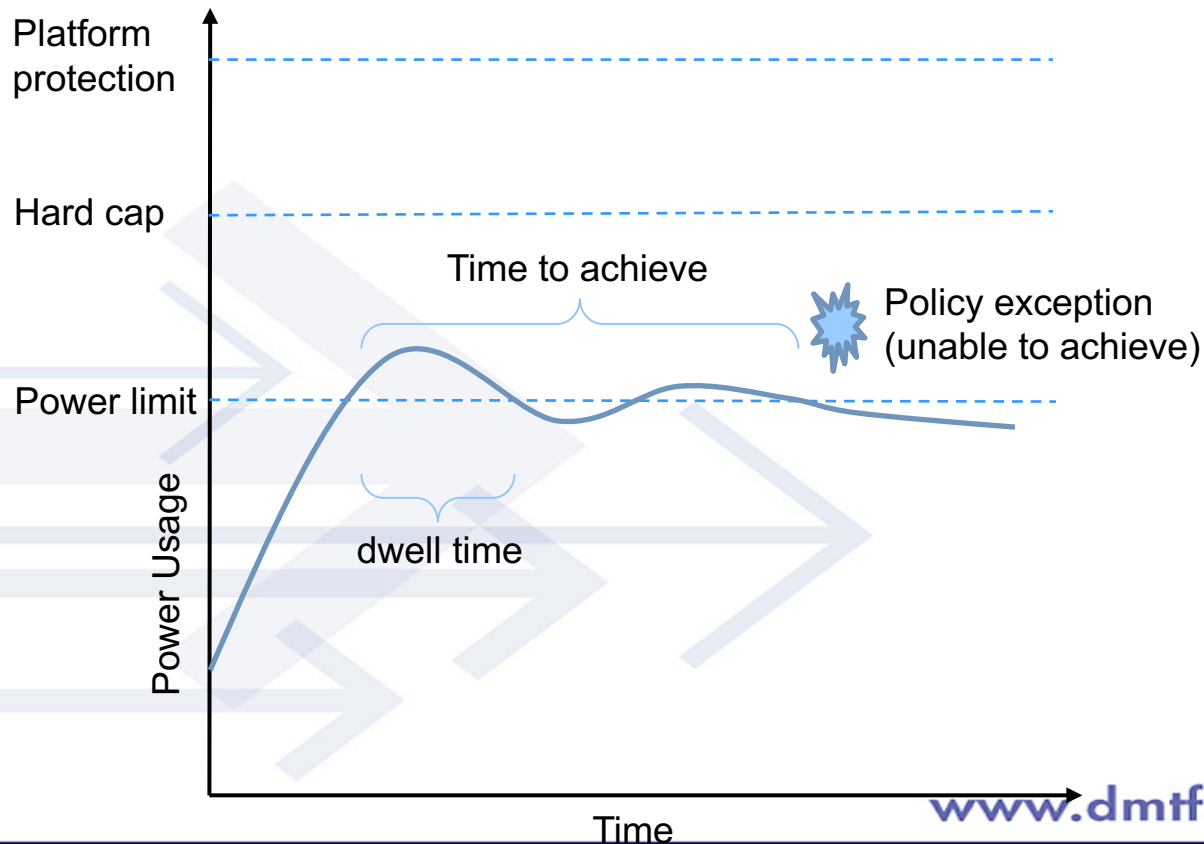


# Power Limiting Policies - nomenclature

Threshold Triggered Action policy  
(reactive)

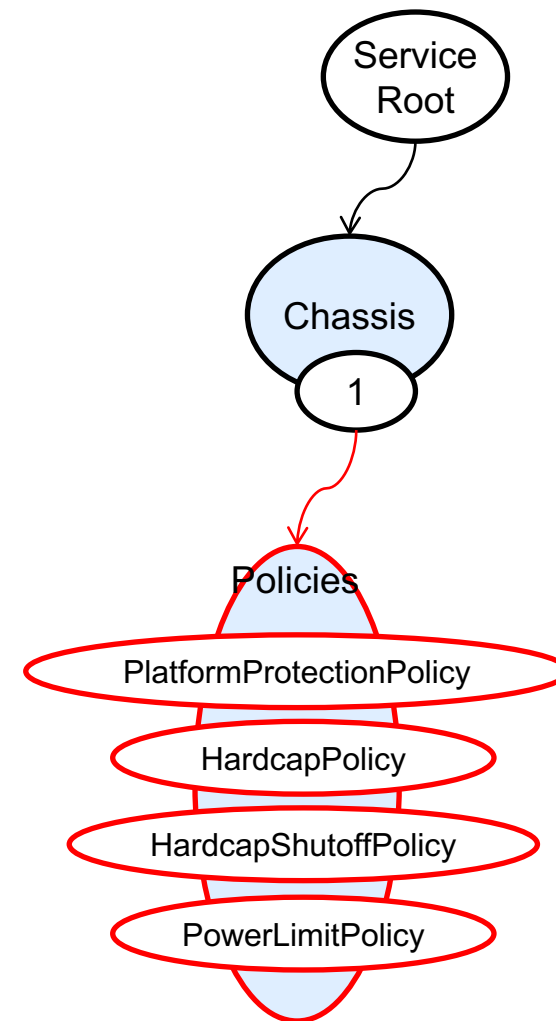


Pre-emptive Action policy  
(predictive)



## Example - Platform Power Policies

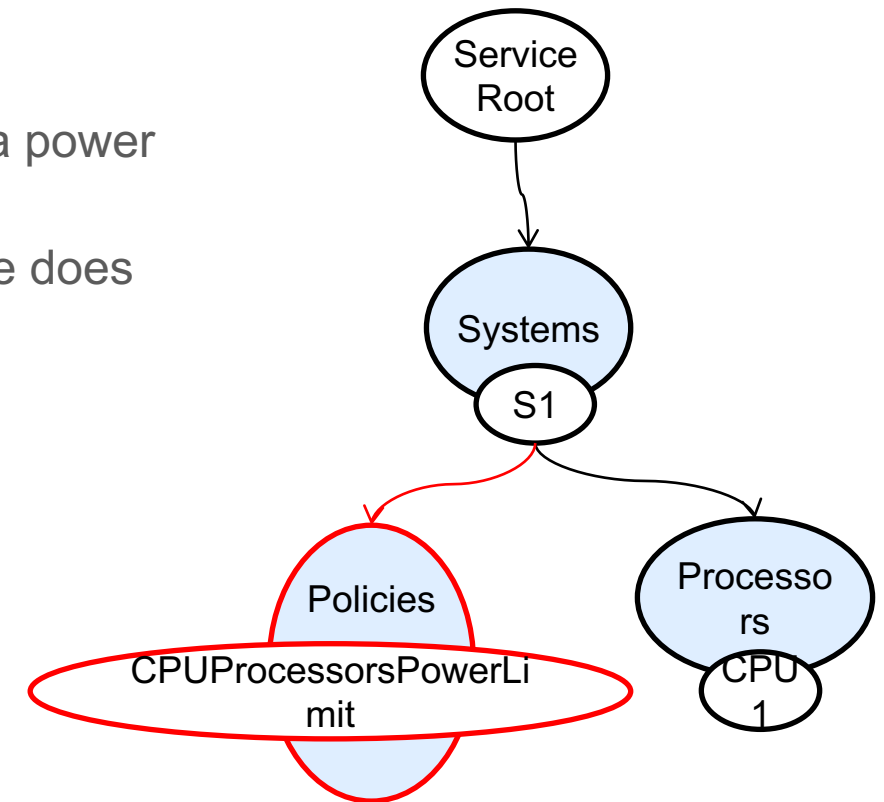
- **Power Limit Policy (read/write)**
  - Platform does not exceed the power limit (on average). The platform uses its power limiting capabilities and sends exception if unable.
- **Hardcap Policy (read/write)**
  - Platform should not exceed the power limit (else breaker is tripped). The platform uses its power limiting capabilities prior to cap and sends excepti.
- **Hardcap Shutoff Policy (read/write)**
  - When platform exceeds a threshold, then power off the platform
  - (Could be the exception action for Hardcap Policy)
- **Platform Protection Policy (read-only)**
  - Configured by OEM/System Integrator





## Example - Processor Domain Power Policies

- Power Limit Policy (read/write)
  - The power consumption of the processors does not exceed a power limit.
  - The power consumption of process of specific processor-type does not exceed a power limit





## Example - Processor Power Policies

- LossOfSensor Policy
  - When readings cannot be obtained from the processor, throttle to a percentage of CPU power usages (e.g. 100%)

