

# Clamp Rest API

# Table of Contents

- 1. Overview ..... 1
  - 1.1. Version information ..... 1
  - 1.2. URI scheme..... 1
- 2. Paths ..... 2
  - 2.1. GET /v1/healthcheck..... 2
    - 2.1.1. Responses ..... 2
    - 2.1.2. Produces ..... 2
  - 2.2. GET /v1/user/getUser ..... 2
    - 2.2.1. Responses ..... 2
    - 2.2.2. Produces ..... 2
  - 2.3. GET /v2/clampInformation..... 2
    - 2.3.1. Responses ..... 2
    - 2.3.2. Produces ..... 2
  - 2.4. GET /v2/dictionary ..... 2
    - 2.4.1. Responses ..... 3
    - 2.4.2. Produces ..... 3
  - 2.5. PUT /v2/dictionary ..... 3
    - 2.5.1. Parameters ..... 3
    - 2.5.2. Responses ..... 3
    - 2.5.3. Consumes ..... 3
    - 2.5.4. Produces ..... 3
  - 2.6. GET /v2/dictionary/secondary/names ..... 3
    - 2.6.1. Responses ..... 3
    - 2.6.2. Produces ..... 4
  - 2.7. GET /v2/dictionary/{dictionaryName}..... 4
    - 2.7.1. Parameters ..... 4
    - 2.7.2. Responses ..... 4
    - 2.7.3. Produces ..... 4
  - 2.8. PUT /v2/dictionary/{name}..... 4
    - 2.8.1. Parameters ..... 4
    - 2.8.2. Responses ..... 4
    - 2.8.3. Consumes ..... 4
    - 2.8.4. Produces ..... 5
  - 2.9. DELETE /v2/dictionary/{name} ..... 5
    - 2.9.1. Parameters ..... 5
    - 2.9.2. Responses ..... 5

2.9.3. Produces .....	5
2.10. DELETE /v2/dictionary/{name}/elements/{shortName} .....	5
2.10.1. Parameters .....	5
2.10.2. Responses .....	5
2.10.3. Produces .....	5
2.11. PUT /v2/loop/addOperationaPolicy/{loopName}/policyModel/{policyType}/{policyVersion} ....	6
2.11.1. Parameters .....	6
2.11.2. Responses .....	6
2.11.3. Produces .....	6
2.12. POST /v2/loop/create/{loopName}?templateName={templateName} .....	6
2.12.1. Parameters .....	6
2.12.2. Responses .....	6
2.12.3. Consumes .....	7
2.12.4. Produces .....	7
2.13. PUT /v2/loop/delete/{loopName}. .....	7
2.13.1. Parameters .....	7
2.13.2. Responses .....	7
2.14. PUT /v2/loop/deploy/{loopName} .....	7
2.14.1. Parameters .....	7
2.14.2. Responses .....	7
2.14.3. Produces .....	8
2.15. GET /v2/loop/getAllNames .....	8
2.15.1. Responses .....	8
2.15.2. Produces .....	8
2.16. GET /v2/loop/getstatus/{loopName} .....	8
2.16.1. Parameters .....	8
2.16.2. Responses .....	8
2.16.3. Produces .....	8
2.17. PUT /v2/loop/refreshMicroServicePolicyJsonSchema/{loopName}/{microServicePolicyName}..	8
2.17.1. Parameters .....	8
2.17.2. Responses .....	9
2.17.3. Produces .....	9
2.18. PUT /v2/loop/refreshOperationalPolicyJsonSchema/{loopName}/{operationalPolicyName} ....	9
2.18.1. Parameters .....	9
2.18.2. Responses .....	9
2.18.3. Produces .....	9
2.19. PUT /v2/loop/removeOperationaPolicy/{loopName}/policyModel/{policyType}/{policyVersion}	10
2.19.1. Parameters .....	10

2.19.2. Responses	10
2.19.3. Produces	10
2.20. PUT /v2/loop/restart/{loopName}	10
2.20.1. Parameters	10
2.20.2. Responses	10
2.20.3. Produces	11
2.21. PUT /v2/loop/stop/{loopName}	11
2.21.1. Parameters	11
2.21.2. Responses	11
2.21.3. Produces	11
2.22. PUT /v2/loop/submit/{loopName}	11
2.22.1. Parameters	11
2.22.2. Responses	11
2.22.3. Produces	11
2.23. GET /v2/loop/svgRepresentation/{loopName}	12
2.23.1. Parameters	12
2.23.2. Responses	12
2.23.3. Produces	12
2.24. PUT /v2/loop/undeploy/{loopName}	12
2.24.1. Parameters	12
2.24.2. Responses	12
2.24.3. Produces	12
2.25. POST /v2/loop/updateGlobalProperties/{loopName}	12
2.25.1. Parameters	12
2.25.2. Responses	13
2.25.3. Consumes	13
2.25.4. Produces	13
2.26. POST /v2/loop/updateMicroservicePolicy/{loopName}	13
2.26.1. Parameters	13
2.26.2. Responses	13
2.26.3. Consumes	13
2.26.4. Produces	14
2.27. POST /v2/loop/updateOperationalPolicies/{loopName}	14
2.27.1. Parameters	14
2.27.2. Responses	14
2.27.3. Consumes	14
2.27.4. Produces	14
2.28. GET /v2/loop/{loopName}	14

2.28.1. Parameters . . . . .	14
2.28.2. Responses . . . . .	14
2.28.3. Produces . . . . .	15
2.29. POST /v2/policyToscaModels . . . . .	15
2.29.1. Parameters . . . . .	15
2.29.2. Responses . . . . .	15
2.29.3. Consumes . . . . .	15
2.29.4. Produces . . . . .	15
2.30. GET /v2/policyToscaModels . . . . .	15
2.30.1. Responses . . . . .	15
2.30.2. Produces . . . . .	15
2.31. GET /v2/policyToscaModels/yaml/{policyModelType}/{policyModelVersion} . . . . .	16
2.31.1. Parameters . . . . .	16
2.31.2. Responses . . . . .	16
2.31.3. Produces . . . . .	16
2.32. GET /v2/policyToscaModels/{policyModelType}/{policyModelVersion} . . . . .	16
2.32.1. Parameters . . . . .	16
2.32.2. Responses . . . . .	16
2.32.3. Produces . . . . .	17
2.33. PUT /v2/policyToscaModels/{policyModelType}/{policyModelVersion} . . . . .	17
2.33.1. Parameters . . . . .	17
2.33.2. Responses . . . . .	17
2.33.3. Consumes . . . . .	17
2.33.4. Produces . . . . .	17
2.34. GET /v2/templates. . . . .	17
2.34.1. Responses . . . . .	18
2.34.2. Produces . . . . .	18
2.35. GET /v2/templates/names . . . . .	18
2.35.1. Responses . . . . .	18
2.35.2. Produces . . . . .	18
2.36. GET /v2/templates/{templateName} . . . . .	18
2.36.1. Parameters . . . . .	18
2.36.2. Responses . . . . .	18
2.36.3. Produces . . . . .	18
2.37. GET /v2/templates/{templateName}/svgRepresentation. . . . .	19
2.37.1. Parameters . . . . .	19
2.37.2. Responses . . . . .	19
2.37.3. Produces . . . . .	19

- 3. Definitions ..... 20
  - 3.1. ClampInformation ..... 20
  - 3.2. CldsHealthCheck ..... 20
  - 3.3. Dictionary ..... 20
  - 3.4. DictionaryElement ..... 21
  - 3.5. ExternalComponent ..... 21
  - 3.6. ExternalComponentState ..... 22
  - 3.7. JsonArray ..... 22
  - 3.8. JsonNull ..... 23
  - 3.9. JsonObject ..... 24
  - 3.10. JsonPrimitive ..... 26
  - 3.11. Loop ..... 27
  - 3.12. LoopElementModel ..... 28
  - 3.13. LoopLog ..... 29
  - 3.14. LoopTemplate ..... 29
  - 3.15. LoopTemplateLoopElementModel ..... 30
  - 3.16. MicroServicePolicy ..... 30
  - 3.17. Number ..... 31
  - 3.18. OperationalPolicy ..... 31
  - 3.19. PolicyModel ..... 32
  - 3.20. Service ..... 33

# Chapter 1. Overview

## 1.1. Version information

*Version* : 5.0.0-SNAPSHOT

## 1.2. URI scheme

*Host* : localhost:46561

*BasePath* : /restservices/clds/

*Schemes* : HTTP

# Chapter 2. Paths

## 2.1. GET /v1/healthcheck

### 2.1.1. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">CldsHealthCheck</a>

### 2.1.2. Produces

- `application/json`

## 2.2. GET /v1/user/getUser

### 2.2.1. Responses

HTTP Code	Schema
200	No Content

### 2.2.2. Produces

- `text/plain`

## 2.3. GET /v2/clampInformation

### 2.3.1. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">ClampInformation</a>

### 2.3.2. Produces

- `application/json`

## 2.4. GET /v2/dictionary



### 2.4.1. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Dictionary</a>

### 2.4.2. Produces

- [application/json](#)

## 2.5. PUT /v2/dictionary

### 2.5.1. Parameters

Type	Name	Schema
Body	<b>body</b> <i>required</i>	<a href="#">Dictionary</a>

### 2.5.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Dictionary</a>

### 2.5.3. Consumes

- [application/json](#)

### 2.5.4. Produces

- [application/json](#)

## 2.6. GET /v2/dictionary/secondary/names

### 2.6.1. Responses

HTTP Code	Description	Schema
200	Output type	< string > array

## 2.6.2. Produces

- `application/json`

## 2.7. GET /v2/dictionary/{dictionaryName}

### 2.7.1. Parameters

Type	Name	Schema
Path	<b>dictionaryName</b> <i>required</i>	string

### 2.7.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Dictionary</a>

### 2.7.3. Produces

- `application/json`

## 2.8. PUT /v2/dictionary/{name}

### 2.8.1. Parameters

Type	Name	Schema
Path	<b>name</b> <i>required</i>	string
Body	<b>body</b> <i>required</i>	<a href="#">Dictionary</a>

### 2.8.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Dictionary</a>

### 2.8.3. Consumes

- `application/json`

## 2.8.4. Produces

- `application/json`

## 2.9. DELETE /v2/dictionary/{name}

### 2.9.1. Parameters

Type	Name	Schema
Path	<b>name</b> <i>required</i>	string

### 2.9.2. Responses

HTTP Code	Schema
200	No Content

### 2.9.3. Produces

- `application/json`

## 2.10. DELETE /v2/dictionary/{name}/elements/{shortName}

### 2.10.1. Parameters

Type	Name	Schema
Path	<b>name</b> <i>required</i>	string
Path	<b>shortName</b> <i>required</i>	string

### 2.10.2. Responses

HTTP Code	Schema
200	No Content

### 2.10.3. Produces

- `application/json`

## 2.11. PUT

**/v2/loop/addOperationaPolicy/{loopName}/policyModel/{policyType}/{policyVersion}**

### 2.11.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Path	<b>policyType</b> <i>required</i>	string
Path	<b>policyVersion</b> <i>required</i>	string

### 2.11.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.11.3. Produces

- [application/json](#)

## 2.12. POST

**/v2/loop/create/{loopName}?templateName={templateName}**

### 2.12.1. Parameters

Type	Name	Schema
Path	<b>loopName}?templateName={templateName</b> <i>required</i>	string

### 2.12.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.12.3. Consumes

- `application/json`

### 2.12.4. Produces

- `application/json`

## 2.13. PUT /v2/loop/delete/{loopName}

### 2.13.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.13.2. Responses

HTTP Code	Schema
200	No Content

## 2.14. PUT /v2/loop/deploy/{loopName}

### 2.14.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.14.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.14.3. Produces

- `application/json`

## 2.15. GET /v2/loop/getAllNames

### 2.15.1. Responses

HTTP Code	Description	Schema
200	Output type	< string > array

### 2.15.2. Produces

- `application/json`

## 2.16. GET /v2/loop/getstatus/{loopName}

### 2.16.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.16.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.16.3. Produces

- `application/json`

## 2.17. PUT

**/v2/loop/refreshMicroServicePolicyJsonSchema/{loopName}/{microServicePolicyName}**

### 2.17.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Path	<b>microServicePolicyName</b> <i>required</i>	string

### 2.17.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.17.3. Produces

- `application/json`

## 2.18. PUT

**`/v2/loop/refreshOperationalPolicyJsonSchema/{loopName}/{operationalPolicyName}`**

### 2.18.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Path	<b>operationalPolicyName</b> <i>required</i>	string

### 2.18.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.18.3. Produces

- `application/json`

## 2.19. PUT

### /v2/loop/removeOperationalPolicy/{loopName}/policyModel/{policyType}/{policyVersion}

#### 2.19.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Path	<b>policyType</b> <i>required</i>	string
Path	<b>policyVersion</b> <i>required</i>	string

#### 2.19.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

#### 2.19.3. Produces

- [application/json](#)

## 2.20. PUT /v2/loop/restart/{loopName}

#### 2.20.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

#### 2.20.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>



### 2.20.3. Produces

- `application/json`

## 2.21. PUT /v2/loop/stop/{loopName}

### 2.21.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.21.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.21.3. Produces

- `application/json`

## 2.22. PUT /v2/loop/submit/{loopName}

### 2.22.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.22.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.22.3. Produces

- `application/json`

## 2.23. GET /v2/loop/svgRepresentation/{loopName}

### 2.23.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.23.2. Responses

HTTP Code	Description	Schema
200	Output type	string

### 2.23.3. Produces

- `application/xml`

## 2.24. PUT /v2/loop/undeploy/{loopName}

### 2.24.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

### 2.24.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.24.3. Produces

- `application/json`

## 2.25. POST /v2/loop/updateGlobalProperties/{loopName}

### 2.25.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Body	<b>body</b> <i>required</i>	JsonObject

### 2.25.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

### 2.25.3. Consumes

- `application/json`

### 2.25.4. Produces

- `application/json`

## 2.26. POST

### /v2/loop/updateMicroservicePolicy/{loopName}

#### 2.26.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Body	<b>body</b> <i>required</i>	MicroServicePolicy

#### 2.26.2. Responses

HTTP Code	Description	Schema
200	Output type	MicroServicePolicy

#### 2.26.3. Consumes

- `application/json`

## 2.26.4. Produces

- `application/json`

## 2.27. POST

### `/v2/loop/updateOperationalPolicies/{loopName}`

#### 2.27.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string
Body	<b>body</b> <i>required</i>	<a href="#">JsonArray</a>

#### 2.27.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

#### 2.27.3. Consumes

- `application/json`

#### 2.27.4. Produces

- `application/json`

## 2.28. GET `/v2/loop/{loopName}`

#### 2.28.1. Parameters

Type	Name	Schema
Path	<b>loopName</b> <i>required</i>	string

#### 2.28.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">Loop</a>

### 2.28.3. Produces

- [application/json](#)

## 2.29. POST /v2/policyToscaModels

### 2.29.1. Parameters

Type	Name	Schema
Body	<b>body</b> <i>required</i>	string

### 2.29.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">PolicyModel</a>

### 2.29.3. Consumes

- [plain/text](#)

### 2.29.4. Produces

- [application/json](#)

## 2.30. GET /v2/policyToscaModels

### 2.30.1. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">PolicyModel</a>

### 2.30.2. Produces

- [application/json](#)

## 2.31. GET

### /v2/policyToscaModels/yaml/{policyModelType}/{policyModelVersion}

#### 2.31.1. Parameters

Type	Name	Schema
Path	<b>policyModelType</b> <i>required</i>	string
Path	<b>policyModelVersion</b> <i>required</i>	string

#### 2.31.2. Responses

HTTP Code	Description	Schema
200	Output type	string

#### 2.31.3. Produces

- `application/json`

## 2.32. GET

### /v2/policyToscaModels/{policyModelType}/{policyModelVersion}

#### 2.32.1. Parameters

Type	Name	Schema
Path	<b>policyModelType</b> <i>required</i>	string
Path	<b>policyModelVersion</b> <i>required</i>	string

#### 2.32.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">PolicyModel</a>

### 2.32.3. Produces

- [application/json](#)

## 2.33. PUT

### /v2/policyToscaModels/{policyModelType}/{policyModelVersion}

#### 2.33.1. Parameters

Type	Name	Schema
Path	<b>policyModelType</b> <i>required</i>	string
Path	<b>policyModelVersion</b> <i>required</i>	string
Body	<b>body</b> <i>required</i>	string

#### 2.33.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">PolicyModel</a>

#### 2.33.3. Consumes

- [plain/text](#)

#### 2.33.4. Produces

- [application/json](#)

## 2.34. GET /v2/templates

### 2.34.1. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">LoopTemplate</a>

### 2.34.2. Produces

- `application/json`

## 2.35. GET /v2/templates/names

### 2.35.1. Responses

HTTP Code	Description	Schema
200	Output type	< string > array

### 2.35.2. Produces

- `application/json`

## 2.36. GET /v2/templates/{templateName}

### 2.36.1. Parameters

Type	Name	Schema
Path	<b>templateName</b> <i>required</i>	string

### 2.36.2. Responses

HTTP Code	Description	Schema
200	Output type	<a href="#">LoopTemplate</a>

### 2.36.3. Produces

- `application/json`



## 2.37. GET

### /v2/templates/{templateName}/svgRepresentation

#### 2.37.1. Parameters

Type	Name	Schema
Path	<b>templateName</b> <i>required</i>	string

#### 2.37.2. Responses

HTTP Code	Description	Schema
200	Output type	string

#### 2.37.3. Produces

- `application/xml`

# Chapter 3. Definitions

## 3.1. ClampInformation

Name	Schema
<b>allPermissions</b> <i>optional</i>	< string > array
<b>cldsVersion</b> <i>optional</i>	string
<b>userName</b> <i>optional</i>	string

## 3.2. CldsHealthCheck

Name	Schema
<b>description</b> <i>optional</i>	string
<b>healthCheckComponent</b> <i>optional</i>	string
<b>healthCheckStatus</b> <i>optional</i>	string

## 3.3. Dictionary

Name	Schema
<b>createdBy</b> <i>optional</i>	string
<b>createdDate</b> <i>optional</i>	integer (int64)
<b>dictionaryElements</b> <i>optional</i>	< <a href="#">DictionaryElement</a> > array
<b>name</b> <i>optional</i>	string
<b>secondLevelDictionary</b> <i>optional</i>	integer (int32)

Name	Schema
<b>subDictionaryType</b> <i>optional</i>	string
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)

### 3.4. DictionaryElement

Name	Schema
<b>createdBy</b> <i>optional</i>	string
<b>createdAt</b> <i>optional</i>	integer (int64)
<b>description</b> <i>optional</i>	string
<b>name</b> <i>optional</i>	string
<b>shortName</b> <i>optional</i>	string
<b>subDictionary</b> <i>optional</i>	string
<b>type</b> <i>optional</i>	string
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)
<b>usedByDictionaries</b> <i>optional</i>	< <a href="#">Dictionary</a> > array

### 3.5. ExternalComponent

Name	Schema
<b>componentName</b> <i>optional</i>	string

Name	Schema
<b>state</b> <i>optional</i>	<a href="#">ExternalComponentState</a>

### 3.6. ExternalComponentState

Name	Schema
<b>description</b> <i>optional</i>	string
<b>level</b> <i>optional</i>	integer (int32)
<b>stateName</b> <i>optional</i>	string

### 3.7. JsonArray

Name	Schema
<b>asBigDecimal</b> <i>optional</i>	number
<b>asBigInteger</b> <i>optional</i>	integer
<b>asBoolean</b> <i>optional</i>	boolean
<b>asByte</b> <i>optional</i>	string (byte)
<b>asCharacter</b> <i>optional</i>	string
<b>asDouble</b> <i>optional</i>	number (double)
<b>asFloat</b> <i>optional</i>	number (float)
<b>asInt</b> <i>optional</i>	integer (int32)
<b>asJSONArray</b> <i>optional</i>	<a href="#">JsonArray</a>
<b>asJsonNull</b> <i>optional</i>	<a href="#">JsonNull</a>

Name	Schema
<b>asJsonObject</b> <i>optional</i>	JsonObject
<b>asJsonPrimitive</b> <i>optional</i>	JsonPrimitive
<b>asLong</b> <i>optional</i>	integer (int64)
<b>asNumber</b> <i>optional</i>	Number
<b>asShort</b> <i>optional</i>	integer (int32)
<b>asString</b> <i>optional</i>	string
<b>jsonArray</b> <i>optional</i>	boolean
<b>jsonNull</b> <i>optional</i>	boolean
<b>jsonObject</b> <i>optional</i>	boolean
<b>jsonPrimitive</b> <i>optional</i>	boolean

### 3.8. JsonNull

Name	Schema
<b>asBigDecimal</b> <i>optional</i>	number
<b>asBigInteger</b> <i>optional</i>	integer
<b>asBoolean</b> <i>optional</i>	boolean
<b>asByte</b> <i>optional</i>	string (byte)
<b>asCharacter</b> <i>optional</i>	string
<b>asDouble</b> <i>optional</i>	number (double)

Name	Schema
<b>asFloat</b> <i>optional</i>	number (float)
<b>asInt</b> <i>optional</i>	integer (int32)
<b>asJSONArray</b> <i>optional</i>	<a href="#">JSONArray</a>
<b>asJsonNull</b> <i>optional</i>	<a href="#">JsonNull</a>
<b>asJsonObject</b> <i>optional</i>	<a href="#">JsonObject</a>
<b>asJsonPrimitive</b> <i>optional</i>	<a href="#">JsonPrimitive</a>
<b>asLong</b> <i>optional</i>	integer (int64)
<b>asNumber</b> <i>optional</i>	<a href="#">Number</a>
<b>asShort</b> <i>optional</i>	integer (int32)
<b>asString</b> <i>optional</i>	string
<b>jsonArray</b> <i>optional</i>	boolean
<b>jsonNull</b> <i>optional</i>	boolean
<b>jsonObject</b> <i>optional</i>	boolean
<b>jsonPrimitive</b> <i>optional</i>	boolean

### 3.9. JsonObject

Name	Schema
<b>asBigDecimal</b> <i>optional</i>	number
<b>asBigInteger</b> <i>optional</i>	integer

<b>Name</b>	<b>Schema</b>
<b>asBoolean</b> <i>optional</i>	boolean
<b>asByte</b> <i>optional</i>	string (byte)
<b>asCharacter</b> <i>optional</i>	string
<b>asDouble</b> <i>optional</i>	number (double)
<b>asFloat</b> <i>optional</i>	number (float)
<b>asInt</b> <i>optional</i>	integer (int32)
<b>asJSONArray</b> <i>optional</i>	<a href="#">JSONArray</a>
<b>asJsonNull</b> <i>optional</i>	<a href="#">JsonNull</a>
<b>asJsonObject</b> <i>optional</i>	<a href="#">JsonObject</a>
<b>asJsonPrimitive</b> <i>optional</i>	<a href="#">JsonPrimitive</a>
<b>asLong</b> <i>optional</i>	integer (int64)
<b>asNumber</b> <i>optional</i>	<a href="#">Number</a>
<b>asShort</b> <i>optional</i>	integer (int32)
<b>asString</b> <i>optional</i>	string
<b>jsonArray</b> <i>optional</i>	boolean
<b>jsonNull</b> <i>optional</i>	boolean
<b>jsonObject</b> <i>optional</i>	boolean
<b>jsonPrimitive</b> <i>optional</i>	boolean

## 3.10. JsonPrimitive

Name	Schema
<b>asBigDecimal</b> <i>optional</i>	number
<b>asBigInteger</b> <i>optional</i>	integer
<b>asBoolean</b> <i>optional</i>	boolean
<b>asByte</b> <i>optional</i>	string (byte)
<b>asCharacter</b> <i>optional</i>	string
<b>asDouble</b> <i>optional</i>	number (double)
<b>asFloat</b> <i>optional</i>	number (float)
<b>asInt</b> <i>optional</i>	integer (int32)
<b>asJsonArray</b> <i>optional</i>	<a href="#">JsonArray</a>
<b>asJsonNull</b> <i>optional</i>	<a href="#">JsonNull</a>
<b>asJsonObject</b> <i>optional</i>	<a href="#">JsonObject</a>
<b>asJsonPrimitive</b> <i>optional</i>	<a href="#">JsonPrimitive</a>
<b>asLong</b> <i>optional</i>	integer (int64)
<b>asNumber</b> <i>optional</i>	<a href="#">Number</a>
<b>asShort</b> <i>optional</i>	integer (int32)
<b>asString</b> <i>optional</i>	string
<b>boolean</b> <i>optional</i>	boolean



Name	Schema
<b>jsonArray</b> <i>optional</i>	boolean
<b>jsonNull</b> <i>optional</i>	boolean
<b>jsonObject</b> <i>optional</i>	boolean
<b>jsonPrimitive</b> <i>optional</i>	boolean
<b>number</b> <i>optional</i>	boolean
<b>string</b> <i>optional</i>	boolean

### 3.11. Loop

Name	Schema
<b>components</b> <i>optional</i>	< string, <a href="#">ExternalComponent</a> > map
<b>createdBy</b> <i>optional</i>	string
<b>createdDate</b> <i>optional</i>	integer (int64)
<b>dcaeDeploymentId</b> <i>optional</i>	string
<b>dcaeDeploymentStatusUrl</b> <i>optional</i>	string
<b>globalPropertiesJson</b> <i>optional</i>	<a href="#">JsonObject</a>
<b>lastComputedState</b> <i>optional</i>	enum (DESIGN, SUBMITTED, DEPLOYED, RUNNING, STOPPED, IN_ERROR, WAITING)
<b>loopLogs</b> <i>optional</i>	< <a href="#">LoopLog</a> > array
<b>loopTemplate</b> <i>optional</i>	<a href="#">LoopTemplate</a>
<b>microServicePolicies</b> <i>optional</i>	< <a href="#">MicroServicePolicy</a> > array

Name	Schema
<b>modelService</b> <i>optional</i>	<a href="#">Service</a>
<b>name</b> <i>optional</i>	string
<b>operationalPolicies</b> <i>optional</i>	< <a href="#">OperationalPolicy</a> > array
<b>svgRepresentation</b> <i>optional</i>	string
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)

### 3.12. LoopElementModel

Name	Schema
<b>blueprint</b> <i>optional</i>	string
<b>createdBy</b> <i>optional</i>	string
<b>createdAt</b> <i>optional</i>	integer (int64)
<b>dcaeBlueprintId</b> <i>optional</i>	string
<b>loopElementType</b> <i>optional</i>	string
<b>name</b> <i>optional</i>	string
<b>policyModels</b> <i>optional</i>	< <a href="#">PolicyModel</a> > array
<b>shortName</b> <i>optional</i>	string
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)

Name	Schema
<b>usedByLoopTemplates</b> <i>optional</i>	< <a href="#">LoopTemplateLoopElementModel</a> > array

### 3.13. LoopLog

Name	Schema
<b>id</b> <i>optional</i>	integer (int64)
<b>logComponent</b> <i>optional</i>	string
<b>logInstant</b> <i>optional</i>	integer (int64)
<b>logType</b> <i>optional</i>	enum (INFO, WARNING, ERROR)
<b>loop</b> <i>optional</i>	<a href="#">Loop</a>
<b>message</b> <i>optional</i>	string

### 3.14. LoopTemplate

Name	Schema
<b>allowedLoopType</b> <i>optional</i>	enum (OPEN, CLOSED, HYBRID)
<b>blueprint</b> <i>optional</i>	string
<b>createdBy</b> <i>optional</i>	string
<b>createdDate</b> <i>optional</i>	integer (int64)
<b>dcaeBlueprintId</b> <i>optional</i>	string
<b>loopElementModelsUsed</b> <i>optional</i>	< <a href="#">LoopTemplateLoopElementModel</a> > array
<b>maximumInstancesAllowed</b> <i>optional</i>	integer (int32)

Name	Schema
<b>modelService</b> <i>optional</i>	<a href="#">Service</a>
<b>name</b> <i>optional</i>	string
<b>svgRepresentation</b> <i>optional</i>	string
<b>uniqueBlueprint</b> <i>optional</i>	boolean
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)

### 3.15. LoopTemplateLoopElementModel

Name	Schema
<b>flowOrder</b> <i>optional</i>	integer (int32)
<b>loopElementModel</b> <i>optional</i>	<a href="#">LoopElementModel</a>
<b>loopTemplate</b> <i>optional</i>	<a href="#">LoopTemplate</a>

### 3.16. MicroServicePolicy

Name	Schema
<b>configurationsJson</b> <i>optional</i>	<a href="#">JsonObject</a>
<b>context</b> <i>optional</i>	string
<b>createdBy</b> <i>optional</i>	string
<b>createdDate</b> <i>optional</i>	integer (int64)
<b>dcaeBlueprintId</b> <i>optional</i>	string

Name	Schema
<b>dcaeDeploymentId</b> <i>optional</i>	string
<b>dcaeDeploymentStatusUrl</b> <i>optional</i>	string
<b>deviceTypeScope</b> <i>optional</i>	string
<b>jsonRepresentation</b> <i>optional</i>	JsonObject
<b>loopElementModel</b> <i>optional</i>	LoopElementModel
<b>name</b> <i>optional</i>	string
<b>pdpGroup</b> <i>optional</i>	string
<b>pdpSubgroup</b> <i>optional</i>	string
<b>policyModel</b> <i>optional</i>	PolicyModel
<b>shared</b> <i>optional</i>	boolean
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)
<b>usedByLoops</b> <i>optional</i>	< Loop > array

## 3.17. Number

Type : object

## 3.18. OperationalPolicy

Name	Schema
<b>configurationsJson</b> <i>optional</i>	JsonObject

Name	Schema
<b>createdBy</b> <i>optional</i>	string
<b>createdDate</b> <i>optional</i>	integer (int64)
<b>jsonRepresentation</b> <i>optional</i>	<a href="#">JsonObject</a>
<b>legacy</b> <i>optional</i>	boolean
<b>loop</b> <i>optional</i>	<a href="#">Loop</a>
<b>loopElementModel</b> <i>optional</i>	<a href="#">LoopElementModel</a>
<b>name</b> <i>optional</i>	string
<b>pdpGroup</b> <i>optional</i>	string
<b>pdpSubgroup</b> <i>optional</i>	string
<b>policyModel</b> <i>optional</i>	<a href="#">PolicyModel</a>
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)

### 3.19. PolicyModel

Name	Schema
<b>createdBy</b> <i>optional</i>	string
<b>createdDate</b> <i>optional</i>	integer (int64)
<b>policyAcronym</b> <i>optional</i>	string
<b>policyModelTosca</b> <i>optional</i>	string

Name	Schema
<b>policyModelType</b> <i>optional</i>	string
<b>policyPdpGroup</b> <i>optional</i>	JsonObject
<b>updatedBy</b> <i>optional</i>	string
<b>updatedAt</b> <i>optional</i>	integer (int64)
<b>usedByElementModels</b> <i>optional</i>	< LoopElementModel > array
<b>version</b> <i>optional</i>	string

## 3.20. Service

Name	Schema
<b>name</b> <i>optional</i>	string
<b>resourceDetails</b> <i>optional</i>	JsonObject
<b>serviceDetails</b> <i>optional</i>	JsonObject
<b>serviceUuid</b> <i>optional</i>	string
<b>version</b> <i>optional</i>	string