

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	7
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	10
/v2/loop/removeOperationaPolicy/{loopName}/policyModel/{policyType}/{policyVersion}/{policyNa	

me}	
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	12
2.23. PUT /v2/loop/undeploy/{loopName}	12
2.23.1. Parameters	12
2.23.2. Responses	12
2.23.3. Produces	12
2.24. POST /v2/loop/updateGlobalProperties/{loopName}	12
2.24.1. Parameters	12
2.24.2. Responses	12
2.24.3. Consumes	12
2.24.4. Produces	13
2.25. POST /v2/loop/updateMicroservicePolicy/{loopName}	13
2.25.1. Parameters	13
2.25.2. Responses	13
2.25.3. Consumes	13
2.25.4. Produces	13
2.26. POST /v2/loop/updateOperationalPolicies/{loopName}	13
2.26.1. Parameters	13
2.26.2. Responses	14
2.26.3. Consumes	14
2.26.4. Produces	14
2.27. GET /v2/loop/{loopName}	14
2.27.1. Parameters	14
2.27.2. Responses	14

2.27.3. Produces	14
2.28. GET /v2/policies.....	14
2.28.1. Responses	14
2.28.2. Produces	15
2.29. PUT /v2/policies/pdpDeployment	15
2.29.1. Parameters	15
2.29.2. Responses	15
2.29.3. Consumes	15
2.30. POST /v2/policies/{policyModelType}/{policyModelVersion}/{policyName}/{policyVersion} ...	15
2.30.1. Parameters	15
2.30.2. Responses	16
2.30.3. Consumes	16
2.30.4. Produces	16
2.31. DELETE /v2/policies/{policyModelType}/{policyModelVersion}/{policyName}/{policyVersion}	16
2.31.1. Parameters	16
2.31.2. Responses	16
2.32. POST /v2/policyToscaModels	16
2.32.1. Parameters	17
2.32.2. Responses	17
2.32.3. Consumes	17
2.32.4. Produces	17
2.33. GET /v2/policyToscaModels	17
2.33.1. Responses	17
2.33.2. Produces	17
2.34. GET /v2/policyToscaModels/yaml/{policyModelType}/{policyModelVersion}	17
2.34.1. Parameters	17
2.34.2. Responses	18
2.34.3. Produces	18
2.35. GET /v2/policyToscaModels/{policyModelType}/{policyModelVersion}	18
2.35.1. Parameters	18
2.35.2. Responses	18
2.35.3. Produces	18
2.36. PUT /v2/policyToscaModels/{policyModelType}/{policyModelVersion}	19
2.36.1. Parameters	19
2.36.2. Responses	19
2.36.3. Consumes	19
2.36.4. Produces	19
2.37. GET /v2/templates.....	19

2.37.1. Responses	19
2.37.2. Produces	19
2.38. GET /v2/templates/names	20
2.38.1. Responses	20
2.38.2. Produces	20
2.39. GET /v2/templates/{templateName}	20
2.39.1. Parameters	20
2.39.2. Responses	20
2.39.3. Produces	20
3. Definitions	21
3.1. ClampInformation	21
3.2. CldsHealthCheck	21
3.3. Dictionary	21
3.4. DictionaryElement	22
3.5. ExternalComponent	22
3.6. ExternalComponentState	23
3.7. JsonArray	23
3.8. JsonElement	24
3.9. JsonNull	25
3.10. JsonObject	27
3.11. JsonPrimitive	28
3.12. Loop	29
3.13. LoopElementModel	30
3.14. LoopLog	31
3.15. LoopTemplate	31
3.16. LoopTemplateLoopElementModel	32
3.17. MicroServicePolicy	32
3.18. Number	33
3.19. OperationalPolicy	34
3.20. PolicyModel	34
3.21. Service	35

Chapter 1. Overview

1.1. Version information

Version : 6.1.0-SNAPSHOT

1.2. URI scheme

Host : localhost:40069

BasePath : /restservices/clds/

Schemes : HTTP

Chapter 2. Paths

2.1. GET /v1/healthcheck

2.1.1. Responses

HTTP Code	Description	Schema
200	Output type	CldsHealthCheck

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	ClampInformation

2.3.2. Produces

- `application/json`

2.4. GET /v2/dictionary

2.4.1. Responses

HTTP Code	Description	Schema
200	Output type	Dictionary

2.4.2. Produces

- [application/json](#)

2.5. PUT /v2/dictionary

2.5.1. Parameters

Type	Name	Schema
Body	body <i>required</i>	Dictionary

2.5.2. Responses

HTTP Code	Description	Schema
200	Output type	Dictionary

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	dictionaryName <i>required</i>	string

2.7.2. Responses

HTTP Code	Description	Schema
200	Output type	Dictionary

2.7.3. Produces

- `application/json`

2.8. PUT /v2/dictionary/{name}

2.8.1. Parameters

Type	Name	Schema
Path	name <i>required</i>	string
Body	body <i>required</i>	Dictionary

2.8.2. Responses

HTTP Code	Description	Schema
200	Output type	Dictionary

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	name <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	name <i>required</i>	string
Path	shortName <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	loopName <i>required</i>	string
Path	policyType <i>required</i>	string
Path	policyVersion <i>required</i>	string

2.11.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.11.3. Produces

- [application/json](#)

2.12. POST

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

2.12.1. Parameters

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

2.12.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

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	loopName <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	loopName <i>required</i>	string

2.14.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

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	loopName <i>required</i>	string

2.16.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.16.3. Produces

- `application/json`

2.17. PUT

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

2.17.1. Parameters

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

2.17.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.17.3. Produces

- `application/json`

2.18. PUT

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

2.18.1. Parameters

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

2.18.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.18.3. Produces

- `application/json`

2.19. PUT

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

2.19.1. Parameters

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

2.19.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.19.3. Produces

- [application/json](#)

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

2.20.1. Parameters

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

2.20.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.20.3. Produces

- [application/json](#)

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

2.21.1. Parameters

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

2.21.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.21.3. Produces

- [application/json](#)

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

2.22.1. Parameters

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

2.22.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.22.3. Produces

- `application/json`

2.23. PUT /v2/loop/undeploy/{loopName}

2.23.1. Parameters

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

2.23.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.23.3. Produces

- `application/json`

2.24. POST /v2/loop/updateGlobalProperties/{loopName}

2.24.1. Parameters

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

2.24.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.24.3. Consumes

- `application/json`

2.24.4. Produces

- `application/json`

2.25. POST

`/v2/loop/updateMicroservicePolicy/{loopName}`

2.25.1. Parameters

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

2.25.2. Responses

HTTP Code	Description	Schema
200	Output type	MicroServicePolicy

2.25.3. Consumes

- `application/json`

2.25.4. Produces

- `application/json`

2.26. POST

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

2.26.1. Parameters

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

2.26.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.26.3. Consumes

- `application/json`

2.26.4. Produces

- `application/json`

2.27. GET /v2/loop/{loopName}

2.27.1. Parameters

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

2.27.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.27.3. Produces

- `application/json`

2.28. GET /v2/policies

2.28.1. Responses

HTTP Code	Description	Schema
200	Output type	JsonObject

2.28.2. Produces

- `application/json`

2.29. PUT /v2/policies/pdpDeployment

2.29.1. Parameters

Type	Name	Schema
Body	body <i>required</i>	JsonElement

2.29.2. Responses

HTTP Code	Schema
200	No Content

2.29.3. Consumes

- `application/json`

2.30. POST

/v2/policies/{policyModelType}/{policyModelVersion}/{policyName}/{policyVersion}

2.30.1. Parameters

Type	Name	Schema
Path	policyModelType <i>required</i>	string
Path	policyModelVersion <i>required</i>	string
Path	policyName <i>required</i>	string
Path	policyVersion <i>required</i>	string
Body	body <i>required</i>	JsonElement

2.30.2. Responses

HTTP Code	Description	Schema
200	Output type	JsonObject

2.30.3. Consumes

- `application/json`

2.30.4. Produces

- `application/json`

2.31. DELETE

`/v2/policies/{policyModelType}/{policyModelVersion}/{policyName}/{policyVersion}`

2.31.1. Parameters

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

2.31.2. Responses

HTTP Code	Schema
200	No Content

2.32. POST `/v2/policyToscaModels`

2.32.1. Parameters

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

2.32.2. Responses

HTTP Code	Description	Schema
200	Output type	PolicyModel

2.32.3. Consumes

- `plain/text`

2.32.4. Produces

- `application/json`

2.33. GET /v2/policyToscaModels

2.33.1. Responses

HTTP Code	Description	Schema
200	Output type	PolicyModel

2.33.2. Produces

- `application/json`

2.34. GET

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

2.34.1. Parameters

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

2.34.2. Responses

HTTP Code	Description	Schema
200	Output type	string

2.34.3. Produces

- `application/json`

2.35. GET

`/v2/policyToscaModels/{policyModelType}/{policyModelVersion}`

2.35.1. Parameters

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

2.35.2. Responses

HTTP Code	Description	Schema
200	Output type	JsonObject

2.35.3. Produces

- `application/json`

2.36. PUT

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

2.36.1. Parameters

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

2.36.2. Responses

HTTP Code	Description	Schema
200	Output type	PolicyModel

2.36.3. Consumes

- [plain/text](#)

2.36.4. Produces

- [application/json](#)

2.37. GET /v2/templates

2.37.1. Responses

HTTP Code	Description	Schema
200	Output type	LoopTemplate

2.37.2. Produces

- [application/json](#)

2.38. GET /v2/templates/names

2.38.1. Responses

HTTP Code	Description	Schema
200	Output type	< string > array

2.38.2. Produces

- `application/json`

2.39. GET /v2/templates/{templateName}

2.39.1. Parameters

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

2.39.2. Responses

HTTP Code	Description	Schema
200	Output type	LoopTemplate

2.39.3. Produces

- `application/json`

Chapter 3. Definitions

3.1. ClampInformation

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

3.2. CldsHealthCheck

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

3.3. Dictionary

Name	Schema
createdBy <i>optional</i>	string
createdDate <i>optional</i>	integer (int64)
dictionaryElements <i>optional</i>	< DictionaryElement > array
name <i>optional</i>	string
secondLevelDictionary <i>optional</i>	integer (int32)

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

3.4. DictionaryElement

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

3.5. ExternalComponent

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

Name	Schema
state <i>optional</i>	ExternalComponentState

3.6. ExternalComponentState

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

3.7. JsonArray

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

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

3.8. JsonElement

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

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

3.9. JsonNull

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

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

3.10. JsonObject

Name	Schema
asBigDecimal <i>optional</i>	number
asBigInteger <i>optional</i>	integer
asBoolean <i>optional</i>	boolean
asByte <i>optional</i>	string (byte)
asCharacter <i>optional</i>	string
asDouble <i>optional</i>	number (double)
asFloat <i>optional</i>	number (float)
asInt <i>optional</i>	integer (int32)
asJSONArray <i>optional</i>	JSONArray
asJsonNull <i>optional</i>	JsonNull
asJsonObject <i>optional</i>	JsonObject
asJsonPrimitive <i>optional</i>	JsonPrimitive
asLong <i>optional</i>	integer (int64)
asNumber <i>optional</i>	Number
asShort <i>optional</i>	integer (int32)
asString <i>optional</i>	string
jsonArray <i>optional</i>	boolean

Name	Schema
jsonNull <i>optional</i>	boolean
jsonObject <i>optional</i>	boolean
jsonPrimitive <i>optional</i>	boolean

3.11. JsonPrimitive

Name	Schema
asBigDecimal <i>optional</i>	number
asBigInteger <i>optional</i>	integer
asBoolean <i>optional</i>	boolean
asByte <i>optional</i>	string (byte)
asCharacter <i>optional</i>	string
asDouble <i>optional</i>	number (double)
asFloat <i>optional</i>	number (float)
asInt <i>optional</i>	integer (int32)
asJsonArray <i>optional</i>	JsonArray
asJsonNull <i>optional</i>	JsonNull
asJsonObject <i>optional</i>	JsonObject
asJsonPrimitive <i>optional</i>	JsonPrimitive
asLong <i>optional</i>	integer (int64)

Name	Schema
asNumber <i>optional</i>	Number
asShort <i>optional</i>	integer (int32)
asString <i>optional</i>	string
boolean <i>optional</i>	boolean
jsonArray <i>optional</i>	boolean
jsonNull <i>optional</i>	boolean
jsonObject <i>optional</i>	boolean
jsonPrimitive <i>optional</i>	boolean
number <i>optional</i>	boolean
string <i>optional</i>	boolean

3.12. Loop

Name	Schema
components <i>optional</i>	< string, ExternalComponent > map
createdBy <i>optional</i>	string
createdDate <i>optional</i>	integer (int64)
dcaeDeploymentId <i>optional</i>	string
dcaeDeploymentStatusUrl <i>optional</i>	string
globalPropertiesJson <i>optional</i>	JsonObject

Name	Schema
lastComputedState <i>optional</i>	enum (DESIGN, SUBMITTED, DEPLOYED, RUNNING, STOPPED, IN_ERROR, WAITING)
loopLogs <i>optional</i>	< LoopLog > array
loopTemplate <i>optional</i>	LoopTemplate
microServicePolicies <i>optional</i>	< MicroServicePolicy > array
modelService <i>optional</i>	Service
name <i>optional</i>	string
operationalPolicies <i>optional</i>	< OperationalPolicy > array
updatedBy <i>optional</i>	string
updatedAt <i>optional</i>	integer (int64)

3.13. LoopElementModel

Name	Schema
blueprint <i>optional</i>	string
createdBy <i>optional</i>	string
createdDate <i>optional</i>	integer (int64)
dcaeBlueprintId <i>optional</i>	string
loopElementType <i>optional</i>	string
name <i>optional</i>	string
policyModels <i>optional</i>	< PolicyModel > array

Name	Schema
shortName <i>optional</i>	string
updatedBy <i>optional</i>	string
updatedAt <i>optional</i>	integer (int64)
usedByLoopTemplates <i>optional</i>	< LoopTemplateLoopElementModel > array

3.14. LoopLog

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

3.15. LoopTemplate

Name	Schema
allowedLoopType <i>optional</i>	enum (OPEN, CLOSED, HYBRID)
blueprint <i>optional</i>	string
createdBy <i>optional</i>	string
createdDate <i>optional</i>	integer (int64)

Name	Schema
dcaeBlueprintId <i>optional</i>	string
loopElementModelsUsed <i>optional</i>	< LoopTemplateLoopElementModel > array
maximumInstancesAllowed <i>optional</i>	integer (int32)
modelService <i>optional</i>	Service
name <i>optional</i>	string
uniqueBlueprint <i>optional</i>	boolean
updatedBy <i>optional</i>	string
updatedAt <i>optional</i>	integer (int64)

3.16. LoopTemplateLoopElementModel

Name	Schema
flowOrder <i>optional</i>	integer (int32)
loopElementModel <i>optional</i>	LoopElementModel
loopTemplate <i>optional</i>	LoopTemplate

3.17. MicroServicePolicy

Name	Schema
configurationsJson <i>optional</i>	JsonObject
context <i>optional</i>	string
createdBy <i>optional</i>	string

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

3.18. Number

Type : object

3.19. OperationalPolicy

Name	Schema
configurationsJson <i>optional</i>	JsonObject
createdBy <i>optional</i>	string
createdDate <i>optional</i>	integer (int64)
jsonRepresentation <i>optional</i>	JsonObject
loop <i>optional</i>	Loop
loopElementModel <i>optional</i>	LoopElementModel
name <i>optional</i>	string
pdpGroup <i>optional</i>	string
pdpSubgroup <i>optional</i>	string
policyModel <i>optional</i>	PolicyModel
updatedBy <i>optional</i>	string
updatedAt <i>optional</i>	integer (int64)

3.20. PolicyModel

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

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

3.21. Service

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