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 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. 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} . . . 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/updateMicroservicePolicy/{loopName} . . . 13
2.26.1. Parameters . . . 13
2.26.2. Responses . . . 13
2.26.3. Consumes . . . 14
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 . . . 15
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 . . . 16
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 . . . 17
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 : 5.1.0-SNAPSHOT

1.2. URI scheme

Host : localhost:37033
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** *required*	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** *required*	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** *required*	string
Body	**body** *required*	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** *required*	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** *required*	string
Path	**shortName** *required*	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** *required*	string
Path	**policyType** *required*	string
Path	**policyVersion** *required*	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}?templateName={templateName** *required*	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** *required*	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** *required*	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** *required*	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** *required*	string
Path	**microServicePolicyName** *required*	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** *required*	string
Path	**operationalPolicyName** *required*	string

2.18.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.18.3. Produces

- `application/json`

2.19. PUT /v2/loop/removeOperationaPolicy/{loopName}/policyModel/{policyType}/{policyVersion}/{policyName}

2.19.1. Parameters

Type	Name	Schema
Path	**loopName** *required*	string
Path	**policyName** *required*	string
Path	**policyType** *required*	string
Path	**policyVersion** *required*	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** *required*	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** *required*	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** *required*	string

2.22.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.22.3. Produces

- `application/json`

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

2.23.1. Parameters

Type	Name	Schema
Path	**loopName** *required*	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	**loopName** *required*	string

2.24.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.24.3. Produces

- `application/json`

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

2.25.1. Parameters

Type	Name	Schema
Path	**loopName** *required*	string
Body	**body** *required*	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	**loopName** *required*	string
Body	**body** *required*	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	**loopName** *required*	string
Body	**body** *required*	JsonArray

2.27.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

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	**loopName** *required*	string

2.28.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.28.3. Produces

- `application/json`

2.29. POST /v2/policyToscaModels

2.29.1. Parameters

Type	Name	Schema
Body	**body** *required*	string

2.29.2. Responses

HTTP Code	Description	Schema
200	Output type	PolicyModel

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	PolicyModel

2.30.2. Produces

- `application/json`

2.31. GET /v2/policyToscaModels/yaml/{policyModelType}/{policyModelVersion}

2.31.1. Parameters

Type	Name	Schema
Path	**policyModelType** *required*	string
Path	**policyModelVersion** *required*	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	**policyModelType** *required*	string
Path	**policyModelVersion** *required*	string

2.32.2. Responses

HTTP Code	Description	Schema
200	Output type	PolicyModel

2.32.3. Produces

- `application/json`

2.33. PUT /v2/policyToscaModels/{policyModelType}/{policyModelVersion}

2.33.1. Parameters

Type	Name	Schema
Path	**policyModelType** *required*	string
Path	**policyModelVersion** *required*	string
Body	**body** *required*	string

2.33.2. Responses

HTTP Code	Description	Schema
200	Output type	PolicyModel

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	LoopTemplate

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	**templateName** *required*	string

2.36.2. Responses

HTTP Code	Description	Schema
200	Output type	LoopTemplate

2.36.3. Produces

- `application/json`

2.37. GET /v2/templates/{templateName}/svgRepresentation

2.37.1. Parameters

Type	Name	Schema
Path	**templateName** *required*	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
allPermissions *optional*	< string > array
cldsVersion *optional*	string
userName *optional*	string

3.2. CldsHealthCheck

Name	Schema
description *optional*	string
healthCheckComponent *optional*	string
healthCheckStatus *optional*	string

3.3. Dictionary

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

Name	Schema
subDictionaryType *optional*	string
updatedBy *optional*	string
updatedDate *optional*	integer (int64)

3.4. DictionaryElement

Name	Schema
createdBy *optional*	string
createdDate *optional*	integer (int64)
description *optional*	string
name *optional*	string
shortName *optional*	string
subDictionary *optional*	string
type *optional*	string
updatedBy *optional*	string
updatedDate *optional*	integer (int64)
usedByDictionaries *optional*	< Dictionary > array

3.5. ExternalComponent

Name	Schema
componentName *optional*	string

Name	Schema
state *optional*	ExternalComponentState

3.6. ExternalComponentState

Name	Schema
description *optional*	string
level *optional*	integer (int32)
stateName *optional*	string

3.7. JsonArray

Name	Schema
asBigDecimal *optional*	number
asBigInteger *optional*	integer
asBoolean *optional*	boolean
asByte *optional*	string (byte)
asCharacter *optional*	string
asDouble *optional*	number (double)
asFloat *optional*	number (float)
asInt *optional*	integer (int32)
asJsonArray *optional*	JsonArray
asJsonNull *optional*	JsonNull

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

3.8. JsonNull

Name	Schema
asBigDecimal *optional*	number
asBigInteger *optional*	integer
asBoolean *optional*	boolean
asByte *optional*	string (byte)
asCharacter *optional*	string
asDouble *optional*	number (double)

Name	Schema
asFloat *optional*	number (float)
asInt *optional*	integer (int32)
asJsonArray *optional*	JsonArray
asJsonNull *optional*	JsonNull
asJsonObject *optional*	JsonObject
asJsonPrimitive *optional*	JsonPrimitive
asLong *optional*	integer (int64)
asNumber *optional*	Number
asShort *optional*	integer (int32)
asString *optional*	string
jsonArray *optional*	boolean
jsonNull *optional*	boolean
jsonObject *optional*	boolean
jsonPrimitive *optional*	boolean

3.9. JsonObject

Name	Schema
asBigDecimal *optional*	number
asBigInteger *optional*	integer

Name	Schema
asBoolean *optional*	boolean
asByte *optional*	string (byte)
asCharacter *optional*	string
asDouble *optional*	number (double)
asFloat *optional*	number (float)
asInt *optional*	integer (int32)
asJsonArray *optional*	JsonArray
asJsonNull *optional*	JsonNull
asJsonObject *optional*	JsonObject
asJsonPrimitive *optional*	JsonPrimitive
asLong *optional*	integer (int64)
asNumber *optional*	Number
asShort *optional*	integer (int32)
asString *optional*	string
jsonArray *optional*	boolean
jsonNull *optional*	boolean
jsonObject *optional*	boolean
jsonPrimitive *optional*	boolean

3.10. JsonPrimitive

Name	Schema
asBigDecimal *optional*	number
asBigInteger *optional*	integer
asBoolean *optional*	boolean
asByte *optional*	string (byte)
asCharacter *optional*	string
asDouble *optional*	number (double)
asFloat *optional*	number (float)
asInt *optional*	integer (int32)
asJsonArray *optional*	JsonArray
asJsonNull *optional*	JsonNull
asJsonObject *optional*	JsonObject
asJsonPrimitive *optional*	JsonPrimitive
asLong *optional*	integer (int64)
asNumber *optional*	Number
asShort *optional*	integer (int32)
asString *optional*	string
boolean *optional*	boolean

Name	Schema
jsonArray *optional*	boolean
jsonNull *optional*	boolean
jsonObject *optional*	boolean
jsonPrimitive *optional*	boolean
number *optional*	boolean
string *optional*	boolean

3.11. Loop

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

Name	Schema
modelService *optional*	Service
name *optional*	string
operationalPolicies *optional*	< OperationalPolicy > array
svgRepresentation *optional*	string
updatedBy *optional*	string
updatedDate *optional*	integer (int64)

3.12. LoopElementModel

Name	Schema
blueprint *optional*	string
createdBy *optional*	string
createdDate *optional*	integer (int64)
dcaeBlueprintId *optional*	string
loopElementType *optional*	string
name *optional*	string
policyModels *optional*	< PolicyModel > array
shortName *optional*	string
updatedBy *optional*	string
updatedDate *optional*	integer (int64)

Name	Schema
usedByLoopTemplates *optional*	< LoopTemplateLoopElementModel > array

3.13. LoopLog

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

3.14. LoopTemplate

Name	Schema
allowedLoopType *optional*	enum (OPEN, CLOSED, HYBRID)
blueprint *optional*	string
createdBy *optional*	string
createdDate *optional*	integer (int64)
dcaeBlueprintId *optional*	string
loopElementModelsUsed *optional*	< LoopTemplateLoopElementModel > array
maximumInstancesAllowed *optional*	integer (int32)

Name	Schema
modelService *optional*	Service
name *optional*	string
svgRepresentation *optional*	string
uniqueBlueprint *optional*	boolean
updatedBy *optional*	string
updatedDate *optional*	integer (int64)

3.15. LoopTemplateLoopElementModel

Name	Schema
flowOrder *optional*	integer (int32)
loopElementModel *optional*	LoopElementModel
loopTemplate *optional*	LoopTemplate

3.16. MicroServicePolicy

Name	Schema
configurationsJson *optional*	JsonObject
context *optional*	string
createdBy *optional*	string
createdDate *optional*	integer (int64)
dcaeBlueprintId *optional*	string

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

3.17. Number

Type : object

3.18. OperationalPolicy

Name	Schema
configurationsJson *optional*	JsonObject
createdBy *optional*	string
createdDate *optional*	integer (int64)
jsonRepresentation *optional*	JsonObject
legacy *optional*	boolean
loop *optional*	Loop
loopElementModel *optional*	LoopElementModel
name *optional*	string
pdpGroup *optional*	string
pdpSubgroup *optional*	string
policyModel *optional*	PolicyModel
updatedBy *optional*	string
updatedDate *optional*	integer (int64)

3.19. PolicyModel

Name	Schema
createdBy *optional*	string
createdDate *optional*	integer (int64)
policyAcronym *optional*	string

Name	Schema
policyModelTosca *optional*	string
policyModelType *optional*	string
policyPdpGroup *optional*	JsonObject
updatedBy *optional*	string
updatedDate *optional*	integer (int64)
usedByElementModels *optional*	< LoopElementModel > array
version *optional*	string

3.20. Service

Name	Schema
name *optional*	string
resourceDetails *optional*	JsonObject
serviceDetails *optional*	JsonObject
serviceUuid *optional*	string
version *optional*	string