

Clamp Rest API

Table of Contents

- 1. Overview 1
 - 1.1. Version information 1
 - 1.2. URI scheme..... 1
- 2. Paths 2
 - 2.1. PUT /v1/clds/action/{action}/{modelName}?test={test} 2
 - 2.1.1. Parameters 2
 - 2.1.2. Responses 2
 - 2.1.3. Consumes 2
 - 2.1.4. Produces 2
 - 2.2. GET /v1/clds/cldsDetails..... 2
 - 2.2.1. Responses 2
 - 2.2.2. Produces 2
 - 2.3. GET /v1/clds/cldsInfo 3
 - 2.3.1. Responses 3
 - 2.3.2. Produces 3
 - 2.4. POST /v1/clds/dcae/event?test={test} 3
 - 2.4.1. Parameters 3
 - 2.4.2. Responses 3
 - 2.4.3. Consumes 3
 - 2.4.4. Produces 3
 - 2.5. PUT /v1/clds/deploy/{modelName} 3
 - 2.5.1. Parameters 3
 - 2.5.2. Responses 4
 - 2.5.3. Consumes 4
 - 2.5.4. Produces 4
 - 2.6. GET /v1/clds/model-names 4
 - 2.6.1. Responses 4
 - 2.6.2. Produces 4
 - 2.7. GET /v1/clds/model/bpmn/{modelName} 4
 - 2.7.1. Parameters 4
 - 2.7.2. Responses 4
 - 2.7.3. Produces 5
 - 2.8. GET /v1/clds/model/image/{modelName} 5
 - 2.8.1. Parameters 5
 - 2.8.2. Responses 5
 - 2.8.3. Produces 5

2.9. GET /v1/clds/model/{modelName}	5
2.9.1. Parameters	5
2.9.2. Responses	5
2.9.3. Produces	5
2.10. PUT /v1/clds/model/{modelName}	5
2.10.1. Parameters	6
2.10.2. Responses	6
2.10.3. Consumes	6
2.10.4. Produces	6
2.11. GET /v1/clds/properties	6
2.11.1. Responses	6
2.11.2. Produces	6
2.12. PUT /v1/clds/undeploy/{modelName}	6
2.12.1. Parameters	6
2.12.2. Responses	7
2.12.3. Consumes	7
2.12.4. Produces	7
2.13. GET /v1/cldsTempate/template-names	7
2.13.1. Responses	7
2.13.2. Produces	7
2.14. GET /v1/cldsTempate/template/bpmn/{templateName}	7
2.14.1. Parameters	7
2.14.2. Responses	7
2.14.3. Produces	8
2.15. GET /v1/cldsTempate/template/image/{templateName}	8
2.15.1. Parameters	8
2.15.2. Responses	8
2.15.3. Produces	8
2.16. GET /v1/cldsTempate/template/{templateName}	8
2.16.1. Parameters	8
2.16.2. Responses	8
2.16.3. Produces	9
2.17. PUT /v1/cldsTempate/template/{templateName}	9
2.17.1. Parameters	9
2.17.2. Responses	9
2.17.3. Consumes	9
2.17.4. Produces	9
2.18. GET /v1/dictionary	9

2.18.1. Responses	9
2.18.2. Produces	9
2.19. PUT /v1/dictionary/{dictionaryName}.....	10
2.19.1. Parameters	10
2.19.2. Responses	10
2.19.3. Consumes	10
2.19.4. Produces	10
2.20. GET /v1/dictionary/{dictionaryName}/items	10
2.20.1. Parameters	10
2.20.2. Responses	10
2.20.3. Produces	10
2.21. PUT /v1/dictionary/{dictionaryName}/items	11
2.21.1. Parameters	11
2.21.2. Responses	11
2.21.3. Consumes	11
2.21.4. Produces	11
2.22. GET /v1/healthcheck	11
2.22.1. Responses	11
2.22.2. Produces	11
2.23. GET /v1/tosca/models.....	11
2.23.1. Responses	11
2.23.2. Produces	12
2.24. GET /v1/tosca/models/policyType/{policyType}	12
2.24.1. Parameters	12
2.24.2. Responses	12
2.24.3. Produces	12
2.25. GET /v1/tosca/models/{toscaModelName}	12
2.25.1. Parameters	12
2.25.2. Responses	12
2.25.3. Produces	13
2.26. PUT /v1/tosca/models/{toscaModelName}	13
2.26.1. Parameters	13
2.26.2. Responses	13
2.26.3. Consumes	13
2.26.4. Produces	13
2.27. GET /v1/user/getUser	13
2.27.1. Responses	13
2.27.2. Produces	13

2.28. PUT /v2/loop/delete/{loopName}	13
2.28.1. Parameters	14
2.28.2. Responses	14
2.29. PUT /v2/loop/deploy/{loopName}	14
2.29.1. Parameters	14
2.29.2. Responses	14
2.29.3. Produces	14
2.30. GET /v2/loop/getAllNames	14
2.30.1. Responses	14
2.30.2. Produces	14
2.31. GET /v2/loop/getstatus/{loopName}	15
2.31.1. Parameters	15
2.31.2. Responses	15
2.31.3. Produces	15
2.32. PUT /v2/loop/restart/{loopName}	15
2.32.1. Parameters	15
2.32.2. Responses	15
2.32.3. Produces	15
2.33. PUT /v2/loop/stop/{loopName}	15
2.33.1. Parameters	15
2.33.2. Responses	16
2.33.3. Produces	16
2.34. PUT /v2/loop/submit/{loopName}	16
2.34.1. Parameters	16
2.34.2. Responses	16
2.34.3. Produces	16
2.35. GET /v2/loop/svgRepresentation/{loopName}	16
2.35.1. Parameters	16
2.35.2. Responses	17
2.35.3. Produces	17
2.36. PUT /v2/loop/undeploy/{loopName}	17
2.36.1. Parameters	17
2.36.2. Responses	17
2.36.3. Produces	17
2.37. POST /v2/loop/updateGlobalProperties/{loopName}	17
2.37.1. Parameters	17
2.37.2. Responses	17
2.37.3. Consumes	18

- 2.37.4. Produces 18
- 2.38. POST /v2/loop/updateMicroservicePolicy/{loopName} 18
 - 2.38.1. Parameters 18
 - 2.38.2. Responses 18
 - 2.38.3. Consumes 18
 - 2.38.4. Produces 18
- 2.39. POST /v2/loop/updateOperationalPolicies/{loopName} 18
 - 2.39.1. Parameters 19
 - 2.39.2. Responses 19
 - 2.39.3. Consumes 19
 - 2.39.4. Produces 19
- 2.40. GET /v2/loop/{loopName} 19
 - 2.40.1. Parameters 19
 - 2.40.2. Responses 19
 - 2.40.3. Produces 19
- 3. Definitions 20
 - 3.1. CldsDictionary 20
 - 3.2. CldsDictionaryItem 20
 - 3.3. CldsEvent 21
 - 3.4. CldsInfo 21
 - 3.5. CldsModel 22
 - 3.6. CldsModelInstance 23
 - 3.7. CldsMonitoringDetails 23
 - 3.8. CldsTemplate 24
 - 3.9. CldsToscaModel 25
 - 3.10. DcaeEvent 25
 - 3.11. ExternalComponent 26
 - 3.12. ExternalComponentState 26
 - 3.13. JsonArray 26
 - 3.14. JsonNull 28
 - 3.15. JsonObject 29
 - 3.16. JsonPrimitive 30
 - 3.17. Loop 31
 - 3.18. LoopLog 32
 - 3.19. MicroServicePolicy 33
 - 3.20. Number 33
 - 3.21. OperationalPolicy 33
 - 3.22. ValueItem 34

Chapter 1. Overview

1.1. Version information

Version : 4.1.1-SNAPSHOT

1.2. URI scheme

Host : localhost:43819

BasePath : /restservices/clds/

Schemes : HTTP

Chapter 2. Paths

2.1. PUT

/v1/clds/action/{action}/{modelName}?test={test}

2.1.1. Parameters

Type	Name	Schema
Path	action <i>required</i>	string
Path	modelName}?test={test <i>required</i>	string
Body	body <i>required</i>	CldsModel

2.1.2. Responses

HTTP Code	Schema
200	No Content

2.1.3. Consumes

- [application/json](#)

2.1.4. Produces

- [application/json](#)

2.2. GET /v1/clds/cldsDetails

2.2.1. Responses

HTTP Code	Description	Schema
200	Output type	CldsMonitoringDetails

2.2.2. Produces

- [application/json](#)

2.3. GET /v1/clds/cldsInfo

2.3.1. Responses

HTTP Code	Description	Schema
200	Output type	CldsInfo

2.3.2. Produces

- [application/json](#)

2.4. POST /v1/clds/dcae/event?test={test}

2.4.1. Parameters

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

2.4.2. Responses

HTTP Code	Schema
200	No Content

2.4.3. Consumes

- [application/json](#)

2.4.4. Produces

- [application/json](#)

2.5. PUT /v1/clds/deploy/{modelName}

2.5.1. Parameters

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

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

2.5.2. Responses

HTTP Code	Schema
200	No Content

2.5.3. Consumes

- [application/json](#)

2.5.4. Produces

- [application/json](#)

2.6. GET /v1/clds/model-names

2.6.1. Responses

HTTP Code	Description	Schema
200	Output type	ValueItem

2.6.2. Produces

- [application/json](#)

2.7. GET /v1/clds/model/bpmn/{modelName}

2.7.1. Parameters

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

2.7.2. Responses

HTTP Code	Schema
200	No Content

2.7.3. Produces

- `text/xml`

2.8. GET /v1/clds/model/image/{modelName}

2.8.1. Parameters

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

2.8.2. Responses

HTTP Code	Schema
200	No Content

2.8.3. Produces

- `text/xml`

2.9. GET /v1/clds/model/{modelName}

2.9.1. Parameters

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

2.9.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsModel

2.9.3. Produces

- `application/json`

2.10. PUT /v1/clds/model/{modelName}

2.10.1. Parameters

Type	Name	Schema
Path	modelName <i>required</i>	string
Body	body <i>required</i>	CldsModel

2.10.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsModel

2.10.3. Consumes

- [application/json](#)

2.10.4. Produces

- [application/json](#)

2.11. GET /v1/clds/properties

2.11.1. Responses

HTTP Code	Schema
200	No Content

2.11.2. Produces

- [application/json](#)

2.12. PUT /v1/clds/undeploy/{modelName}

2.12.1. Parameters

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

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

2.12.2. Responses

HTTP Code	Schema
200	No Content

2.12.3. Consumes

- [application/json](#)

2.12.4. Produces

- [application/json](#)

2.13. GET /v1/cldsTempate/template-names

2.13.1. Responses

HTTP Code	Description	Schema
200	Output type	ValueItem

2.13.2. Produces

- [application/json](#)

2.14. GET

/v1/cldsTempate/template/bpmn/{templateName}

2.14.1. Parameters

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

2.14.2. Responses

HTTP Code	Schema
200	No Content

2.14.3. Produces

- `text/xml`

2.15. GET

`/v1/cldsTempate/template/image/{templateName}`

2.15.1. Parameters

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

2.15.2. Responses

HTTP Code	Schema
200	No Content

2.15.3. Produces

- `text/xml`

2.16. GET `/v1/cldsTempate/template/{templateName}`

2.16.1. Parameters

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

2.16.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsTemplate

2.16.3. Produces

- `application/json`

2.17. PUT /v1/cldsTempate/template/{templateName}

2.17.1. Parameters

Type	Name	Schema
Path	templateName <i>required</i>	string
Body	body <i>required</i>	CldsTemplate

2.17.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsTemplate

2.17.3. Consumes

- `application/json`

2.17.4. Produces

- `application/json`

2.18. GET /v1/dictionary

2.18.1. Responses

HTTP Code	Description	Schema
200	Output type	CldsDictionary

2.18.2. Produces

- `application/json`

2.19. PUT /v1/dictionary/{dictionaryName}

2.19.1. Parameters

Type	Name	Schema
Path	dictionaryName <i>required</i>	string
Body	body <i>required</i>	CldsDictionary

2.19.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsDictionary

2.19.3. Consumes

- [application/json](#)

2.19.4. Produces

- [application/json](#)

2.20. GET /v1/dictionary/{dictionaryName}/items

2.20.1. Parameters

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

2.20.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsDictionary

2.20.3. Produces

- [application/json](#)

2.21. PUT /v1/dictionary/{dictionaryName}/items

2.21.1. Parameters

Type	Name	Schema
Path	dictionaryName <i>required</i>	string
Body	body <i>required</i>	CldsDictionaryItem

2.21.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsDictionaryItem

2.21.3. Consumes

- [application/json](#)

2.21.4. Produces

- [application/json](#)

2.22. GET /v1/healthcheck

2.22.1. Responses

HTTP Code	Schema
200	No Content

2.22.2. Produces

- [application/json](#)

2.23. GET /v1/tosca/models

2.23.1. Responses

HTTP Code	Description	Schema
200	Output type	CldsToscaModel

2.23.2. Produces

- `application/json`

2.24. GET /v1/tosca/models/policyType/{policyType}

2.24.1. Parameters

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

2.24.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsToscaModel

2.24.3. Produces

- `application/json`

2.25. GET /v1/tosca/models/{toscaModelName}

2.25.1. Parameters

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

2.25.2. Responses

HTTP Code	Description	Schema
200	Output type	CldsToscaModel

2.25.3. Produces

- `application/json`

2.26. PUT /v1/tosca/models/{toscaModelName}

2.26.1. Parameters

Type	Name	Schema
Path	toscaModelName <i>required</i>	string
Body	body <i>required</i>	CldsToscaModel

2.26.2. Responses

HTTP Code	Schema
200	No Content

2.26.3. Consumes

- `application/json`

2.26.4. Produces

- `application/json`

2.27. GET /v1/user/getUser

2.27.1. Responses

HTTP Code	Schema
200	No Content

2.27.2. Produces

- `text/plain`

2.28. PUT /v2/loop/delete/{loopName}

2.28.1. Parameters

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

2.28.2. Responses

HTTP Code	Schema
200	No Content

2.29. PUT /v2/loop/deploy/{loopName}

2.29.1. Parameters

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

2.29.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.29.3. Produces

- `application/json`

2.30. GET /v2/loop/getAllNames

2.30.1. Responses

HTTP Code	Description	Schema
200	Output type	< string > array

2.30.2. Produces

- `application/json`

2.31. GET /v2/loop/getstatus/{loopName}

2.31.1. Parameters

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

2.31.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.31.3. Produces

- `application/json`

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

2.32.1. Parameters

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

2.32.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.32.3. Produces

- `application/json`

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

2.33.1. Parameters

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

2.33.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.33.3. Produces

- `application/json`

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

2.34.1. Parameters

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

2.34.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.34.3. Produces

- `application/json`

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

2.35.1. Parameters

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

2.35.2. Responses

HTTP Code	Description	Schema
200	Output type	string

2.35.3. Produces

- `application/xml`

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

2.36.1. Parameters

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

2.36.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.36.3. Produces

- `application/json`

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

2.37.1. Parameters

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

2.37.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.37.3. Consumes

- [application/json](#)

2.37.4. Produces

- [application/json](#)

2.38. POST

/v2/loop/updateMicroservicePolicy/{loopName}

2.38.1. Parameters

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

2.38.2. Responses

HTTP Code	Description	Schema
200	Output type	MicroServicePolicy

2.38.3. Consumes

- [application/json](#)

2.38.4. Produces

- [application/json](#)

2.39. POST

/v2/loop/updateOperationalPolicies/{loopName}

2.39.1. Parameters

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

2.39.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.39.3. Consumes

- [application/json](#)

2.39.4. Produces

- [application/json](#)

2.40. GET /v2/loop/{loopName}

2.40.1. Parameters

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

2.40.2. Responses

HTTP Code	Description	Schema
200	Output type	Loop

2.40.3. Produces

- [application/json](#)

Chapter 3. Definitions

3.1. CldsDictionary

Name	Schema
cldsDictionaryItems <i>optional</i>	< CldsDictionaryItem > array
createdBy <i>optional</i>	string
dictionaryId <i>optional</i>	string
dictionaryName <i>optional</i>	string
lastUpdatedDate <i>optional</i>	string
updatedBy <i>optional</i>	string

3.2. CldsDictionaryItem

Name	Schema
createdBy <i>optional</i>	string
dictElementDesc <i>optional</i>	string
dictElementId <i>optional</i>	string
dictElementName <i>optional</i>	string
dictElementShortName <i>optional</i>	string
dictElementType <i>optional</i>	string
dictionaryId <i>optional</i>	string

Name	Schema
lastUpdatedDate <i>optional</i>	string
updatedBy <i>optional</i>	string

3.3. CldsEvent

Name	Schema
actionCd <i>optional</i>	string
actionStateCd <i>optional</i>	string
id <i>optional</i>	string
processInstanceId <i>optional</i>	string
userid <i>optional</i>	string

3.4. CldsInfo

Name	Schema
cldsVersion <i>optional</i>	string
permissionReadCl <i>optional</i>	boolean
permissionReadTemplate <i>optional</i>	boolean
permissionReadTosca <i>optional</i>	boolean
permissionUpdateCl <i>optional</i>	boolean
permissionUpdateTemplate <i>optional</i>	boolean
permissionUpdateTosca <i>optional</i>	boolean

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

3.5. CldsModel

Name	Schema
blueprintText <i>optional</i>	string
bpmnText <i>optional</i>	string
cldsModelInstanceList <i>optional</i>	< CldsModelInstance > array
controlName <i>optional</i>	string
controlNamePrefix <i>optional</i>	string
controlNameUuid <i>optional</i>	string
deploymentId <i>optional</i>	string
deploymentStatusUrl <i>optional</i>	string
docText <i>optional</i>	string
errorMessageForUi <i>optional</i>	string
event <i>optional</i>	CldsEvent
id <i>optional</i>	string
imageText <i>optional</i>	string
name <i>optional</i>	string
permittedActionCd <i>optional</i>	< string > array

Name	Schema
propText <i>optional</i>	string
status <i>optional</i>	string
templateId <i>optional</i>	string
templateName <i>optional</i>	string
typeId <i>optional</i>	string
typeName <i>optional</i>	string

3.6. CldsModelInstance

Name	Schema
location <i>optional</i>	string
modelInstanceId <i>optional</i>	string
vmName <i>optional</i>	string

3.7. CldsMonitoringDetails

Name	Schema
action <i>optional</i>	string
closeLoopName <i>optional</i>	string
deploymentId <i>optional</i>	string
modelId <i>optional</i>	string
modelName <i>optional</i>	string

Name	Schema
serviceTypeId <i>optional</i>	string
templateName <i>optional</i>	string
timestamp <i>optional</i>	string
userid <i>optional</i>	string

3.8. CldsTemplate

Name	Schema
bpmnId <i>optional</i>	string
bpmnText <i>optional</i>	string
bpmnUserId <i>optional</i>	string
controlNamePrefix <i>optional</i>	string
controlNameUuid <i>optional</i>	string
id <i>optional</i>	string
imageId <i>optional</i>	string
imageText <i>optional</i>	string
imageUserId <i>optional</i>	string
name <i>optional</i>	string
propId <i>optional</i>	string
propText <i>optional</i>	string

Name	Schema
propUserId <i>optional</i>	string
userAuthorizedToUpdate <i>optional</i>	boolean

3.9. CldsToscaModel

Name	Schema
createdDate <i>optional</i>	string
id <i>optional</i>	string
lastUpdatedDate <i>optional</i>	string
policyType <i>optional</i>	string
revisionId <i>optional</i>	string
toscaModelJson <i>optional</i>	string
toscaModelName <i>optional</i>	string
toscaModelYaml <i>optional</i>	string
userId <i>optional</i>	string
version <i>optional</i>	number (double)

3.10. DcaeEvent

Name	Schema
artifactName <i>optional</i>	string
cldsActionCd <i>optional</i>	string

Name	Schema
controlName <i>optional</i>	string
event <i>optional</i>	string
instances <i>optional</i>	< CldsModelInstance > array
resourceUUID <i>optional</i>	string
serviceUUID <i>optional</i>	string

3.11. ExternalComponent

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

3.12. ExternalComponentState

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

3.13. JsonArray

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

3.14. JsonNull

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

3.15. 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)

Name	Schema
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

3.16. 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

Name	Schema
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
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

3.17. Loop

Name	Schema
blueprint <i>optional</i>	string
components <i>optional</i>	< string, ExternalComponent > map

Name	Schema
dcaeBlueprintId <i>optional</i>	string
dcaeDeploymentId <i>optional</i>	string
dcaeDeploymentStatusUrl <i>optional</i>	string
globalPropertiesJson <i>optional</i>	JsonObject
lastComputedState <i>optional</i>	enum (DESIGN, SUBMITTED, DEPLOYED, RUNNING, STOPPED, IN_ERROR, WAITING)
loopLogs <i>optional</i>	< LoopLog > array
microServicePolicies <i>optional</i>	< MicroServicePolicy > array
modelPropertiesJson <i>optional</i>	JsonObject
name <i>optional</i>	string
operationalPolicies <i>optional</i>	< OperationalPolicy > array
svgRepresentation <i>optional</i>	string

3.18. 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

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

3.19. MicroServicePolicy

Name	Schema
jsonRepresentation <i>optional</i>	JsonObject
modelType <i>optional</i>	string
name <i>optional</i>	string
policyTosca <i>optional</i>	string
properties <i>optional</i>	JsonObject
shared <i>optional</i>	boolean
usedByLoops <i>optional</i>	< Loop > array

3.20. Number

Type : object

3.21. OperationalPolicy

Name	Schema
configurationsJson <i>optional</i>	JsonObject
jsonRepresentation <i>optional</i>	JsonObject
loop <i>optional</i>	Loop
name <i>optional</i>	string

3.22. ValueItem

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