**VirIoT control commands (JSON) - MQTT binding**

{command:[value], [arg1]:[value], [argN]:[value]}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Command** | **arguments:value** | **Publisher** | **Subscriber** | **Topic** | **Description of Actions** |
| addVThing | vSiloID: <vSiloID>  vThingID: <ThingID> | Master-controller | vSilo controller | vSilo/<vSiloID>/c\_in | Add a vThing to the silo whose ID is vSiloID |
| deleteVThing | vSiloID: <vSiloID>  vThingID: <ThingID> | Master-controller | vSilo controller | vSilo/<vSiloID>/c\_in | Delete the vThing from the silo whose ID is vSiloID |
| createVThing | thingVisorID: <thingVisorID>  vThing:{ label: <vThingLabel>  id: <vThingID>  description: <vThingDescription> } | Thing Visor | Master-controller | TV/<thingVisorID>/c\_out | Notify the creation of a new virtual thing  Note: <vThingID> must be equal to  <thingVisorID>/<vThingLID>  where vThingLID is a local identifier, e.g. a random number |
| destroyTV | thingVisorID: <thingVisorID> | Master-controller | Thing Visor | TV/<thingVisorID>/c\_in | Destroy thing Visor request |
| destroyTVAck | thingVisorID: <thingVisorID> | Thing Visor | Master Controller | TV/<thingVisorID>/c\_out | Destroy thing Visor ack |
| destroyVSilo | vSiloID: <vSiloID> | Master-controller | vSilo controller | vSilo/<vSiloID>/c\_in | Notify the silo of an impending shutdown of the container |
| destroyVSiloAck | vSiloID: <vSiloID> | Vsilo Controller | Master-controller | vSilo/<vSiloID>/c\_out | Destroy virtual silo ack |
| deleteVThing | vSiloID: ALL  vThingID: <ThingID> | Thing Visor | vSilo controller | vThing/<vThingID>/c\_out | Delete from all silos the vThing whose ID is <thingVisorID>/<vThingLID> |
| getContextRequest | vSiloID: <vSiloID>  vThingID: <ThingID> | vSilo Controller | Thing Visor | vThing/<vThingID>/c\_in | Get the current status of the virtual Thing whose id is vThingID |
| getContextResponse | data: [NGSI-LD Entity array]  meta: { vThingID: <ThingID> } | Thing Visor | vSilo Controller | vSilo/<vSiloID>/c\_in | Response to getContextRequest message. The payload of the message is an array of NGSI-LD entities |
| updateTV | thingVisorID: <thingVisorID>  tvDescription: <tvDescription>  params: { key: value, ... }  updateInfo: { key: value, ... } | Master Controller | Thing Visor | TV/<thingVisorID>/c\_in | Send update information to the ThingVisor |
| setVThingEndpoint | vThingID: <ThingID>  endpoint: <http endpoint> | Master Controller | Thing Visor | TV/<thingVisorID>/c\_in | Set virtual thing http endpoint for streaming services |
| delVThingEndpoint | vThingID: <ThingID> | Master Controller | Thing Visor | TV/<thingVisorID>/c\_in | Delete virtual thing http endpoint for streaming services |