

SDDC

Software Defined Data Center

Generische API

Silvan Adrian
Fabian Binna

1 Änderungshistorie

Datum	Version	Änderung	Autor
17.10.15	1.00	Erstellung des Dokuments	Gruppe
17.10.15	1.01	Operationen Übersicht	Silvan Adrian
17.10.15	1.02	Contracts für die wichtigsten Operationen	Silvan Adrian

Inhaltsverzeichnis

1	Änderungshistorie	2
2	Einführung	4
2.1	Zweck	4
2.2	Gültigkeit	4
2.3	Übersicht	4
3	Operationen	4
3.1	Allgemein	4
3.1.1	Contracts	4
3.2	Compute	5
3.2.1	Contracts	5
3.3	Storage	6
3.3.1	Contracts	6
3.4	Network	7
3.4.1	Contracts	7

2 Einführung

2.1 Zweck

Dieses Dokument beschreibt die generische API

2.2 Gültigkeit

Dieses Dokument ist während des ganzen Projekts gültig und wird laufend erweitert

2.3 Übersicht

Dieses Dokument soll eine Übersicht über die Funktionen der generischen API bieten.

3 Operationen

3.1 Allgemein

- connect(uri: String, ReadOnly: Boolean)
- disconnect()

3.1.1 Contracts

Operation	connect(uri : String, ReadOnly: Boolean)
Cross References	UC01
Preconditions	Eine URI wurde mitgegeben und ReadOnly True oder False gesetzt
Postconditions	<ul style="list-style-type: none">• Client ist mit dem Hypervisor verbunden
Operation	disconnect()
Cross References	UC01
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Verbindung zum Hypervisor wurde geschlossen

3.2 Compute

- createCompute(Config: String)
- deleteCompute(comp: Compute)
- getCompute(name: String)

3.2.1 Contracts

Operation	createCompute(Config: String)
Cross References	UC01
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Compute Instanz wurde erstellt
Operation	deleteCompute(comp: Compute)
Cross References	UC01
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Compute Instanz wurde gelöscht
Operation	getCompute(name: String)
Cross References	UC01
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Hypervisor gibt ComputeObjekt zurück

3.3 Storage

- createStorage(Config: String)
- deleteStorage(storage: Storage)
- getStorage(name: String)

3.3.1 Contracts

Operation	createStorage(Config: String)
Cross References	UC01
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Storage wurde erstellt
Operation	deleteStorage(storage: Storage)
Cross References	UC02
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Storage wurde gelöscht
Operation	getStorage(name: String)
Cross References	UC02
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Hypervisor gibt Storage Objekt zurück

3.4 Network

- createNetwork(Config: String)
- deleteNetwork(network: Network)
- getNetwork(name: String)

3.4.1 Contracts

Operation	createNetwork(Config: String)
Cross References	UC01
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Netzwerk wurde erstellt
Operation	deleteNetwork(network: Network)
Cross References	UC02
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Netzwerk wurde gelöscht
Operation	getNetwork(name: String)
Cross References	UC02
Preconditions	Client ist mit Hypervisor verbunden
Postconditions	<ul style="list-style-type: none">• Hypervisor gibt Network Objekt zurück