

TESTING THE OVIRT TERRAFORM PROVIDER

1. CODE QUALITY

1. CODE QUALITY

2. WHY?

1. CODE QUALITY
2. WHY?
3. MAKING TESTS ROBUST

1. CODE QUALITY
2. WHY?
3. MAKING TESTS ROBUST
4. REFACTORING EVERYTHING

1. CODE QUALITY
2. WHY?
3. MAKING TESTS ROBUST
4. REFACTORING EVERYTHING
5. ???

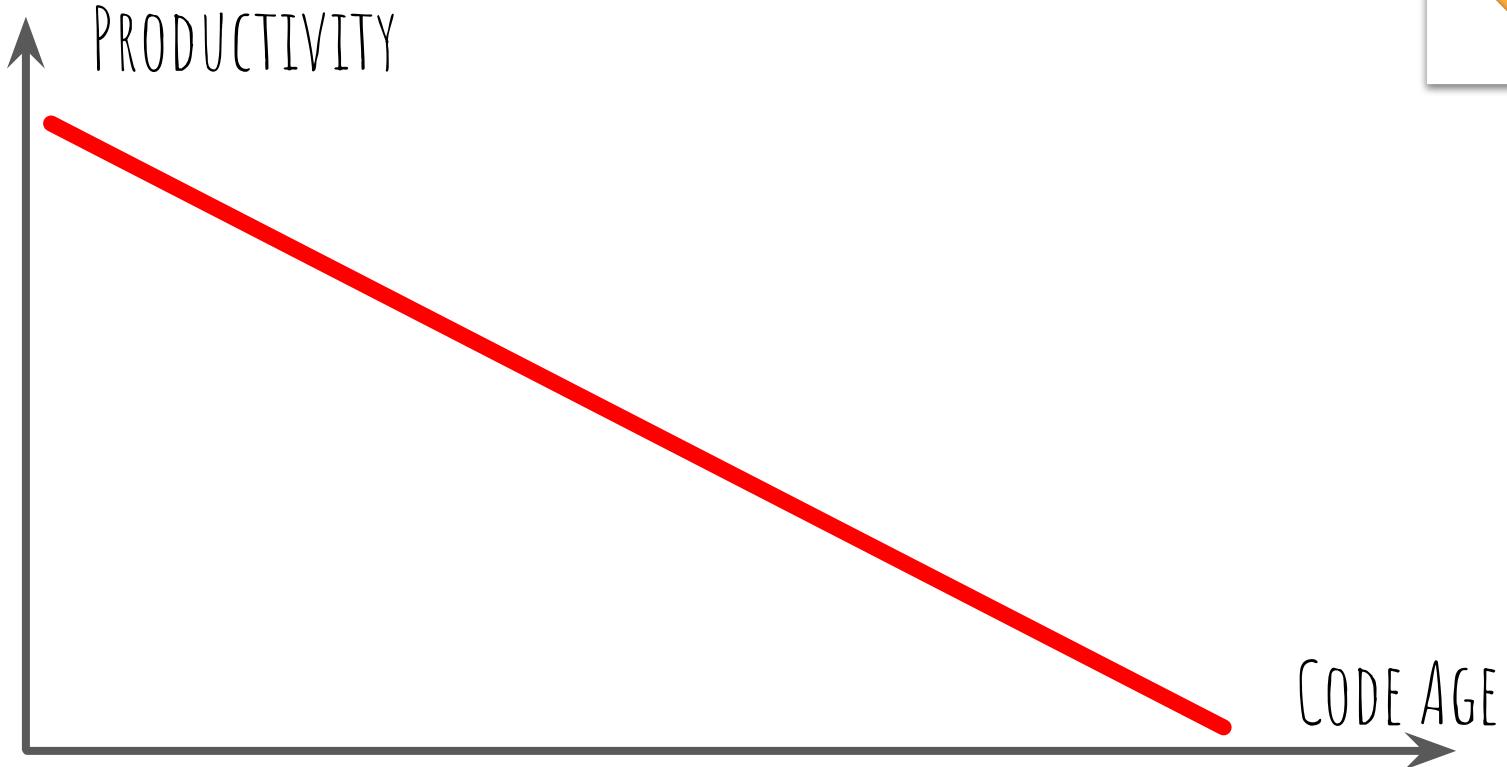
1. CODE QUALITY
2. WHY?
3. MAKING TESTS ROBUST
4. REFACTORING EVERYTHING
5. ???
6. PROFIT!

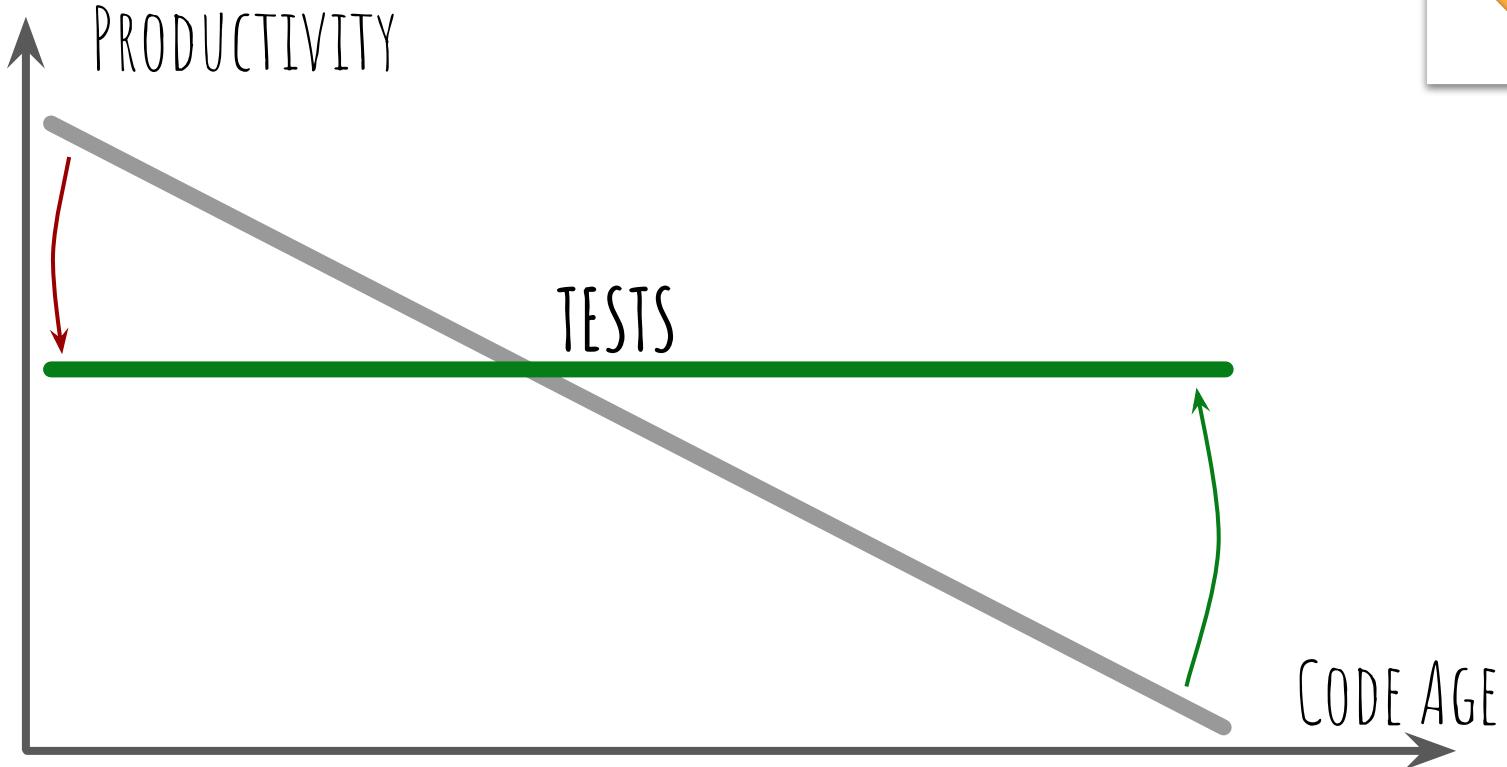


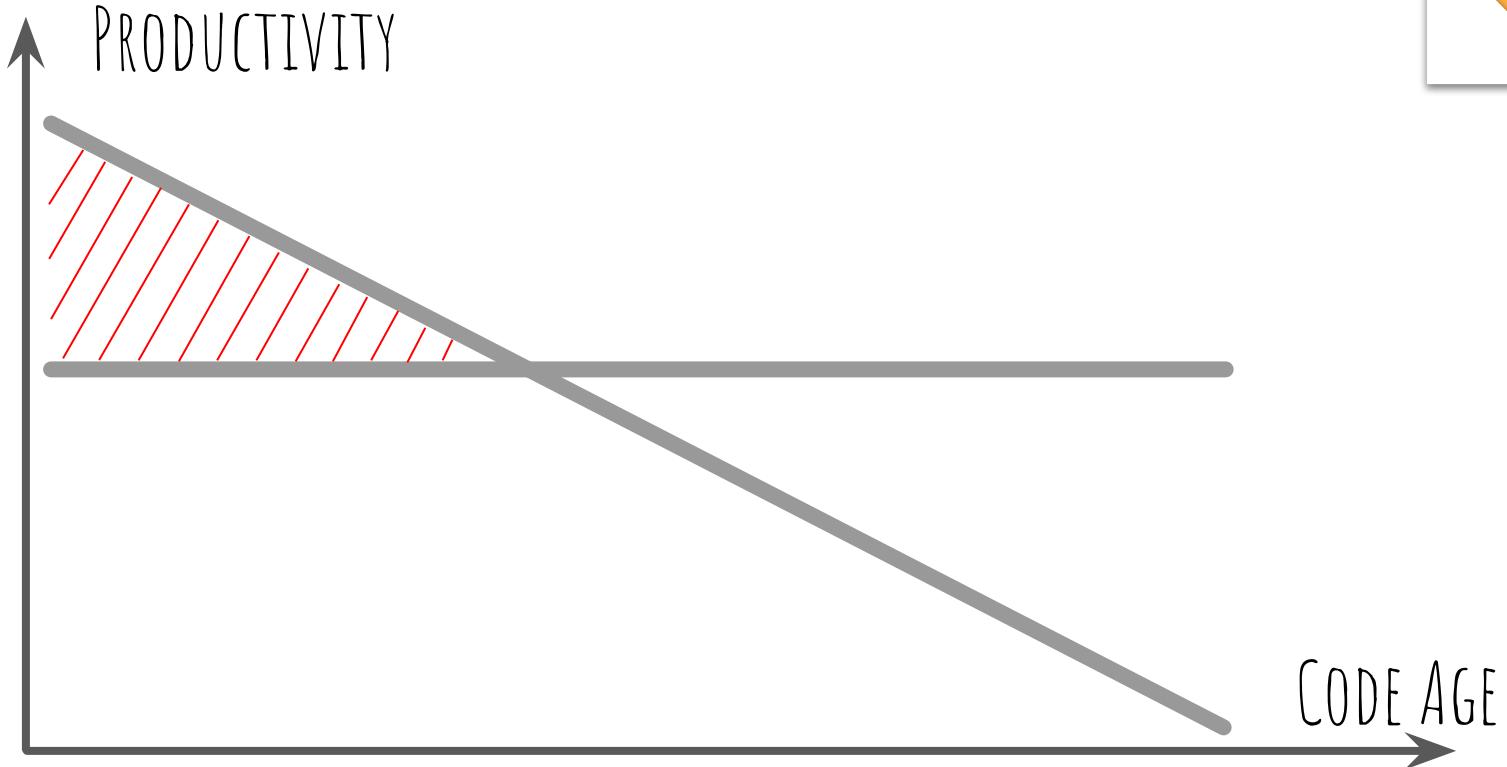
PRODUCTIVITY

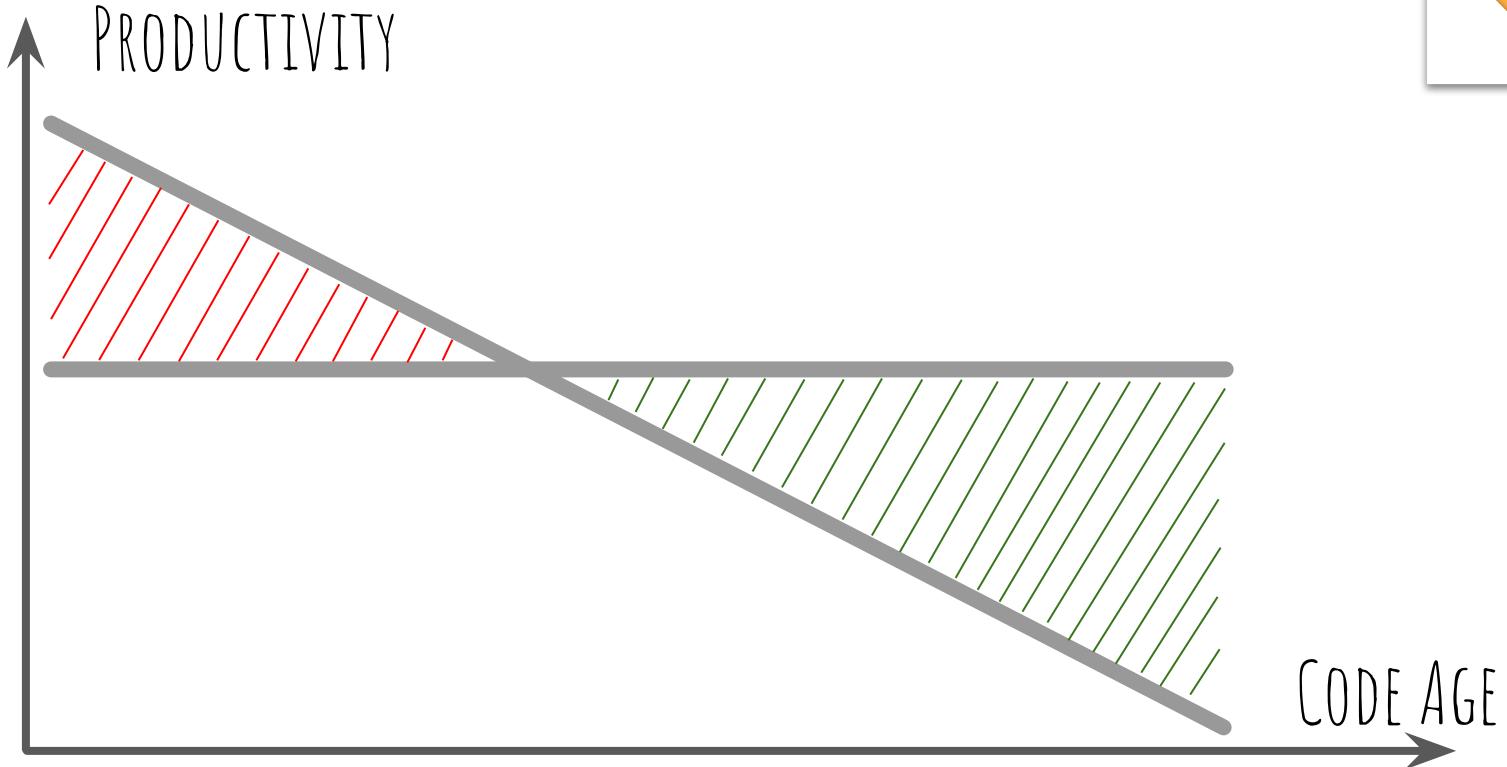


CODE AGE





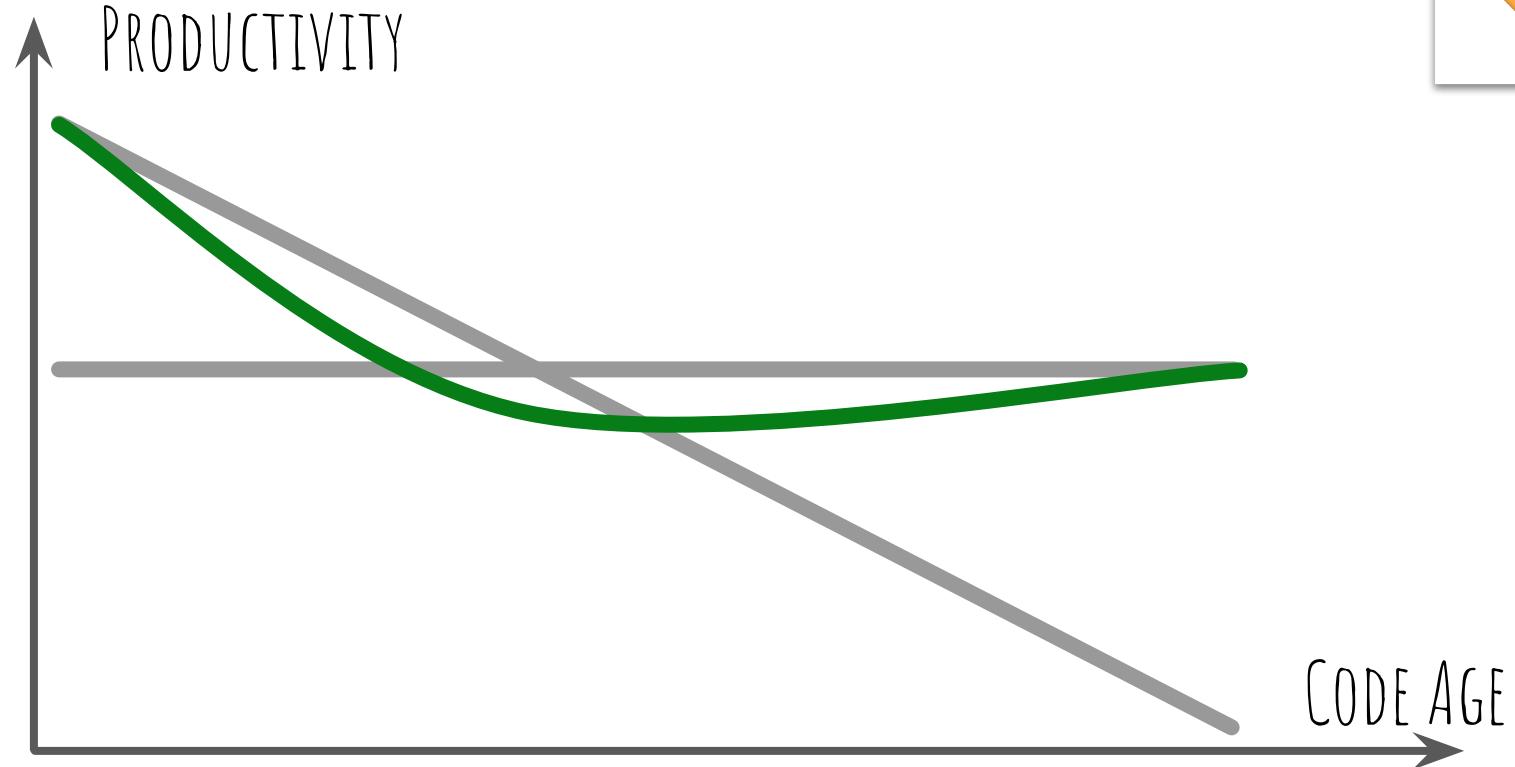


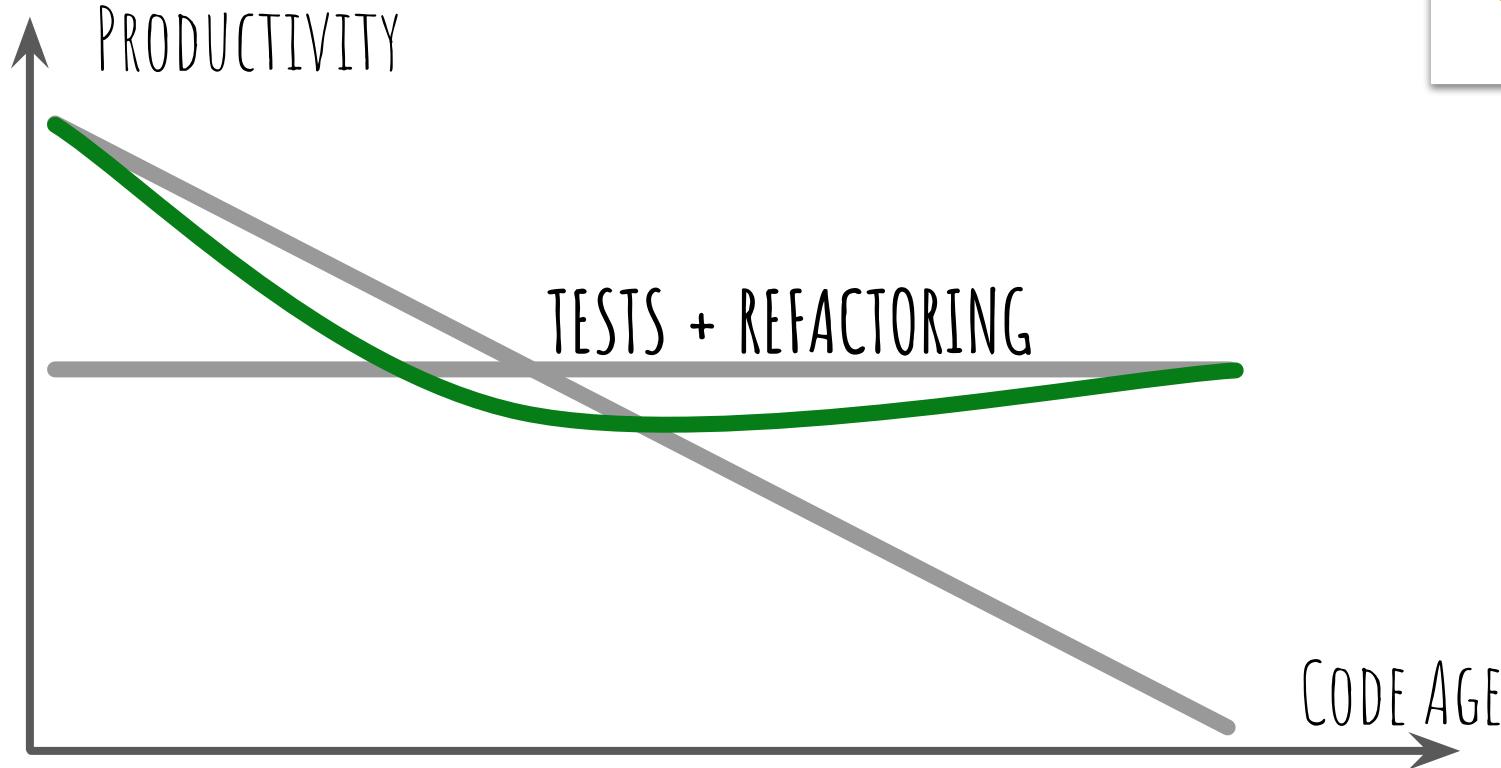


PROJECTS WITHOUT TECH DEBT



JOHN TRAVOLTA IN PULP FICTION, © MIRAMAX FILMS





OPENShift INSTALLER

OPENSHIFT INSTALLER

OVIRT TERRAFORM
PROVIDER



OCPBUGSM-23696

[1919980] oVirt installer fails due to terraform error "Failed to wait for Template(...)



OCPBUGSM-23415

[1917898] [ovirt] install fails: due to terraform error "Tag not matched: expect <fault> but

⋮



OCPBUGSM-23147

[1916279] [OCPonRHV] Sometimes terraform installation fails on -failed to fetch Cluster(a)



OCPBUGSM-20311

[1896181] [ovirt] install fails: due to terraform error "Cannot run VM. VM is being updated



OCPBUGSM-23414

[1917893] [ovirt] install fails: due to terraform error "Cannot attach Virtual Disk: Disk is loc



OCPBUGSM-30984

[1972527] [RFE] add option to set VMs as highly available

- OCPBUGSM-23696 [1919980] oVirt installer fails due to terraform error "Failed to wait for Template(...)"
- OCPBUGSM-23415 [1917898] [ovirt] install fails: due to terraform error "Tag not matched: expect <fault> but
- OCPBUGSM-23147 [1916279] [OCPonRHV] Sometimes terraform installation fails on -failed to fetch Cluster(a)
- OCPBUGSM-20311 [1896181] [ovirt] install fails: due to terraform error "Cannot run VM. VM is being updated"
- OCPBUGSM-23414 [1917893] [ovirt] install fails: due to terraform error "Cannot attach Virtual Disk: Disk is loc
- OCPBUGSM-30984 [1972527] [RFE] add option to set VMs as highly available





Search or jump to...



Pull requests Issues Marketplace Explore

oVirt / terraform-provider-ovirt

forked from EMSL-MSC/terraform-provider-ovirt



Watch

14



Star

116

Code

Issues 21

Pull requests 16

Actions

Security

Insights

Settings

master ▾

7 branches

10 tags

Go to file

Add file ▾

Code ▾

About



Terraform provider for oVirt 4.x

terraform

terraform-provider

ovirt

Readme

View license

Releases

10

v0.99.0

Latest

on 18 Mar

+ 9 releases

Packages

This branch is 346 commits ahead, 1 commit behind EMSL-MSC:master.

Contribute ▾ Fetch upstream ▾

emesika Merge pull request #286 from emesika/issue-268 ...

a9e23f6 7 days ago 367 commits

.github

Revert to Go 1.14 (waiting for Terraform #27257)

5 months ago

docs

use the new auto pinning naming

2 months ago

examples

Added option storage_domain parameter. Related only for build VM fro...

9 months ago

ovirt

set insecure default value to false

7 days ago

scripts

Prune the unnecessary scripts

2 years ago

vendor

bump go mod to 1.14

5 months ago

.gitignore

Added live checks for either of the certificate options to be set

5 months ago

master ▾

[terraform-provider-ovirt](#) / .github / PULL_REQUEST_TEMPLATE.md

15 lines (10 sloc) | 353 Bytes

Fixes #0000

Changes proposed in this pull request:

- Change 1
- Change 2

Output from acceptance testing:

```
$ make testacc TEST=./ovirt TESTARGS='-run=TestAccOvirtDataCenter_'
```

...

master ▾

[terraform-provider-ovirt](#) / .github / PULL_REQUEST_TEMPLATE.md

15 lines (10 sloc) | 353 Bytes

Fixes #0000

Changes proposed in this pull request:

- Change 1
- Change 2

Output from acceptance testing:

```
$ make testacc TEST=./ovirt TESTARGS='-run=TestAccOvirtDataCenter_'
```

...

IT HAS TESTS!



master ▾

terraform-provider-ovirt / .github / PULL_REQUEST_TEMPLATE.md

15 lines (10 sloc) | 353 Bytes

Fixes #0000

Changes proposed in this pull request:

- Change 1
- Change 2

Output from acceptance testing:

```
$ make testacc TEST=./ovirt TESTARGS='-run=TestAccOvirtDataCenter_'
```

...

EXPECTING CONTRIBUTORS
TO RUN TESTS?



master ▾

terraform-provider-ovirt / .github / PULL_REQUEST_TEMPLATE.md

15 lines (10 sloc) | 353 Bytes

Fixes #0000

Changes proposed in this pull request:

- Change 1
- Change 2

Output from acceptance testing:

```
$ make testacc TEST=./ovirt TESTARGS='-run=TestAccOvirtDataCenter_'
```

...

RUNNING ONLY ONE TEST?



```
func TestAccOvirtDiskAttachment_basic (t *testing.T) {
    var diskAttachment ovirtsdk4.DiskAttachment
    vmID := "437d0f69-d1eb-441f-bf6b-0e97797fe11e"
    diskID := "230349f6-59a9-47e9-bc90-7c1221645b07"
    resource.Test(t, resource.TestCase{
        PreCheck:     func() { testAccPreCheck(t) },
        Providers:    testAccProviders,
        IDRefreshName: "ovirt_disk_attachment.attachment",
        CheckDestroy: testAccCheckDiskAttachmentDestroy,
        Steps: []resource.TestStep{
            {
                Config: testAccDiskAttachmentBasic (vmID, diskID),
                Check: resource.ComposeTestCheckFunc (
                    testAccCheckOvirtDiskAttachmentExists (
                        "ovirt_disk_attachment.attachment" , &diskAttachment),
                    resource.TestCheckResourceAttr (
                        "ovirt_disk_attachment.attachment" , "interface" , "virtio"),
                    resource.TestCheckResourceAttr (
                        "ovirt_disk_attachment.attachment" , "bootable" , "true"),
                    resource.TestCheckResourceAttr (
                        "ovirt_disk_attachment.attachment" , "read_only" , "true"),
                    resource.TestCheckResourceAttr (
                        "ovirt_disk_attachment.attachment" , "active" , "true"),
                ),
            },
        },
    })
}
```

OH MY DOG!



PHOTO BY [MATTHEW HENRY](#) ON [UNSPLASH](#)

```
func TestAccOvirtDiskAttachment_basic(t *testing.T) {
    var diskAttachment ovirtsdk4.DiskAttachment
    vmID := "437d0f69-d1eb-441f-bf6b-0e97797fe11e"
    diskID := "230349f6-59a9-47e9-bc90-7c1221645b07"
    resource.Test(t, resource.TestCase{
        PreCheck: func() { testAccPreCheck(t) },
        Providers: testAccProviders,
        IDRefreshName: "ovirt_disk_attachment.attachment",
        CheckDestroy: testAccCheckDiskAttachmentDestroy,
        Steps: []resource.TestStep{
            {
                Config: testAccDiskAttachmentBasic(vmID, diskID),
                Check: resource.ComposeTestCheckFunc(
                    testAccCheckOvirtDiskAttachmentExists(
                        "ovirt_disk_attachment.attachment", &diskAttachment),
                    resource.TestCheckResourceAttr(
                        "ovirt_disk_attachment.attachment", "interface", "virtio"),
                    resource.TestCheckResourceAttr(
                        "ovirt_disk_attachment.attachment", "bootable", "true"),
                    resource.TestCheckResourceAttr(
                        "ovirt_disk_attachment.attachment", "read_only", "true"),
                    resource.TestCheckResourceAttr(
                        "ovirt_disk_attachment.attachment", "active", "true"),

```

HARD-CODED ID'S?

TODO

1. MAKE TESTS ROBUST
2. ADD A CI SYSTEM

TODO

1. MAKE TESTS ROBUST
- ~~2. ADD A CI SYSTEM~~

```
var testAccCheckOvirtAuthzsDataSourceNameRegexConfig = `  
data "ovirt_authzs" "name_regex_filtered_authz" {  
    name_regex = "^internal-*"  
    search {  
        max = 1  
    }  
}  
`
```

```
var testAccCheckOvirtAuthzsDataSourceNameRegexConfig = `  
data "ovirt_authzs" "name_regex_filtered_authz" {  
    name_regex = "^internal-*"  
    search {  
        max = 1  
    }  
}  
`
```



```
func TestAccOvirtAuthzsDataSource_nameRegexFilter (t *testing.T) {
    resource.Test(t, resource.TestCase{
        PreCheck: func() { testAccPreCheck(t) },
        Providers: testAccProviders,
        Steps: []resource.TestStep{
            {
                Config: testAccCheckOvirtAuthzsDataSourceNameRegexConfig,
                Check: resource.ComposeTestCheckFunc(
                    testAccCheckOvirtDataSourceID("data.ovirt_authzs.name_regex_filtered_authz"),
                    resource.TestCheckResourceAttr(
                        "data.ovirt_authzs.name_regex_filtered_authz",
                        "authzs.#",
                        "1"),
                    resource.TestCheckResourceAttr(
                        "data.ovirt_authzs.name_regex_filtered_authz",
                        "authzs.0.name",
                        regexp.MustCompile(`^internal-*`)),
                ),
            },
        },
    })
}
```

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    resource.Test(t, resource.TestCase{
        PreCheck: func() { testAccPreCheck(t) },
        Providers: testAccProviders,
        Steps: []resource.TestStep{
            {
                Config: testAccCheckOvirtAuthzsDataSourceNameRegexConfig,
                Check: resource.ComposeTestCheckFunc(
                    testAccCheckOvirtDataSourceID("data.ovirt_authzs.name_regex_filtered_authz"),
                    resource.TestCheckResourceAttr(
                        "data.ovirt_authzs.name_regex_filtered_authz",
                        "authzs.#",
                        "1"),
                    resource.TestCheckResourceAttr(
                        "data.ovirt_authzs.name_regex_filtered_authz",
                        "authzs.0.name",
                        regexp.MustCompile(`^internal-*`)),
                    ),
                },
            },
        },
    })
}
```



NOT SOFTWARE



PHOTO BY [PATRICIO GIUSTI](#) ON [UNSPLASH](#)

```
func TestAccOvirtAuthzsDataSource_nameRegexFiltered(t *testing.T) {
    resource.Test(t, resource.TestCase{
        PreCheck:  func() { testAccPreCheck(t) },
        Providers: testAccProviders,
        Steps: []resource.TestStep{
            {
                Config: testAccCheckOvirtAuthzsDataSourceNameRegexConfig,
                Check:  resource.ComposeTestCheckFunc(
                    testAccCheckOvirtDataSourceID("data.ovirt_authzs.name_regex_filtered_authz"),
                    resource.TestCheckResourceAttr(
                        "data.ovirt_authzs.name_regex_filtered_authz",
                        "authzs.#",
                        "1"),
                    resource.TestCheckResourceAttr(
                        "data.ovirt_authzs.name_regex_filtered_authz",
                        "authzs.0.name",
                        regexp.MustCompile(`^internal-*`)),
                ),
            },
        },
    })
}
```

SOFTWARE

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
```

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    suite := getOvirtTestSuite(t)
```

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    suite := getOvirtTestSuite(t)
    authzName := regexp.QuoteMeta(suite.GetTestAuthzName())
```

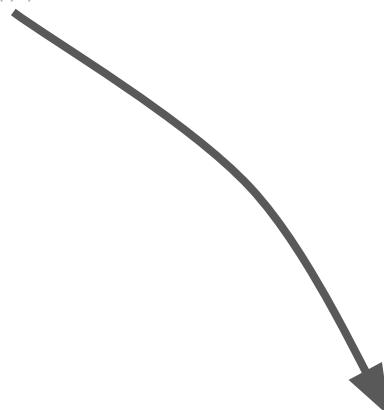
```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    suite := getOvirtTestSuite(t)
    authzName := regexp.QuoteMeta(suite.GetTestAuthzName())
    resource.Test(t, resource.TestCase{
        PreCheck:  suite.PreCheck,
        Providers: suite.Providers(),
        Steps: []resource.TestStep{
```

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    suite := getOvirtTestSuite(t)
    authzName := regexp.QuoteMeta(suite.GetTestAuthzName())
    resource.Test(t, resource.TestCase{
        PreCheck:  suite.PreCheck,
        Providers: suite.Providers(),
        Steps: []resource.TestStep{
            {
                Config: fmt.Sprintf(`

data "ovirt_authzs" "name_regex_filtered_authz" {
    name_regex = "^ %s$"
    search = {
        max = 1
    }
}
`, authzName),
            },
        },
    })
}
```

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    suite := getOvirtTestSuite(t)
    authzName := regexp.QuoteMeta(suite.GetTestAuthzName())
    resource.Test(t, resource.TestCase{
        PreCheck:  suite.PreCheck,
        Providers: suite.Providers(),
        Steps: []resource.TestStep{
            {
                Config: fmt.Sprintf(`

data "ovirt_authzs" "name_regex_filtered_authz" {
    name_regex = "^ %s$"
    search = {
        max = 1
    }
}
`, authzName),
            },
        },
    })
}
```



authzName: internal-something

```
func TestAccOvirtAuthzsDataSource_nameRegexFilter(t *testing.T) {
    suite := getOvirtTestSuite(t)
    authzName := regexp.QuoteMeta(suite.GetTestAuthzName())
    resource.Test(t, resource.TestCase{
        PreCheck:  suite.PreCheck,
        Providers: suite.Providers(),
        Steps: []resource.TestStep{
            {
                Config: fmt.Sprintf(`  

data "ovirt_authzs" "name_regex_filtered_authz" {  

    name_regex = "^ %s$"  

    search = {  

        max = 1  

    }  

}  
`), authzName),
        },
    })
}
```

authzName: internal-something

 Search or jump to... / Pull requests Issues Marketplace Explore You have no unread notifications

oVirt / **terraform-provider-ovirt** Watch 14 Star 116

forked from EMSL-MSC/terraform-provider-ovirt

Code Issues 21 Pull requests 16 Actions Security Insights Settings

Test refactor #277

 janosdebugs wants to merge 23 commits into `oVirt:master` from `janosdebugs:test-refactor` 

Conversation 11 Commits 23 Checks 1 Files changed 114 +4,651 -1,470 

 janosdebugs commented on 6 Jul Member  

This PR replaces [#267](#) because of a GitHub account switch.

This pull request makes most of the tests runnable again by implementing methods that allow it to run on any oVirt cluster that has a working host.

The test suite supports the following environment variables (to be added to the README):

- `OVIRT_USERNAME`
- `OVIRT_PASSWORD`
- `OVIRT_URL` (Should point to <https://api-host/ovirt-engine/api>)
- `OVIRT_INSECURE` (Set to 1 to connect the oVirt engine without certificate verification)

Reviewers Gal-Zaidman 

Still in progress? Convert to draft

Assignees No one—assign yourself 

Labels None yet 

Search or jump to... / Pull requests Issues Marketplace Explore You have no unread notifications

oVirt / **terraform-provider-ovirt** Watch 14 Star 116

forked from EMSL-MSC/terraform-provider-ovirt

Code Issues 21 Pull requests 16 Actions Security Insights Settings

Test refactor #277

Open janosdebugs wants to merge 23 commits into `ovirt:master` from `janosdebugs:test-refactor`

Conversation 11 Commits 23 Checks 1 Files changed 114 +4,651 -1,470

janosdebugs commented on 6 Jul

This PR replaces #267 because of a GitHub account switch.

This pull request makes most of the tests runnable again by implementing methods that allow it to run on any oVirt cluster that has a working host.

The test suite supports the following environment variables (to be added to the README):

- `OVIRT_USERNAME`
- `OVIRT_PASSWORD`
- `OVIRT_URL` (Should point to <https://api-host/ovirt-engine/api>)
- `OVIRT_INSECURE` (Set to 1 to connect the oVirt engine without certificate verification)

Reviewers G. Zaidman Still in progress? Convert to draft

Assignees No one—assign yourself

Labels None yet





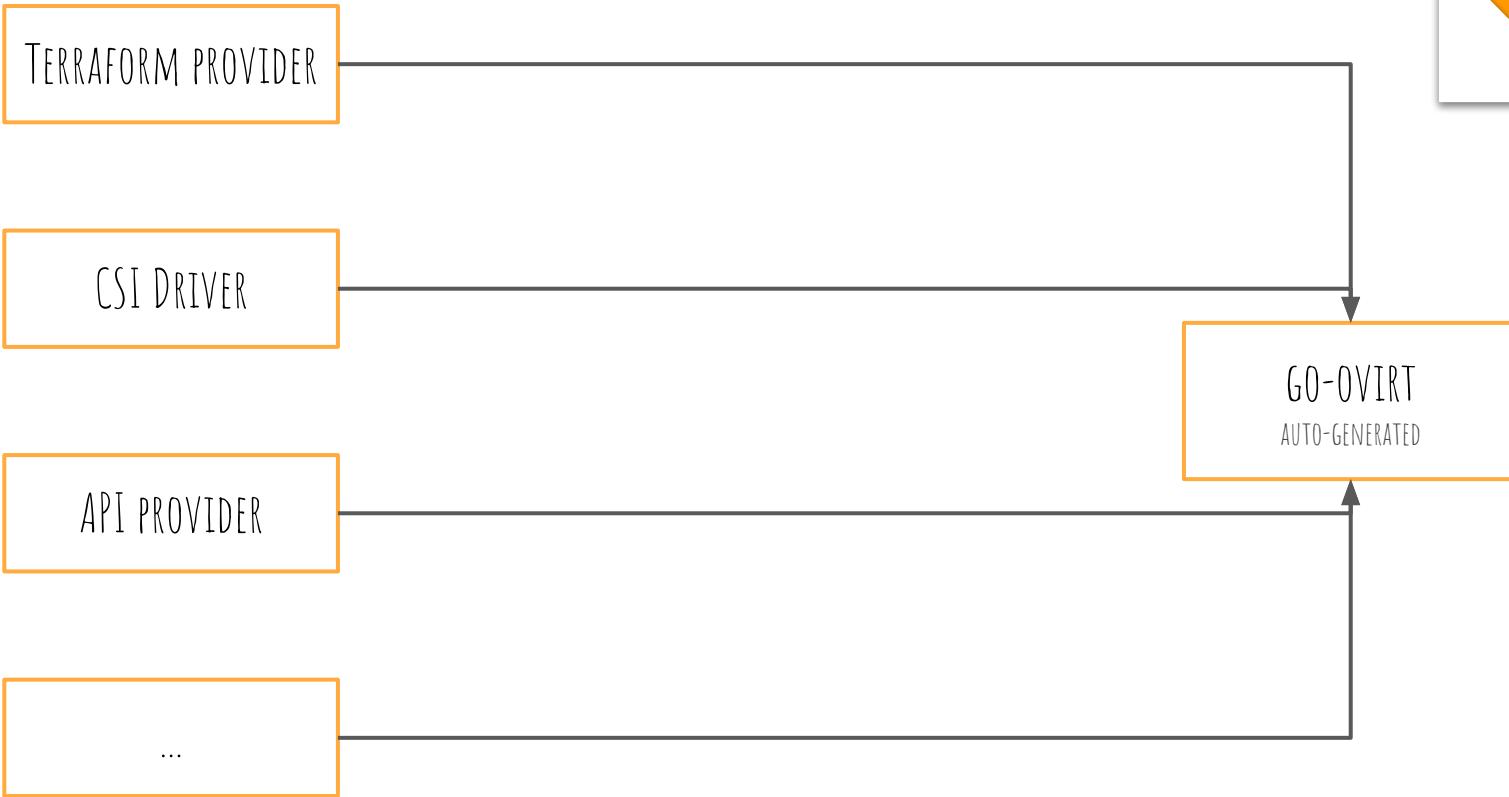
PHOTO BY [CHRISTINE SANDU](#) ON [UNSPLASH](#)

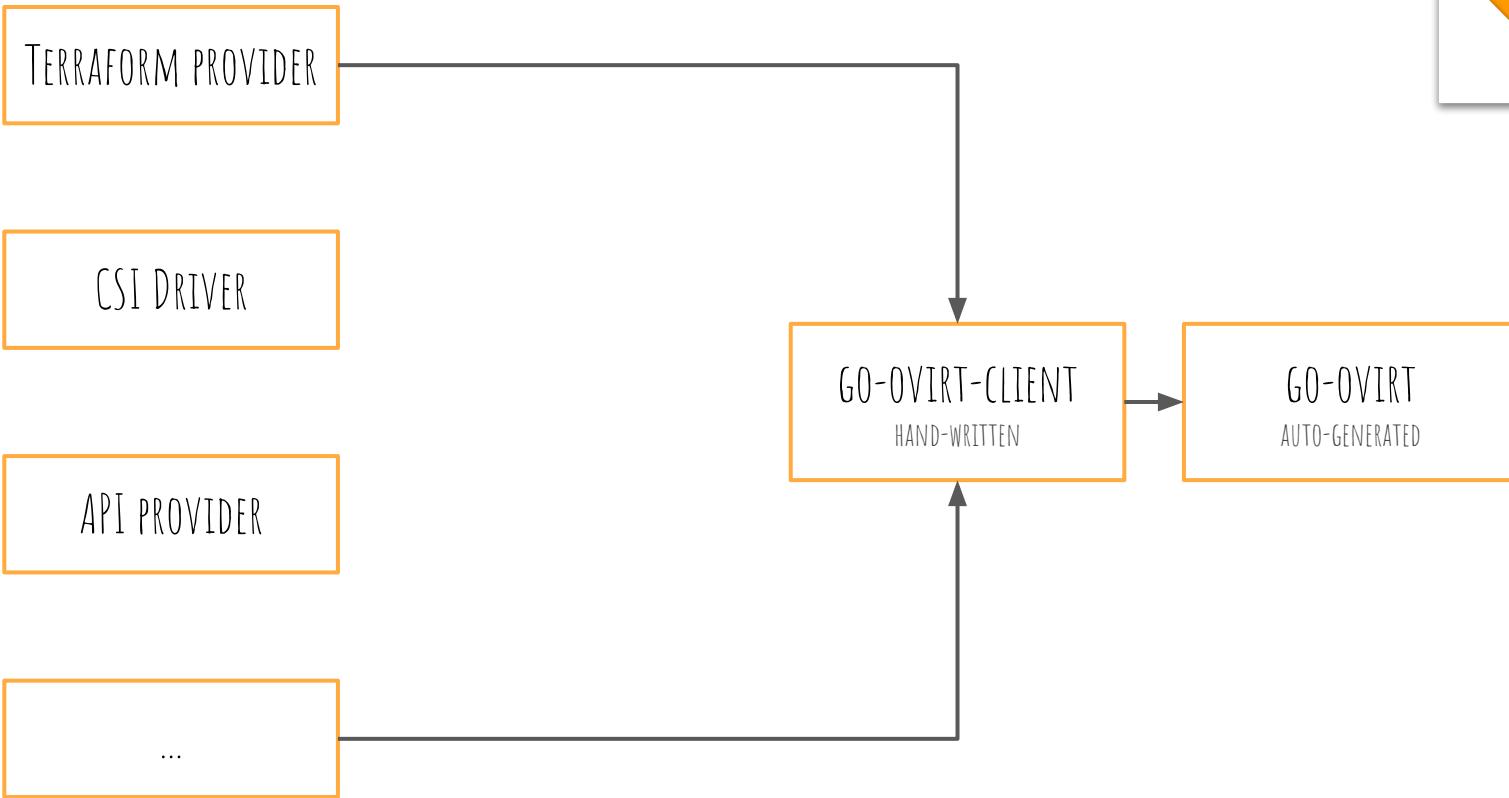
SIMILAR CODE

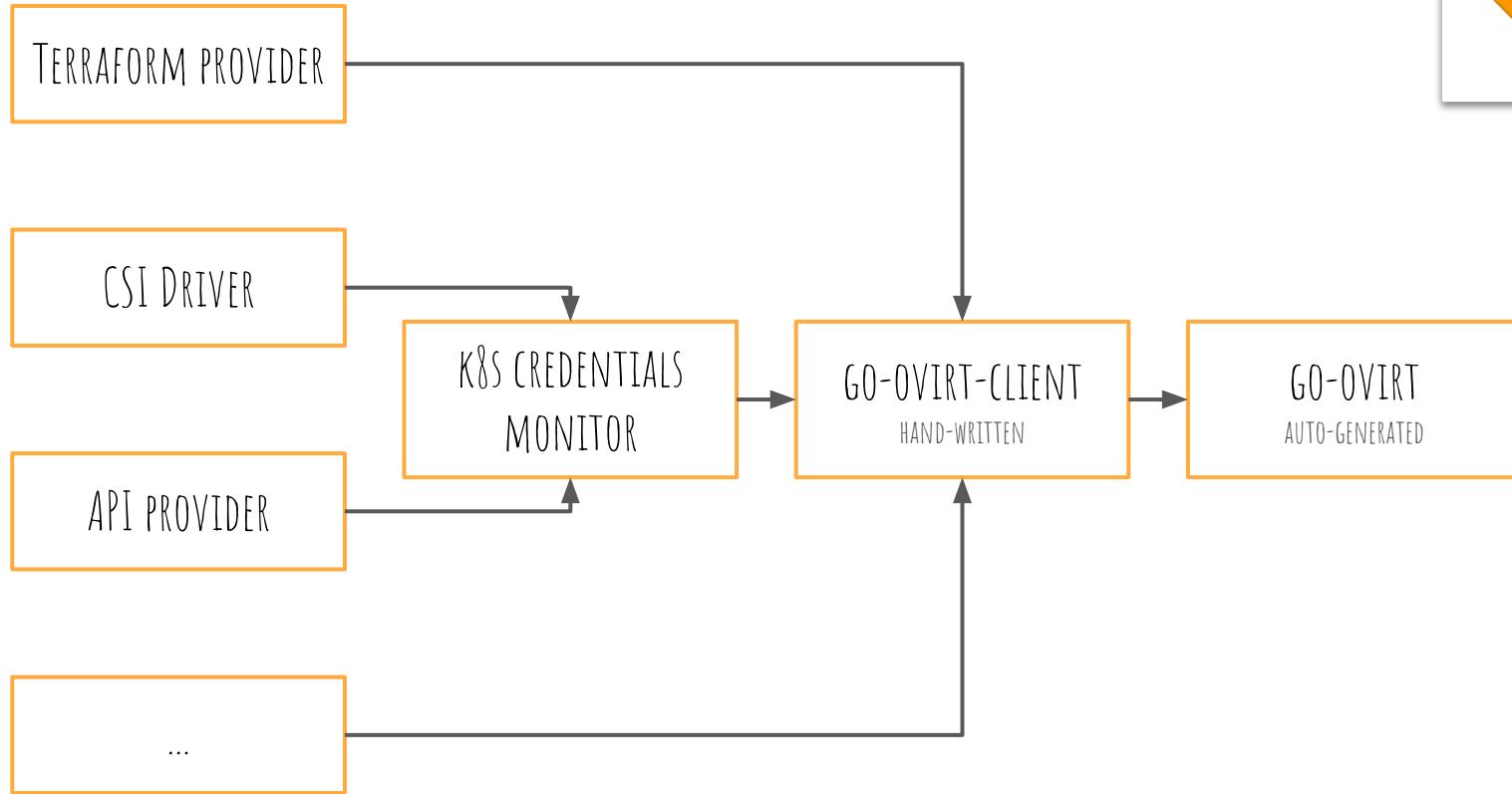
1. TERRAFORM PROVIDER,
2. OPENSHIFT OVIRT CSI DRIVER,
3. OPENSHIFT OVIRT API PROVIDER,
4. ...AND A FEW MORE PLACES.

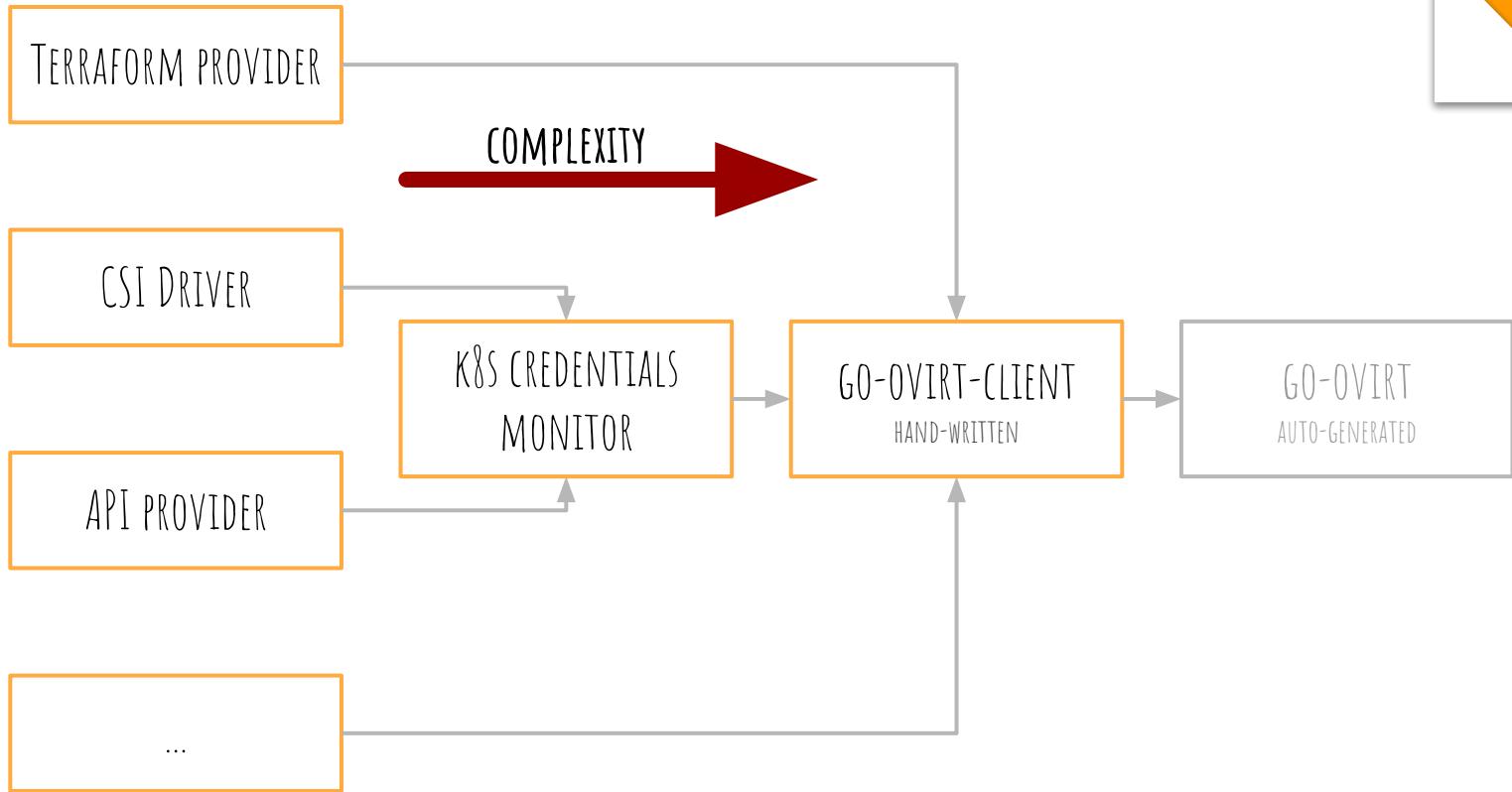
TODO

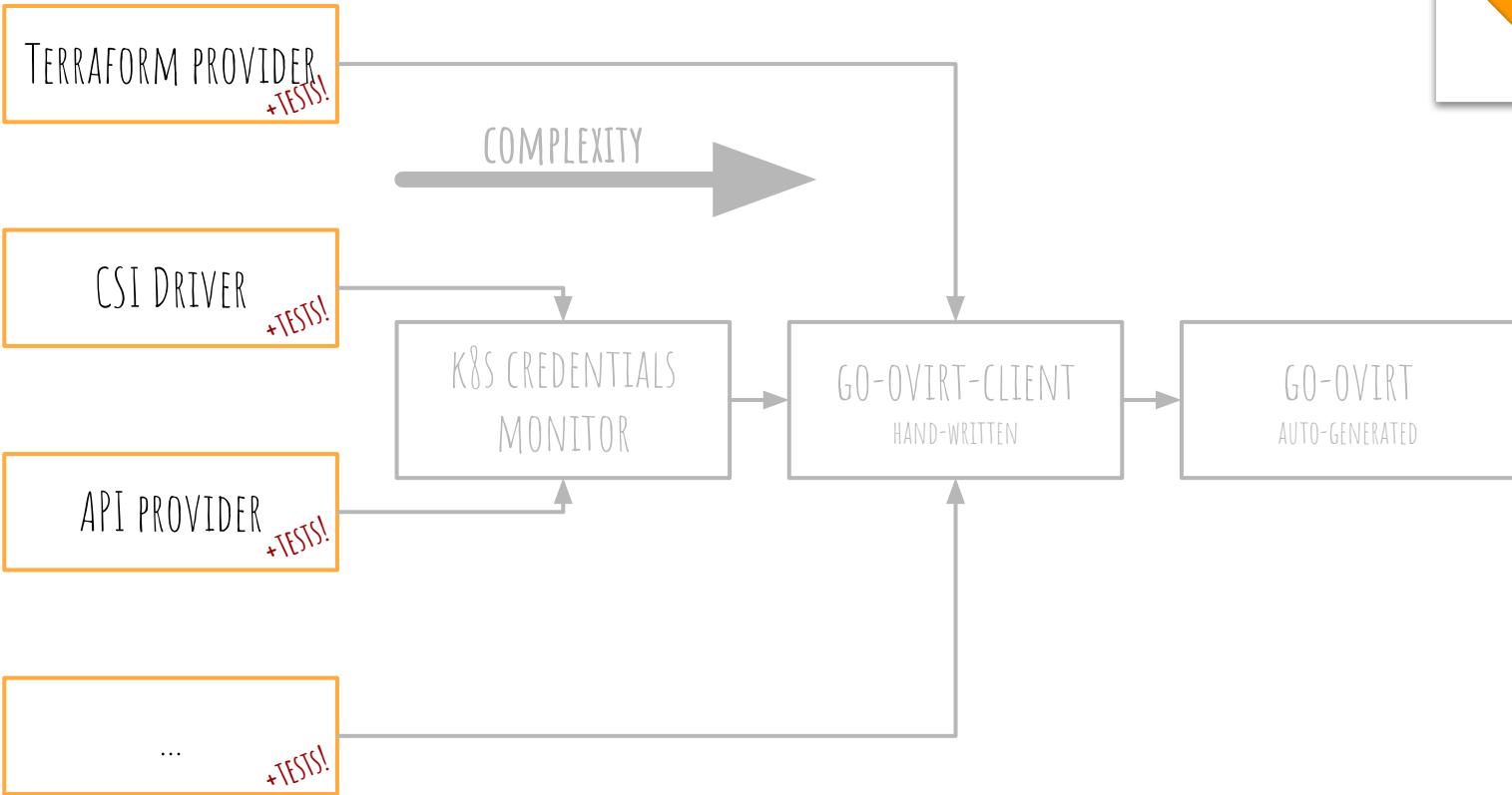
1. ~~MAKE TESTS DYNAMIC~~
2. ~~ADD A CI SYSTEM~~
3. UNIFIED OVERLAY!











```
OVIRTCLIENT.NEW(
```

```
)
```

```
OVIRTCLIENT.NEW(  
    MOCK: TRUE  
)
```

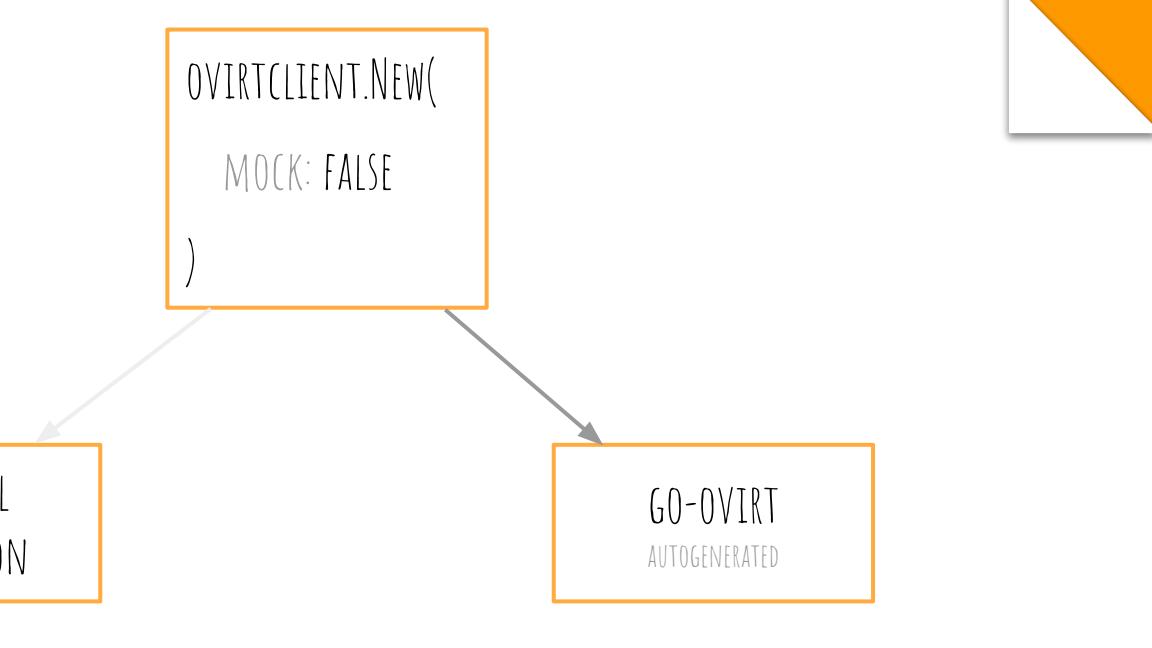
```
INTERNAL  
SIMULATION
```

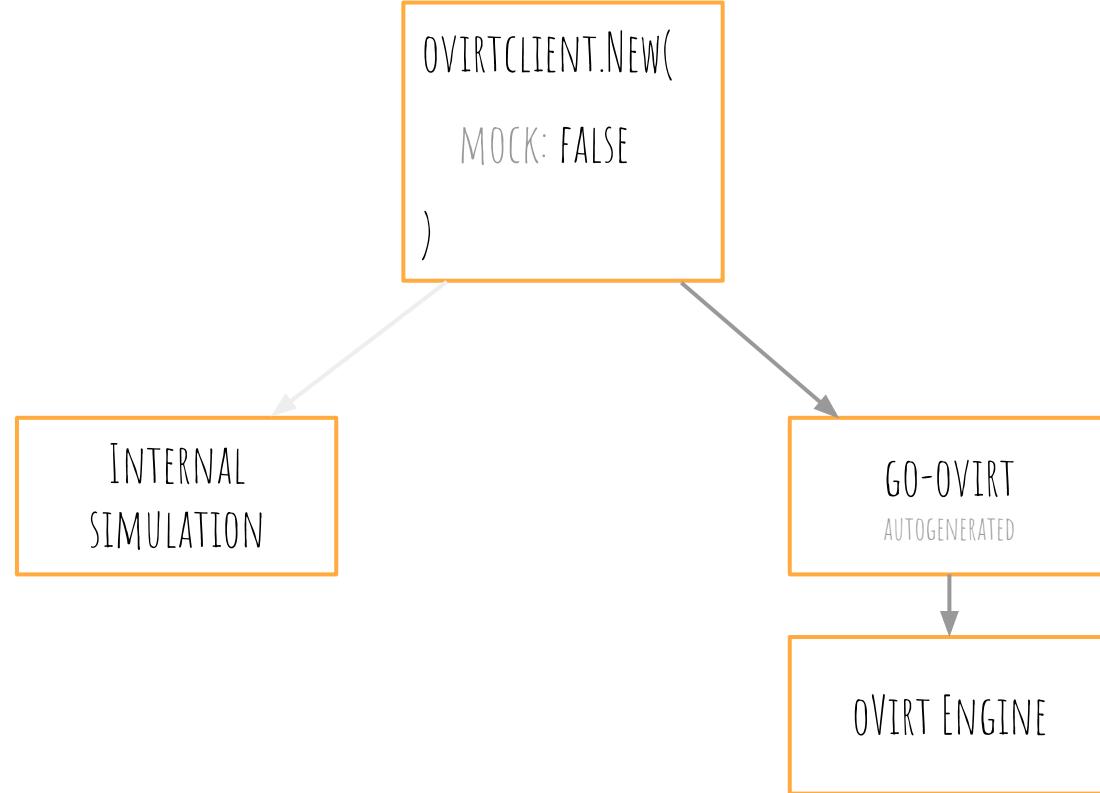


```
OVIRTCLIENT.New(  
    MOCK: FALSE  
)
```

INTERNAL
SIMULATION

GO-OVIRT
AUTOGENERATED





```
func TestImageDownload(t *testing.T) {  
}  
}
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()
    helper: ovirtclient.TestHelper
    client: ovirtclient.Client
}

}
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    yourApplication := yourapp.New(client)

}

    helper: ovirtclient.TestHelper
    client: ovirtclient.Client
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    testImageFile := "./testimage/image.raw"
    testImageSize := 512
    imageName := fmt.Sprintf(
        "client test %s",
        helper.GenerateRandomID(5),
    )
}

}

}
```

helper: ovirtclient.TestHelper
client: ovirtclient.Client

imageName: client_test_asdf1

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    testImageFile := "./testimage/image.raw"
    testImageSize := 512
    imageName     := fmt.Sprintf(
        "client test %s",
        helper.GenerateRandomID(5),
    )

    fileHandle, err := os.Open(testImageFile)
    helper: ovirtclient.TestHelper
    client: ovirtclient.Client
    imageName: client_test_asdf1
    testImageFile: ./testimage/image.raw
}

}
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    testImageFile := "./testimage/image.raw"
    testImageSize := 512
    imageName := fmt.Sprintf(
        "client test %s",
        helper.GenerateRandomID(5),
    )

    fileHandle, err := os.Open(testImageFile)
    uploadResult, err := client.UploadImage (
        alias:           imageName,
        storageDomainID: helper.GetStorageDomainID(),
        sparse:          true,
        size:            testImageSize,
        reader:          fileHandle,
    )
}

func TestImageDownload(t *testing.T) {
    helper := ovirtclient.TestHelper
    client := ovirtclient.Client

    imageName := client_test_asdf1
    testImageFile := ./testimage/image.raw
    uploadResult := ovirtclient.UploadResult
    imageName := client_test_asdf1
    testImageSize := 512
    fileHandle := pointer to open file
}
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    testImageFile := "./testimage/image.raw"
    testImageSize := 512
    imageName := fmt.Sprintf(
        "client test %s",
        helper.GenerateRandomID(5),
    )

    fileHandle, err := os.Open(testImageFile)

    uploadResult, err := client.UploadImage(
        alias:           imageName,
        storageDomainID: helper.GetStorageDomainID(),
        sparse:          true,
        size:            testImageSize,
        reader:          fileHandle,
    )

    imageDownload, err := client.DownloadImage (
        diskID: uploadResult.Disk().ID(),
        format: ovirtclient.ImageFormatRaw,
    )
}

func TestImageDownload(t *testing.T) {
    helper := ovirtclient.TestHelper
    client := ovirtclient.Client

    imageName: client_test_asdf1
    testImageFile: ./testimage/image.raw
    uploadResult: ovirtclient.UploadResult
    imageName: client_test_asdf1
    testImageSize: 512
    fileHandle: pointer to open file
    imageDownload: ovirtclient.ImageDownload
    diskID: oVirt Engine identifier for the disk
}
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    testImageFile := "./testimage/image.raw"
    testImageSize := 512
    imageName := fmt.Sprintf(
        "client test %s",
        helper.GenerateRandomID(5),
    )

    fileHandle, err := os.Open(testImageFile)

    uploadResult, err := client.UploadImage(
        alias:           imageName,
        storageDomainID: helper.GetStorageDomainID(),
        sparse:          true,
        size:            testImageSize,
        reader:          fileHandle,
    )

    imageDownload, err := client.DownloadImage(
        diskID: uploadResult.Disk().ID(),
        format: ovirtclient.ImageFormatRaw,
    )

    data, err := ioutil.ReadAll(imageDownload)

}

    helper: ovirtclient.TestHelper
    client: ovirtclient.Client

    imageName: client_test_asdf1

    testImageFile: ./testimage/image.raw

    uploadResult: ovirtclient.UploadResult
    imageName: client_test_asdf1

    testImageSize: 512
    fileHandle: pointer to open file

    imageDownload: ovirtclient.ImageDownload
    diskID: oVirt Engine identifier for the disk

    data: contents of the downloaded image
```

```
func TestImageDownload(t *testing.T) {
    helper := getHelper(t)
    client := helper.GetClient()

    testImageFile := "./testimage/image.raw"
    testImageSize := 512
    imageName := fmt.Sprintf(
        "client test %s",
        helper.GenerateRandomID(5),
    )

    fileHandle, err := os.Open(testImageFile)

    uploadResult, err := client.UploadImage(
        alias:           imageName,
        storageDomainID: helper.GetStorageDomainID(),
        sparse:          true,
        size:            testImageSize,
        reader:          fileHandle,
    )

    imageDownload, err := client.DownloadImage(
        diskID: uploadResult.Disk().ID(),
        format: ovirtclient.ImageFormatRaw,
    )

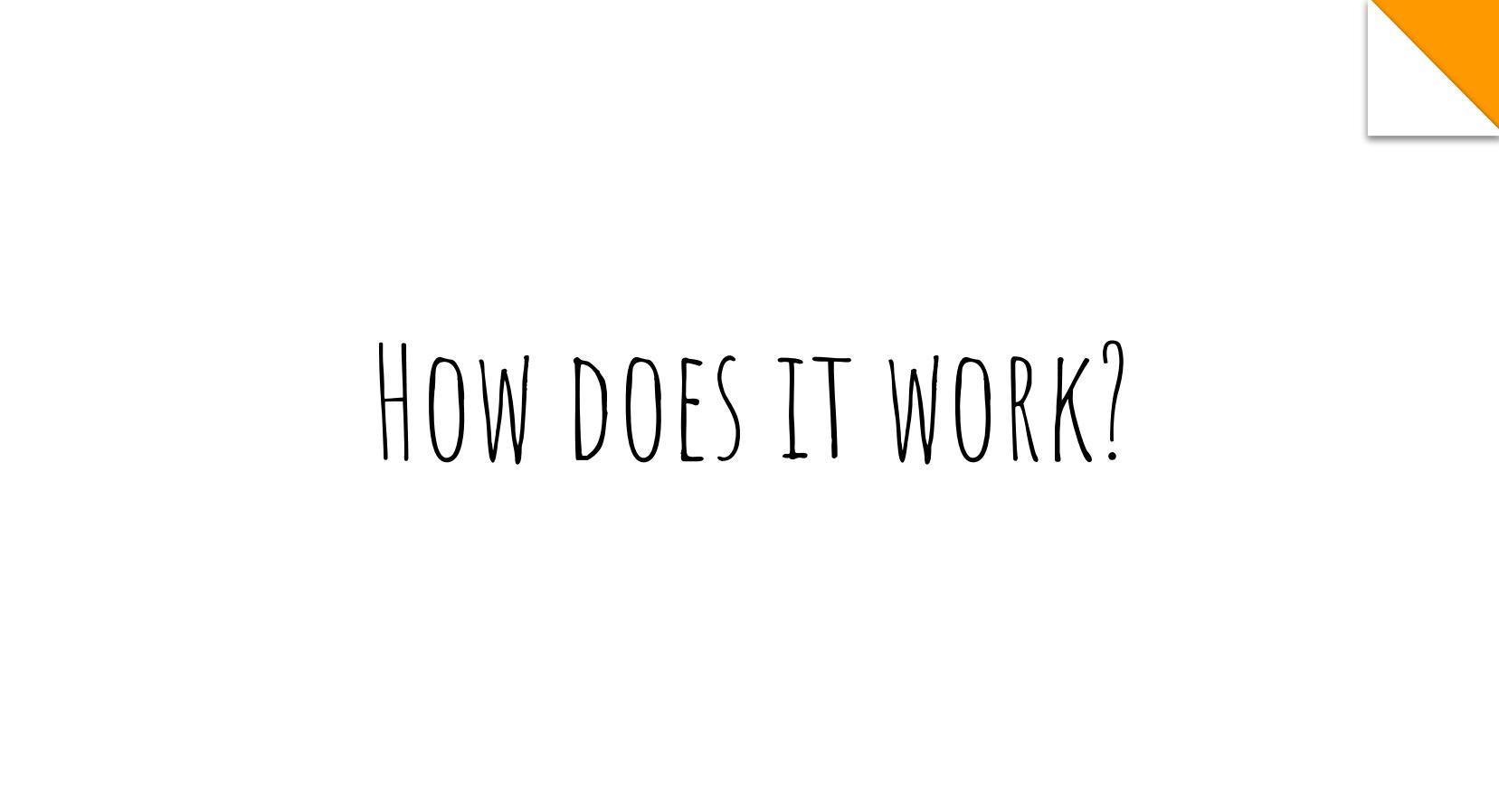
    data, err := ioutil.ReadAll(imageDownload)

    if !bytes.Equal(data[:len(testImageData)], testData) {
        t.Fatal(fmt.Errorf("the downloaded image did not match the original upload"))
    }
}

helper: ovirtclient.TestHelper
client: ovirtclient.Client

imageName: client_test_asdf1
testImageFile: ./testimage/image.raw
uploadResult: ovirtclient.UploadResult
imageName: client_test_asdf1
testImageSize: 512
fileHandle: pointer to open file

imageDownload: ovirtclient.ImageDownload
diskID: oVirt Engine identifier for the disk
data: contents of the downloaded image
```



HOW DOES IT WORK?

```
type Client interface {
    DiskClient
    VMClient
    NICClient
    VNICProfileClient
    NetworkClient
    DatacenterClient
    ClusterClient
    StorageDomainClient
    HostClient
    TemplateClient
    TestConnectionClient
}
```

```
func NewMock() Client {  
}  
}
```

```
func NewMock() Client {
    testCluster := generateTestCluster()
    testHost := generateTestHost(testCluster)
    testStorageDomain := generateTestStorageDomain()

    // ...
}
```

```
func New(  
) (Client, error) {  
}
```

```
func New(  
    url string,  
    username string,  
    password string,  
    tls TLSProvider,  
    logger Logger,  
) (Client, error) {  
    // ...  
}
```

```
type Client interface {
    DiskClient
    VMClient
    NICClient
    VNICProfileClient
    NetworkClient
    DatacenterClient
    ClusterClient
    StorageDomainClient
    HostClient
    TemplateClient
    TestConnectionClient
}
```

```
type DiskClient interface {  
}  
}
```

```
type DiskClient interface {
    UploadImage(
        ) (UploadImageResult, error)
}

}
```

```
type DiskClient interface {
    UploadImage(
        alias string,
        storageDomainID string,
        sparse bool,
        size uint64,
        reader readSeekCloser,
        retries ...RetryStrategy,
    ) (UploadImageResult, error)
}

}
```

```
type DiskClient interface {
    UploadImage(
        alias string,
        storageDomainID string,
        sparse bool,
        size uint64,
        reader readSeekCloser,
        retries ...RetryStrategy,
    ) (UploadImageResult, error)
}

}
```



HOLD YOUR HORSES! ;)

```
type DiskClient interface {
    UploadImage(
        alias string,
        storageDomainID string,
        sparse bool,
        size uint64,
        reader readSeekCloser,
        retries ...RetryStrategy,
    ) (UploadImageResult, error)

    DownloadImage(
        diskID string,
        format ImageFormat,
        retries ...RetryStrategy,
    ) (ImageDownloadReader, error)
}
```

```
type Disk interface {  
}  
}
```

```
type Disk interface {
    ID() string
    Alias() string
    ProvisionedSize() uint64
    TotalSize() uint64
    Format() ImageFormat
    StorageDomainID() string
    Status() DiskStatus
}
```

RLY?



PHOTO BY [MATTHEW HENRY](#) ON [UNSPLASH](#)

```
type Disk struct {
    ID          string
    Alias       string
    ProvisionedSize uint64
    Format      ImageFormat
    StorageDomainID string
    Status       DiskStatus
    TotalSize    uint64
}
```

```
type diskWithData struct {
    disk
    data []byte
}
```

```
type Disk interface {
    ID() string
    Alias() string
    ProvisionedSize() uint64
    TotalSize() uint64
    Format() ImageFormat
    StorageDomainID() string
    Status() DiskStatus

    Download(
        format ImageFormat,
        retries ...RetryStrategy,
    ) (ImageDownloadReader, error)
}
```

```
type disk struct {
    client           Client
    id               string
    alias            string
    provisionedSize uint64
    format           ImageFormat
    storageDomainID string
    status            DiskStatus
    totalSize         uint64
}
```

```
func (d disk) Download(  
    format ImageFormat,  
    retries ...RetryStrategy,  
) (ImageDownloadReader, error) {  
  
}  
}
```

```
func (d disk) Download(
    format ImageFormat,
    retries ...RetryStrategy,
) (ImageDownloadReader, error) {
    return d.client.DownloadImage(
        d.id,
        format,
        retries...,
    )
}
```

RETRIES

```
func (o *oVirtClient) GetDisk(  
    id string,  
    retries ...RetryStrategy,  
) (result Disk, err error) {  
  
}  
}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
}

}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
    err = retry(
        func() error {
            return o.VirtClient.Disk.Get(id)
        },
        &retryOptions{
            Retries:     retries,
            MaxWait:    10 * time.Second,
            MaxAttempts: len(retries),
        },
    )
    if err != nil {
        return result, err
    }
    return result, nil
}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
    err = retry(
        fmt.Sprintf("getting disk %s", id),
    )
}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
    err = retry(
        fmt.Sprintf("getting disk %s", id),
        o.logger,
    )
}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
    err = retry(
        fmt.Sprintf("getting disk %s", id),
        o.logger,
        retries,
    )

}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
    err = retry(
        fmt.Sprintf("getting disk %s", id),
        o.logger,
        retries,
        func() error {
            // ...
        } )
}
```

```
func (o *oVirtClient) GetDisk(
    id string,
    retries ...RetryStrategy,
) (result Disk, err error) {
    retries = defaultRetries(retries, defaultReadTimeouts())
    err = retry(
        fmt.Sprintf("getting disk %s", id),
        o.logger,
        retries,
        func() error {
            // ...
        })
    return result, err
}
```

BENEFITS

1. SIMPLER CLIENT CODE

BENEFITS

1. SIMPLER CLIENT CODE
2. OFFLINE TESTING / MOCKING

BENEFITS

1. SIMPLER CLIENT CODE
2. OFFLINE TESTING / MOCKING
3. LESS DUPLICATED CODE

BENEFITS

1. SIMPLER CLIENT CODE
2. OFFLINE TESTING / MOCKING
3. LESS DUPLICATED CODE
4. DOCUMENTATION!



Search for a package



Why Go

Get Started

Docs

Package

Discover Packages > [github.com/ovirt/go-ovirt-client](#)

ovirtclient

package

module

Version: v0.6.0 Published: Aug 23, 2021 | License: [BSD-3-Clause](#) | Imports: 23 | Imported by: 0

Jump to ...



README

README

[Using this library](#)

[Test helper](#)

[Retries](#)

[Mock client](#)

[FAQ](#)

[Documentation](#)

[Source Files](#)

[Directories](#)

goVirt: an easy-to-use overlay for the oVirt Go SDK

This library is early in development.

This library is early in development, and the API may change at any time until version 1.0.0. We hope to stabilize the API soon, providing the core functionality on an as-needed basis. If you need an API integrated, please open an issue.

This library provides an easy-to-use overlay for the automatically generated [Go SDK for oVirt](#). It does not replace the Go SDK. It implements the functions of the SDK only partially and is primarily used by the [oVirt Terraform provider](#).

Using this library

[Expand ▾](#)

Documentation

Overview

[Reading this documentation](#)

[Creating a client](#)

[Mock client](#)

[Test helper](#)

[Logging](#)

[TLS verification](#)

[Retries](#)

Package ovirtclient provides a human-friendly Go client for the oVirt Engine. It provides an abstraction layer for the oVirt API, as well as a mocking facility for testing purposes.

Details

Valid [go.mod file](#)

Redistributable license

Tagged version

Stable version

[Learn more](#)

Repository

github.com/ovirt/go-ovirt-client

MORE CLEAN CODE

**The Clean Code Talks
Unit Testing**

**Miško Hevery
October 30, 2008**



**The Clean Code Talks
"Don't Look For Things!"**

**Miško Hevery
November 6, 2008**



GITHUB.COM/OVIRT/GO-OVIRT-CLIENT

GITHUB.COM/OVIRT/GO-OVIRT-CLIENT

DEBUGGED.IT/TAGS/GOLANG/