





Ganeti

Ganeti Core Team - Google
LISA '13 - 5 Nov 2013



Ganeti ExtStorage Interface

- Guido Trotter <ultrotter@google.com>
- Helga Velroyen <helgav@google.com>
- Slides contributed by Constantinos Venetsanopoulos <cven@grnet.gr>

Latest version of these slides

Please find the latest version of these slides at:

<https://code.google.com/p/ganeti/wiki/LISA2013>

State before the ExtStorage Interface

- Non-mirrored templates: `plain`, `file`
- Internally mirrored templates: `drbd`
- Externally mirrored templates: `sharedfile`, `rbd`, `blockdev`, `diskless`

Ganeti and external SAN/NAS appliances

- Instance disks residing inside an external SAN/NAS appliance visible by all Ganeti nodes (e.g. NetApp, EMC, IBM)
- Instances should be able to migrate/failover to any node that can access the appliance.
- Ganeti should integrate with external SAN/NAS appliances in a generic way, independent of the appliance itself in the easiest possible way from the admin's perspective.

Introducing the 'ExtStorage Interface'

- A simple interface inspired by the Ganeti OS interface
- To plug an appliance to Ganeti, we need a corresponding 'ExtStorage provider' which is a set of scripts residing under a directory.
- e.g. `/usr/share/ganeti/extstorage/provider1/`

ExtStorage provider methods

Every ExtStorage provider should provide the following methods:

- Create a disk on the appliance
- Remove a disk from the appliance
- Grow a disk on the appliance
- Attach a disk to a given Ganeti node
- Detach a disk from a given Ganeti node
- SetInfo on a disk (add metadata)
- Verify the provider's supported parameters

ExtStorage provider scripts

The methods are implemented in the corresponding 7 executable scripts, using appliance-specific tools:

```
# ls -l /usr/share/ganeti/extstorage/provider1
```

```
create  
remove  
grow  
attach  
detach  
setinfo  
verify
```

attach returns a block device path on success

Input via environment variables, e.g. VOL_NAME, VOL_SIZE

The new 'ext' template

- Introduce a new externally mirrored disk template: **ext**
- Introduce a new disk option: **provider**

Using the interface

Example

Assuming two appliances visible by a Ganeti cluster and their two ExtStorage providers installed on all Ganeti nodes:

```
/usr/share/ganeti/extstorage/emc/*  
/usr/share/ganeti/extstorage/ibm/*  
  
# gnt-instance add -t ext --disk=0:size=2G,provider=emc  
  
# gnt-instance add -t ext --disk=0:size=2G,provider=emc \  
                    --disk=1:size=1G,provider=emc \  
                    --disk=2:size=10G,provider=ibm  
  
# gnt-instance modify --disk 3:add,size=20G,provider=ibm  
  
# gnt-instance migrate testvm1  
  
# gnt-instance migrate n- nodeX.example.com testvm1
```

ExtStorage Interface dynamic parameters

Support for dynamic passing of arbitrary parameters to ExtStorage providers during instance creation/modification per-disk:

```
# gnt-instance add -t ext --disk=0:size=2G,provider=emc,param1=value1,param2=value2  
                        --disk=1:size=10G,provider=ibm,param3=value3,param4=value4
```

```
# gnt-instance modify --disk 2:add,size=3G,provider=emc,param5=value5
```

The above parameters will be exported to the ExtStorage provider's scripts as environment variables:

```
EXTP_PARAM1 = str(value1)  
EXTP_PARAM2 = str(value2)  
...
```

The new 'gnt-storage' client

Inspired by `gnt-os`:

```
# gnt-storage diagnose  
# gnt-storage info
```

Docs on the ExtStorage interface

`man gnt-extstorage-interface`

`man gnt-storage`

`man gnt-instance {add,modify,migrate,failover}`

Design doc: `design-shared-storage.rst`

Thank You!

Questions?

Survey at <https://www.usenix.org/lisa13/training/survey>



- © 2010 - 2013 Google
- Use under GPLv2+ or CC-by-SA

