
Arrikto

# STORAGE REFACTORING AND RELATED HYPERVISOR ENHANCEMENTS.

DIMITRIS ARAGIORGIS
VANGELIS KOUKIS
ARRIKTO INC.

# About Us

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

We Arr Arrikto.

- Stealth-mode storage startup
- Building an innovative storage product from scratch
- Offices: Silicon Valley, Athens

- You'll be hearing details soon, but not in this GanetiCon ☺

# About Us

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

## Who we were

- Created the Synnefo cloud management platform
- Heavy Ganeti contributors
  - Ceph RBD support
  - ExtStorage interface
  - Current Ganeti networking
  - UUID-and-name for all objects
  - Disk and NIC hotplugging
  - Disks as top-level objects

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Overview

## KVMHypervisor

- Old hotplug and SCSI support
- QEMU device model
- New enhancements

## Storage refactoring

- Current status
- Proposed extension

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Old hotplug support

## Shortcomings

- Works only with paravirtualized devices
  - virtio-net-pci, virtio-blk-pci
- Only knows about the PCI bus

## Problem

- A hot-added disk will **always** be a virtio disk

dimara@arrikto.com
vkoukis@arrikto.com

# Old SCSI support

Using the "`scsi`" `disk_type` hvparam

- `disk_type=scsi` ⇒ `-drive file=<path>,if=scsi`

Shortcomings

- Can only use QEMU-based SCSI disk emulation
- SCSI disks on single, implicit SCSI controller (QEMU default)
- No hotplug support
- No SCSI-passthrough support

Arrikto

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Old QEMU device model

The `-drive` and `-net` options

- NIC: `-net nic,model=virtio -net tap`
- Disk: `-drive file=<path>,if=virtio`

Shortcomings

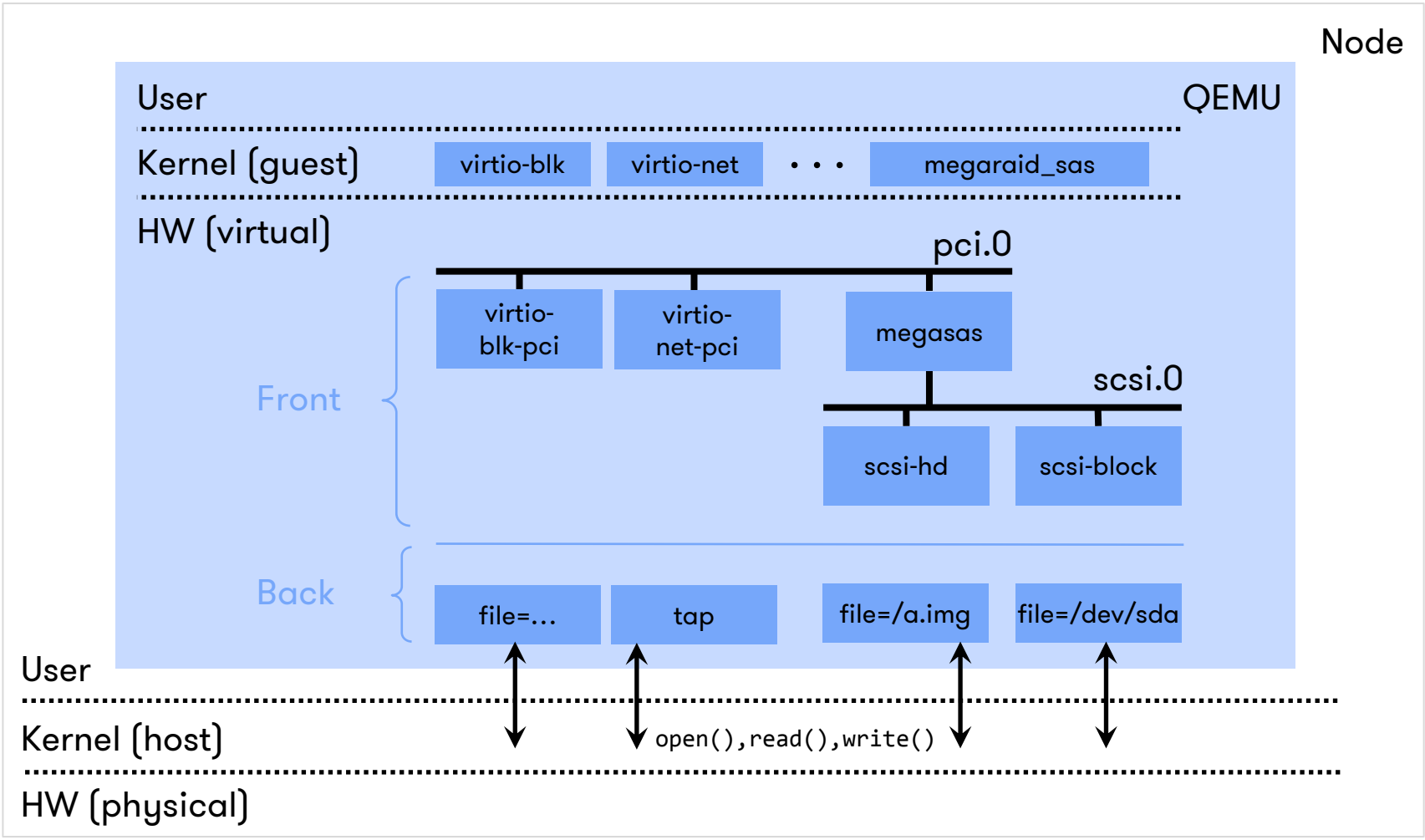
- Deprecated in recent QEMU versions
- Very narrow selection of device types
- Devices have no explicit IDs in the QEMU device tree
- There is no way to manage the PCI, SCSI buses

Use the `-device` option instead!

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# New QEMU device model



GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# New QEMU device model

Paravirtual devices

- Front: `-device virtio-blk-pci,id=disk0,bus=pci.0,addr=0x6,drive=drive0`
- Back: `-drive file=...,if=none,id=drive0`

Explicit SCSI controller

- Front: `-device virtio-scsi-pci,id=scsi`

SCSI passthrough

- Front: `-device scsi-block,id=disk1,drive=drive1,bus=scsi.0,channel=0,scsi-id=1,lun=0`
- Back: `-drive file=/dev/sdb,if=none,id=drive1`

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Hotplug refactoring

Take advantage of the latest QEMU device model

Support all paravirtual and SCSI disk types

- virtio-blk-pci
- scsi-hd, scsi-cd, scsi-block, scsi-generic

How?

- All devices get an ID (based on Ganeti's UUID)
- All devices placed explicitly on a specific bus (e.g. `pci.0`)
- All info needed for `-device` stored in runtime file (`hvinfo` attr)

dimara@arrikto.com
vkoukis@arrikto.com

# QEMU buses

Let Ganeti manage allocations on buses explicitly. Why?

PCI bus

- QEMU creates one by default (`pci.0`)
- First 16 slots left to QEMU-managed devices (e.g., balloon, spice)
- Ganeti adds devices (disks, NICs) from the 17$^{th}$ slot onwards

SCSI bus

- Created by a SCSI controller that Ganeti adds (`-device megasas`)
- Ganeti uses a separate target per disk [`0:0:X:0`]

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Overview

## KVMHypervisor

- Old hotplug and SCSI support
- QEMU device model
- New enhancements

## Storage refactoring

- Current status
- Proposed extension

dimara@arrikto.com
vkoukis@arrikto.com

# Missing features

## Adoption

- Works only for blockdev (no-op) and plain (rename)

## Disk handling

- Cannot manage Disk objects separately
- clone, snapshot, copy

dimara@arrikto.com
vkoukis@arrikto.com

# Goals

Disks as stand-alone top-level objects: **Done!**

bdev

- Push template-specific logic down to bdev
- Handle adoption for all disk templates
- Support `Snapshot()` and `Clone()` methods

cmdlib

- Separate LUs for Disk handling (gnt-disk)
- LUDiskSnapshot/LUDiskClone should create new Disk objects
- Support copying of disks between templates (generalize conversion)

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Refactoring

## Adoption

```
bdev.Create(disk, adopt)

@type adopt: string
@param adopt: Interpreted by code of specific template
```

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Refactoring

## Clone and Snapshot

```
r_bdev = _RecursiveFindBD(src_disk)
r_bdev.Clone(dst_disk)
r_bdev.Snapshot(dst_disk)

@type dst_disk: L{objects.Disk}
@param dst_disk: The resulting disk object
```

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Refactoring

gnt-disk

- Implement `LUDisk*` that map one-to-one with bdev
- Create, Remove, Snapshot, Clone, Modify

Arrikto

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Toward disk-template unification

Long shot

- ExtStorage-ization of all disk templates

Gain

- Get rid of DRBD-specific logic inside cmdlib

Blocker

- Primary nodes cannot talk directly to secondaries


Arrikto

GanetiCon 2015

dimara@arrikto.com
vkoukis@arrikto.com

# Thanks!

http://www.arrikto.com