Vdsm

Virtual Desktop and Server Management Daemon
KVM Forum 2011, Vancouver

Red Hat

Dan Kenigsberg

August 15, 2011

Agenda

1 Vdsm is Free
- What is Vdsm
- Responsibilities
- Why Vdsm
- Architecture
- API
- Storage Architecture
- Storage API
- Thin Provisioning
- Bright Future

- Vdsm is a node management API.

- High level API for managing a hypervisor node.

- Abstracts low level details of underlying Linux environments

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

What is Vdsm?

- Vdsm is a node management API.
- High level API for managing ahypervisor node.
- Abstracts low level details of underlying Linux environments

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

What is Vdsm?

- Vdsm is a node management API.
- High level API for managing ahypervisor node.
- Abstracts low level details of underlying Linux environments
  - Today: RHEL-5 RHEL6 RHEV-H
  - Tomorrow: Fedora and other Linux distros (patches welcome)

What is Vdsm?

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

- Vdsm is a node management API.
- High level API for managing ahypervisor node.
- Abstracts low level details of underlying Linux environments
  - Today: RHEL-5 RHEL6 RHEV-H
  - Tomorrow: Fedora and other Linux distros (patches welcome)

What is Vdsm?

Vdsm

Dan Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

- Vdsm is a node management API.
- High level API for managing ahypervisor node.
- Abstracts low level details of underlying Linux environments
    - Today: RHEL-5 RHEL6 RHEV-H
    - Tomorrow: Fedora and other Linux distros (patches welcome)

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Vdsm Responsibilities

- VM lifecycle (libvirt + guest agent).
- Host bootstrap/registration.
- Monitoring host and VMs.
- Network configuration.
- Storage management.
- Policy management
    - scheduler
    - ksm
    - thin provisioning ("a running VM should keep running")
    - *storage/network/cpu QoS*
    - *balooning*.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Vdsm Responsibilities

- VM lifecycle (libvirt + guest agent).

- Host bootstrap/registration.

- Monitoring host and VMs.

- Network configuration.

- Storage management.

- Policy management

    - scheduler
    - ksm
    - thin provisioning ("a running VM should keep running")
    - storage/network/cpu QoS
    - balooning.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Vdsm Responsibilities

- VM lifecycle (libvirt + guest agent).

- Host bootstrap/registration.

- Monitoring host and VMs.

- Network configuration.

- Storage management.

- Policy management

    - scheduler

    - ksm

    - thin provisioning ("a running VM should keep running")

    - storage/network/cpu QoS

    - balooning.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Vdsm Responsibilities

- VM lifecycle (libvirt + guest agent).

- Host bootstrap/registration.

- Monitoring host and VMs.

- Network configuration.

- Storage management.

- Policy management

  - scheduler

  - ksm

  - thin provisioning ("a running VM should keep running")

  - storage/network/cpu QoS

  - balooning.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Vdsm Responsibilities

- VM lifecycle (libvirt + guest agent).

- Host bootstrap/registration.

- Monitoring host and VMs.

- Network configuration.

- Storage management.

- Policy management

  - scheduler

  - ksm

  - thin provisioning ("a running VM should keep running")

  - storage/network/cpu QoS

  - balooning.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Vdsm Responsibilities

- VM lifecycle (libvirt + guest agent).

- Host bootstrap/registration.

- Monitoring host and VMs.

- Network configuration.

- Storage management.

- Policy management
  - scheduler
  - ksm
  - thin provisioning ("a running VM should keep running")
  - *storage/network/cpu QoS*
  - *balooning*.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Why Vdsm

- qemu is good:
  $ qemu-kvm & yay! we have a virtual machine (but see the fine print).

```
/usr/libexec/qemu-kvm -S -M rhel6.0.0 -cpu Conroe -enable-kvm -m 2048
-smp 1,sockets=1,cores=1,threads=1 -name z-win7x86-1 -uuid e3e19b36-f6b7-4ab9-b604-1f8b5c47lbda
-smbios type=1,manufacturer=Red Hat,product=RHEL,version=6Server-6.1.0.2.el6_1,serial=50C1C6F0-E
-nodefconfig -nodefaults -chardev socket,id=charmonitor,path=/var/lib/libvirt/qemu/z-win7x86-1.m
-mon chardev=charmonitor,id=monitor,mode=control -rtc base=2011-08-04T06:17:36 -boot
cdn -device virtio-serial-pci,id=virtio-serial0,max_ports=16,bus=pci.0,addr=0x6
-drive file=....
```

- Libvirt is even better:
  To manage more than a handful of virtual machines you
  would need libvirt: virsh, virt-manager.

- Managing a virtualization node has much more than firing
  up a (K)VM.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Why Vdsm

- qemu is good:
  $ qemu-kvm & yay! we have a virtual machine (but see
  the fine print).

```
/usr/libexec/qemu-kvm -S -M rhel6.0.0 -cpu Conroe -enable-kvm -m 2048
-smp 1,sockets=1,cores=1,threads=1 -name z-win7x86-1 -uuid e3e19b36-f6b7-4ab9-b604-1f8b5c471bda
-smbios type=1,manufacturer=Red Hat,product=RHEL,version=6Server-6.1.0.2.el6_1,serial=50C1C6F0-E
-nodefconfig -nodefaults -chardev socket,id=charmonitor,path=/var/lib/libvirt/qemu/z-win7x86-1.m
-mon chardev=charmonitor,id=monitor,mode=control -rtc base=2011-08-04T06:17:36 -boot
cdn -device virtio-serial-pci,id=virtio-serial0,max_ports=16,bus=pci.0,addr=0x6
-drive file=....
```

- Libvirt is even better:
  To manage more than a handful of virtual machines you
  would need libvirt: virsh, virt-manager.

- Managing a virtualization node has much more than firing
  up a (K)VM.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Why Vdsm

- qemu is good:
  $ qemu-kvm & yay! we have a virtual machine (but see
  the fine print).

```
/usr/libexec/qemu-kvm -S -M rhel6.0.0 -cpu Conroe -enable-kvm -m 2048
-smp 1,sockets=1,cores=1,threads=1 -name z-win7x86-1 -uuid e3e19b36-f6b7-4ab9-b604-1f8b5c471bda
-smbios type=1,manufacturer=Red Hat,product=RHEL,version=6Server-6.1.0.2.el6_1,serial=50C1C6F0-E
-nodefconfig -nodefaults -chardev socket,id=charmonitor,path=/var/lib/libvirt/qemu/z-win7x86-1.m
-mon chardev=charmonitor,id=monitor,mode=control -rtc base=2011-08-04T06:17:36 -boot
cdn -device virtio-serial-pci,id=virtio-serial0,max_ports=16,bus=pci.0,addr=0x6
-drive file=....
```

  - Libvirt is even better:
    To manage more than a handful of virtual machines you
    would need libvirt: virsh, virt-manager.

  - Managing a virtualization node has much more than firing
    up a (K)VM.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Why Vdsm

- qemu is good:
  $ qemu-kvm & yay! we have a virtual machine (but see the fine print).

```
/usr/libexec/qemu-kvm -S -M rhel6.0.0 -cpu Conroe -enable-kvm -m 2048
-smp 1,sockets=1,cores=1,threads=1 -name z-win7x86-1 -uuid e3e19b36-f6b7-4ab9-b604-1f8b5c471bda
-smbios type=1,manufacturer=Red Hat,product=RHEL,version=6Server-6.1.0.2.el6_1,serial=50C1C6F0-E
-nodefconfig -nodefaults -chardev socket,id=charmonitor,path=/var/lib/libvirt/qemu/z-win7x86-1.m
-mon chardev=charmonitor,id=monitor,mode=control -rtc base=2011-08-04T06:17:36 -boot
cdn -device virtio-serial-pci,id=virtio-serial0,max_ports=16,bus=pci.0,addr=0x6
-drive file=....
```

- Libvirt is even better:
  To manage more than a handful of virtual machines you would need libvirt: virsh, virt-manager.

- Managing a virtualization node has much more than firing up a (K)VM.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Why Vdsm(-18)

With thousands of machines, a cluster of hundreds of hosts,
multiple storage exports — what to do?

- One solution is Red Hat Enterprise Virtualization (RHEV).

- Vdsm is its node agent, tailored for its needs.

- Agile, KVM-specific.

- RHEV is being open sourced.

- Vdsm is moving to a more general use case, applicable to
  other management platforms.

Why Vdsm(-18)

With thousands of machines, a cluster of hundreds of hosts,
multiple storage exports — what to do?

- One solution is Red Hat Enterprise Virtualization (RHEV).

- Vdsm is its node agent, tailored for its needs.

- Agile, KVM-specific.

- RHEV is being open sourced.

- Vdsm is moving to a more general use case, applicable to
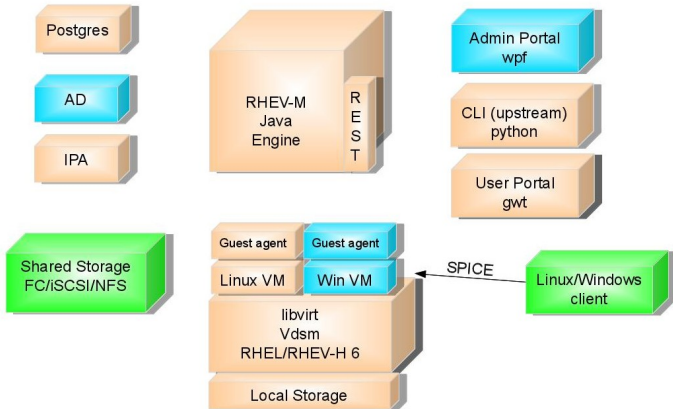 other management platforms.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Why Vdsm(-18)

With thousands of machines, a cluster of hundreds of hosts,
multiple storage exports — what to do?

- One solution is Red Hat Enterprise Virtualization (RHEV).

- Vdsm is its node agent, tailored for its needs.

- Agile, KVM-specific.

- RHEV is being open sourced.

- Vdsm is moving to a more general use case, applicable to
  other management platforms.

Why Vdsm(-18)

With thousands of machines, a cluster of hundreds of hosts, multiple storage exports — what to do?

- One solution is Red Hat Enterprise Virtualization (RHEV).

- Vdsm is its node agent, tailored for its needs.

- Agile, KVM-specific.

- RHEV is being open sourced.

- Vdsm is moving to a more general use case, applicable to other management platforms.

Why Vdsm(-18)

With thousands of machines, a cluster of hundreds of hosts,
multiple storage exports — what to do?

- One solution is Red Hat Enterprise Virtualization (RHEV).

- Vdsm is its node agent, tailored for its needs.

- Agile, KVM-specific.

- RHEV is being open sourced.

- Vdsm is moving to a more general use case, applicable to
  other management platforms.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Place within oVirt

Freeing Vdsm is another step in freeing RHEV-M within the oVirt framework.

Architecture and Implementation

By Daniel Schwen, from Wikipedia

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Architecture and Implementation

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...

Architecture and Implementation

- Written in Python.

- Multithreaded, Multi-processes.

- Used to speak directly to qemu, now uses libvirt for that.

- Speaks with its guest agent via virtio-serial.

- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...

Architecture and Implementation

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...

Architecture and Implementation

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Architecture and Implementation

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...
  - Maintains that on a multihost system, only one concurrent metadata writer exists.
  - Scales linearly in data writers.
  - Image-aware API. (import, templates, qcow, raw)

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Architecture and Implementation

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...
  - Maintains that on a multihost system, only one concurrent metadata writer exists.
  - Scales linearly in data writers.
  - Image-aware API. (import, templates, qcow, raw)

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Architecture and Implementation

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...
    - Maintains that on a multihost system, only one concurrent metadata writer exists.
    - Scales linearly in data writers.
    - Image-aware API. (import, templates, qcow, raw)

Architecture and Implementation

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

- Written in Python.
- Multithreaded, Multi-processes.
- Used to speak directly to qemu, now uses libvirt for that.
- Speaks with its guest agent via virtio-serial.
- Implements a distributed image repository over the supported storage types: Local directory, FCP, FCoE, iSCSI, NFS...
    - Maintains that on a multihost system, only one concurrent metadata writer exists.
    - Scales linearly in data writers.
    - Image-aware API. (import, templates, qcow, raw)

Vdsm

Dan Kenigsberg

Robustness as a design goal:

- Evaporated NFS exports.

- Faulty multipaths.

- Node crash.

- Live-locked qemu.

- Internal Python exceptions.

- Self-fencing of metadata writer.

Robustness as a design goal:

- Evaporated NFS exports.
- Faulty multipaths.
- Node crash.
- Live-locked qemu.
- Internal Python exceptions.
- Self-fencing of metadata writer.

Vdsm

Dan
Kenigsberg

Robustness as a design goal:

- Evaporated NFS exports.

- Faulty multipaths.

- Node crash.

- Live-locked qemu.

- Internal Python exceptions.

- Self-fencing of metadata writer.

Robustness as a design goal:

- Evaporated NFS exports.

- Faulty multipaths.

- Node crash.

- Live-locked qemu.

- Internal Python exceptions.

- Self-fencing of metadata writer.

Vdsm

Dan
Kenigsberg

Robustness as a design goal:

- Evaporated NFS exports.

- Faulty multipaths.

- Node crash.

- Live-locked qemu.

- Internal Python exceptions.

- Self-fencing of metadata writer.

Vdsm

Dan
Kenigsberg

Robustness as a design goal:

- Evaporated NFS exports.

- Faulty multipaths.

- Node crash.
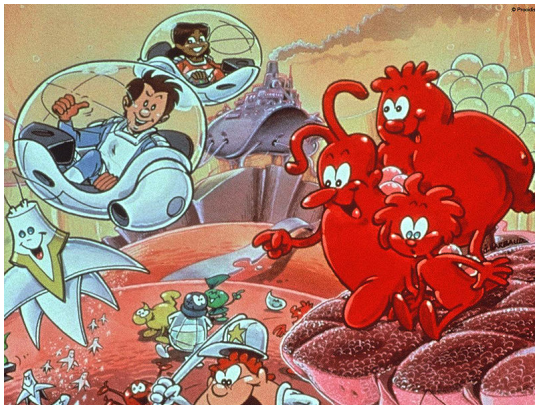
- Live-locked qemu.

- Internal Python exceptions.
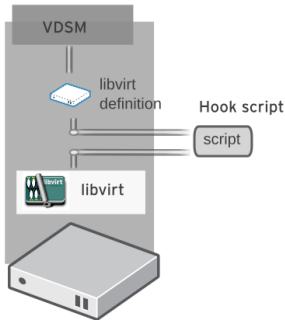
- Self-fencing of metadata writer.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Host Monitoring API

- getVdsCapabilities

- getVdsStats

- ping

- fenceNode

Il était une fois... la Vie, Albert Barillé

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

VM Lifecycle API

- create

- destroy

- pause

- continue

- setVmTicket

- changeCD

- changeFloppy

- migrate (downtime, timeout)

- hibernate

- hooks

VM Lifecycle API (agent-dependent)

- shutdown
- desktopLogin
- desktopLogoff
- desktopLock

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

VM Monitoring API

- list

- getAllVmStats

- getVmStats

  - Interesting applications installed
  - Logged in users
  - CPU consumption
  - Memory usage
  - ...

Network Configuration API
(agent-dependent)

- addNetwork
- delNetwork
- editNetwork
- setSafeNetworkConfig
- setupNetworks
    - connectivityCheck

Async Tasks API

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

- getAllTasksStatuses

- getTaskStatus

- clearTask

- revertTask

- stopTask

(currently used only by the storage subsystem)

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
**Storage
Architecture**
Storage API
Thin
Provisioning
Bright Future

Storage Architecture

- A *Storage Domain* is one of: Local directory, NFS export, VG over iSCSI LUNs, VG over FCP LUNs. Vdsm (tries to) abstract the differences away.

- A *Storage Pool* aggregates several storage domains. We try to remove this concept from the Vdsm level.

- Single meta data writer:
  - SPM lease mechanism (Chockler and Malkhi 2004, Light-Weight Leases for Storage-Centric Coordination).
  - Storage-centric mailbox.

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

Storage API

- connectStorageServer

- getDeviceList

- createStorageDomain

- attachStorageDomain

- createImage

- prepareVolume

-

-

- (and many, many more)
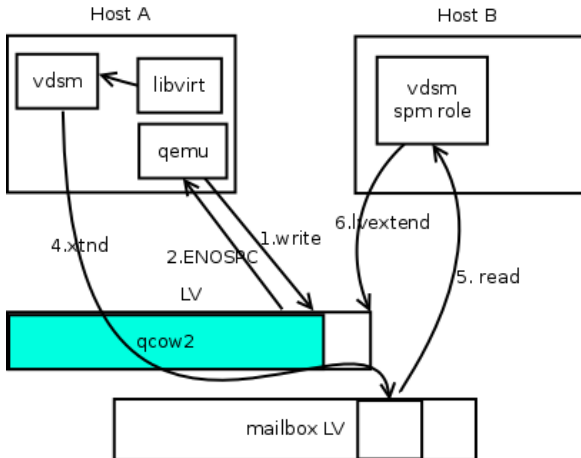
- repoStats

- getSpmStatus

- spmStart

- extendVolume

- Disk image is created as a big empty qcow2 file within a small LV.
- When the guest writes, qcow2 grows.
- If qcow2 file fills the LV, guest stops.
- Local Vdsm senses that, and issues an extend request
- SPM extends LV, and returns success message
- Local Vdsm continue guest execution.

Future

Vdsm

Dan
Kenigsberg

Vdsm is Free
What is Vdsm
Responsibilities
Why Vdsm
Architecture
API
Storage
Architecture
Storage API
Thin
Provisioning
Bright Future

- New, stable API. RESTful(?), similar to that of rhevm-api
    - Current API is not very clean (createVG, createStorageDomain)
- Split Vdsm up to reusable autonomous parts.
    - Spin off as a generic image repository.
- SANLock, Vepa, VN-Link, storage network, storage migration, ??
- Other use cases:
    - Single-host
    - Linux containers?

Contribution

- vdsm-devel@fedorahosted.org

- http://git.fedorahosted.org/git/?p=vdsm.git

- Ping us on #vdsm at irc.freenode.org:
  Saggi Mizrahi, Federico Simoncelli, Igor Lvovsky, Eduardo
  Warszawasky, Yotam Oron, Ayal Baron, yours truly.

- Come over to see our demo at Andy's!

The end.

Thanks for listening.