

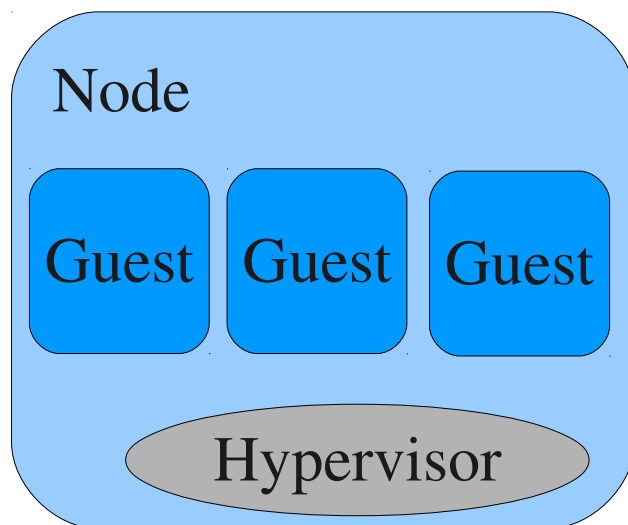


# **Libvirt presentation and perspectives**

Daniel Veillard  
veillard@redhat.com

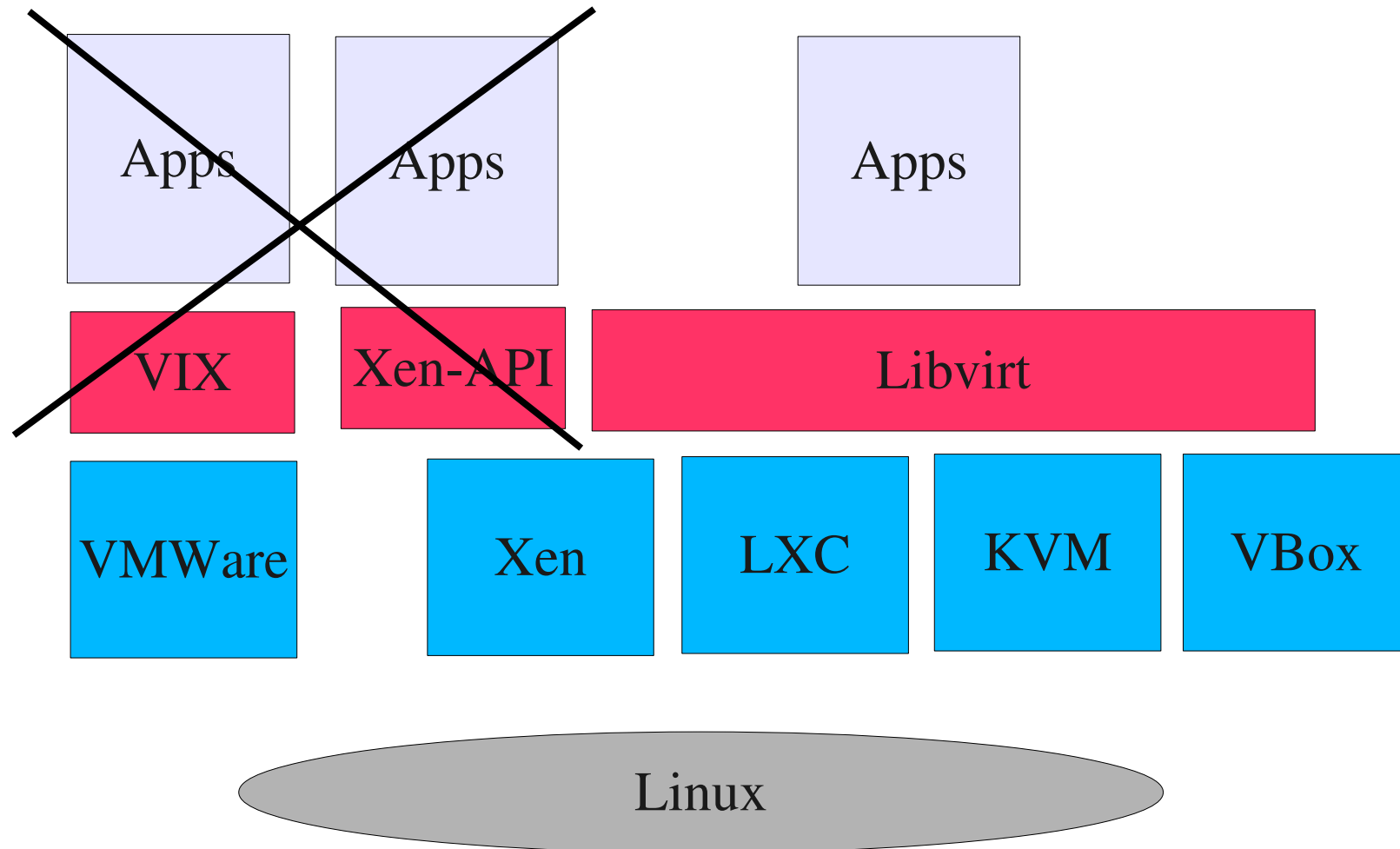
# Libvirt project goals

- Web site: [libvirt.org](http://libvirt.org)
- Virtualization library: manage guest on one node
- Share the application stack between hypervisors
- Long term stability and compatibility of API and ABI
- Provide security and remote access “out of the box”
- Expand to management APIs (Node, Storage, Network)



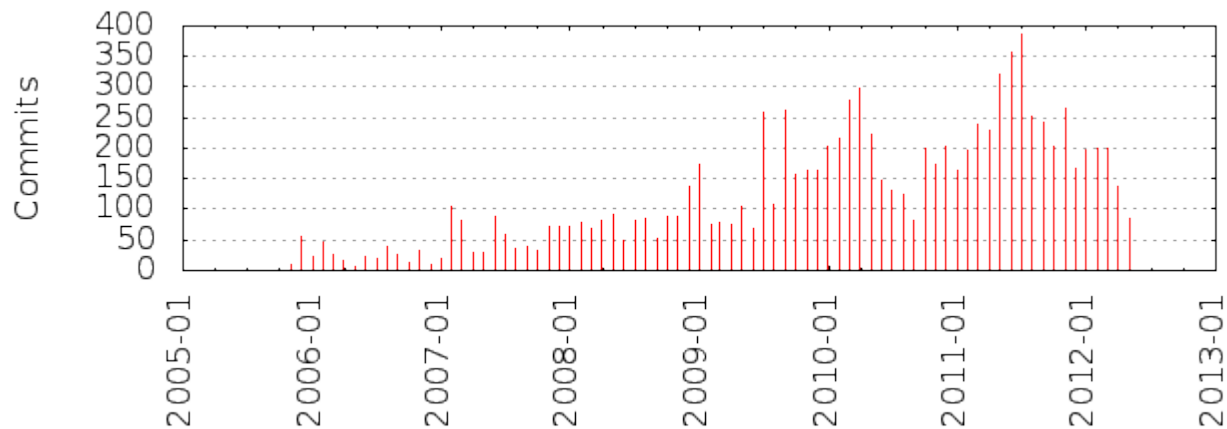
# Limit duplication of efforts

Applications are costly to write and maintain !



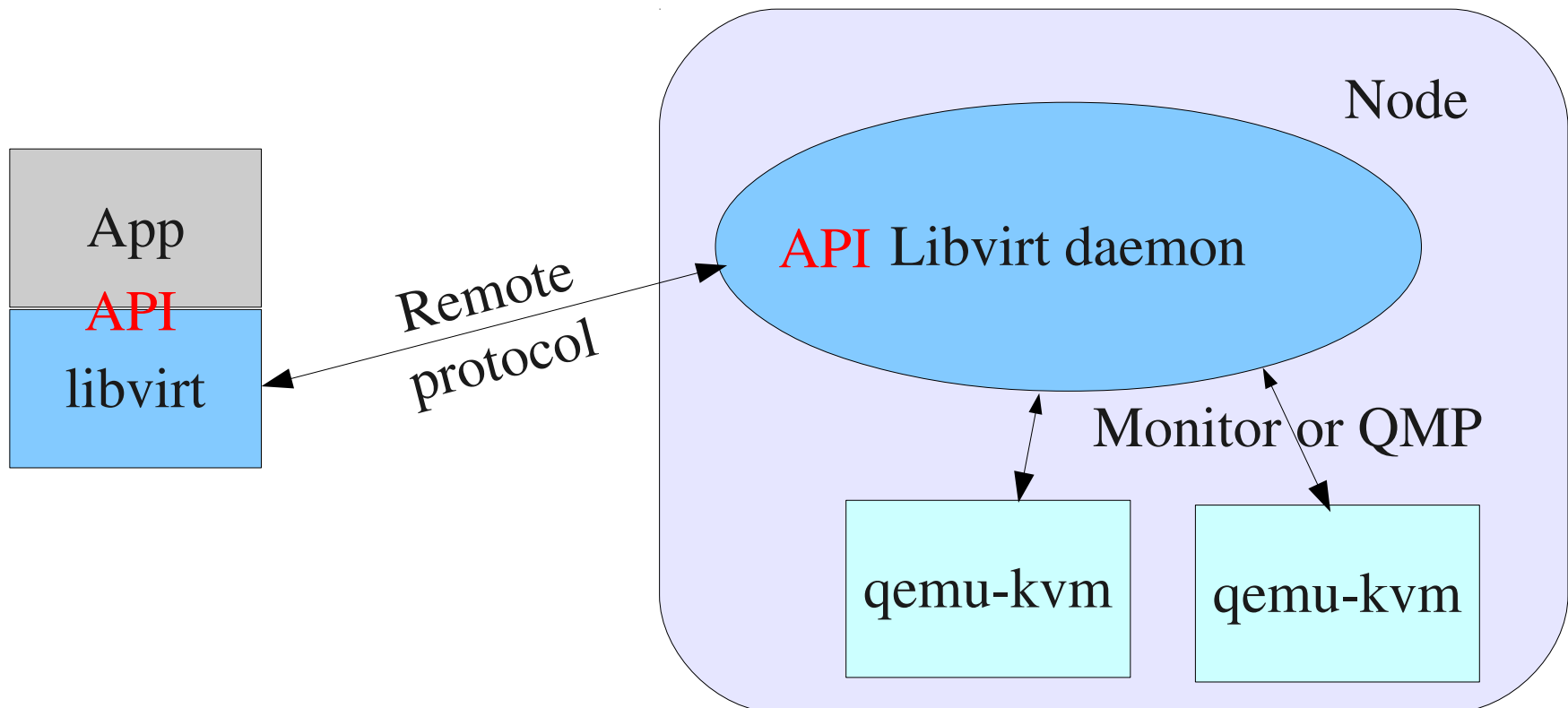
# Project current status

- Started 2005
  - 25 committers, 10 full time Red Hat persons
  - [Active list libvir-list@redhat.com](mailto:libvir-list@redhat.com)
  - A release every month
  - 200-300+ commits/month, 0.9.12
- Support for most hypervisors



# Architecture of libvirt

- Application links to the library
- The libvirt daemon talks to the hypervisor on the node
- Remote protocol access is secure



## Current set of APIs (libvirt.h)

See the [hypervisor support page](#) for the full list

- 1) Domain state handling (save, restore, migration, core...)
- 2) Node and guests resource usage (memory, network, disk)
- 3) Security, audit and credential handling
- 4) Domain control (define, create, shutdown...)
- 5) Tuning (scheduler, memory, I/O, vcpu)

## **Current set of APIs (continued)**

- 5) NUMA support (placement, topology, cells usage, pinning, automatic placement)
- 6) Dynamic or cold device attach and removal
- 7) Networking (virtual network, interfaces, filtering)
- 8) Storage handling (pools and volume)
- 9) Devices handling (enumeration, attach, detach, reset)
- 10) Asynchronous events callbacks

## Since last year ...

- 1) Hypervisors: vSphere5, VBox4.1, LXC, Xen updates, PPC Qemu, Hyper-V (basic)
- 2) Network: OpenVSwitch, filtering, IP snooping
- 3) APIs: guest control, I/O control, screenshot, migration, new events, reset/NMI, stats, audit
- 4) guests: USB2.0, guest agent, S3/S4, reboot
- 5) storage: sheepdog, block and snapshots, formatting, lock management

- 1) Desktop integration: [Boxes](#) , [virt-sandbox](#)
- 2) [Openstack](#) integration improvements
- 3) [Ovirt](#) was released



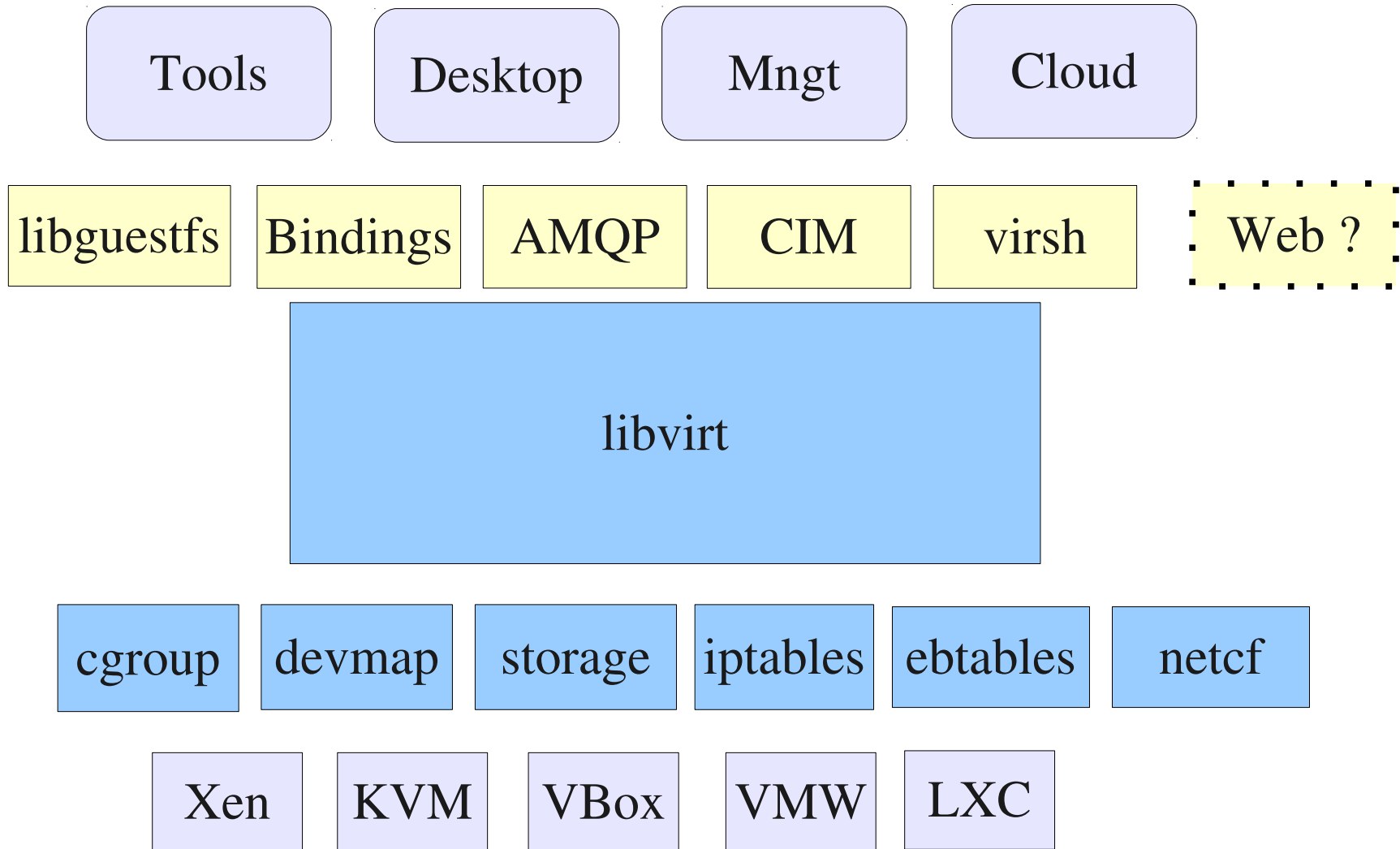
## On the work and TODO

- Fine grained ACL for access control
- Snapshot API and block migration (live/offline)
- Parallels driver
- Storage driver improvements
- Split the daemon into multiple processes
- Solve the problem with iptables :-)
- more dynamic operation (live plug/unplug)
- More LXC improvements
- Platform improvements (PPC, ARM64 ...)
- Someone finish Hyper-V support ?

# Classic libvirt applications

- **Virt-manager** (graphical GUI):
  - manage guests on a few hosts
  - Xen, Qemu/KVM support, LXC coming
- **Gnome Boxes**
  - direct desktop integration
- **Virsh** (CLI for libvirt)
- **Libguestfs**:
  - Read/modify guest disk images
  - Guestfish shell client
  - API with many bindings

# Libvirt stack overview



# Cloud engines relationship

- They nearly all use libvirt hypervisor support:
  - OpenStack, Eucalyptus, OpenNebula, Nimbus ...
- Usually their use of libvirt is limited to:
  - hypervisor support: KVM/Xen/LXC
  - minimal networking and storage settings
- Libvirt APIs and cloud APIs are at a different level
  - OpenNebula attempt
  - cloud API assume a number of design decisions
  - libvirt does not make policy decisions
  - libvirt APIs are focusing on one node
  - REST/SOAP/XML-RPC oriented

=> Cloud APIs battle is ongoing, standardization ?

# Ovirt.org

- Umbrella project for various open source projects:
  - KVM virtualization management (upstream of RHEV)
  - Open source
  - Open governance
- Multiple projects
  - Core engine
  - API
  - Node
  - Vdsm
  - Guest agent
- Go to the workshop Tomorrow for full informations !

# Testing

- 3 public test suites
  - Libvirt-test-API: Python, from Red Hat QE, libvir-list@
  - TCK: Perl framework, libvirt upstream, libvir-list@
  - Autotest: Python, kernel upstream, [autotest@t.k.o](mailto:autotest@t.k.o)
- Upstream has (nearly) no hardware
  - Have tests done by partners
  - Gather and display results on libvirt.org
- Need infrastructure setup
  - Mechanism to report raw test results to libvirt.org
  - Analysis, display, forward of results on libvirt.org

- **Libvirt is mature** <http://libvirt.org/>
- **It is still growing**
- **Feedback is important**
- **Let's share code !**

**<http://veillard.com/talks/LinuxConJapan2012.pdf>**

**Questions?**