Ganeti, "how we did it"

A cluster virtualization manager.

Guido Trotter <ultrotter@google.com>

- Google, Ganeti, Debian

Ganeti at FOSDEM 2012

Saturday, 14:00 Janson, Internals (here and now)

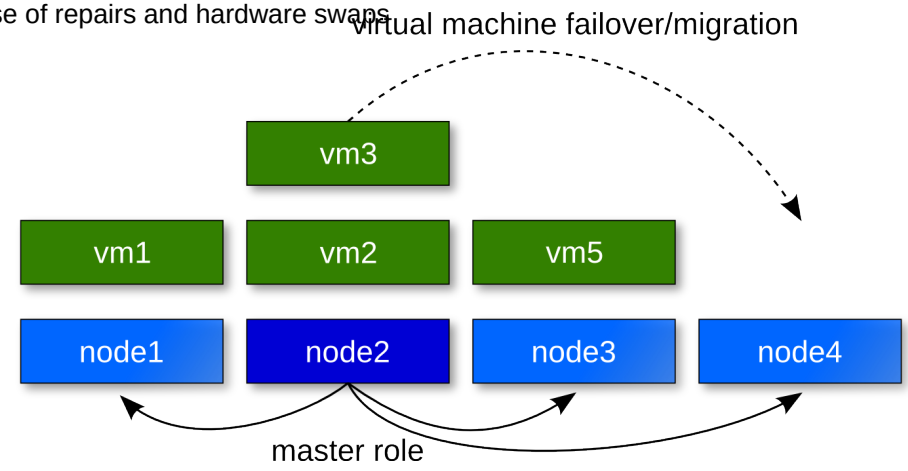
Sunday, 10:00 Chavanne, Getting Started (tomorrow morning)

Outline

- Introduction to Ganeti
- Ganeti internals
- Customizing Ganeti

What can it do?

- Manage clusters of physical machines
- Deploy Xen/KVM/lxc virtual machines on them
  - Live migration
  - Resiliency to failure (data redundancy over DRBD)
  - Cluster balancing
  - Ease of repairs and hardware swaps



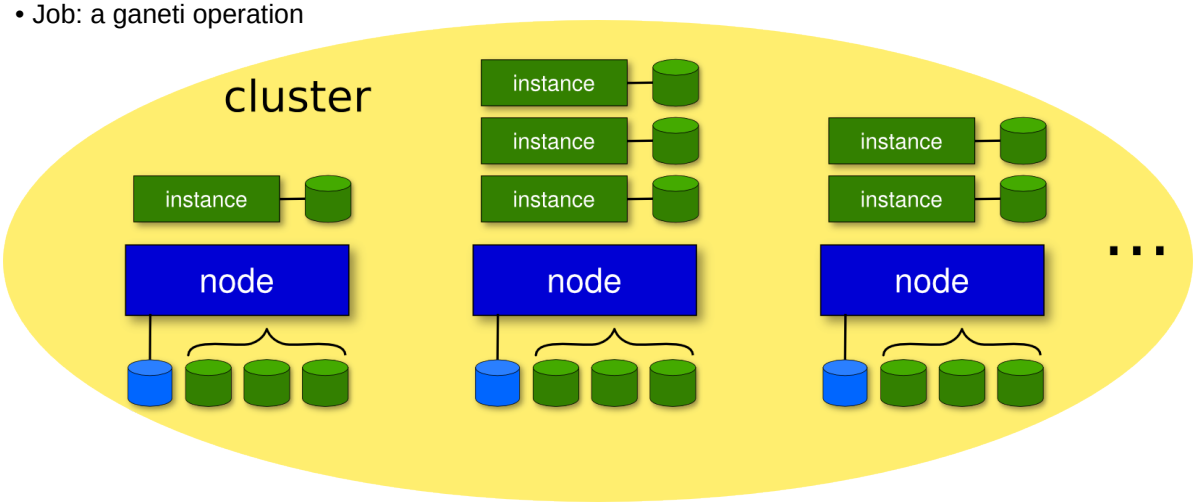
Ideas

- Making the virtualization entry level as low as possible
  - Easy to install/manage

    - No specialized hardware needed (eg. SANs)
    - Lightweight (no "expensive" dependencies)
- Scale to enterprise ecosystems
    - Manage symultaneously from 1 to ~200 host machines
    - Access to advanced features (drbd, live migration)
- Be a good open source citizen
    - Design and code discussions are open
    - External contributions are welcome
    - Cooperate with other "big scale" Ganeti users

Terminology

- Node: a virtualization host
- Nodegroup: an omogeneous set of nodes
- Instance: a virtualization guest
- Cluster: a set of nodes, managed as a collective
- Job: a ganeti operation



Technologies

- Linux and standard utils (iproute2, bridge-utils, ssh)
- KVM/Xen/LXC
- DRBD, LVM, or SAN
- Python (plus a few modules)
- socat
- Haskell (optional)

Node roles (management level)

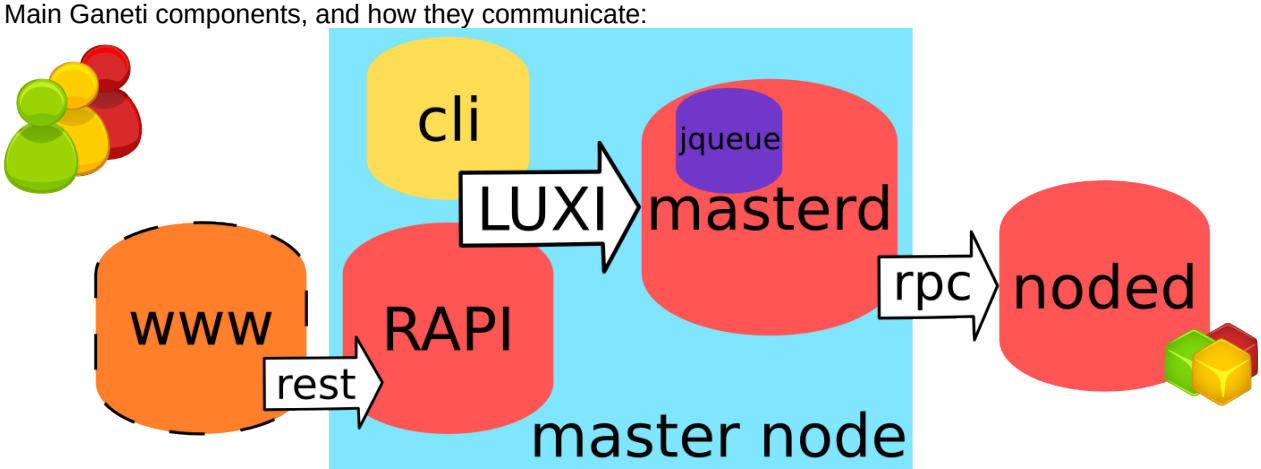
- Master Node
  - runs ganeti-masterd, rapi, noded and confd
- Master candidates
  - have a full copy of the config, can become master
  - run ganeti-confd and noded
- Regular nodes
  - cannot become master
  - get only part of the config
- Offline nodes, are in repair

Node roles (instance hosting level)

- VM capable nodes
  - can run virtual machines
- Drained nodes
  - are being evacuated
- Offlined nodes, are in repair

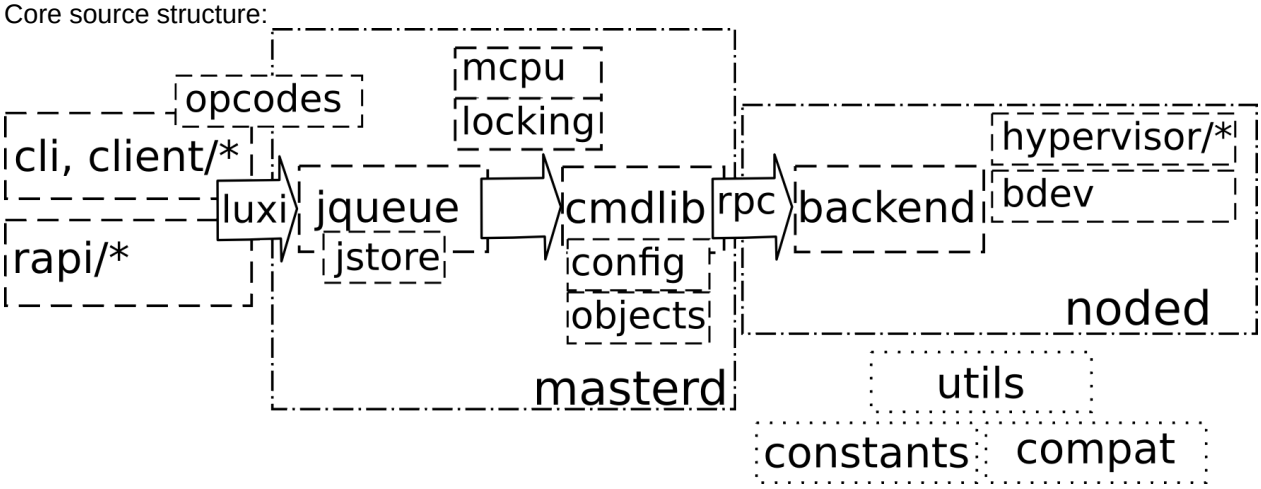
Ganeti Components

Main Ganeti components, and how they communicate:



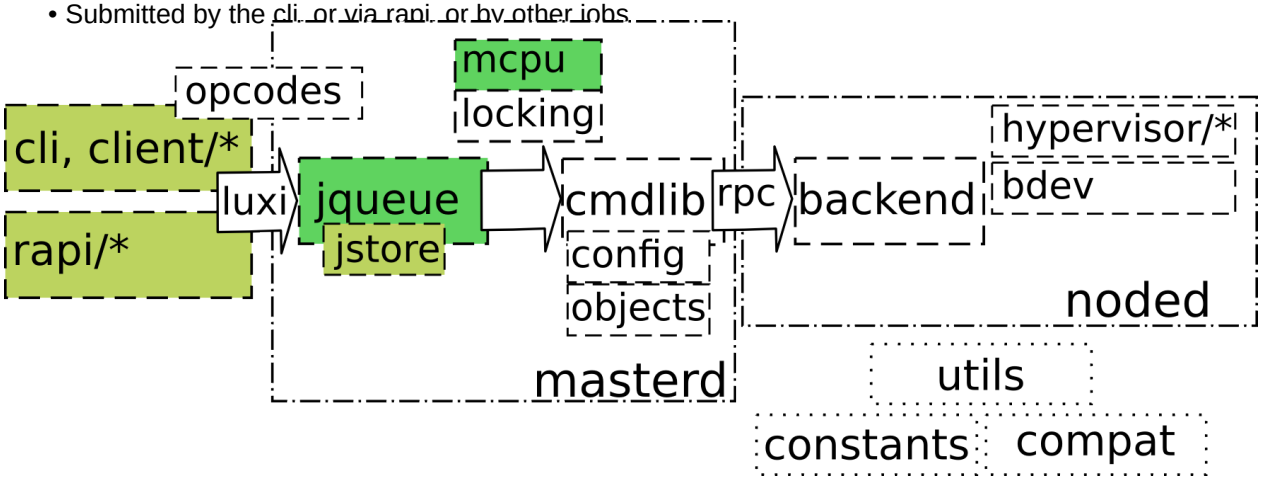
Ganeti Core Structure

Core source structure:



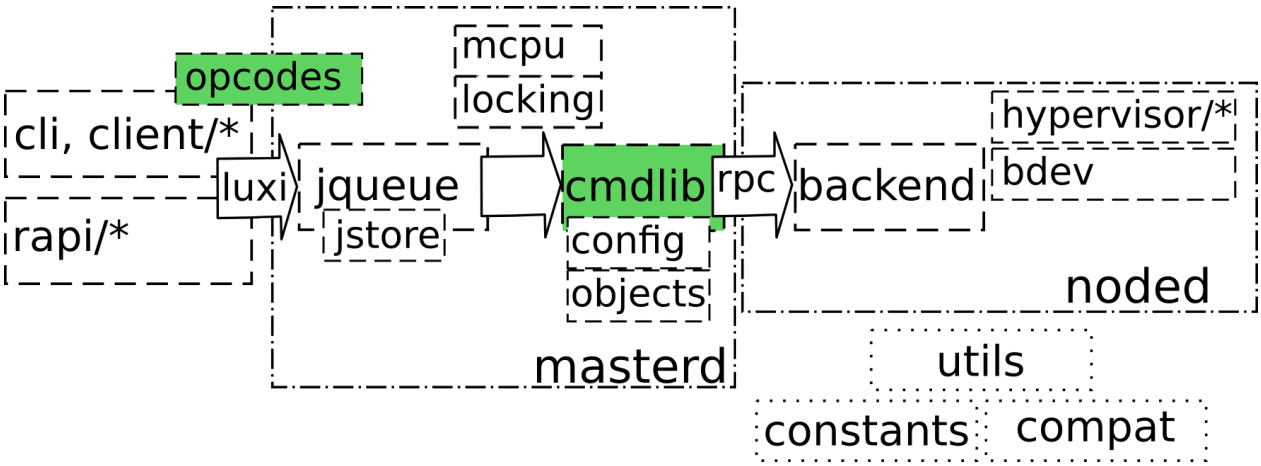
Jobs

- List of opcodes, executed in sequence
- Submitted by the cli, or via rapi, or by other jobs



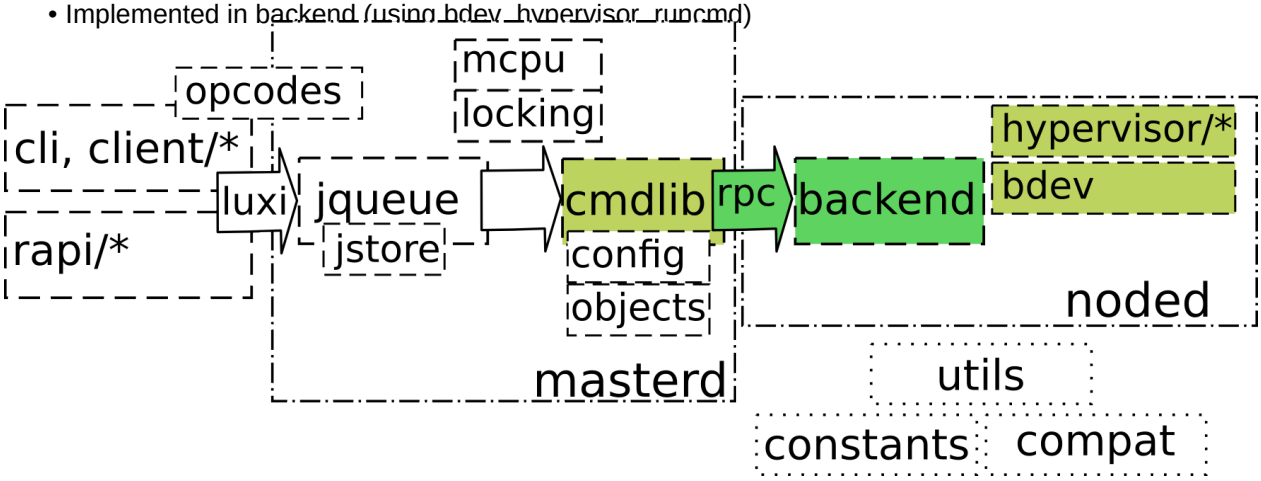
Opcodes

- Cluster business logic
- Implemented in cmdlib



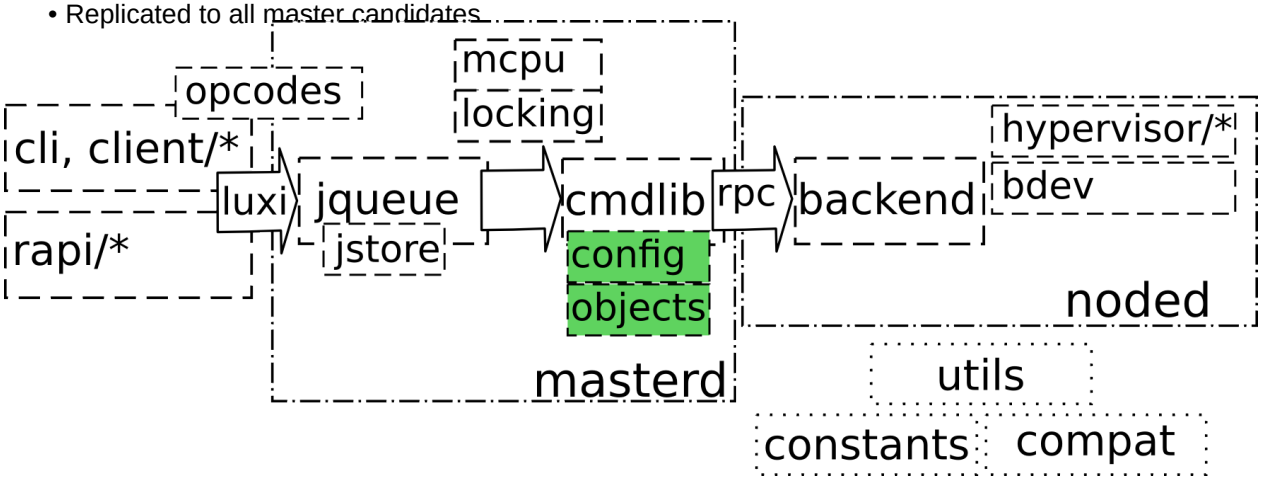
RPCs

- Per-node business logic
- Implemented in backend (using bdev, hypervisor, runcmd)



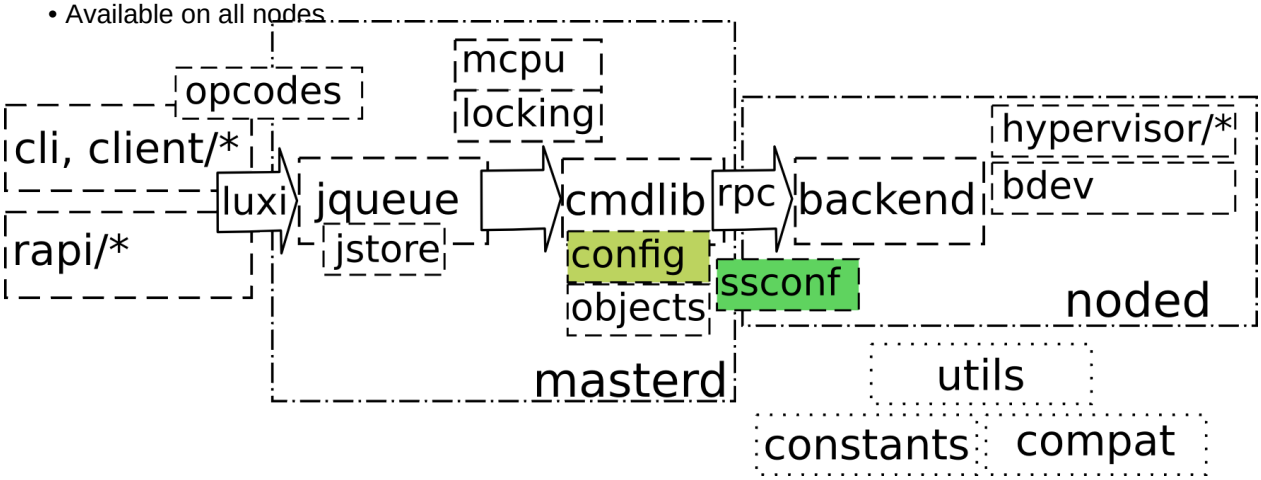
Config

- Tree of "objects" with all the cluster entities
- Replicated to all master candidates



ssconf

- Flat-file export of parts of the config
- Available on all nodes



Customizing Ganeti

Most common customizations:

- Altering hypervisor behavior
- Adding an hypervisor parameter
- Altering cluster business logic
- Adding an option to the cluster business logic
- Adding a backend storage
- Adding a new hypervisor

Altering hypervisor behavior (simple)

- Edit the logic in your hypervisor's file
- For example add a command line flag to kvm, or a config value for xen

Adding an hypervisor parameter (simple)

- Add the parameter in constants.py (eg: HV_KVM_SPICE_USE_VDAGENT)
- Edit the logic in your hypervisor's file
- eg: migration bandwith and downtime control: commit e43d4f9f

Altering cluster business logic (medium)

- Change the logic in cmdlib.py
- Be careful w.r.t. locking (do you need more? less?)
- Add any rpc to backend.py, rpc_defs.py, server/noded.py
- If the hypervisor interface changes, update all hypervisors

Adding opcode level options (medium)

- Add the option field to opcodes.py, use the right type (see ht.py)
- Use the option in cmdlib (see "altering cluster business logic")
- Add the command line flag to cli.py and the right utility in client/*

Adding a backend storage (hard)

- Implement the BlockDev interface in bdev.py
- Add the logic in cmdlib (eg. migration, verify)
- Add the new storage type name to constants
- Add any parameter the new storage needs to constants
- Modify objects.Disk to suppport your storage type
- eg: adding support for RBD: commit 7181fba

Adding a new hypervisor (medium)

- "just" implement the hypervisor API (easy)
- Add the hypervisor name and parameters to contants.py
- Alter cmdlib as needed for supporting it
- Alter the hypervisor API as needed for supporting it

From Development to Deployment

Please come tomorrow, at 10:00 in Chavanne

- Latest Features
- Practical usage examples
- How to bring it to enterprise level

People running Ganeti

- Google (Corporate Computing Infrastructure)
- grnet.gr (Greek Research & Technology Network)
- osuosl.org (Oregon State University Open Source Lab)
- fsffrance.org (according to docs on their website and trac)
- ...

Conclusion

- Check us out at http://code.google.com/p/ganeti.
- Or just search for "Ganeti".
- Try it. Love it. Improve it. Contribute back (CLA required).

Questions? Feedback? Ideas? Flames?