Autocluster

Martin Schwenke &
Andrew Tridgell

Background – Clustered Samba

- SoFS = Scale-out File Services
  - Scalable NAS (petabytes and beyond)
  - Built on RHEL5
  - Uses cluster of x86_64 blades
  - Clustered version of Samba
  - CTDB for clustering and failover
- See last year's LCA talk for details!

Clustering – hard to test

- Lots of problems with testing cluster code
  - There is never enough hardware
  - Hard to reproduce exact setups
  - “bit-rot” - clusters tend to degrade quickly

How long would it take to ....

- How many coffees to do this?
  - Setup 4 machines in a cluster
  - Install RHEL on them
  - Setup 3 NICs on each
  - Setup SAN shared storage
  - Install a cluster filesystem
  - Install clustered Samba
  - Install a TSM server for HSM
  - Setup NTP
  - ... etc etc

Virtual Clusters?

- Test on virtual machines?
  - Everyone on the team can have a cluster
  - Make better use of testing hardware budgets
- Problems ....
  - No good for performance testing
  - Virtual clusters bit-rot as fast as real clusters
  - Still a major pain to setup
  - Take a lot of disk space

Disposable Clusters!

- Can we?
  - Create a new cluster 'instantly'?
  - Create a cluster for each bug?
  - Avoid update problems?
  - Minimise disk usage?
  - Test effectively on a laptop?

Autocluster Demo

- Steps to a new cluster
  - Choose config
  - Create base image (one time only)
  - Create cluster
  - Boot it!

How does Autocluster work?

- Create base image:
  - Empty disk image file
  - Floppy image
  - libvirt XML file for "install guest" = template + options
  - Kickstart file = template + options + postinstall
  - Boot the guest
  - Watch and enjoy... or grab a coffee...
  - .... and another coffee (about 15 minutes total)
  - Make base immutable - "chattr +i foo-base.img"

How does Autocluster work?

- Create cluster – for each node:
  - Copy-on-write disk image, backed by base image
  - Mount image
  - Copy base files from templates (e.g. for config files)
  - Tweak some things ;-)
  - Unmount image
  - libvirt XML file for each node = template + options
- Shared disks
  - Cluster filesystem requires shared storage
  - Multipath! :-)
    - /sbin/scsi_id_autocluster.sh

Things of beauty!

- substitute_vars()
  - Much of autocluster is templates + config
  - This does the template substituton
  - 32 lines of bash
- config.default, defconf
  - autocluster has very few built-in config variable
  - Add a new variable by adding to config.default
  - It becomes a command-line option
  - ... and a config file option
  - ... and a template option
- vircmd
  - Simple layer on top of virsh
  - Do things to all cluster nodes

CTDB test suite

- Was *ad hoc* with local daemons
- Now structured with a bunch of simple tests
- More complex tests coming
- Simple tests still run against local daemons
  - make test
- Run on real/virtual cluster with no changes
  - Use "onnode" command to abstract cluster
  - Need "nodes" file, so need IP address of a node
  - Nodes need access to some support files

Problems

- Timing bugs
  - Do not sleep!
  - Monitor status changes
- KVM issues
  - Corruption: RPMs, RPM database, ...
  - NTP/time-sychronisation problems
- Lockups with nbd in daily run
  - "autocluster create cluster foo" – lock up under cron
  - Create cluster once, creating basic OS images
  - Additional level of copy-on-write
  - Defer the post-install to avoid waste
  - Slower

Customising autocluster

- Currently built for SoFS
  - Is there interest in retargeting it?
- Virtualisation engine
  - Currently kvm and libvirt
  - Should be fairly easy to retarget to Xen
- Distribution
  - Currently RHEL
  - Should be easy to modify templates for other Linux distros

The code

- Get it with git
  - git clone git://git.samba.org/tridge/autocluster.git autocluster

Questions?