Introduction to Large Scale Machine Management (first part)

DAAD Summer School: Aspects of Large Scale High Speed Computing

15th March 2011

Dr. Dirk von Suchodoletz

Faculty of Engineering, University Freiburg

UNI FREIBURG

Overview of this Lecture

- Lecture is split into four parts: two introductory/theoretical blocks (90 min) and two practical oriented ones (180 min, whole afternoon)
- Theoretical lectures:
  - Tuesday, 15th March: Introduction to Large Scale Linux Machine Management (Managing Clouds)
  - Thursday, 17th March: Machine Virtualization for better hardware utilization and efficient resource management

Overview of this Lecture

- Practical blocks:
  "Hands on" with support from two colleagues from the professorship in Freiburg
- Both take place in the computer lab #4
  - Monday, 21st March:
    Traditional LAN booting
  - Thursday, 31st March:
    Advanced and Flexible Wide Area Network booting

My Background

- Lecturer/Researcher at the Chair in Communication Systems (Prof. Schneider, linked to the universities computer center)
- Using Linux since kernel 0.81 (1993)
- Involved into to the development of a stateless Linux project used in pool systems to run Windows in VM-Player, identitiy management
- Project manager of OpenSLX
- Much of my research focus on practial issues of computer operation
- Some of the presented topics were researched on in bachelor or master thesis projects at the professorship

Structure of This Lecture

- Introduction to the topic
- Motivation, background for network based system administration
- Concept and ideas of Stateless Booting
- Client and server sides in network booting
- Network planning and network boot protocols
- Client side root filesystem, options and challenges for Read-write configuration and runtime data
- System monitoring

Goal of this Part of Summer School

- Topic resides above the base machine and software layers and below the cluster/cloud application and strategic/organizational management level
- Provides practical background of cluster and cloud operation
- Focus on system administration with lots of practical aspects

Background of This Topic

- Linux administration usually not in main focus of university teaching, but computer center operation demonstrates a range of open issues
- Despite the very practical matters, the topic triggered some nice research
  - Identity and system management
  - Efficient machine monitoring on different levels
  - Special purpose network block devices
  - Test suites for (automated) machine, network and system evaluation

Background and History of Netboot

- Efficient administration of larger numbers clients is a common problem around for a while
  - Triggered by the rise of the PC – paradigm shift from mainframe to autonomous machines
  - Comparably cheap machine in relation to Unix workstations and Mainframe computers heavily increased the installed number of networked machines significantly
- Bit of history of netbooting
  - Novell BootROMs for DOS and Windows
  - Sun Microsystem Diskless Workstations promoted with BOOTP and NFS
  - General ability of Unix workstations to netboot

System Management

Simplifying Administration

Pre-Requisites and Approaches

Managing Linux Pools

- My background originally management of larger numbers of computer / pools since 1996
  - All Linux based because Windows 95 was not affordable regarding licensing and maintenance cost
  - Extended to 400 clients administrated by few people
  - Similar concept like LTSP
  - Later on changed to client based operation
- Coming from Linux desktop pool operation – just deploying same principles of pool OS distribution into the cluster operation and cloud domain

Managing Linux Pools

UNI FREIBURG

- Linux is a very popular cluster computing operating system
  - Open Source, no license fees (relevant cost issue if talking of larger number of nodes)
  - Very adaptable, easy to extend
- Thus all relevant cluster nodes in Freiburg Computer Center or at the Faculties run different versions of Linux

More Motivations

- Just think of your own IT life
  - Your own machine at home – no problem
  - Think of the machines your non-IT friends ask you to manage their Windows boxes
  - (Windows) machines of your family and wider relatives
- Requirements of a Linux installation are mostly the same – tasks become very repetitive
- Just for system installation: Compute cluster operation is easily on a magnitude of this
- Same for clouds nowadays but on a even larger, more complex scale

More Motivations

- Typical additional administration tasks
  - OS roll-out of exactly the same system to large number of nodes
  - Permanent updates of these machines
- Tests and experiments
  - If OS directly installed difficult to handle
    - Partitioning of the disk
    - Handling different OS installation on same disk
    - More complex bootloader setup

Fast Deployment and Different OS

- Even more requirements for cluster operation – number of machines a relevant factor
  - Fast deployment is crucial – loosing real money if it takes several weeks to setup and customize 100+ machines
- Easy exchange of the installed OS
  - Check machines before buying, deploying
  - General hardware testing in failure cases
  - Do pre-production tests because of OS or program fixes, new versions available
  - Exchange of OS to run optimized version for certain compute jobs / different customer needs

Simplifying Administration

- Lots of software distribution and update systems available for popular operating systems but a number of shortcomings
  - Often costly
  - Peaks in network capacity consumption when rolling out packages
  - More complex to handle different OS installations on same machine
  - Machines might not be in proper state to receive package
  - Different machine configurations, especially hard disk setups and capacity

Linux Approach

- Linux per-se faces same challenges as other operating systems too
  - Tedious tasks to manage hundreds machines manually
  - Management frameworks for disk based installations available but omitted in this course
- Major advantage – Open Source approach, no license fees, no restrictions to scale up (massively)
  - Open for modifications
  - Easy to add new kernel components like special protocols or block devices

Ingredients for Network Booting

- Pre-requisites to simplify administrative tasks discussed in the beginning: Fast computer network
- Ingredients to talk of during this course
  - Network capacity
  - Network booting capability and protocols
  - Network filesystems or block devices
  - Proper filesystem configuration to cater for shared root filesystems without interference of machines

Further Aspects

- More general aspects (not limited to the environments we are focusing on here)
  - System integration for user identity and data management
  - Accounting of services provided
    - Network bandwidth utilized
    - CPU cycles used
    - Filespace consumed
  - System monitoring

Concept and Idea

Client and Server Sides

Stateless Computer Operation

- Why do we do this?
- Re-centraliziation after the era of the Autonomous PC paradigm
- Idea: Dramatic decreased administration because of centralization
  - Attendance of central servers instead of de-central nodes
  - New clients are simple to add
  - Easy replacement of failing machines
  - Rather different operating systems and or operations could be run on just same machine (just rebooted into other system)

Stateless and Diskless

- Stateless good idea and rather easily achievable – see e.g. Linux Live CD/DVDs
- You often find the term "diskless"
  - Sub domain of stateless as disk often not required or simply omitted to save on investments in earlier days
- Today installed hard disks used for scratch space
- Goal: Try to avoid to “personalize” machines and store any machine specific data on the nodes

Stateless Linux Clients

- Client side
  - Does not need any installation to hard disk, but could use any disk for local scratch
  - All clients share the same root filesystem, which is stored on one/some servers
  - Bootup speed is despite network transfer often better then in disk based installations
  - Clients are configured automatically during bootup
  - Stateless system administration packages offer despite shared rootfile-system per client configuration (different to CD/DVD based solutions)

Stateless Linux Clients

- Configuration challenge
  - Configuration is to be renewed with every reboot
  - Should be generic to cover a wider range of different hardware
  - Might consume a certain amount of boot time – usually minor compared to disk stored fixed configuration
  - But: Bootup optimizations work for all attached clients
- Configuration advantage
  - Configuration easily becomes hardware independent
  - Makes it easier to add/exchange nodes in the cluster/ cloud

Stateless Machines Server

UNI FREIBURG

- Server side
  - One server is able to host several different root filesystems and large number of clients
  - Using redundant servers and failover it is easy to have simple maintenance
  - Using standard Internet protocols, like DHCP, TFTP, NFS or Network Block Device Servers

Stateless Machines Server

- Moderate server hardware requirements: Fast-IO and network components
  - Average hardware requirements, lots of RAM, fast disks, broad network connection to the backbone improve performance
  - Strategic placement within the organization
  - Configuration dependent on number of clients served
  - Number of different client operating systems (Linux versions and variants) provided
  - Optimization for large, distributed cloud installations by using cache and proxy servers

Network Planning

Network Boot Protocols

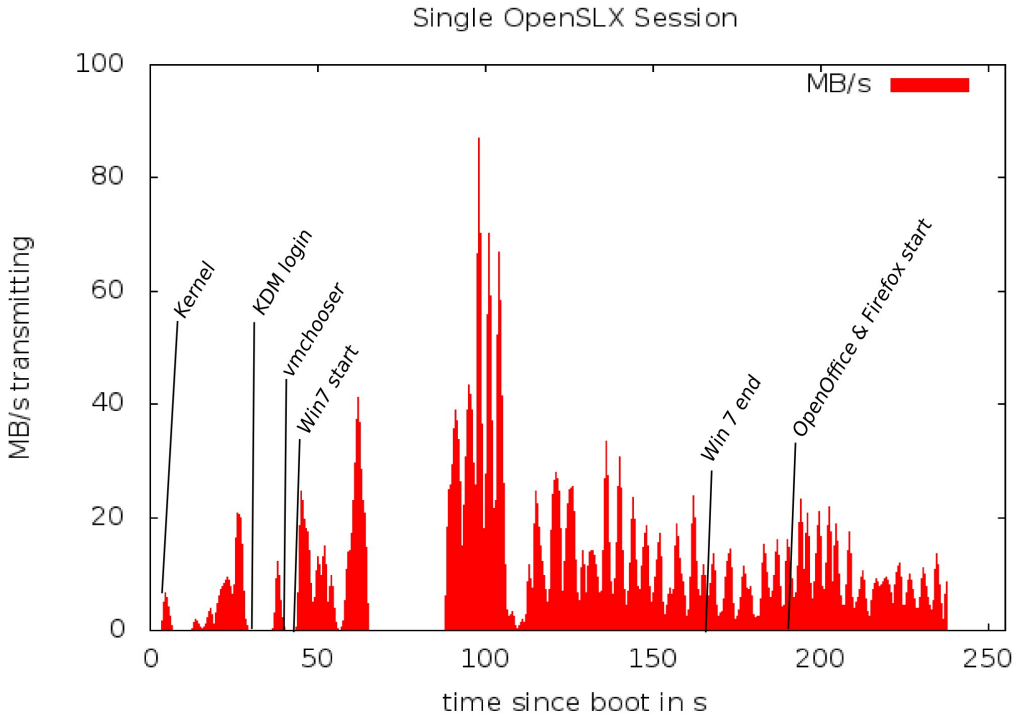
- Concept of multiple compute cluster nodes instead of one super computer around for a while
- Nodes have to be interconnected
  - Cheap solution based on Ethernet with TCP/IP
  - Special purpose solutions use Infiniband or alike
- Plenty of Local Area Network capacity available
  - Mostly copper infrastructure of cables with 4*2 wires of a certain shielding and quality
  - Costs of Gigabit network adapters starting from ~50 Reais
  - Cost of ports on a Gigabit switch with proper stacking ability and uplink capacity from ~100 Reais

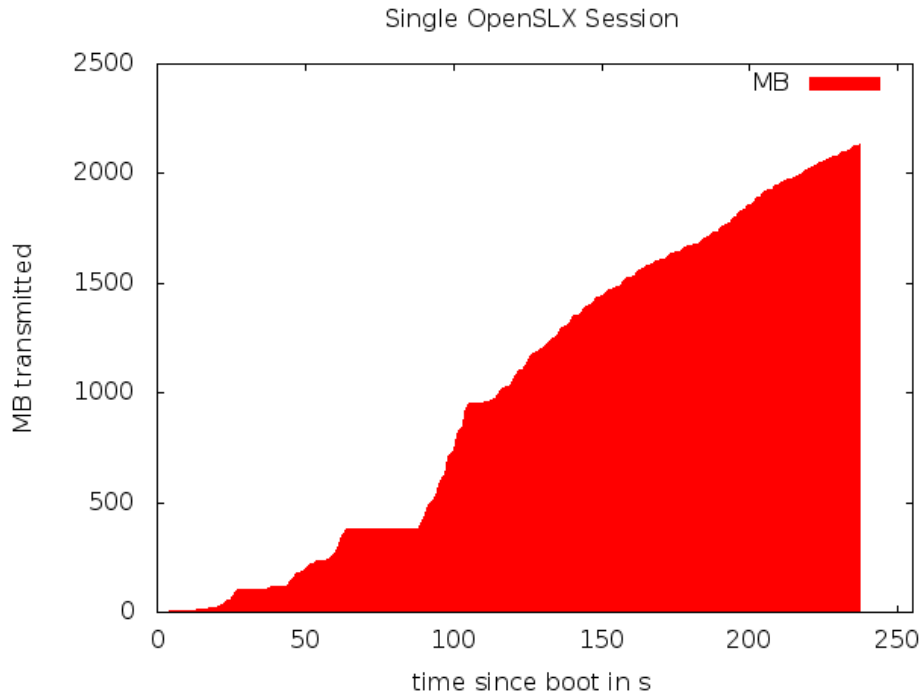
- In most setups data and network booting + filesystem share the same infrastructure
  - Network demand of data absolutely job dependent
  - Jobs which just load a (huge) bunch of data and then do number crunching without sharing any/much data with other nodes, send back results to the manager
  - Jobs like hosting services for e.g. Online stores: Typical bandwidth usage profile of a server machine: Customer data plus database/filesystem interaction
  - Parallel computing jobs loading rather few data upfront but exchange lots of results during runtime with neighboring nodes, often near-real time critical

- Network booting + filesystem is to be accommodated with jobs/cloud application needs
  - DHCP packets no issue
  - Depending on configuration and setup: Kernel + Initial Ram Filesystem few Megabytes
  - After mounting the root filesystem several hundred Megabytes initially
  - Depending on applications during runtime more filesystem data generated

Network Traffic Analysis

- Network booting and operation produces certain network traffic patterns for each client, e.g. booting a Linux desktop with Windows 7 in VM





Break

- Short break, then continue with
  - Network booting
  - Network filesystems and block devices
  - Overlay/union filesystems for read-write runtime data
  - System monitoring