

Bachelor-Thesis

Effiziente Speicherung virtueller Festplatten mit bestehender OpenSource-Software (Arbeitstitel)

Bastian de Groot

1. November 2010

Prüfer Prof. Dr. Jörg Thomaschewski

Zweitprüfer Dr. Arvid Requate

Inhaltsverzeichnis

1	Einleitung	5
1.1	Zieldefinition	5
1.2	Vorgehen und Kurzzusammenfassung	6
1.2.1	Quellen	6
2	Analyse Copy-on-Write	8
2.1	Sparse-Dateien	8
2.2	qcow2	9
2.2.1	Vorteile	9
2.2.2	Nachteile	9
2.3	vhd	9
2.3.1	Vorteile	9
2.3.2	Nachteile	10
2.4	dm-snapshots	10
2.4.1	Vorteile	10
2.4.2	Nachteile	10
2.5	LVM-Snapshots	11
2.5.1	Vorteile	11
2.5.2	Nachteile	11
2.6	Benchmarks	11
2.6.1	IOzone	12
2.6.2	Bonnie++	12
3	Analyse Verteilung von Images	13
4	Synthese	14

4.1	Konzept	14
4.2	Realisierung einer Komplettlösung	14
5	Zusammenfassung und Ausblick	15

1 Einleitung

Betriebssystemvirtualisierung ist, wenn sich mehrere virtuelle Betriebssysteminstanzen Hardwareressourcen wie CPU, RAM oder Festplatten teilen. Der Virtualisierungskern (Hypervisor) stellt den virtuellen Betriebssysteminstanzen eine in Software und Hardware realisierte Umgebung zur Verfügung, die für die darin laufenden Instanzen kaum von einer echten Hardwareumgebung unterscheidbar sind. Es gibt unterschiedliche technische Ansätze der Virtualisierung, wie Paravirtualisierung oder Vollvirtualisierung. Diese Kategorisierung bezieht sich darauf, wie der Hypervisor die vorhandene Hardware der virtuellen Instanz bereitstellt. Auf diesem Gebiet gibt es eine sehr aktive Entwicklung. [Prz] [Bau]

Wenig beachtet bei der Entwicklung von Virtualisierungssoftware ist jedoch die Speicherung von virtuellen Festplatten. In dieser Arbeit wird dieser Punkt aufgegriffen und die Möglichkeit der Optimierung mit der Copy-on-Write Strategie beleuchtet.

Copy-on-Write ist eine Optimierungsstrategie, die dazu dient unnötiges Kopieren zu vermeiden und somit die für Bereitstellung einer geklonten virtuellen Maschine benötigte Zeit zu minimieren.

1.1 Zieldefinition

Ziel dieser Arbeit soll es sein, Möglichkeiten zur effizienten Speicherung von virtuellen Festplatten aufzuzeigen. Hierbei wird in dieser Arbeit ausschließlich auf beste-

hende Open Source Lösungen zurückgegriffen. Diese Lösungen werden miteinander verglichen und eine effiziente Lösung herausgearbeitet. Außerdem wird betrachtet, wie die für das Copy-on-Write benötigten Master-Images im Netzwerk effizient verteilt werden können.

1.2 Vorgehen und Kurzzusammenfassung

Zunächst werden die vorhandenen Softwarelösungen für Copy-on-Write und für die Verteilung der Master-Images erläutert. Danach werden diese anhand verschiedener anwendungsrelevanter Kriterien miteinander verglichen. Nachdem die besten Lösungen beider Kategorien gefunden wurden, werden Softwaretools erstellt, die die Nutzung der gefundenen Lösung ohne tiefgreifende Vorkenntnisse ermöglicht.

1.2.1 Quellen

Diese Arbeit enthält neben den herkömmlichen Quellen auch Mailinglisten- und Forenbeiträge, sowie Blogeinträge. Hierdrauf soll im Folgenden eingegangen werden.

Quellenangaben sind im Bereich der Open Source Software schwer zu machen. Es gibt keine einheitliche Dokumentation der Software. Häufig sind die Informationen nicht an einer zentralen Stelle vereint, sondern liegen verstreut im Internet in Foren, Blogs, Mailinglisten oder auch in Manpages und den Quelltexten selbst. Die Relevanz und die Richtigkeit einer solcher Quellen ist schwer zu bewerten, da Blogs, Mailinglisten und Foren keinen Beschränkungen unterliegen. Das heißt, jeder der Willens ist zu einem Thema etwas zu schreiben, kann dies auch tun.

Die oben genannte Verstreuer birgt, neben der schwierigen Bewertbarkeit der Rich-

tigkeit und Relevanz, ein weiteres Problem. Da sehr viele Autoren zum einem Thema etwas schreiben, werden unterschiedliche Begriffe synonym verwendet oder sind mehrdeutig.

Alle Quellen werden mit der zu Grunde liegenden Erfahrung des Autors dieser Arbeit ausgewählt und überprüft, können aber aus den oben genannten Gründen keine absolute Richtigkeit für sich beanspruchen.

2 Analyse Copy-on-Write

2.1 Sparse-Dateien

Eine Sparse-Datei ist eine Datei, die nicht vom Anfang bis zum Ende beschrieben ist. Sie enthält also Lücken. Um Speicherplatz zu sparen, werden diese Lücken bei Sparse-Dateien nicht auf den Datenträger geschrieben. Eine Sparse-Datei ist kein eigenes Imageformat sondern eine Optimierungsstrategie. [spa]

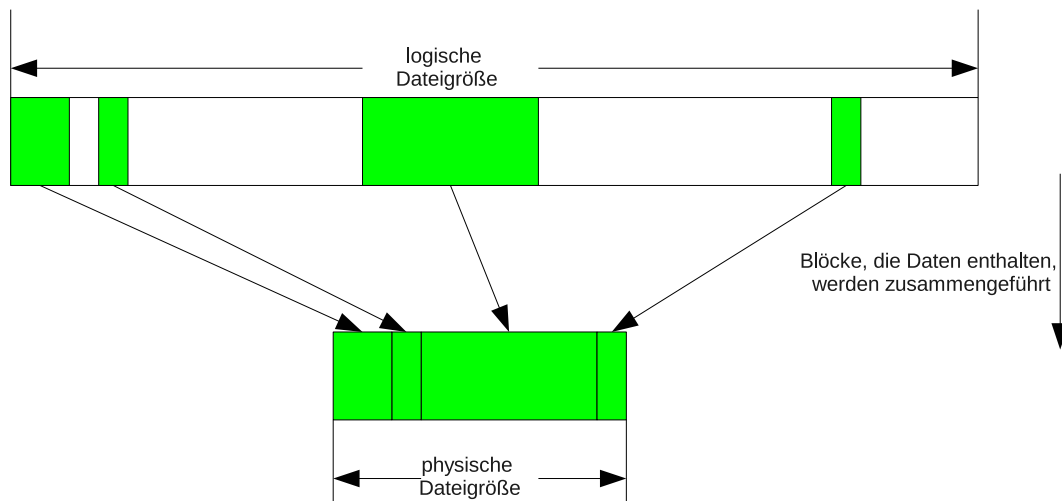


Abbildung 2.1: Sparse-Datei

2.2 qcow2

Das Imageformat qcow2 ist im Rahmen des qemu Projekts entwickelt wurde. Es ist der Nachfolger des ebenfalls aus dem qemu Projekt stammenden Formats qcow. [McL] [qem]

2.2.1 Vorteile

- einfache Einrichtung

2.2.2 Nachteile

- Kompatibilitätsprobleme mit Xen und anderen offenen Virtualisierungstechniken (z.B. VirtualBox)

2.3 vhd

Das Format vhd ist von Conectix und Microsoft entwickelt worden. Die Spezifikation des Imageformats wurde von Microsoft im Zuge des “Microsoft Open Specification Promise” freigegeben. Seit der Freigabe der Spezifikation bieten einige Open Source Virtualisierungstechnologien wie qemu, Xen oder VirtualBox die Möglichkeit dieses Format zu verwenden. [mso] [vhd]

2.3.1 Vorteile

- einfache Einrichtung

2.3.2 Nachteile

- Weiterentwicklung ist fragwürdig
- Verwendung von Copy on Write mit KVM nicht möglich

2.4 dm-snapshots

Die dm-snapshots sind eine Funktion des Device Mappers. Device Mapper ist ein Treiber im Linux-Kernel. Er erstellt virtuelle Gerätedateien, die mit bestimmten Features wie zum Beispiel Verschlüsselung ausgestattet sind. Bei dm-snapshots wird eine solche virtuelle Gerätedatei erstellt. Sie wird aus zwei anderen Gerätedateien zusammengesetzt. Die erste Gerätedatei ist der Ausgangspunkt, wenn an daran Änderungen vorgenommen werden, werden sie als Differenz in der zweiten Gerätedatei gespeichert. [Bro] [dmk]

2.4.1 Vorteile

- hohes Entwicklungsstadium
- sichere Weiterentwicklung

2.4.2 Nachteile

- Aufwendige Einrichtung
- unabhängig von Virtualisierungstechnik

2.5 LVM-Snapshots

LVM-Snapshots sind ein Teil des Logical Volume Managers. LVM ist eine Software-Schicht die über den eigentlichen Hardware-Festplatten einzuordnen ist. Es basiert auf Device Mapper. LVM ermöglicht das Anlegen von virtuellen Partitionen (logical volumes). Diese können sich über mehrere Festplatten-Partitionen erstrecken und Funktionen wie Copy-on-Write bereitstellen. [lvmc] [lvma] [lvmb]

2.5.1 Vorteile

- hohes Entwicklungsstadium
- sichere Weiterentwicklung
- unabhängig von Virtualisierungstechnik

2.5.2 Nachteile

- Aufwendige Einrichtung
- Live-Migration nicht möglich
- Nutzung von Sparse-Dateien schwer umsetzbar

2.6 Benchmarks

Ein wichtiger Punkt für die Entscheidung welche Copy-on-Write Implementierung optimal ist, ist die Lese- und Schreibgeschwindigkeit. Hierbei gibt es zwei Zugriffsarten, einmal den sequentiellen Zugriff und den wahlfreien oder auch zufälligen Zugriff.

Die Testergebnisse werden in diesem Kapitel zusammenfassend aufgeführt. Die kompletten Testergebnisse befinden sich im Anhang.

2.6.1 IOzone

IOzone ist ein Tool mit dem in einer Reihe von unterschiedlichen Tests die Lese- und Schreib-Geschwindigkeit überprüft werden kann. Es wird hier zur Überprüfung der sequentiellen Lese- und Schreibgeschwindigkeit verwendet.

2.6.2 Bonnie++

Bonnie++ dient wie IOzone als Tool zum Testen von Festplatten. Es wird hier zur Überprüfung der sequentiellen Lese- und Schreibgeschwindigkeit sowie zum Testen des wahlfreien Zugriffs verwendet.

3 Analyse Verteilung von Images

4 Synthese

4.1 Konzept

4.2 Realisierung einer Komplettlösung

5 Zusammenfassung und Ausblick

Glossar

Open Source

Open Source Software.

Literaturverzeichnis

- [Bau] BAUN, Christian: *Vorlesung Systemsoftware*. http://jonathan.sv.hs-mannheim.de/~c.baun/SYS0708/Skript/folien_sys_vorlesung_13_WS0708.pdf, Abruf: 31.10.2010
- [Bro] BROŽ, Milan: *Device mapper*. <http://mbroz.fedorapeople.org/talks/DeviceMapperBasics/dm.pdf>, Abruf: 17.10.2010
- [dmk] *Device-mapper snapshot support*. <http://www.kernel.org/doc/Documentation/device-mapper/snapshot.txt>, Abruf: 17.10.2010
- [lvma] *Linux LVM-HOWTO*. <http://www.selflinux.org/selflinux/html/lvm01.html>, Abruf: 18.10.2010
- [lvmb] *LVM2 Resource Page*. <http://sourceware.org/lvm2/>, Abruf: 18.10.2010
- [lvmc] *What is Logical Volume Management?* <http://tldp.org/HOWTO/LVM-HOWTO/whatisvolman.html>, Abruf: 18.10.2010
- [McL] McLOUGHLIN, Mark: *The QCOW2 Image Format*. <http://people.gnome.org/~markmc/qcow-image-format.html>, Abruf: 18.10.2010
- [mso] *Microsoft Open Specification Promise*. <https://www.microsoft.com/interop/osp/default.aspx>, Abruf: 18.10.2010
- [Prz] PRZYWARA, André: *Virtualization Primer*. <http://www.andrep.de/virtual/>, Abruf: 01.11.2010

[qem] *QEMU Emulator User Documentation*. http://wiki.qemu.org/download/qemu-doc.html#disk_005fimages, Abruf: 18.10.2010

[spa] *Sparse files*. <http://www.lrdev.com/lr/unix/sparsefile.html>, Abruf: 18.10.2010

[vhd] *Virtual Hard Disk Image Format Specification*. <http://technet.microsoft.com/en-us/virtualserver/bb676673.aspx>, Abruf: 18.10.2010

Tabellenverzeichnis

Abbildungsverzeichnis

2.1	Sparse-Datei	8
-----	------------------------	---

Listings