

---

# Goggle Summer of Code Project Proposal

## NetBSD Volume Manager Project

Adam Hamšík

This work is distributed under the BSD license.

### Abstract

This is project proposal for Google Summer of Code project.



Copyright © 2007 Adam Hamšík

## Table of Contents

1. General .....	2
2. Project Proposal .....	2
3. NetBSD implementation .....	2
3.1. Linux Device Mapper driver .....	3
3.2. User land Part .....	4
3.3. Documentation Part .....	4
4. Project Tasks .....	4
Bibliography .....	5

# 1. General

Logical Volume Manager LVM is a method of allocating space on mass storage devices like disks. LVM is more flexible than standard partitioning schemes. In particular a Volume Manager can concentrate, concatenate parts of disks together into larger ones. These virtual partitions can be re sized, moved, snapshotted, while possibly being used.

Logical Volume Manager is form of virtualization because it turns storage into a more easily allocated, and more manageable parts.

# 2. Project Proposal

Volume Manager is part of all standard Unix like/Unix OS's:

- Linux  
Logical Volume Manager based on devmapper
- FreeBSD  
Vinum
- AiX  
Logical Volume Manager
- Solaris  
Solaris Volume Manager

## 1. Physical extends PV

Are created on non continuous mass storage partitions. PV's are split into small chunks called Physical Extends PE. PE's can have even size like Linux and HPUX Volume Manager's or variable-size like in the AiX LVM or in Veritas.

## 2. Logical Volume LV

PE's can be grouped into virtual partitions called Logical Volumes. LV's can be re sized, snapshotted, renamed, removed etc. Some Volume Managers allow only on line growing others resizing in both directions.

## 3. Volume Group VG

PV's may be organized into the groups called Volume Groups. Volume Groups are used for redundancy. Administrator can mirror PE's from one LV to another PE in same VG.

# 3. NetBSD implementation

How to accomplish my goals.

- Port of Linux LVM

Christian Limbach already ported lvm to NetBSD 1.6 long time ago. Port of Linux LVM can be done in less time and probably it will be more feature full then complete rewrite of Volume Manager. against this way. This Port include BSD rewrite of linux device-mapper driver. It will be 1:1 port so our port will be fully interface compatible with linux driver.

## 3.1. Linux Device Mapper driver

I want to implement functional 1:1 port of Linux dm driver as backend to NetBSD Volume Manager. Linux device-mapper support more targets. Target define how are blocks on virtual block devices managed.

### List of generally available targets.

- Linear Target
- Crypt Target
- Zero Target
- Error Target
- Mirror Target
- Strip Target

I want use wedges as future NetBSD partition scheme. Wedges are used to lineally map disk blocks to user defined disk wedge. Wedges are funcionally equivalent to Linux device-mapper linear target.

### 3.1.1. Kernel Part of my project

#### 3.1.1.1. Basic kernel driver

Write new kernel driver compatible with Linux device-mapper. Wedges will work as Linear target. NetBSD device driver will be able to address sectors from physical devices to logically defined volumes.

Device mapper driver uses tables to define logical devices. Each table define blocks and physical device which are mapped to selected Logical device.

```
Logical Volume Lvm1
    sectors 0-1000    linear  /dev/wd0a
    sectors 0-1000    linear  /dev/wd1a
    sectors 3000-5000 linear  /dev/wd2a
```

Our device-mapper driver will create in-kernel wedges on devices which were send to it from libdev-mapper library. This wedges are grouped into one Logical device called lvm1. LVM1 will be

regular device on which file system can be created and which can be mounted. In my example code Logical Volume is LV virtual partition, PV's are created on wd0a, wd1a and wd2a and these wedges are split into the smaller chunks called sectors/PE's.

This task should be almost complete before coding part of SoC starts.

### 3.1.1.2. Improve kernel-userspace ioctl protocol

For communication between libdevmapper library and device-mapper kernel driver complicated ioctl protocol was defined. ioctl interface copy one structure `dm_ioctl` into kernel with ioctl command. Kernel driver then need to find userspace address of this structure and copy one memory chunk its size is find in `dm_ioctl->data_size` and copy it into kernel.

As my task for SoC project I want to implement new proplib based kernel ioctl protocol. This task will include some changes to userland code to but this will be related mostly to one function in libdevmapper `_dm_do_ioctl`.

### 3.1.1.3. LVM snapshots, multipath, targets

There is no support for snapshots neither multipath in our device-mapper driver now. As my SoC project I want to change this. Another targets are also possible task e.g. linux crypt target can be implemented with our cgd device.

## 3.2. User land Part

As userland part of my work I want to port Linux Libdevmapper library and lvm2 tools(this was already done by agc@). Which are well tested and used tools. Next userland task is import of lvm2-tools and libdevmapper library into netbsd base system.

## 3.3. Documentation Part

Documentation part of my SoC project will consist from these tasks:

- Write good user documentation
- Add developer documentation into NetBSD internals book
- Properly comment our dm driver code.

# 4. Project Tasks

## Procedure 1. Kernel, Userland and documentation tasks

### 1. Kernel

New proplib ioctl interface, multipath, snapshots, new targets.

### 2. Userland

Import of libdevmapper and lvm2-tools into NetBSD source tree.

### 3. Documentation

Write user, developer documentation and properly comment my code.

## Bibliography

Laurent Vanel, Ronald van der Knaap, Dugald Foreman (2000). “ [AIX Logical Volume Manager from A to Z: Introduction and Concepts](http://www.redbooks.ibm.com/redbooks.nsf/0/1f80d43a7a4e475d862568200078bb54?OpenDocument). [http://www.redbooks.ibm.com/redbooks.nsf/0/1f80d43a7a4e475d862568200078bb54?OpenDocument] ”

“ [Logical Volume Manager HOWTO](http://tldp.org/HOWTO/LVM-HOWTO/index.html). [http://tldp.org/HOWTO/LVM-HOWTO/index.html] ”

“ [Device-mapper presentation](http://people.redhat.com/agk/talks/FOSDEM_2005/). [http://people.redhat.com/agk/talks/FOSDEM\_2005/] ”