

[Tutorials](#)[Tags](#)[Forums](#)[Linux Commands](#)[Subscribe](#)[ISPConfig](#)[News](#)[Q](#) Tutorial search[Home](#)[Learn LVM on Linux with graphical tools](#)

# Learn LVM on Linux with graphical tools

The LVM, as a storage model, has been widely used on machines with Linux architectures, including PCS, NAS, and servers. Its value is self-evident. As a widely used, mature software, the tutorials of how to use LVM are already all over the internet.

## On this page

- [CONCEPT](#)
  - [1. LVM](#)
  - [2. PV](#)
  - [3. VG](#)
  - [4. LV](#)
  - [5. PE](#)
  - [6. LE](#)
- [OPERATIONS](#)

However, these tutorials are all based on command-line tools. The command-line tool, while powerful and complete, is complex and very unintuitive, making it difficult for users to understand LVM, especially for beginners.

Just like partitioning tools, various graphical or semi-graphical (character-based) tools make partitioning easy to read in a very intuitive way. Learning LVM concepts and operations through graphical tools also greatly reduces the cost of user learning.

Let's guide you on the LVM concepts & LVM managing through graphical tools.

## CONCEPT

## 1. LVM

---

LVM Logical Volume Management is a storage Management solution in Linux. It provides users with an extensible and highly flexible storage Management model. With LVM users can achieve server-level storage performance and security inexpensively.

For common users and small-sized server users, LVM supports RAID 0, 1, 4, 5, 6, 10, and JBOD volumes for creating snapshots and storage pools.

For advanced users, LVM supports highly reliable access and can easily implement distributed storage services.

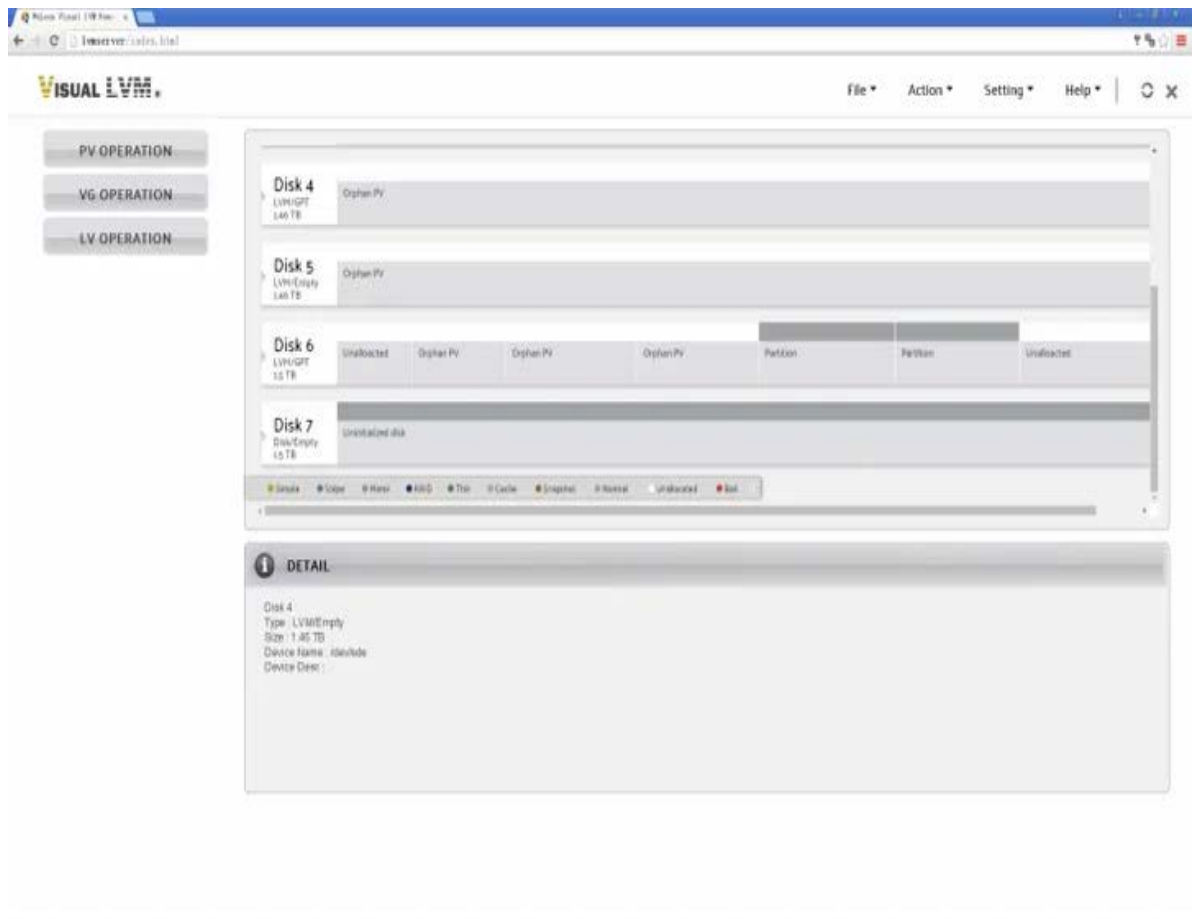
At the same time, volumes created by LVM can be extended and migrated very easily. Replacing/adding devices is easy.

The LVM model consists of three layers: Physical Volume (PV), Volume Group (VG), and Logical Volume (LV).

## 2. PV

---

The PV layer is the physical layer of LVM and the basis of LVM's existence. The PV can be regarded as disk in a RAID array. Users can create PVs using disks or partitions. LVM configuration information is usually stored in each PV. The Linux kernel reads configuration information to detect LVM, and when the LVM is damaged, the repair tool also uses this information to restore the LVM configuration.



The host devices of PVs

There is only one partition on disk 4, which use entire space

There is no partition table(neither MBR nor GPT) on disk5

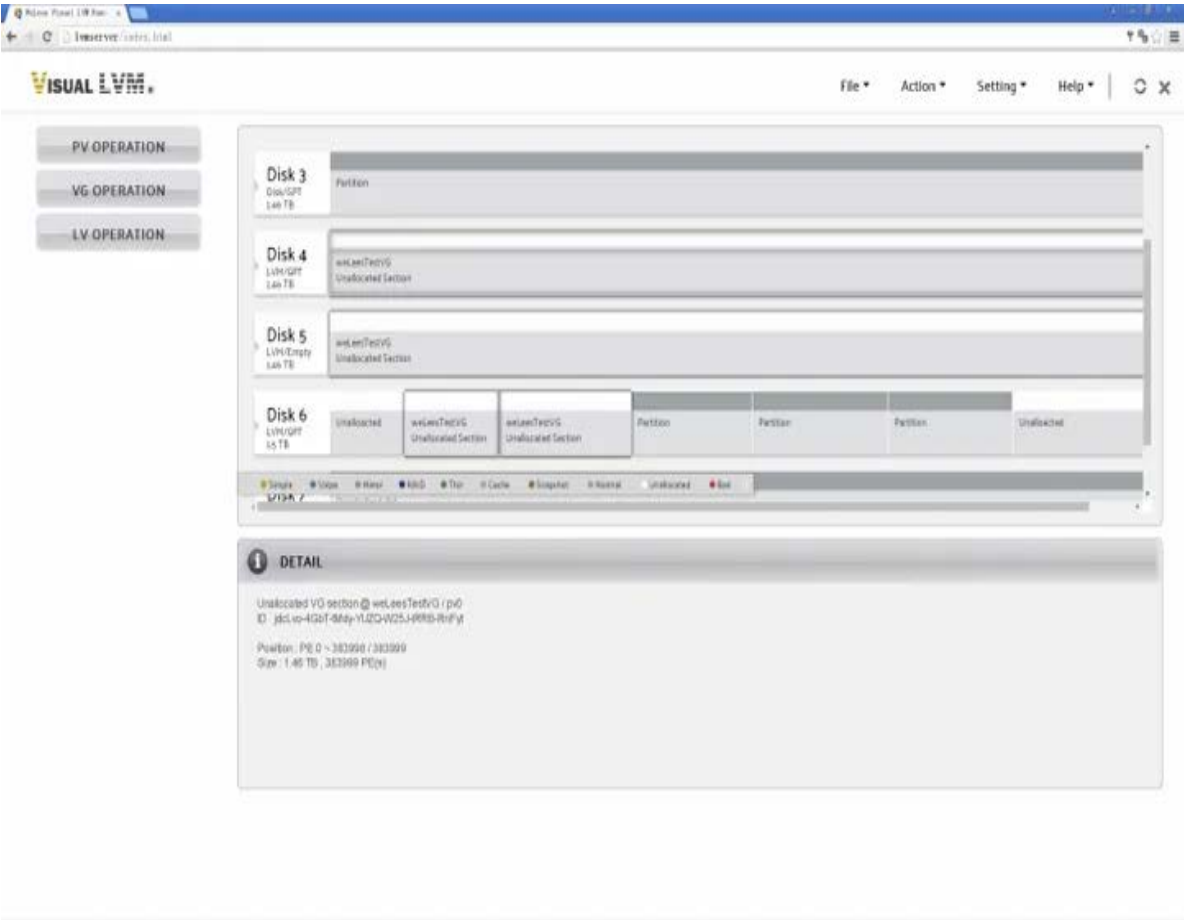
2 partitions on disk 6

### 3. VG

VG is the management unit of LVM and is used to configure a group of LVs and PVs.

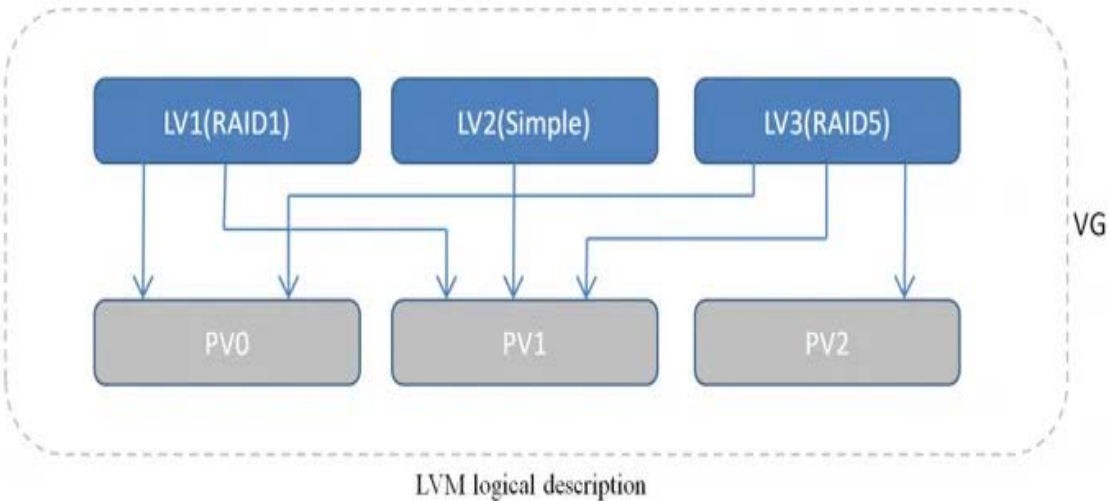
VG consists of one or more PVs. All PVS that make up a VG can be migrated to another machine without additional configuration. One or more LVs can be created in a VG.

There can be multiple VGs in a system.



The VG consists with disk4/part1, disk5, isk6/part1 & disk6/part2

All PVs belong to a VG are highlighted when the mouse points over a member of the VG for user identification

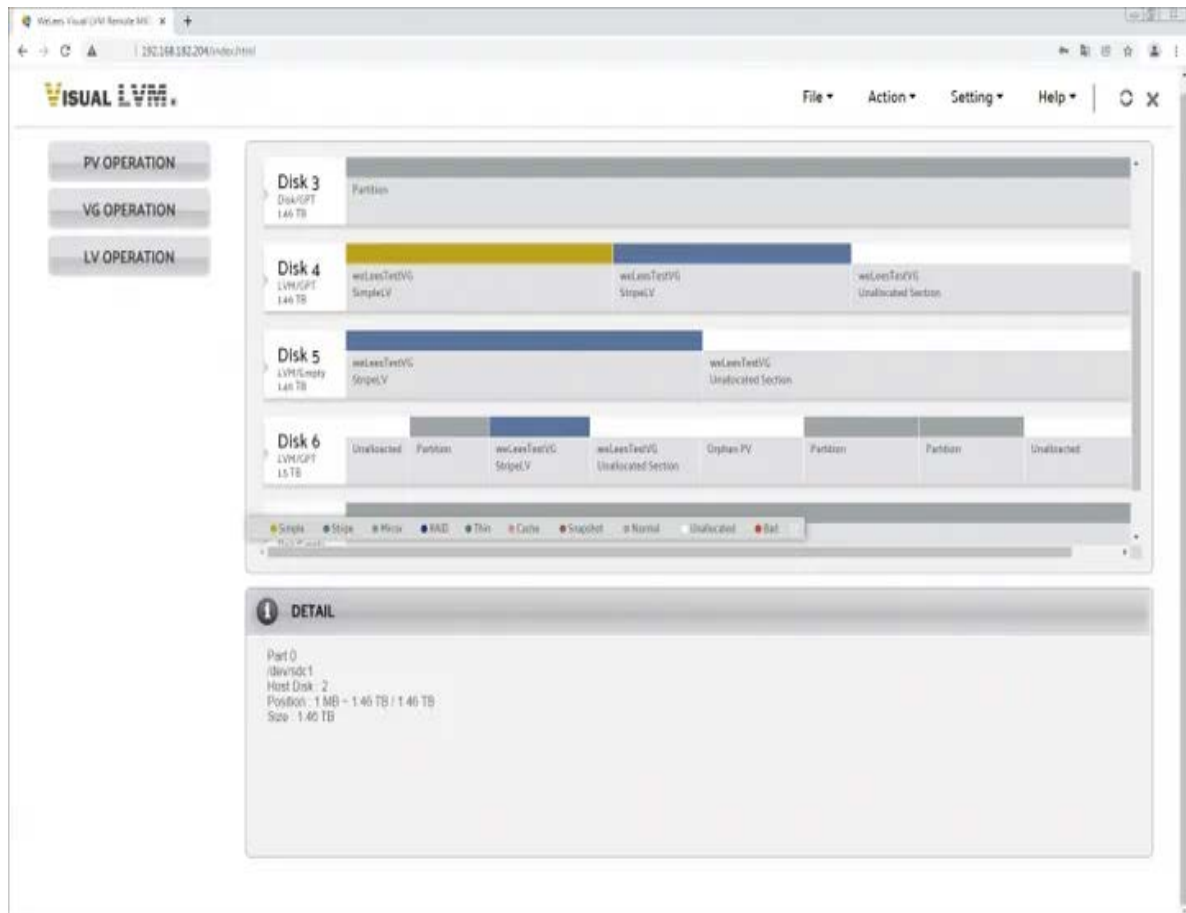


LVM logical description

In the LVM model, VG is responsible for organization and management, LV interacts with users, and PV interacts with storage devices.

## 4. LV

LV is a volume created based on user requirements. It can be mounted directly. LVM provides users with a wealth of features, ranging from high access performance to high security. We'll cover the various features in more detail later.



VG with 2 LVs, one is simple type, and the other is Stripe/RAID 0 type

## 5. PE

PE, Physical Extent is a unit of PV space allocation, similar to a disk cluster. The default value is 4MB. Users can set different values as needed.

## 6. LE

LE, Logical Extent is the space allocation unit of an LV. The size of a LE is the same as PE in a VG. When users create LV, the size of LV must be an integer multiple of LE.

## OPERATIONS

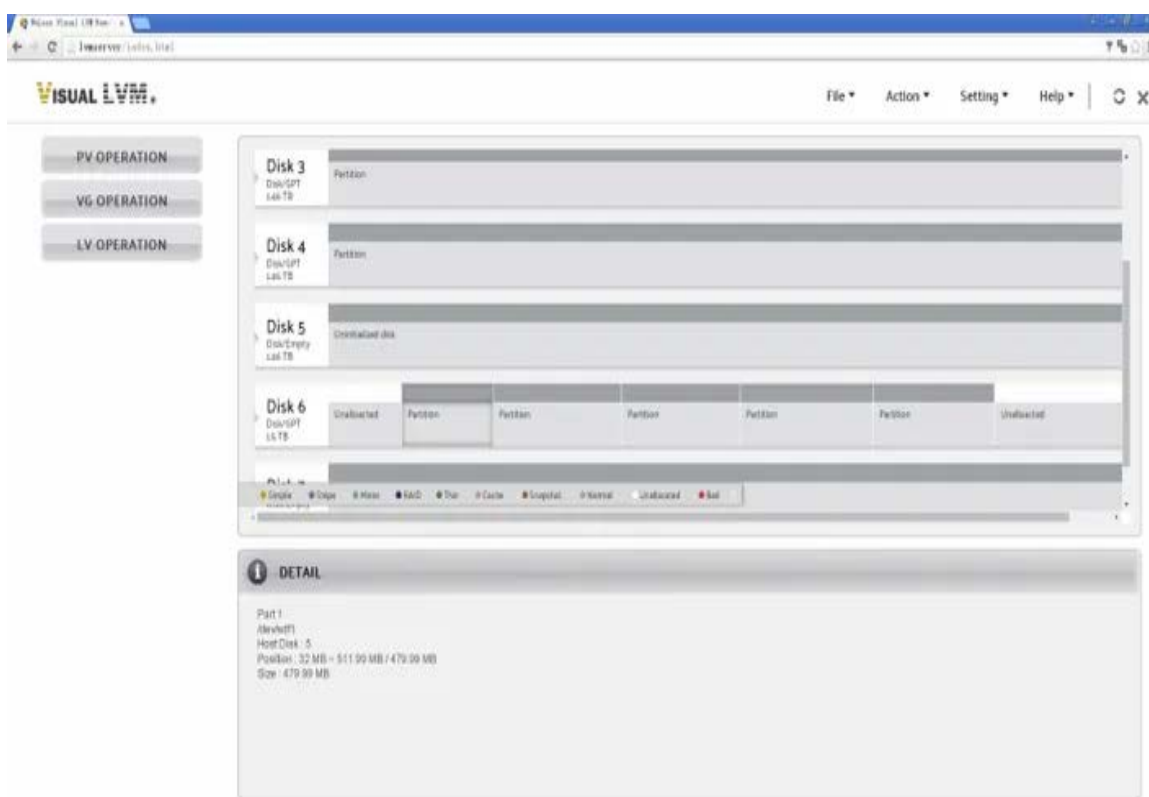
To use LVM, you first need to create, then manage, delete, and so on. Let's explain how to do each of these in turn.

### 1. PV Operations

#### 1.1. Create PV

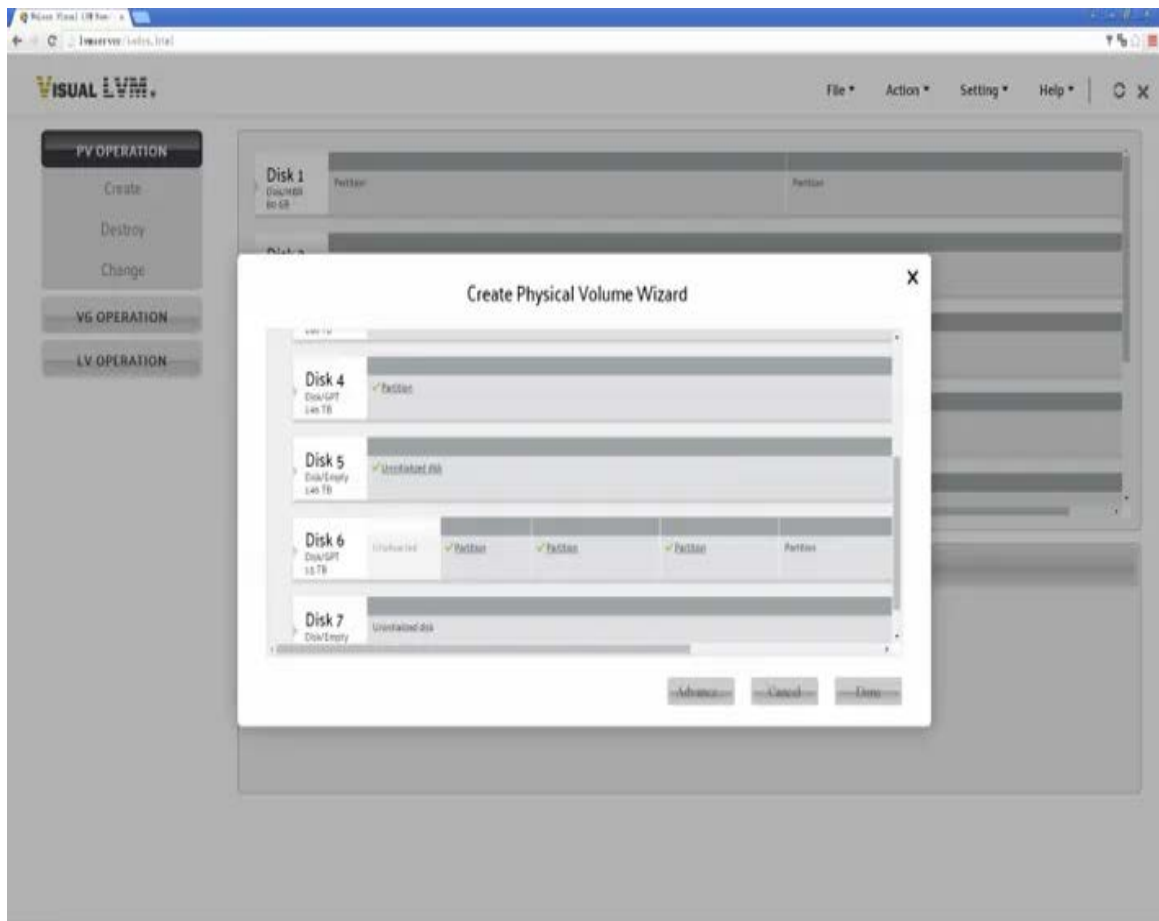
PV is the lowest level of LVM, and to use LVM you must start by building PVs.

First, a "clean" system looks something like this.



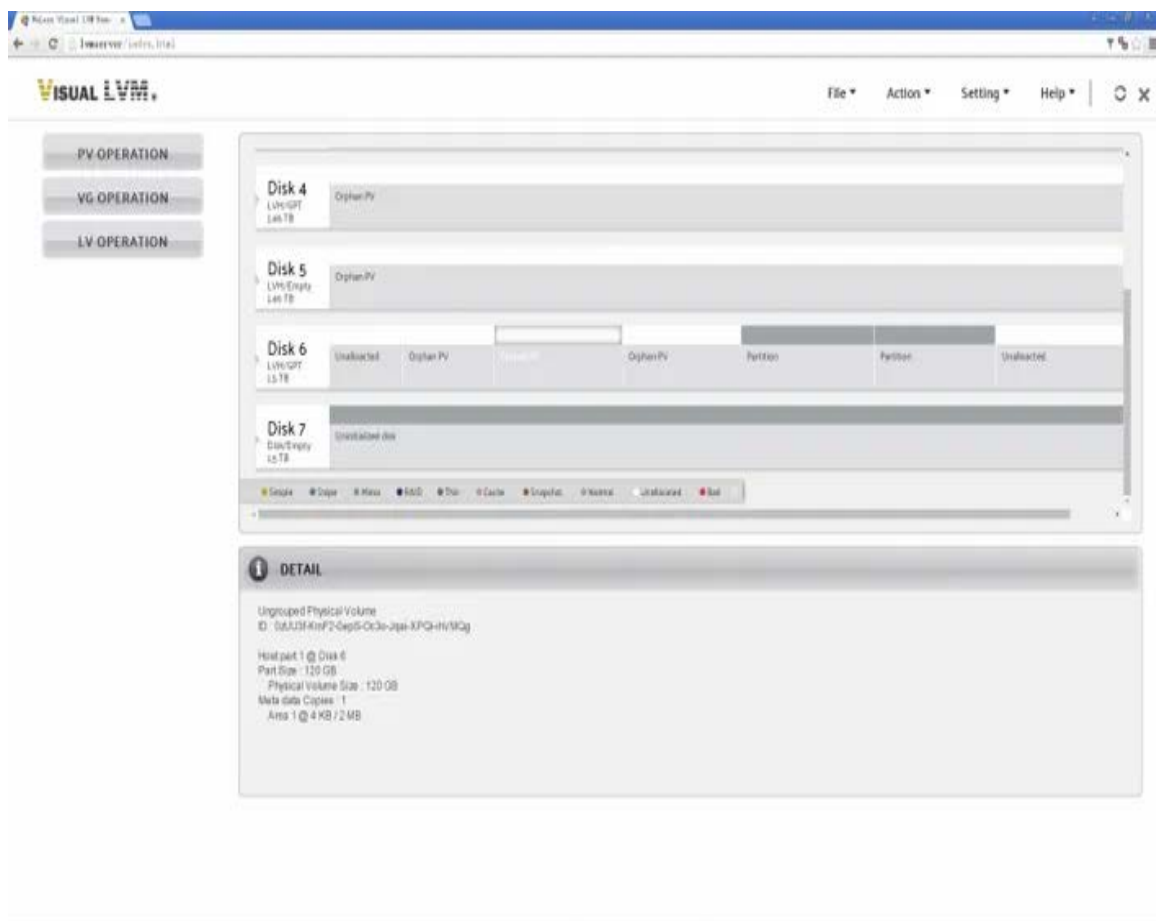
Disks without LVM configuration

Let's create PVs with /dev/sdd1, /dev/sdf1, /dev/sdf2, /dev/sdf3 & /dev/sde.



Select disk/partition to create PV(s)

After the creating operation, four PVs appear in the system. At this time, no VG exists in the system, so the four PVs exist in isolation and are marked as orphan PV in GUI.

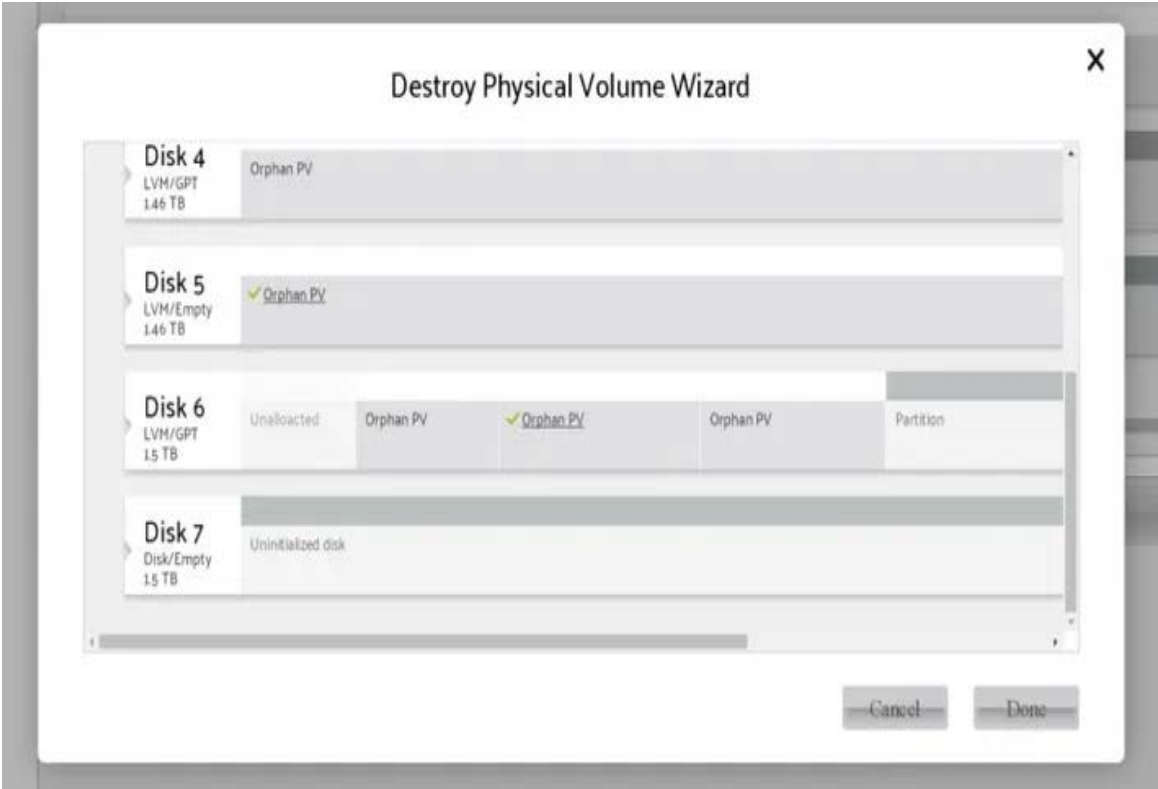


The GUI showing PV information when mouse pointer over the `/dev/sdf2`

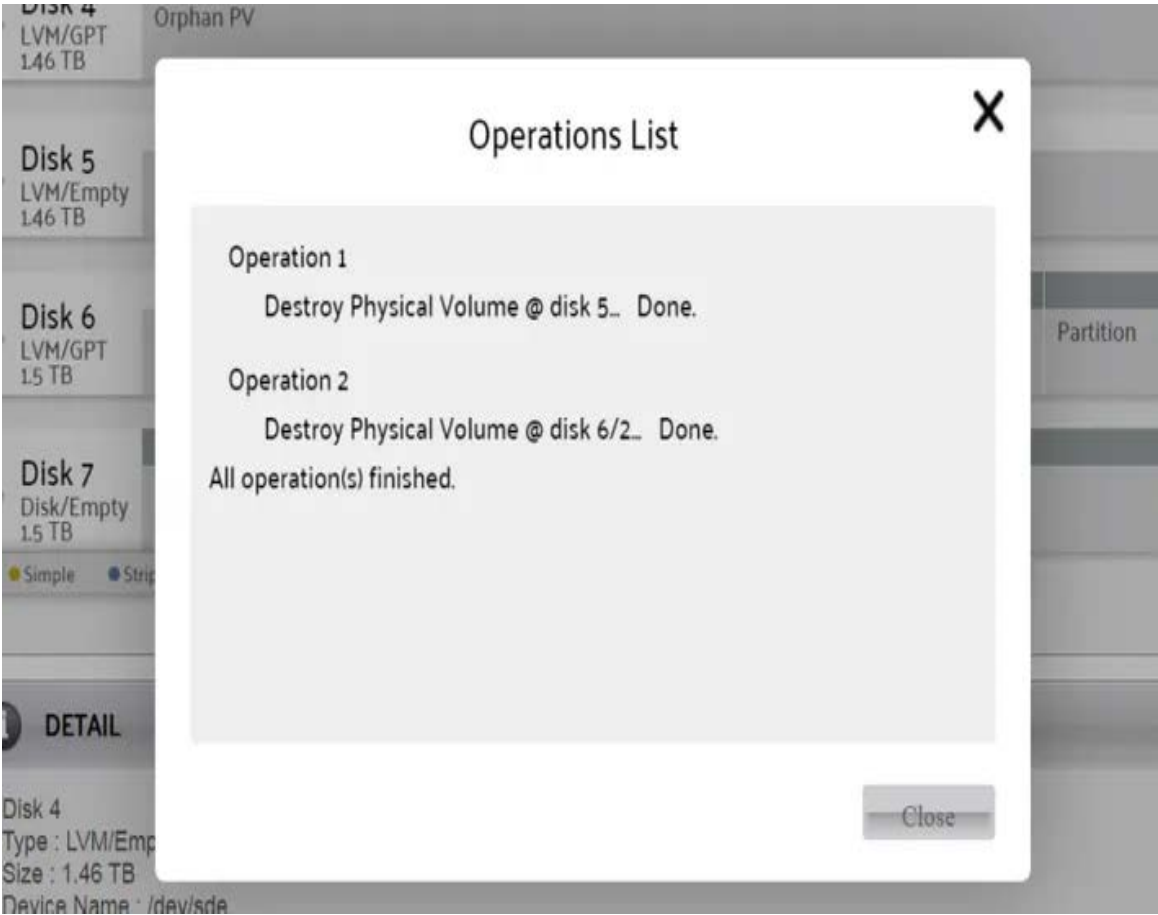
## 1.2. Delete PV

Selecting PV OPERATION->Delete on menu, and click the PV user want to delete.

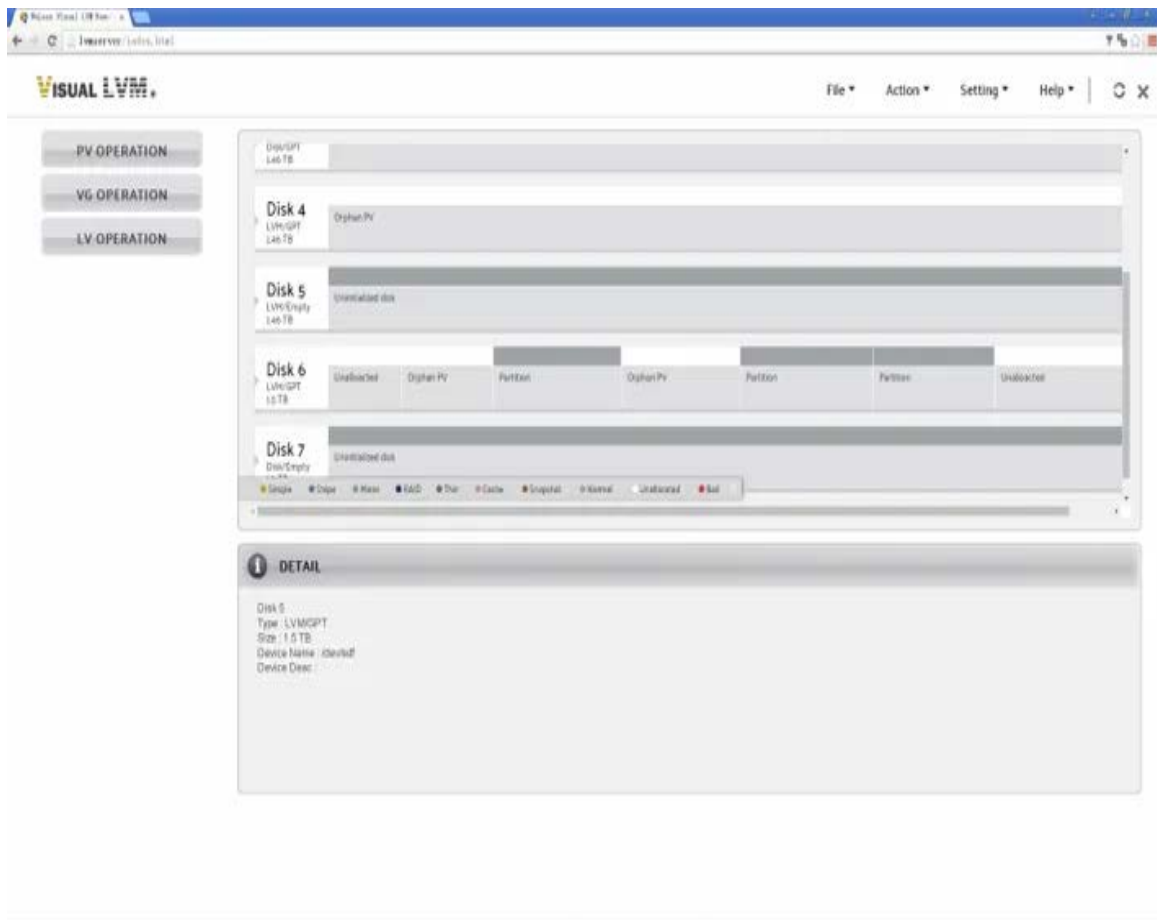




Delete 2 PVs -- /dev/sde & /dev/sdf1



## Operation completed

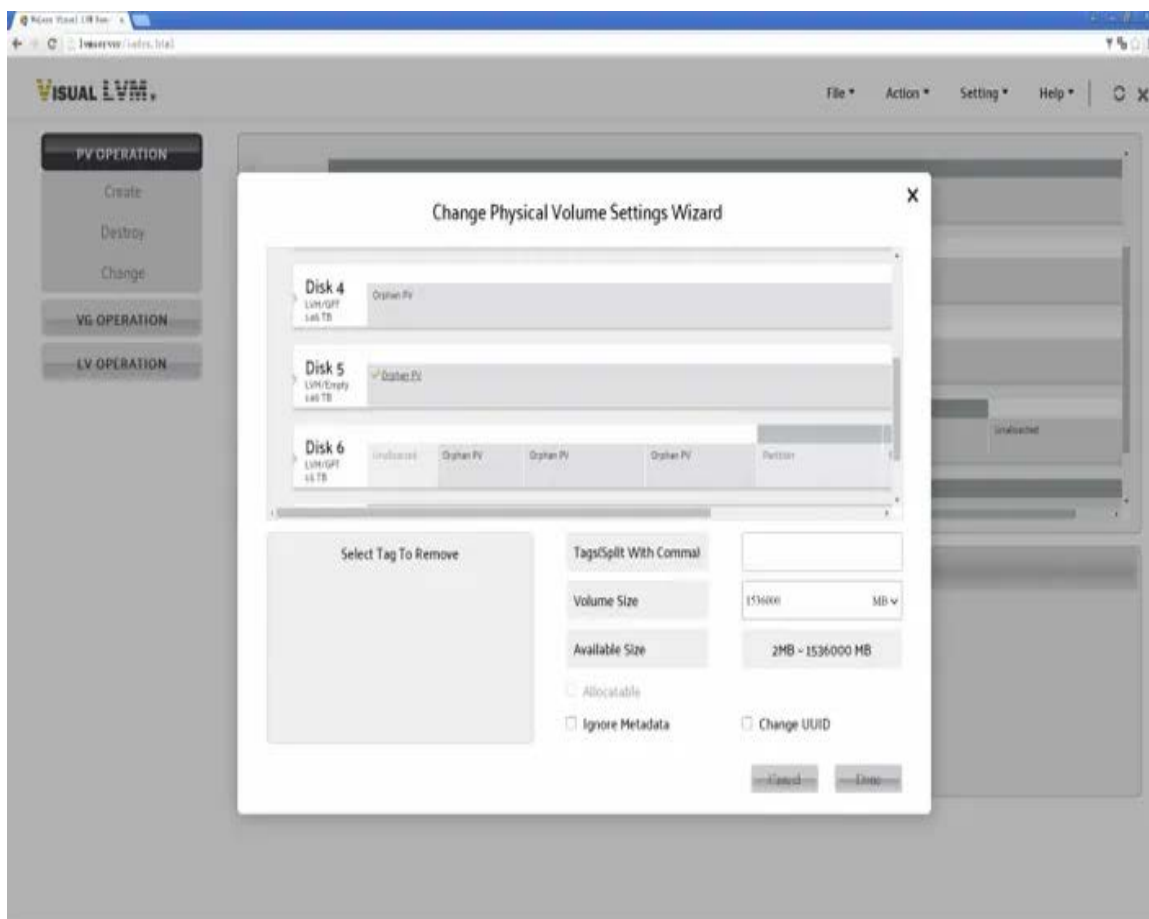


Here we can see the selected PVs are normal partitions

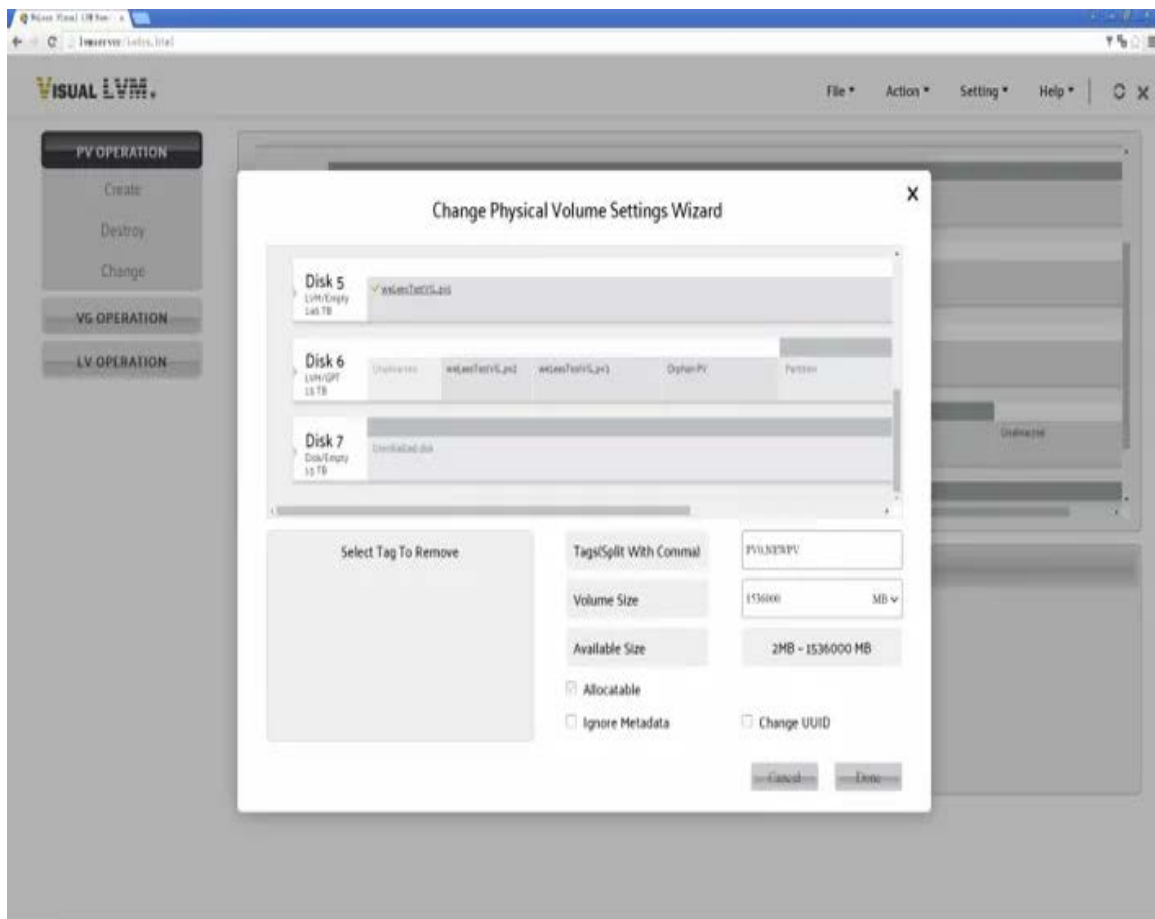
A PV belongs to a VG cannot be deleted. If you want to delete a PV, you must first remove it from the VG. It can only be removed as an orphan PV.

### 1.3. Change configuration of PV

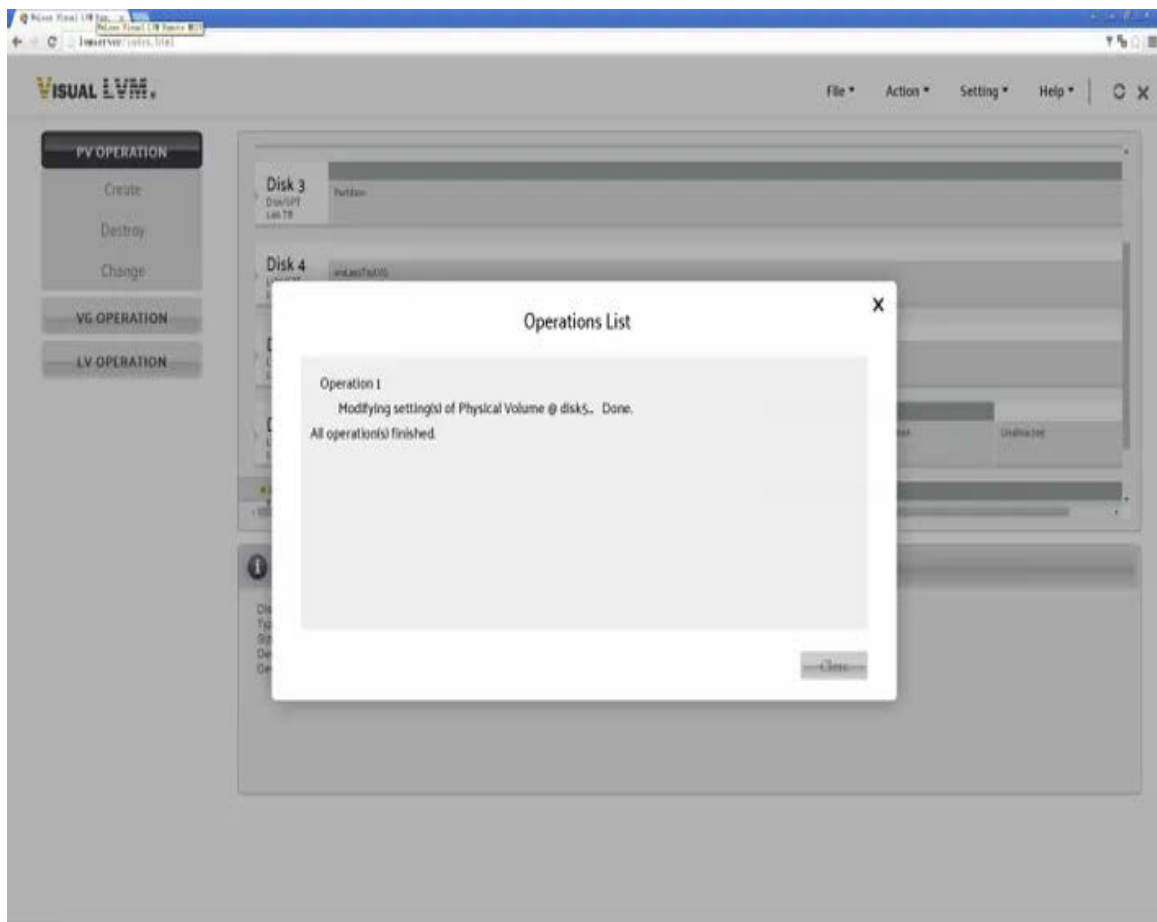
The configurations that can be modified are different for PVs that are added to VG and orphan PVs. Since there is no metadata in orphan PV, only the copies of metadata, metadata area size, and boot area size can be modified.



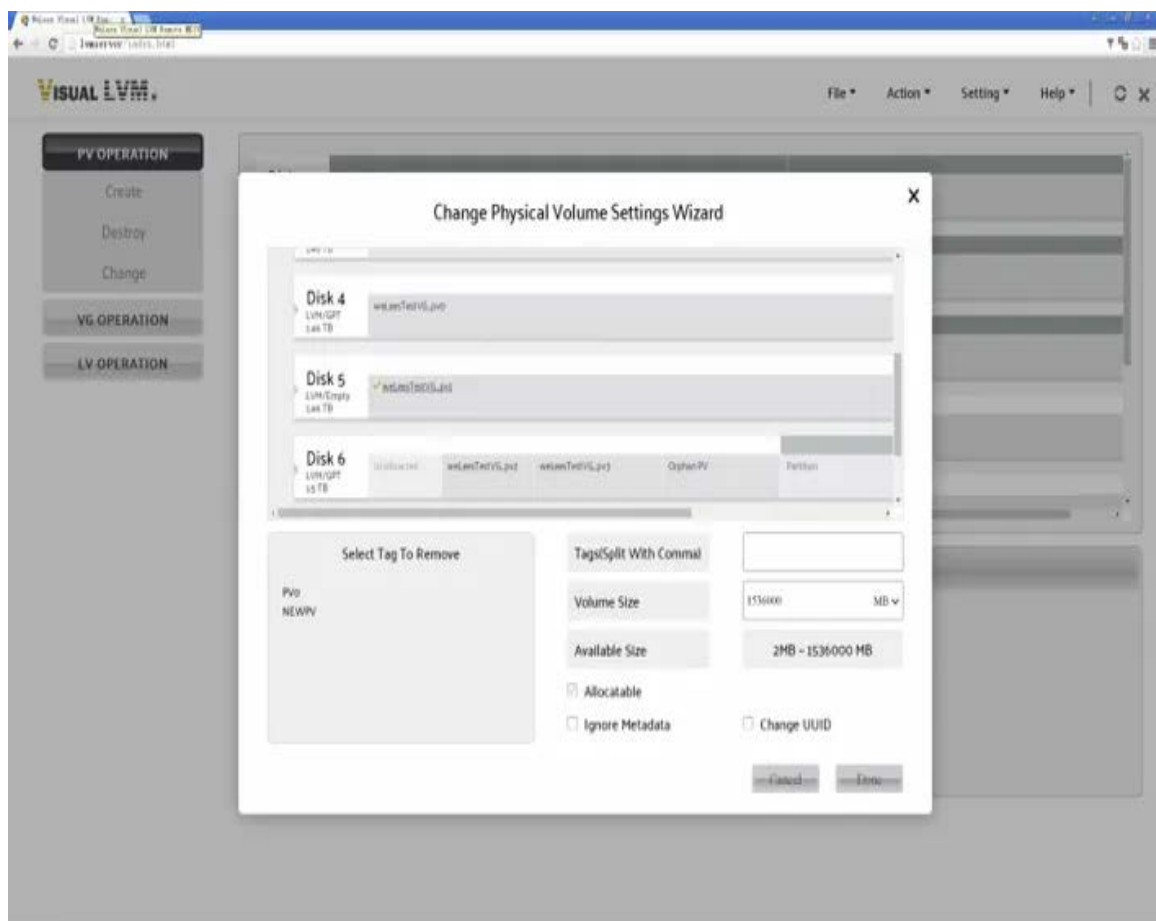
Orphan PV, user cannot change allocation flag and tag



The PV in VG, user can change allocation flag and tag



Operation completed

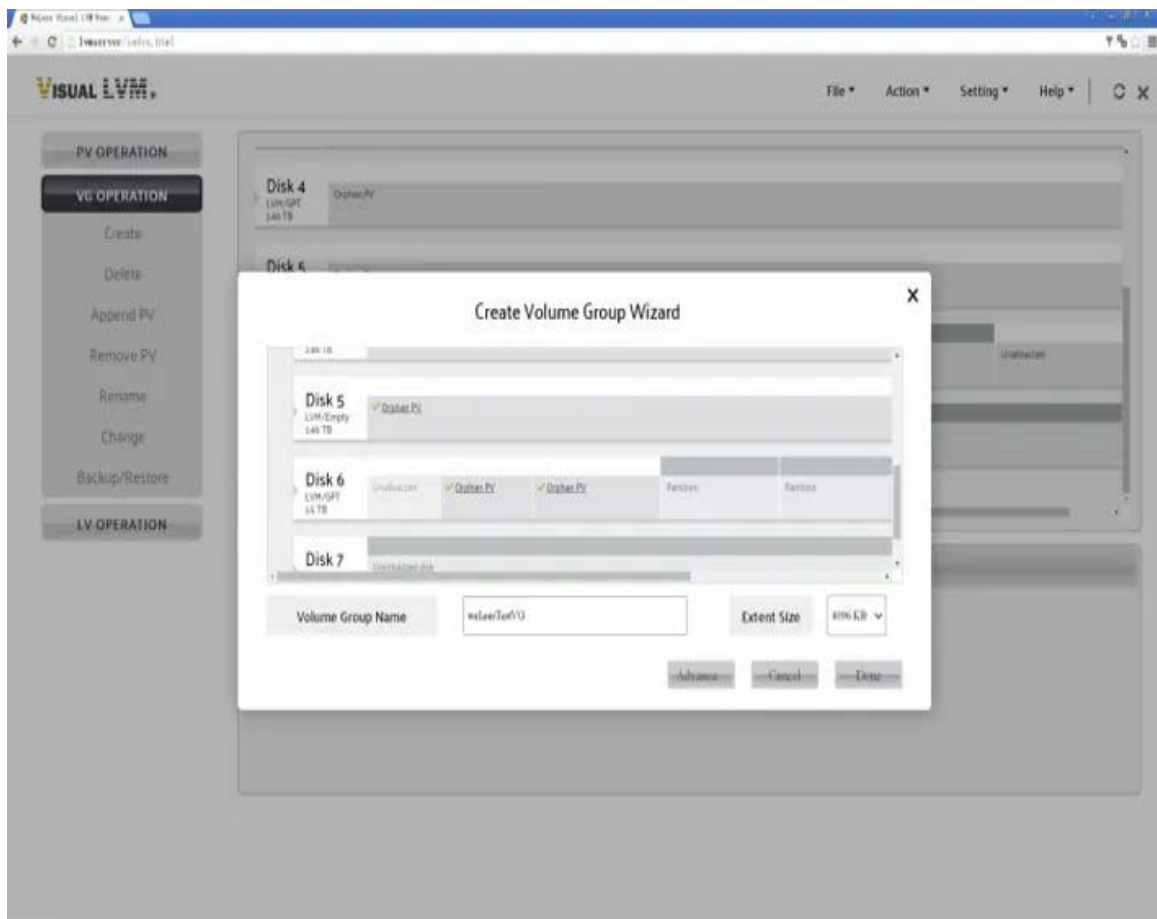


After refreshing, we can found the tags in PV

## 2. VG Operations

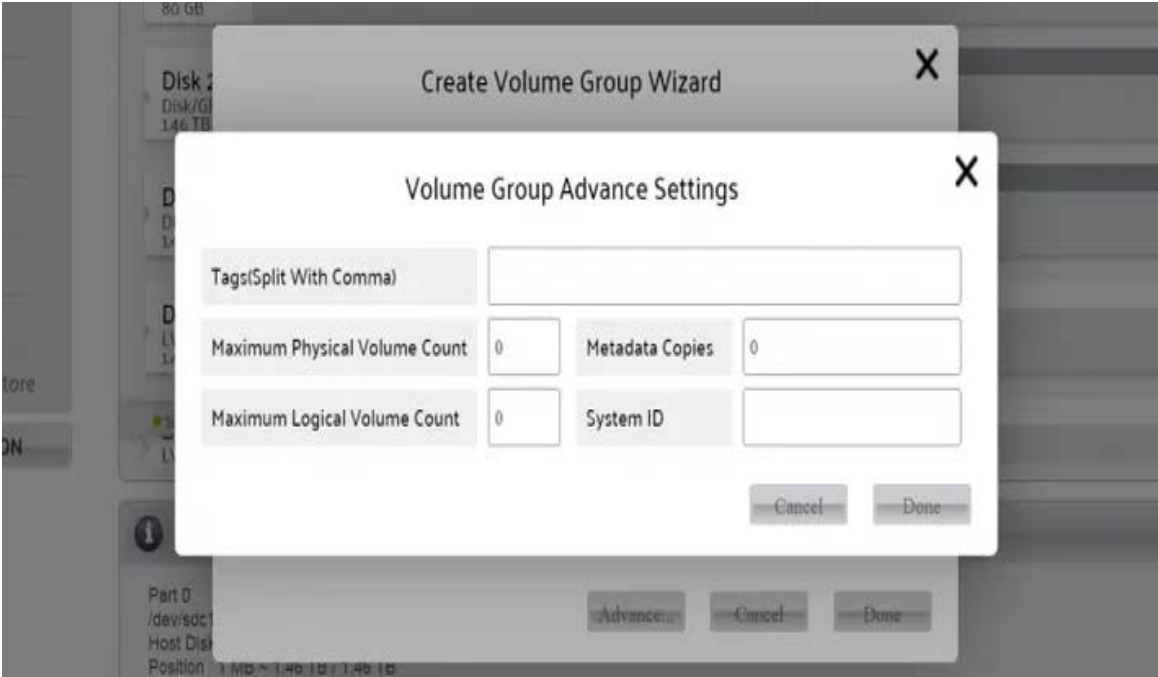
### 2.1. Create VG

Selecting VG Operation->Create, and create VG with all 4 PVs.



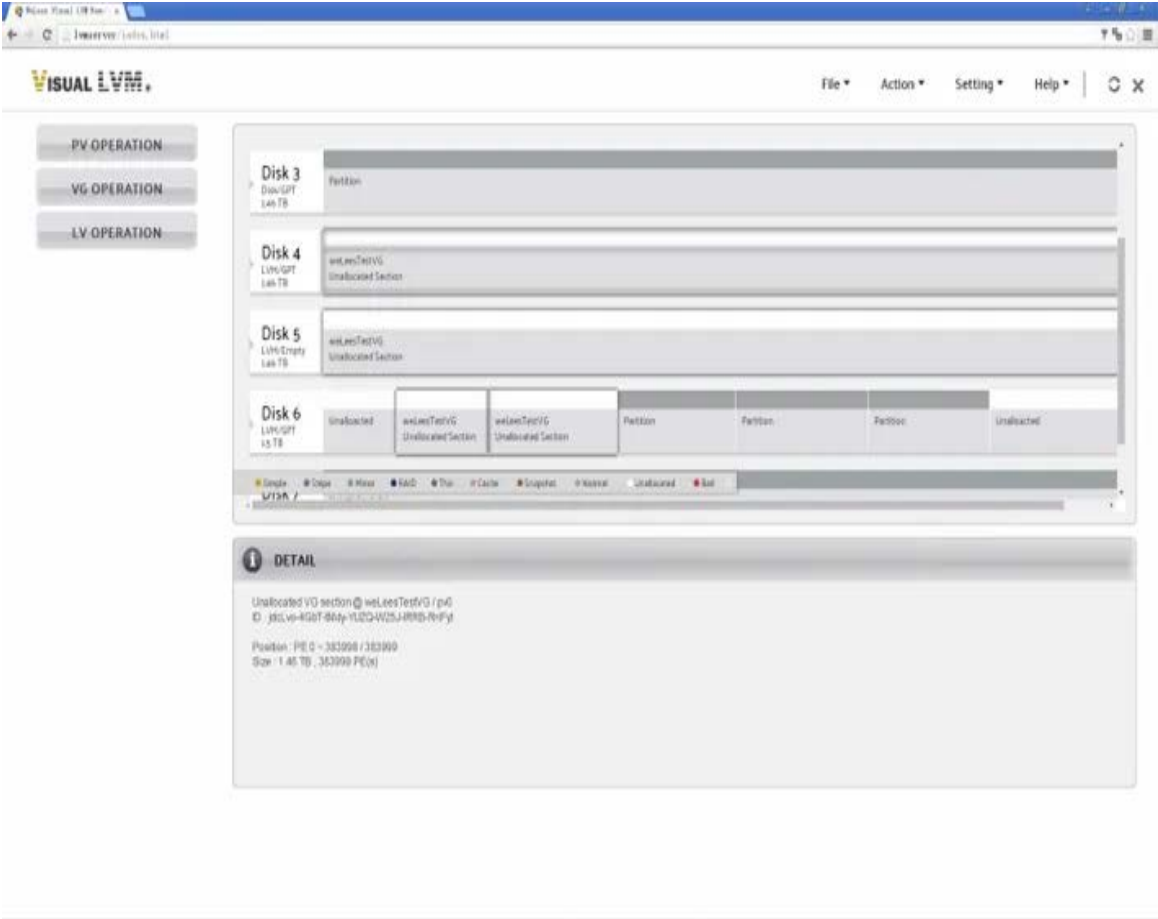
Selecting all 4 PVs and specify the size of LE, name of VG. Use can click Advance button to configure advance settings

User can specify the tags of VG, maximum PV/LV count and the copies of metadata settings in advanced setting of creating VG dialog.



Advanced setting of creating VG

After the VG created we can get the layout.

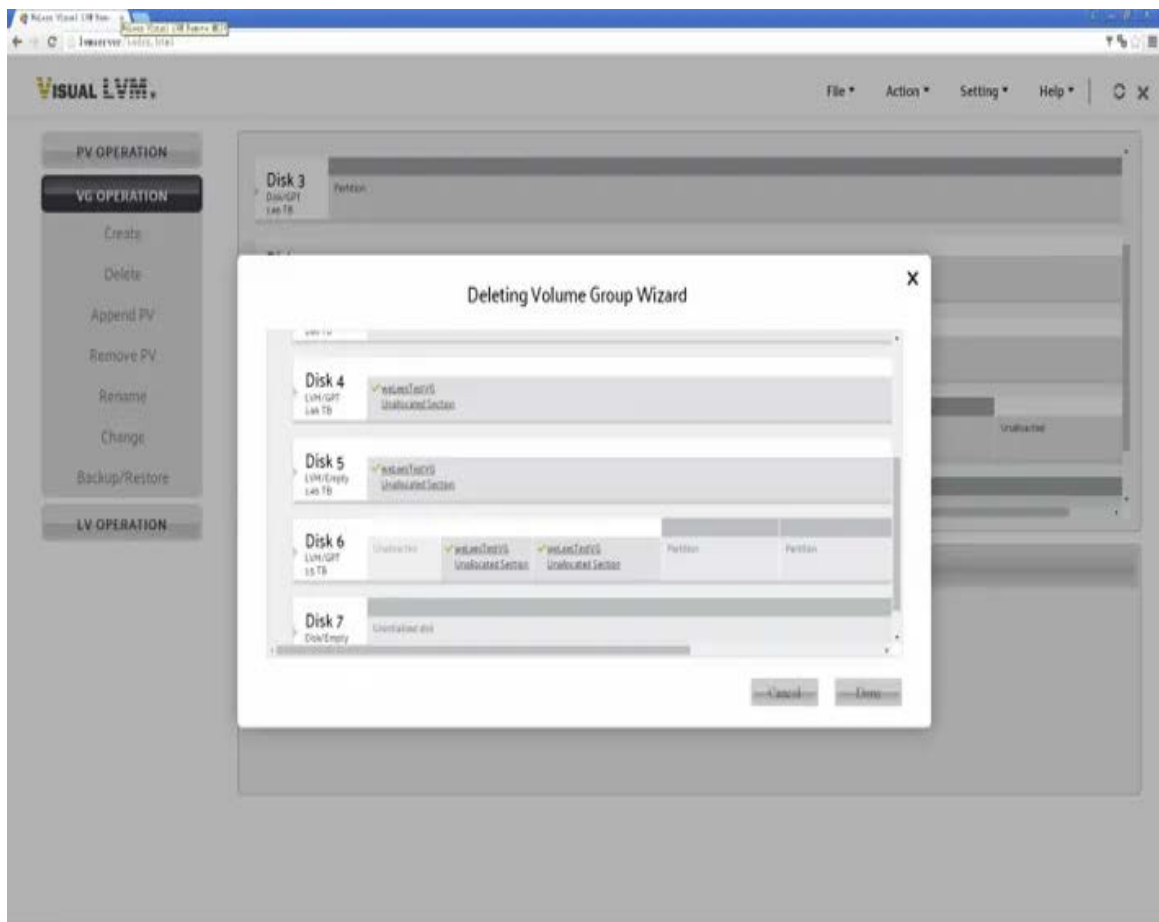




There are 4 PVs in VG weLeesTestVG. The PVs belong to the same VG will show highlight to prompt user when mouse pointer over any PV.

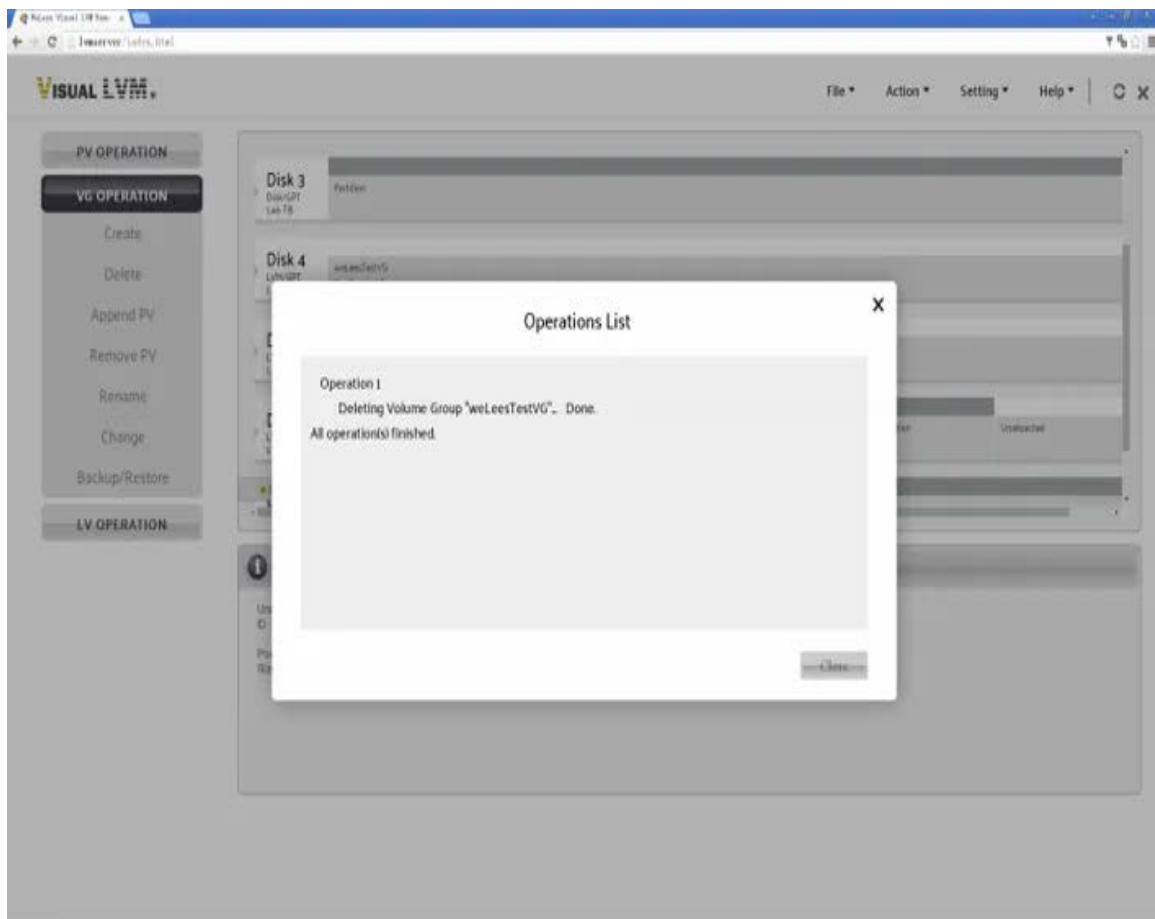
## 2.2. Delete VG

Selecting VG OPERATION->Delete



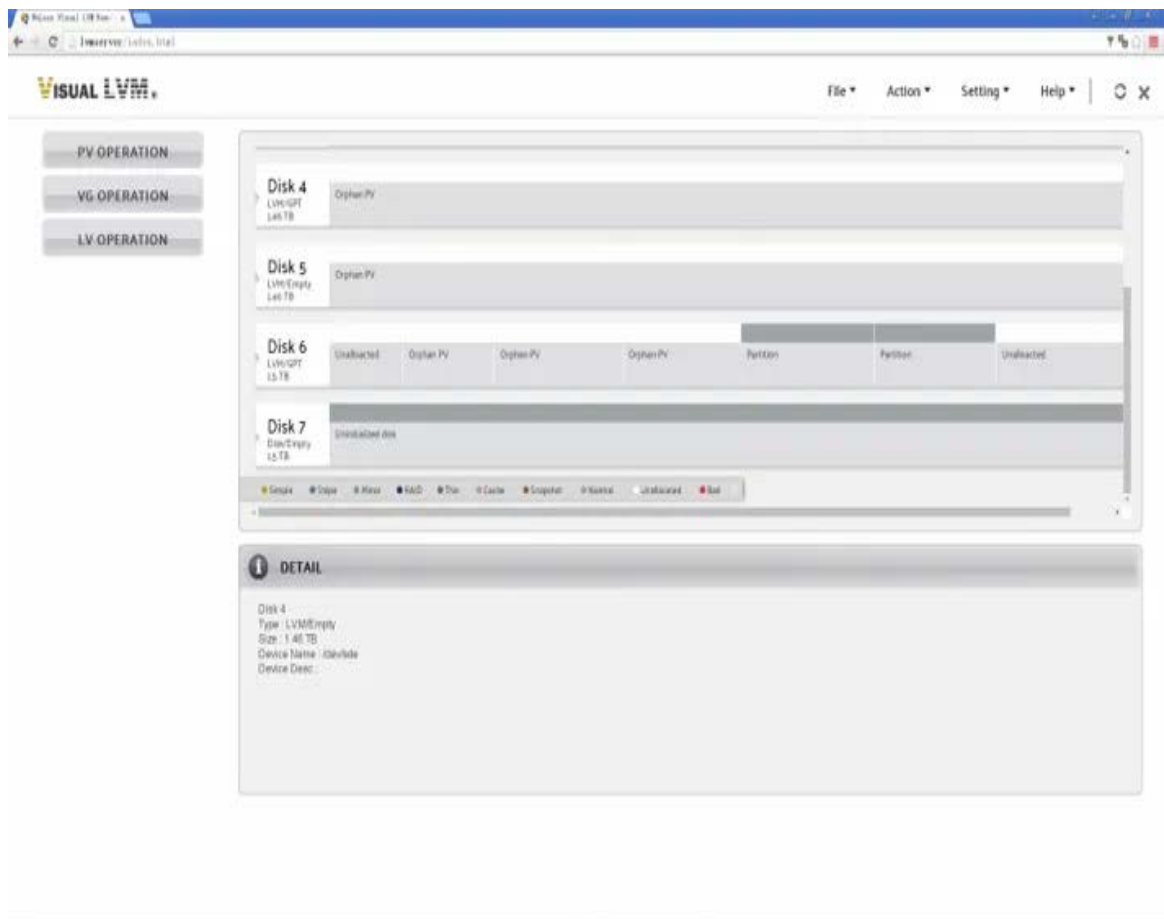
Delete VG dialog

Click any PV in VG you want to delete, and then press the Done button to delete the VG. Note That to protect user data, VG with LV cannot be deleted. Users need to delete all LVs in VG before deleting it.



VG deleted, click Close button to refresh information

At this point, we can see that the 4 PVs have returned to the status of orphan PVs.



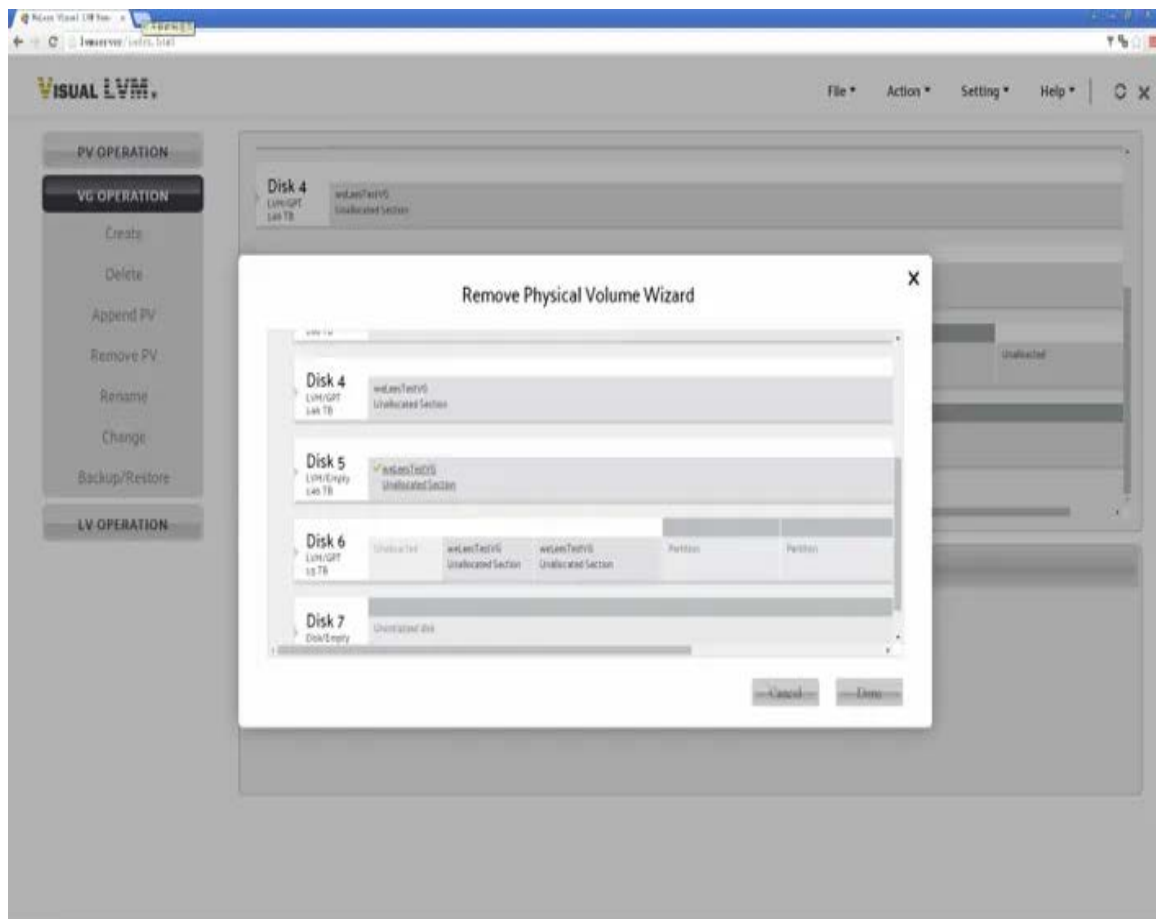
A LVM system without VG

## 2.3. Remove PV

Let's recovery the VG first.

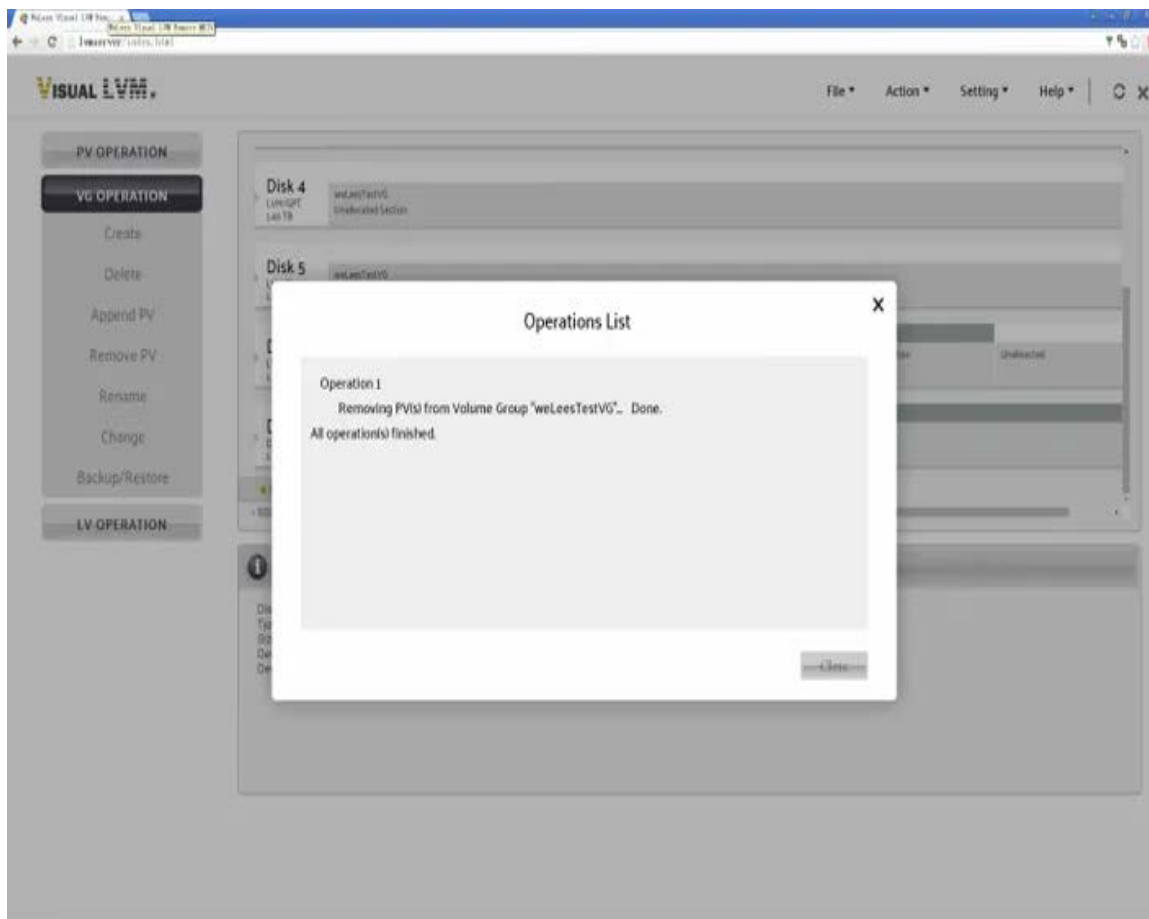
This operation should have been placed after the appending PV, but I changed the order to reduce the operations.

Choose VG OPERATION->Remove PV



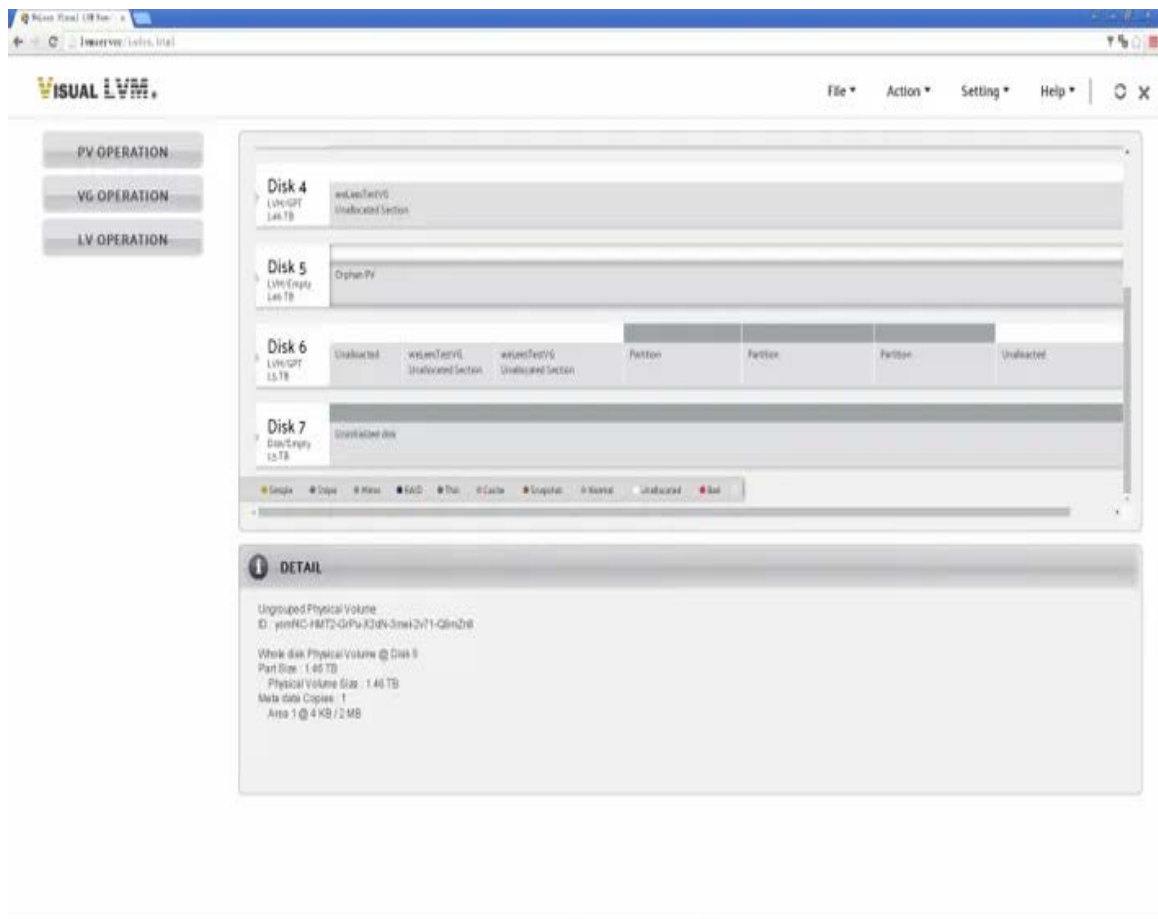
### Remove PV dialog

Selecting disk5 and click done button to remove it.



Operation completed

We can see the disk5 is a orphan PV now. It does not belong to VG weLeesTestVG.



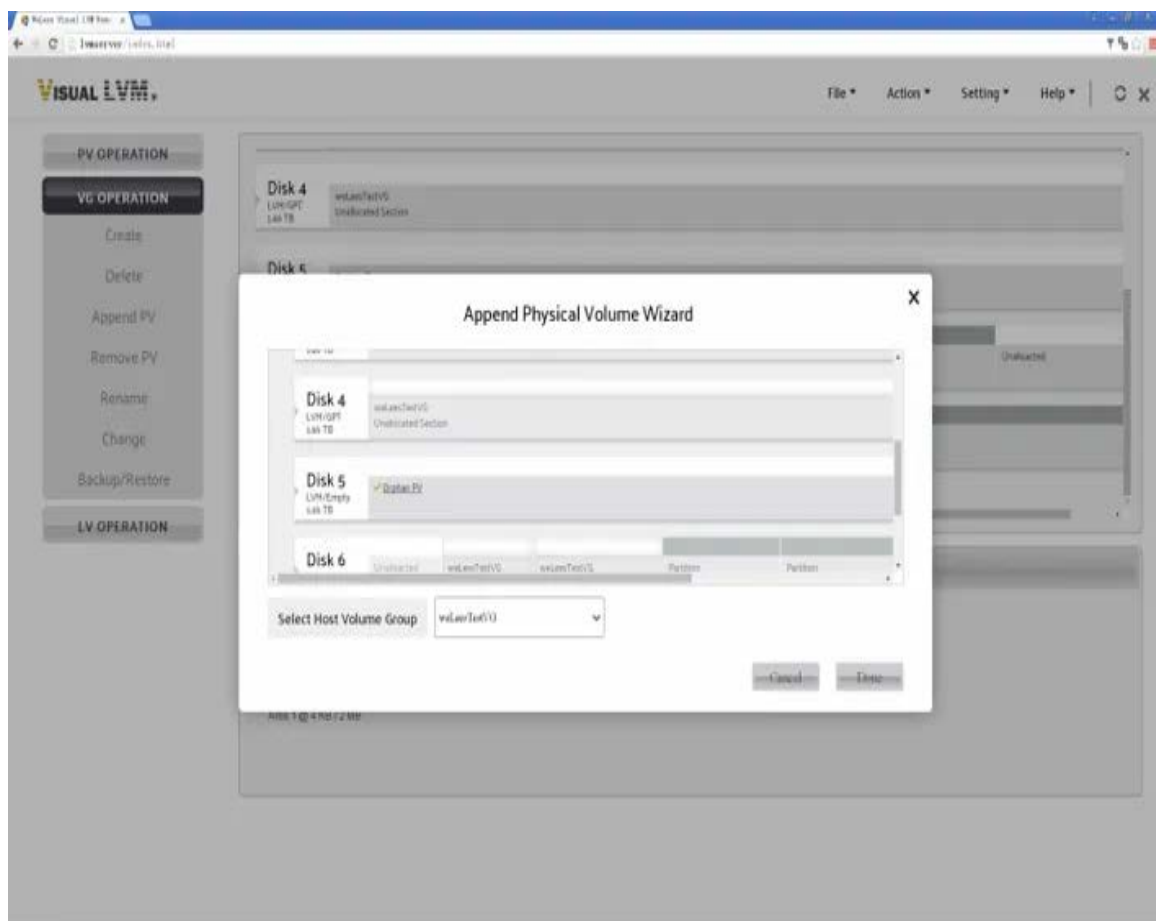
The VG with one PV removed

## 2.4. Append PV

After adding a new device, we need to add it to the existing VG, and then we need to use the appending PV operation.

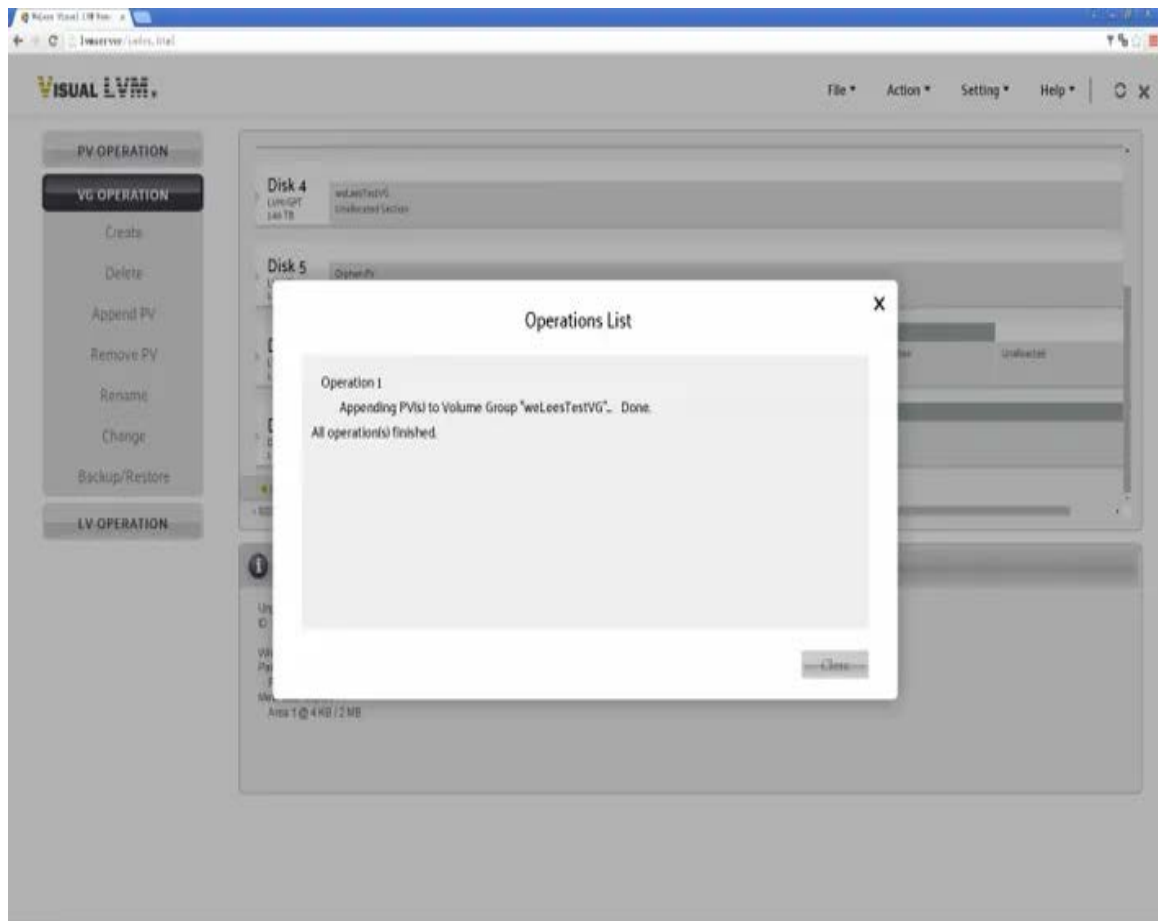
Let's add the PV removed from the previous operation back to the weLeesTestVG.

First, choose VG OPERATION->Append PV



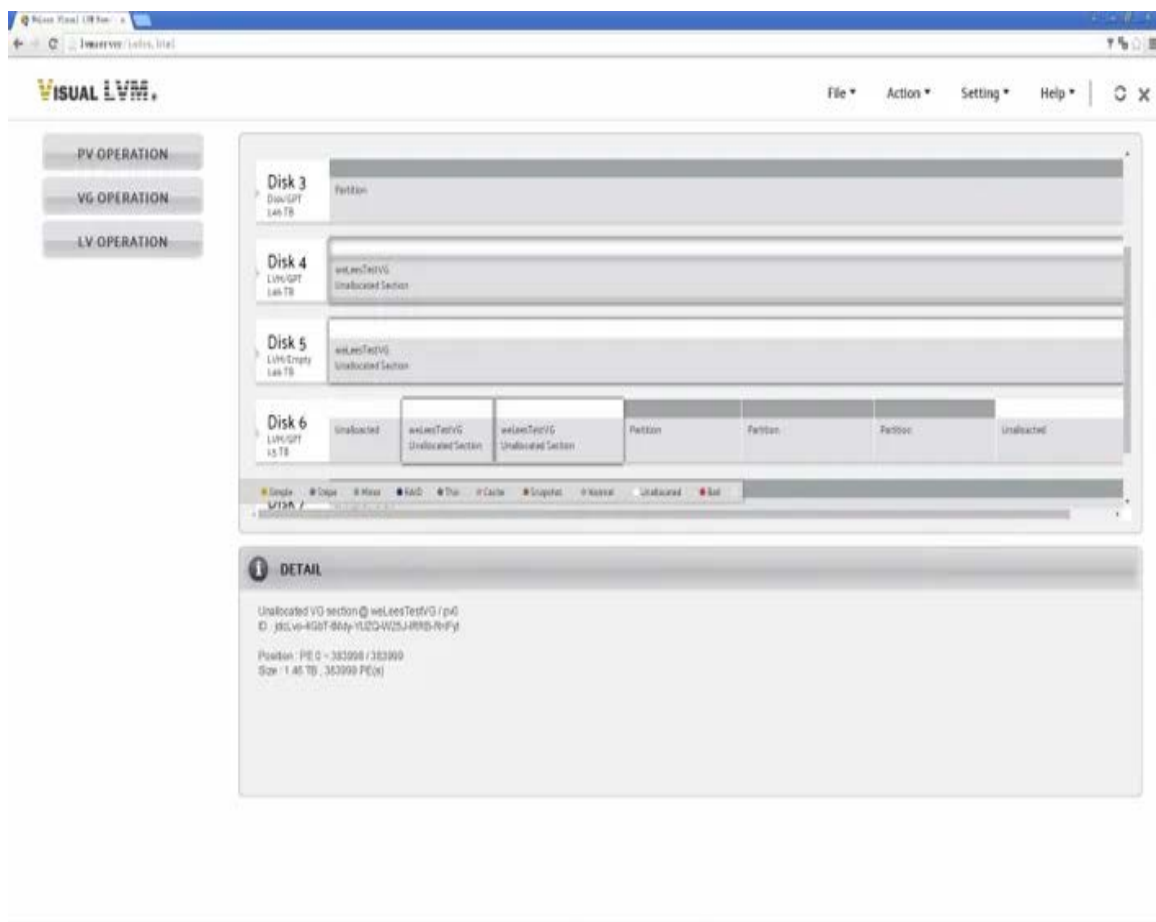
Append PV dialog

Clicking on disk5 for selecting, and select weLeesTestVG in VG list followed(There is only one VG now, of course)



Operation completed

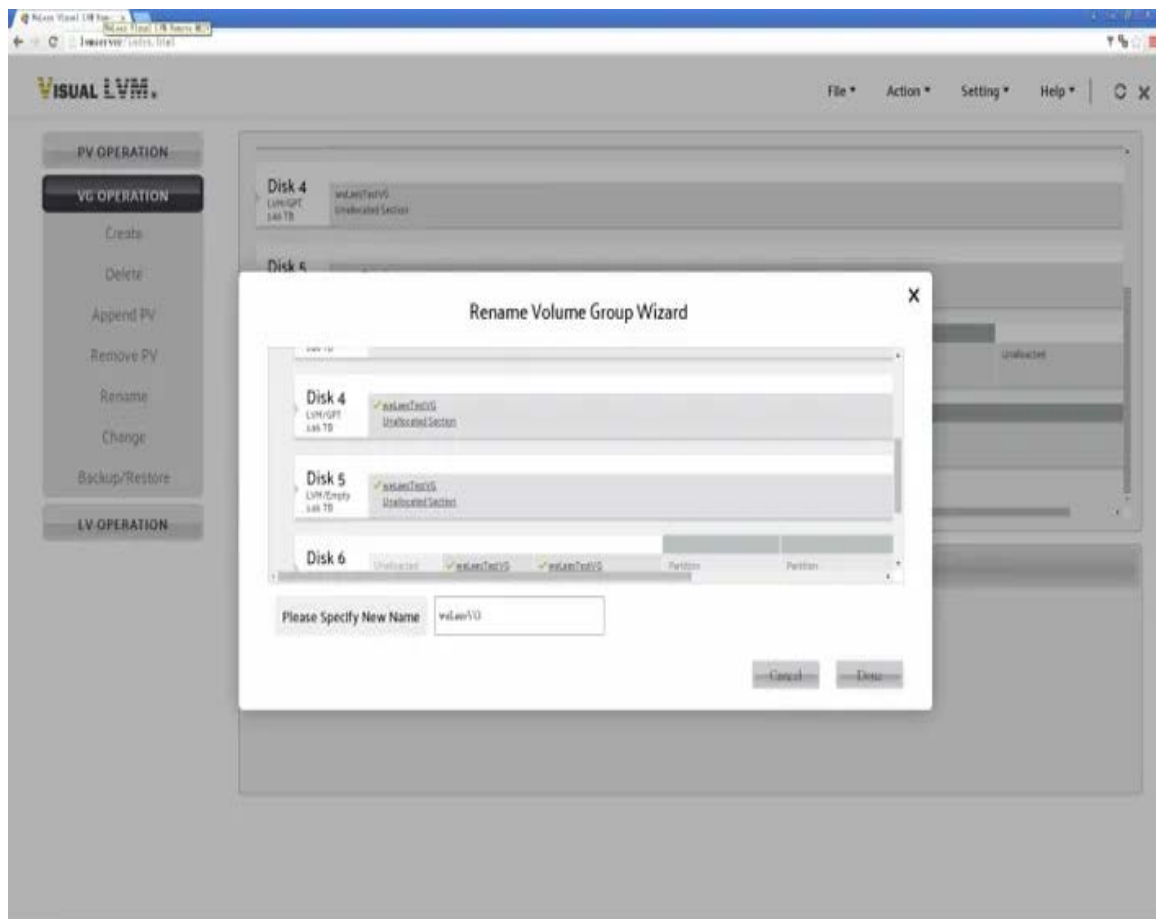




The orphan PV returns to the VG now

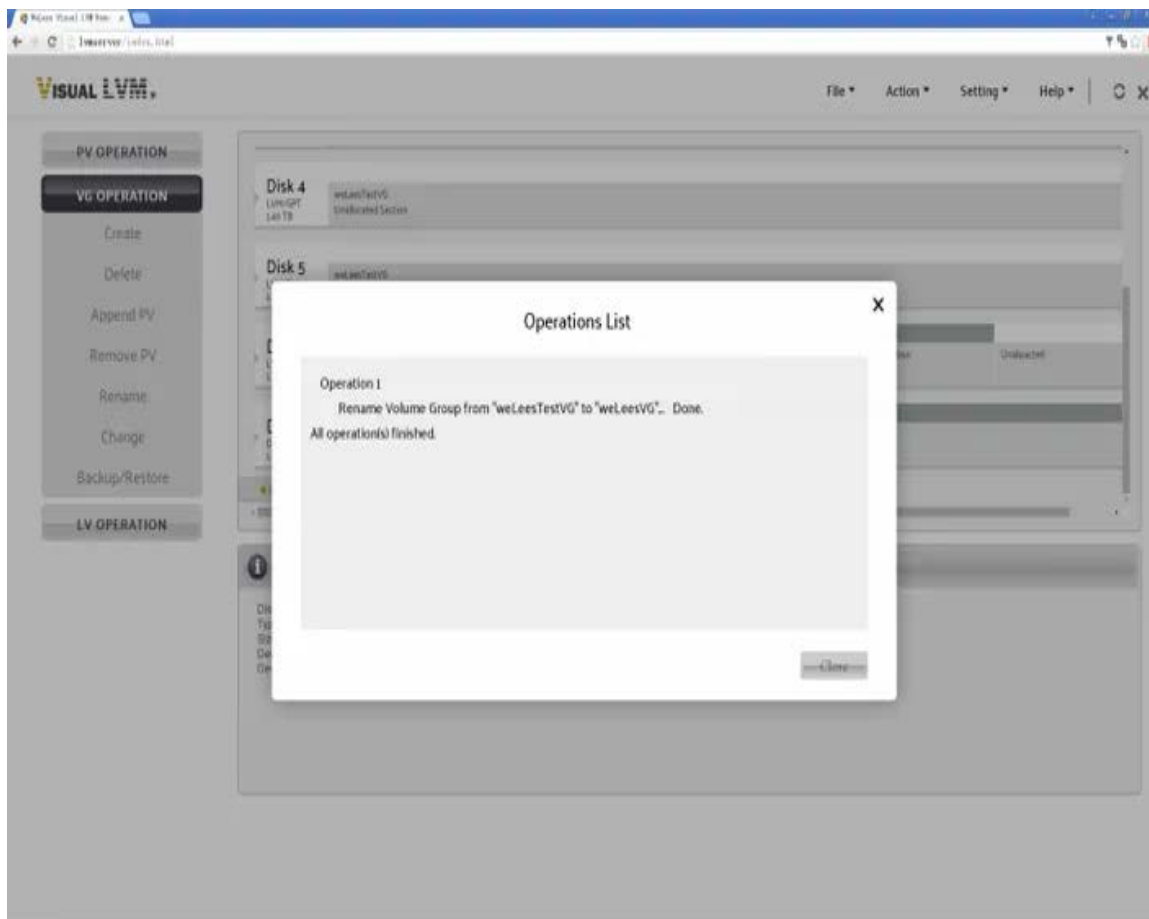
## 2.5. Rename VG

The rename operation is easy, selecting VG OPERATION->Rename.



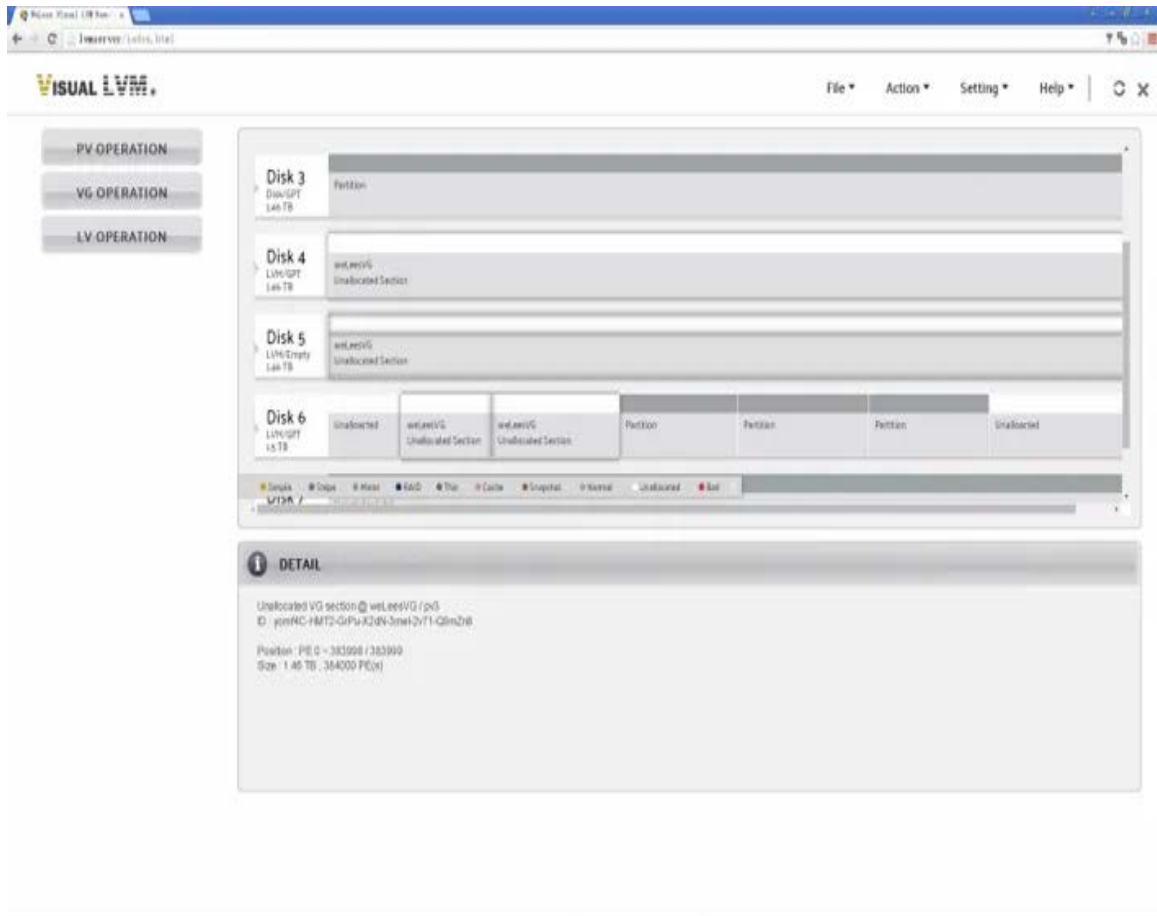
Click the VG we want to rename and type the new name in input box, and click Done button.

CAUTION: The LVM has strict restrictions on VG and LV names. Only a-z, A-z, 0-9, and \_+ characters can be used.



Operation completed

We can see the new name of VG after refreshing.

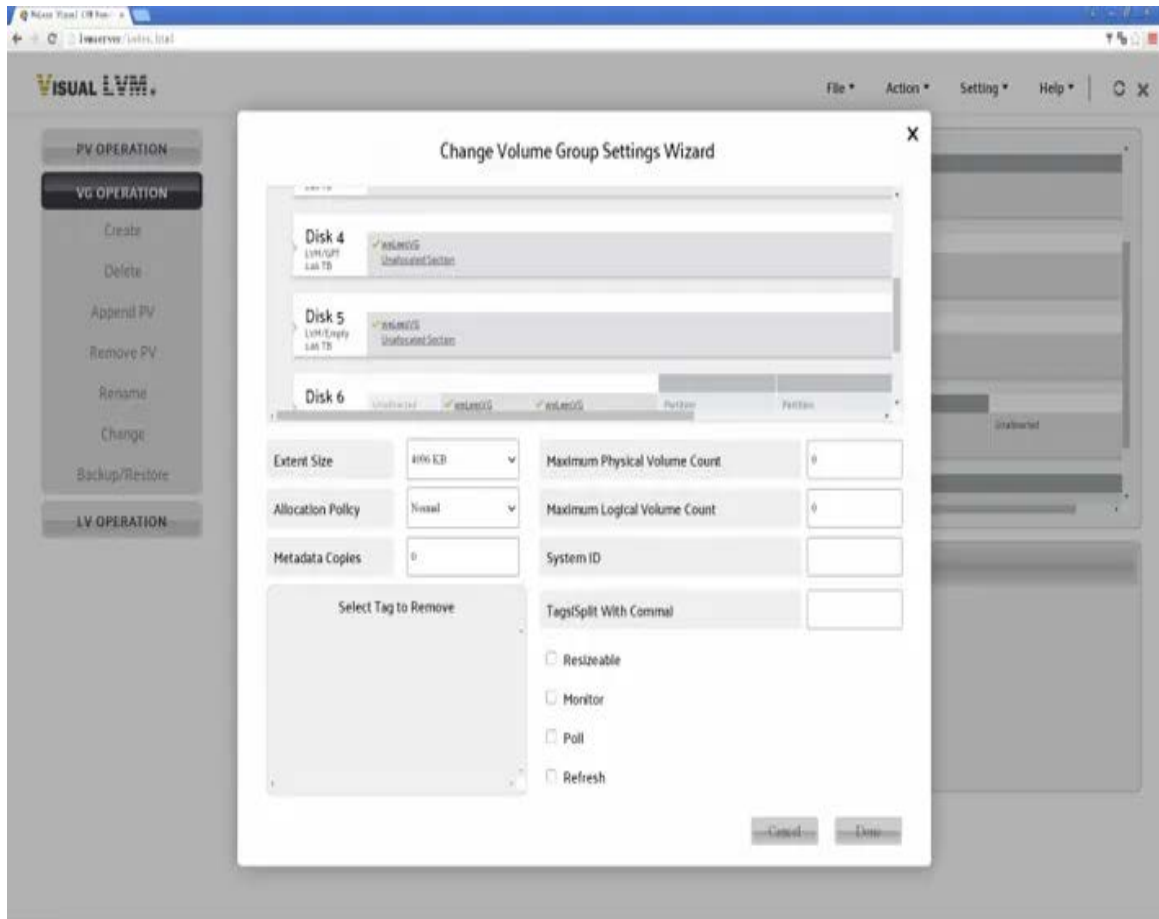


The VG with new name

## 2.6. Change VG configuration

The user may need to change the VG configuration usually, which is complicated on the command line but very simple on the GUI. Now let's look at how to change the VG configuration.

Select VG OPERATION->Change



Change VG configuration dialog

If there is no VG selected, all settings are disabled. Now select weLeesVG. The configurations that can be changed include LE/PE dimensions. Allocation policy, number of metadata copies, maximum number of PVs & LVs and tags, and some configuration related to LVM control.

It is easy to change the PE/LE size to a smaller value. However, to change the value to a larger value, you need to check whether the start position of the LV in the VG can be exactly divided by the desired value. If not, you cannot set the value.

## 2.7. Backup/Restore VG

It's a good habit to back up the VG's configuration, and if something goes wrong, it's much easier to recover with backup data.

Visual LVM has automatic backup function. Users can turn on/off automatic backup function in setting menu. The backup VG configuration information is

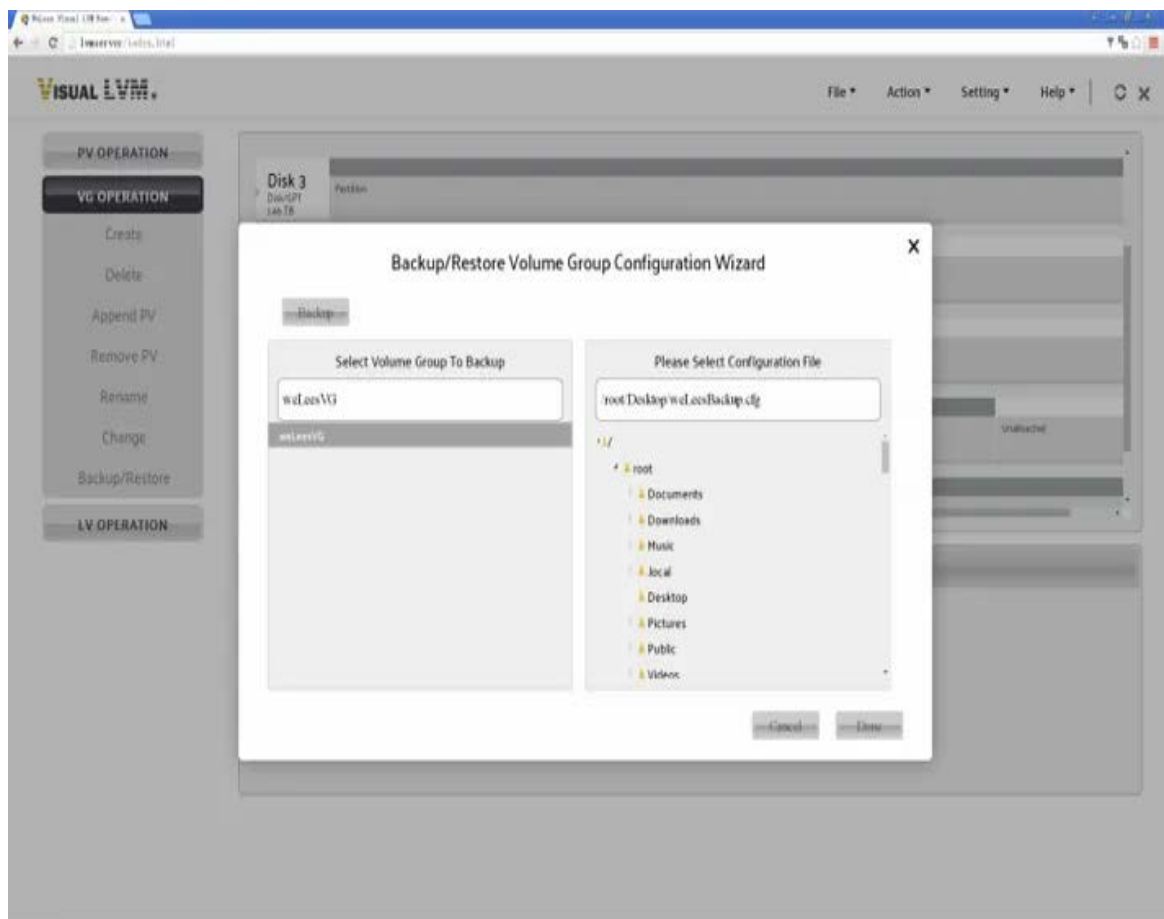
stored in the backup directory under the Visual LVM installation directory. The backup files are named by operation & time.



Setting menu of Visual LVM, user can enable/disable auto-backup, enable/disable access from specified IPs, change login password and port

Now let's talk about how to manually backing up VG configuration information.

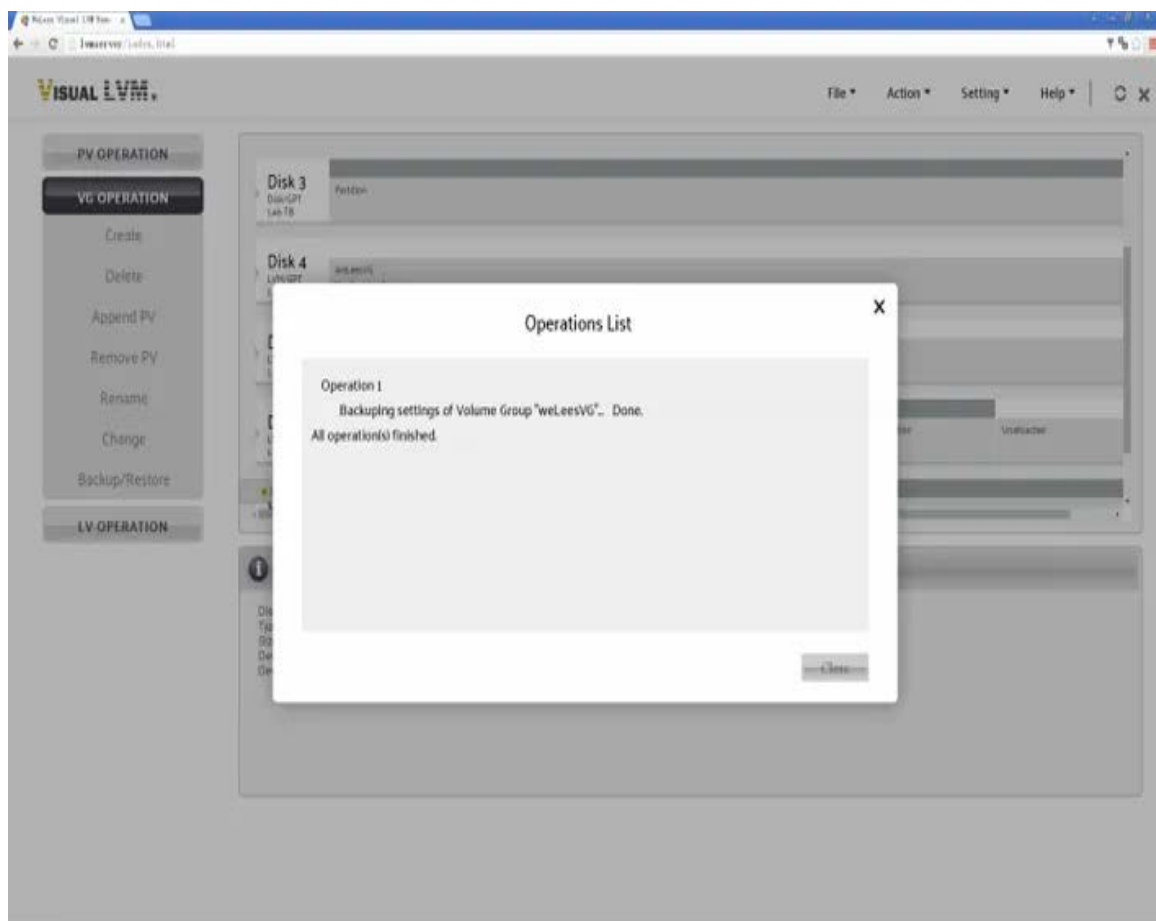
Select VG OPERATION->Backup/Restore



Backup/Restore VG dialog

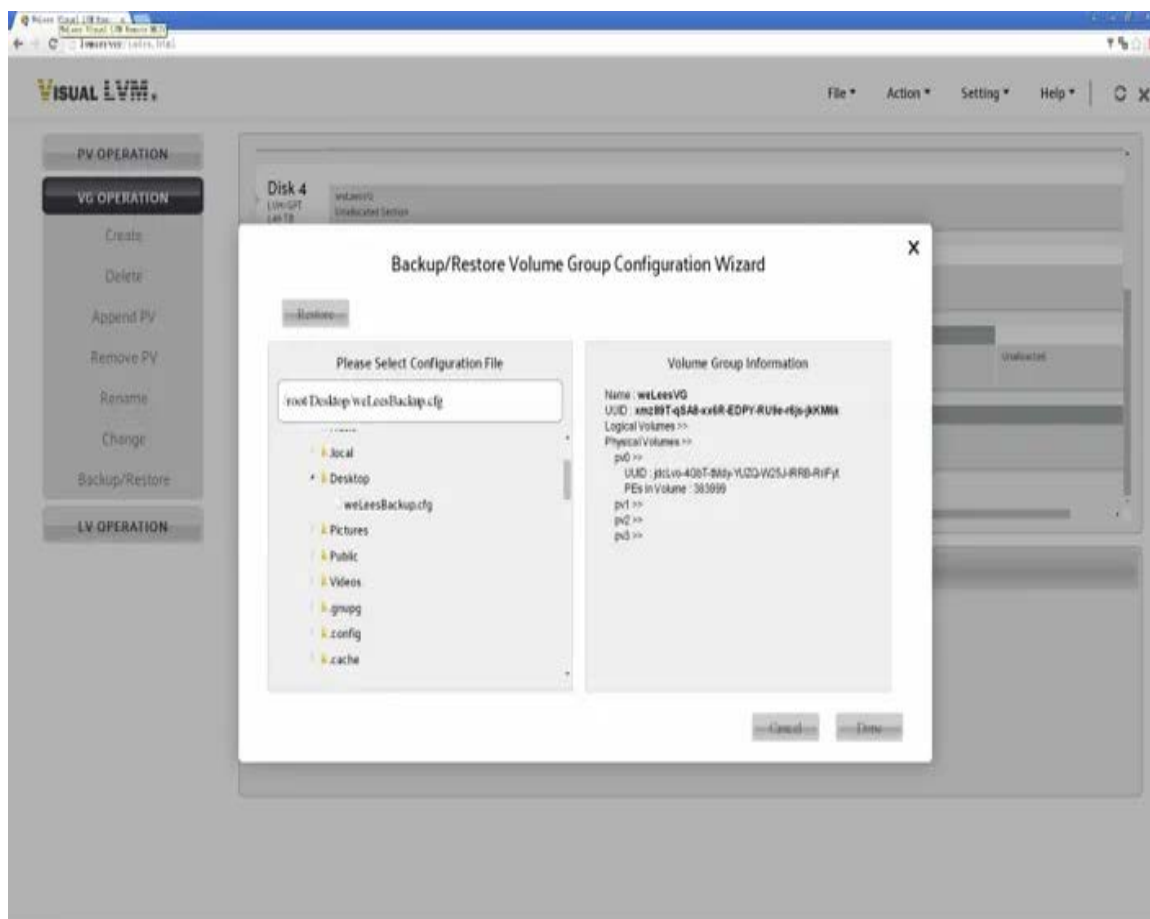
Select the VG to back up in the list of left side in backup dialog, and specify the directory and name of the backup file on the right. Click the Done button when

finished.



Operation completed

User can restore VG by click Backup button in left top side in dialog.



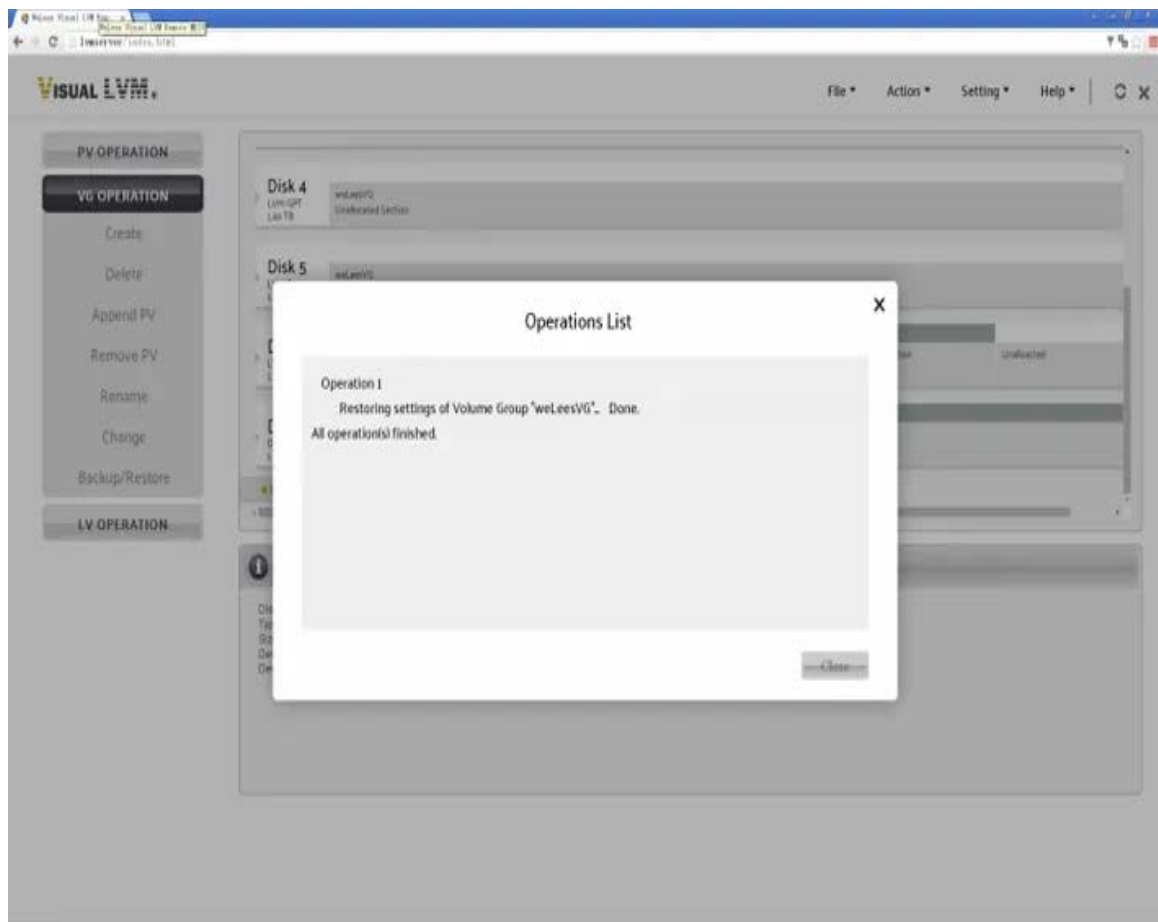
### Restore VG dialog

Select the configuration file to be restored from path tree in the left.

Double-click the configuration file. The information in the configuration file is displayed in the information bar on the right, helping you select a correct file for restoration.

Click the Done button after you determine the backup file.





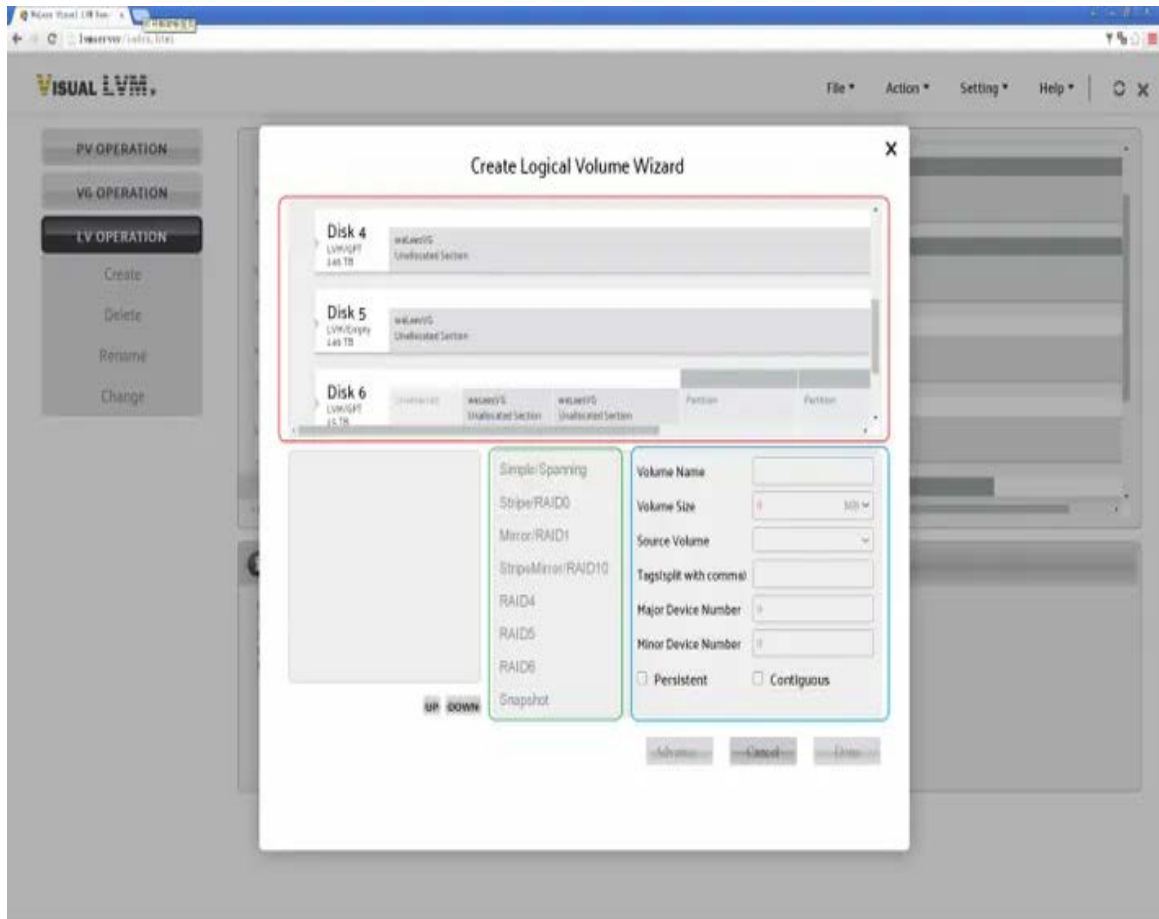
Operation completed

### 3. LV Operations

#### 3.1. Create LV

Now that the PV & VG have been created, let's start creating LV.

The interface of LV creating is relatively complex, but it does not mean that the operation of creating LV is complicated. It just supports multiple types of LV in one dialog box.



=

### Creating LV dialog

The Creating LV dialog is divided into three main sections. In the red box is the space selection area, where the user chooses the PV and size to use.

The green box is the LV type selection area, with a total of 8 types to choose from. It should be noted here that each LV has certain limits.

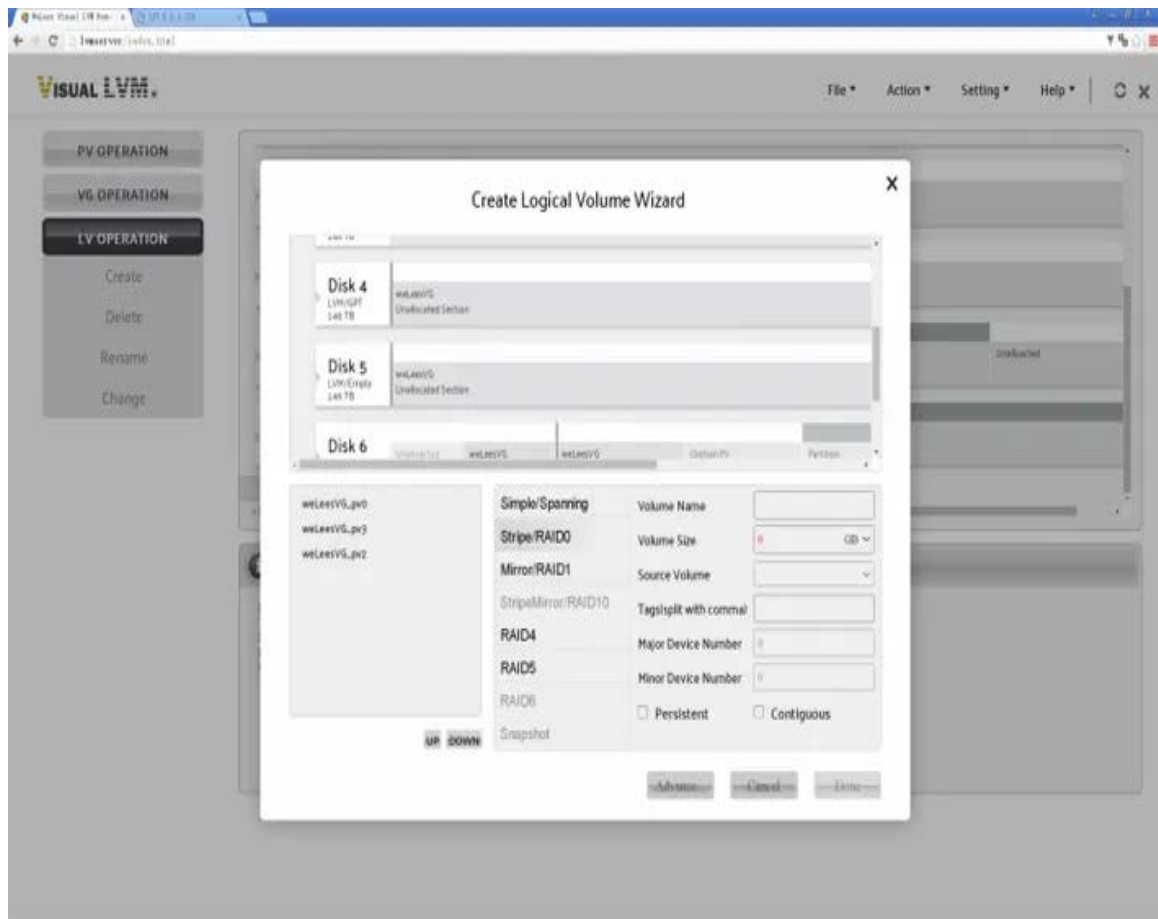
- Simple/Spawning/Stripe: It can be created in any case.
- Mirror: Using at least 2 PVs.
- StripeMirror: Using at least 4PVs, the amount of PVs must be an even number.
- RAID4/RAID5: Using at least 3 PVs.
- RAID6: Using at least 5 PVs.
- Snapshot: There is at least 1 existing LV in VG.
- Certain types are forbidden to use except the restriction conditions are met.

The blue box shows LV parameters. Certain configurations depend on other configurations & LV types, they are enabled only after the dependency conditions are met.

Source Volume is used only for Snapshot.

The Major/Minor Device Number can be configured only after Persistent is selected.

First, select PVs for LV. The selected PV displays a flashing vertical bar for dragging to adjust the size to be used. You can also enter the desired size in the Volume Size box.

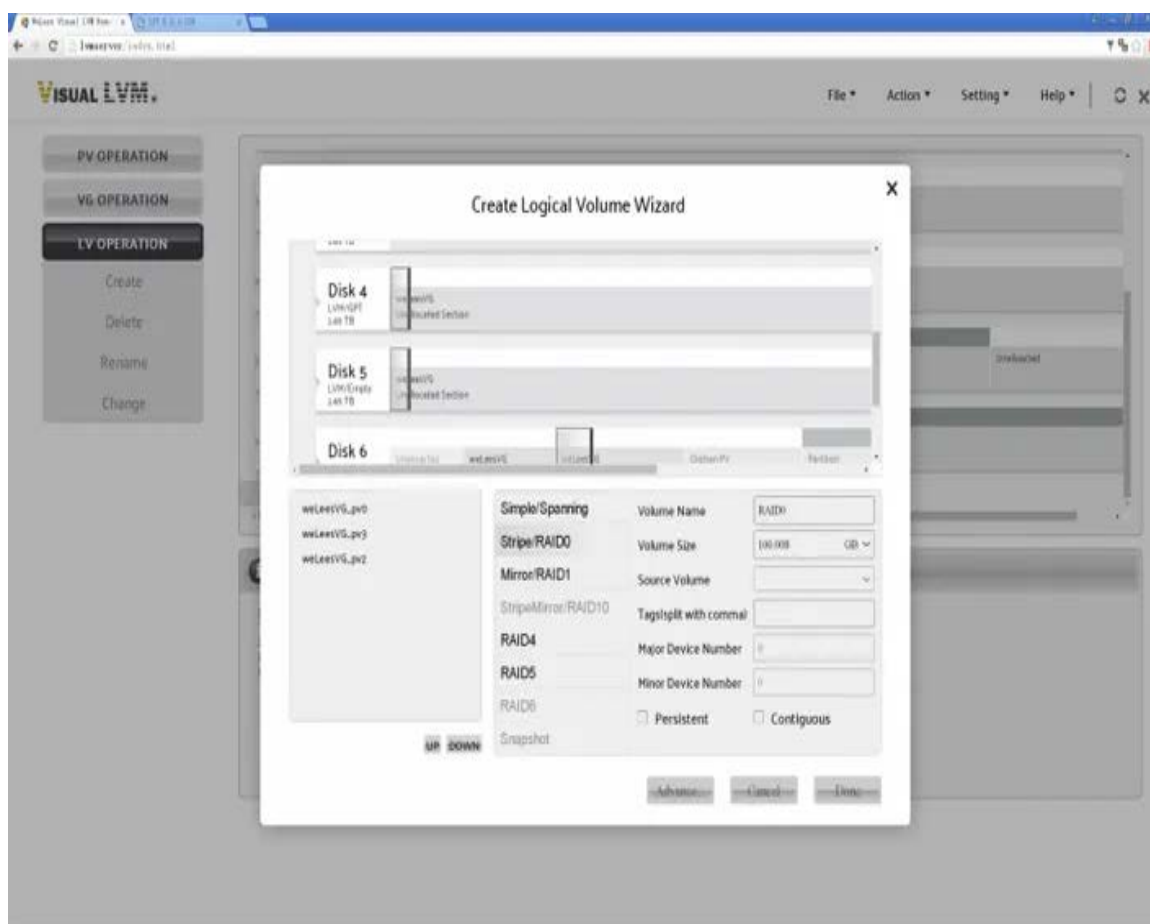


Click to select PV

After the PV is selected, the LV type that meets the requirements is enabled. In addition, users can drag to set the LV size to see the difference in space allocation between RAID LV and non-RAID LV.

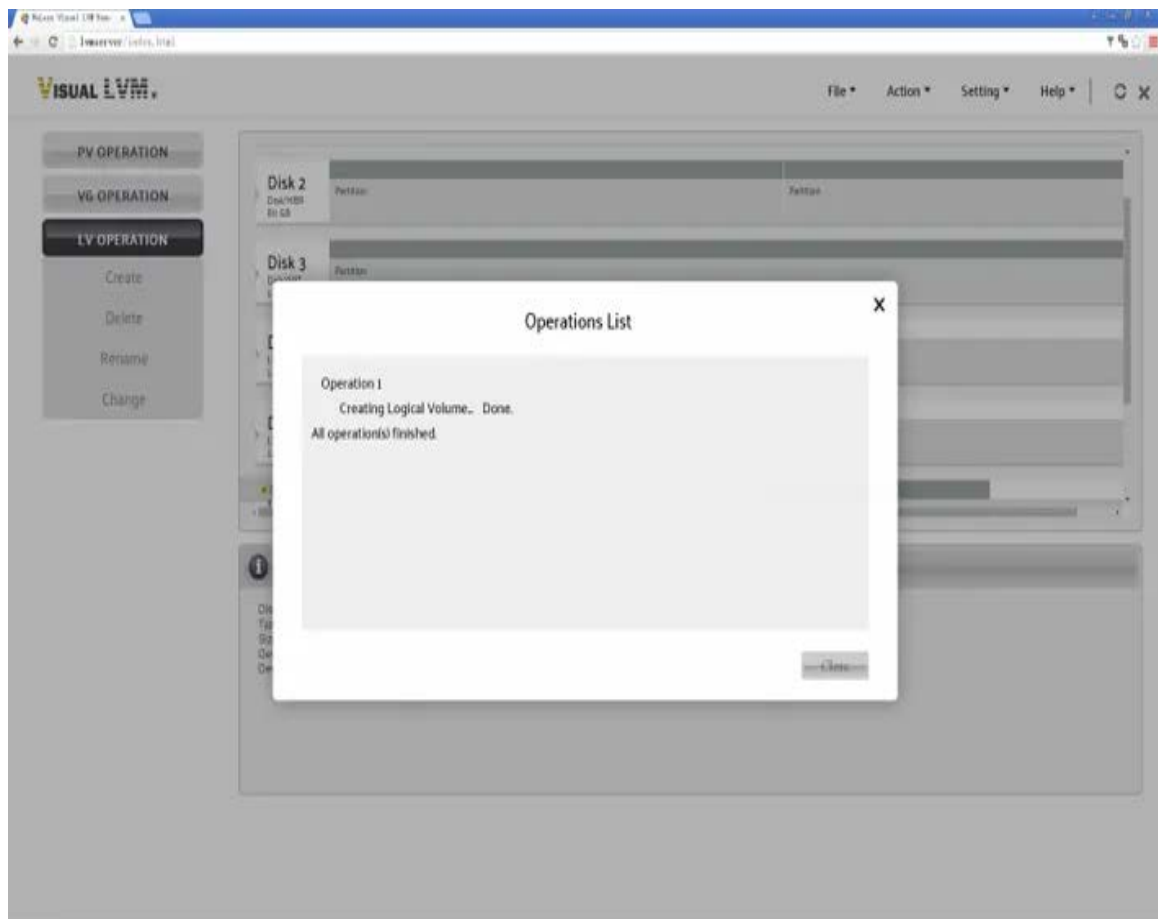
For the Simple/Spanning/Snapshot, the user needs to drag the resize bars in each PV. For Stripe/Mirror/StripeMirror/RAID4/RAID5/RAID6, it needs to allocate the same size of space from all PVs in use, and it also needs to access data via stripe. So when user drags resize bar in any PV, the size in other selected PVs change correspondingly.

This reduces the complexity of LV creating.

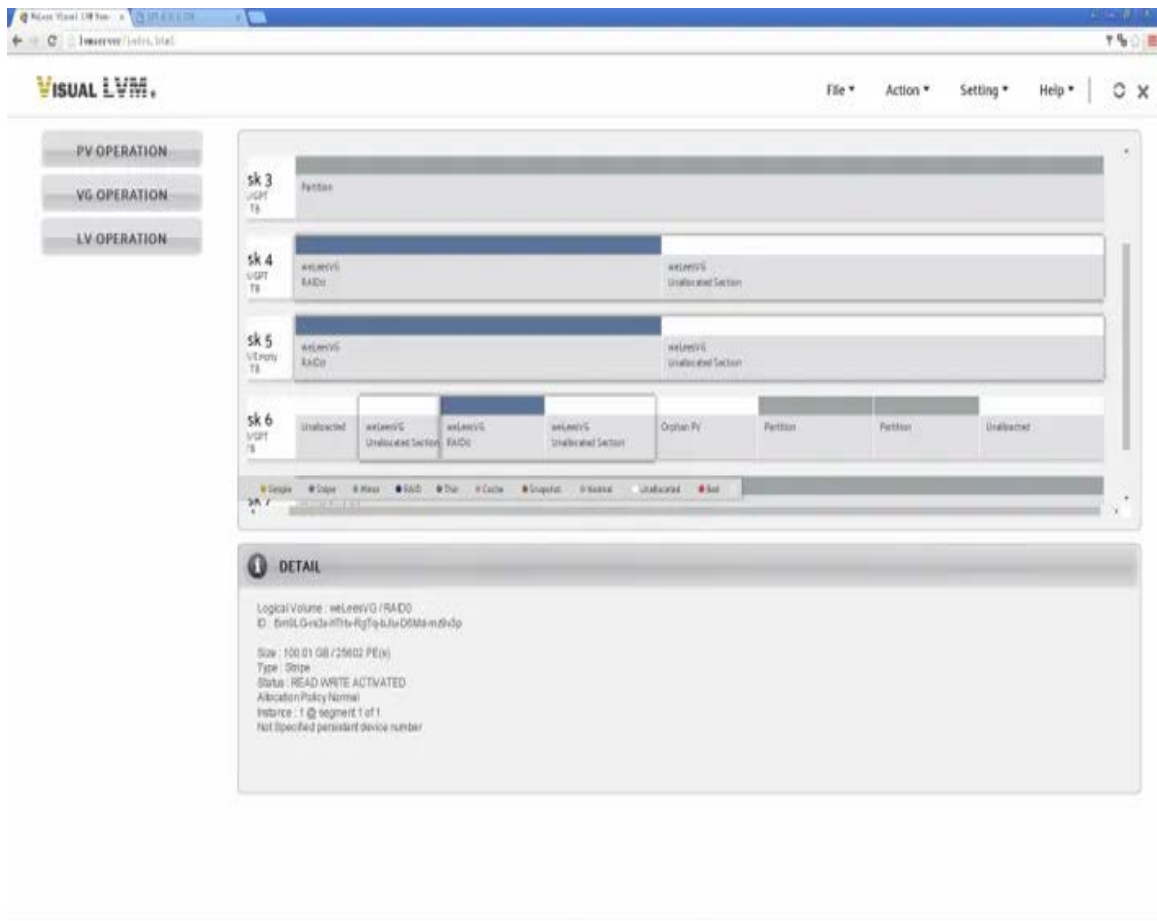


LV configured, ready for creating

Click Done button to create



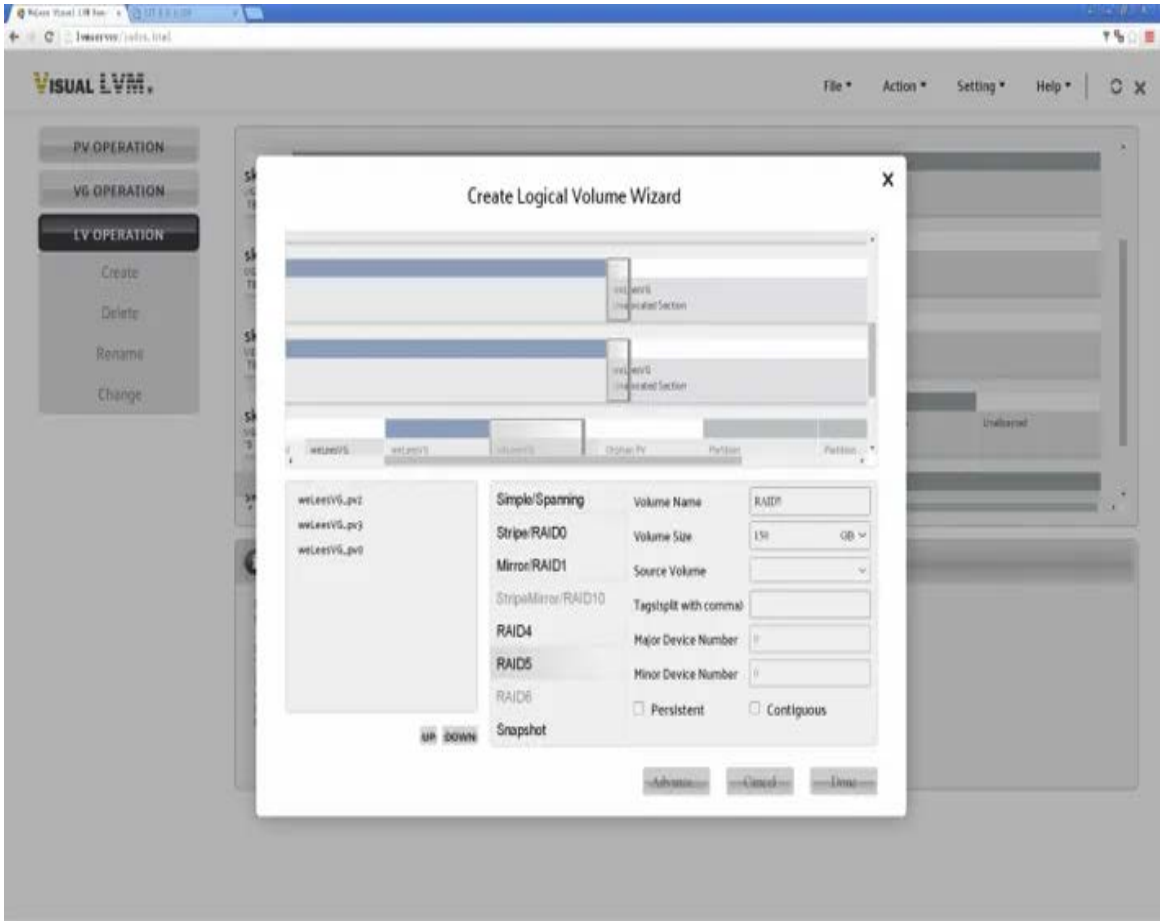
A RAID0 LV created



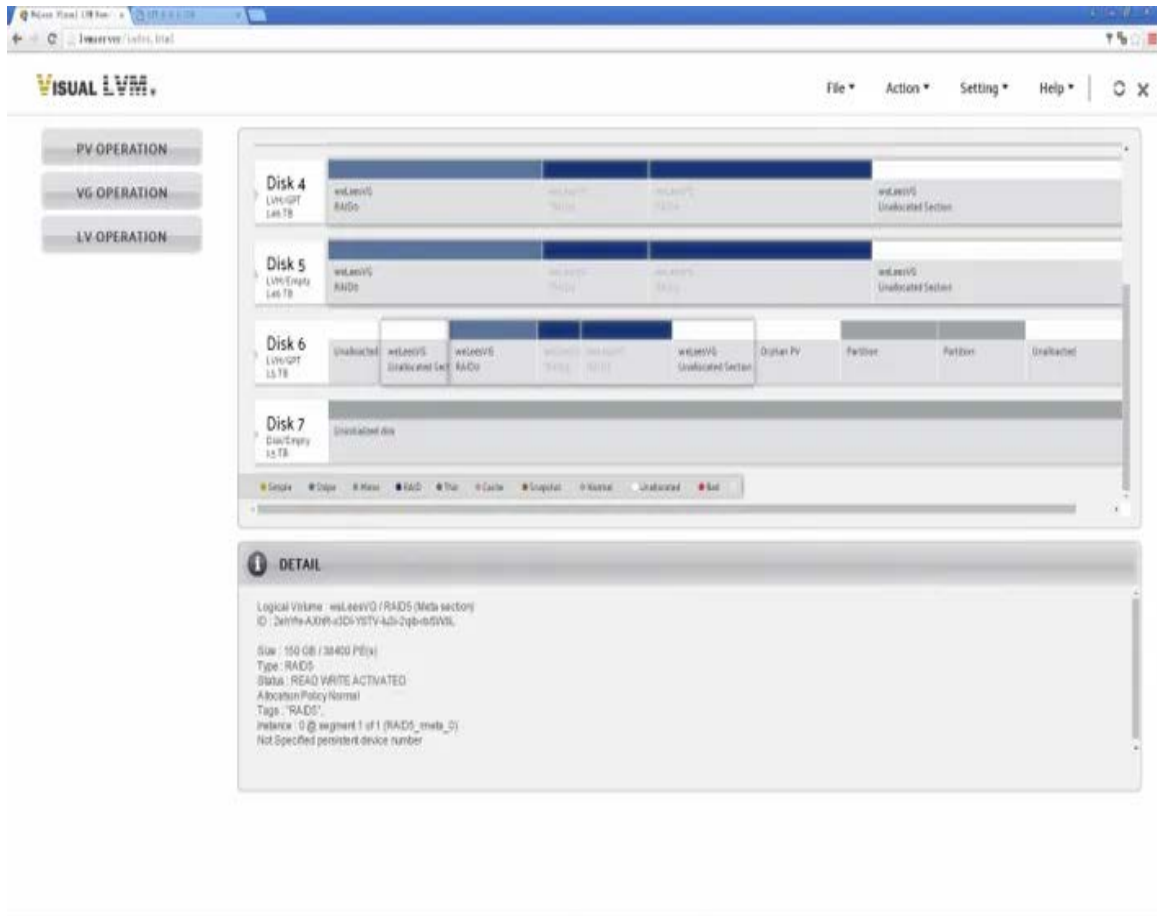
Yes we get the first LV

Its information will be shown in detail bar when mouse pointer over any section of LV. And all sections of LV are shown highlighted to help user views entire LV.

Next, let's create a RAID5 LV.



Create RAID5 LV



### VG with 2 LVs

RAID5 & RAID0 are marked with different colors by type. See the tip bar below the disk layout view for the colors used by different types of LV.

The LVM will also establish a meta area for RAID type (except RAID0) in all used PVs, so RAID5 has one more meta section in each PV and there is a '\*' before its name in meta area. User can view the information from detail bar by mouse pointer over metaarea section.

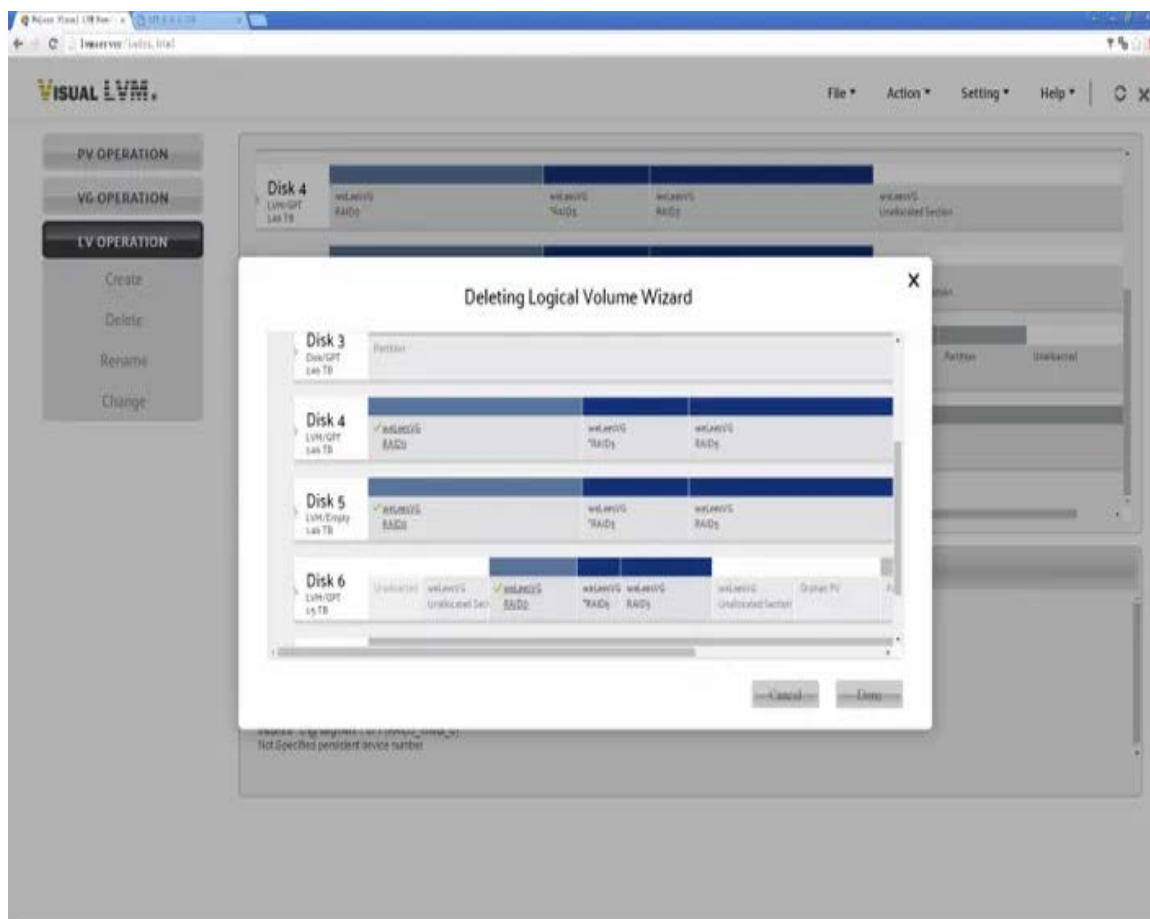
### 3.2. Delete LV

Now let's delete the LV RAID0.

First, select LV OPERATION->Delete.

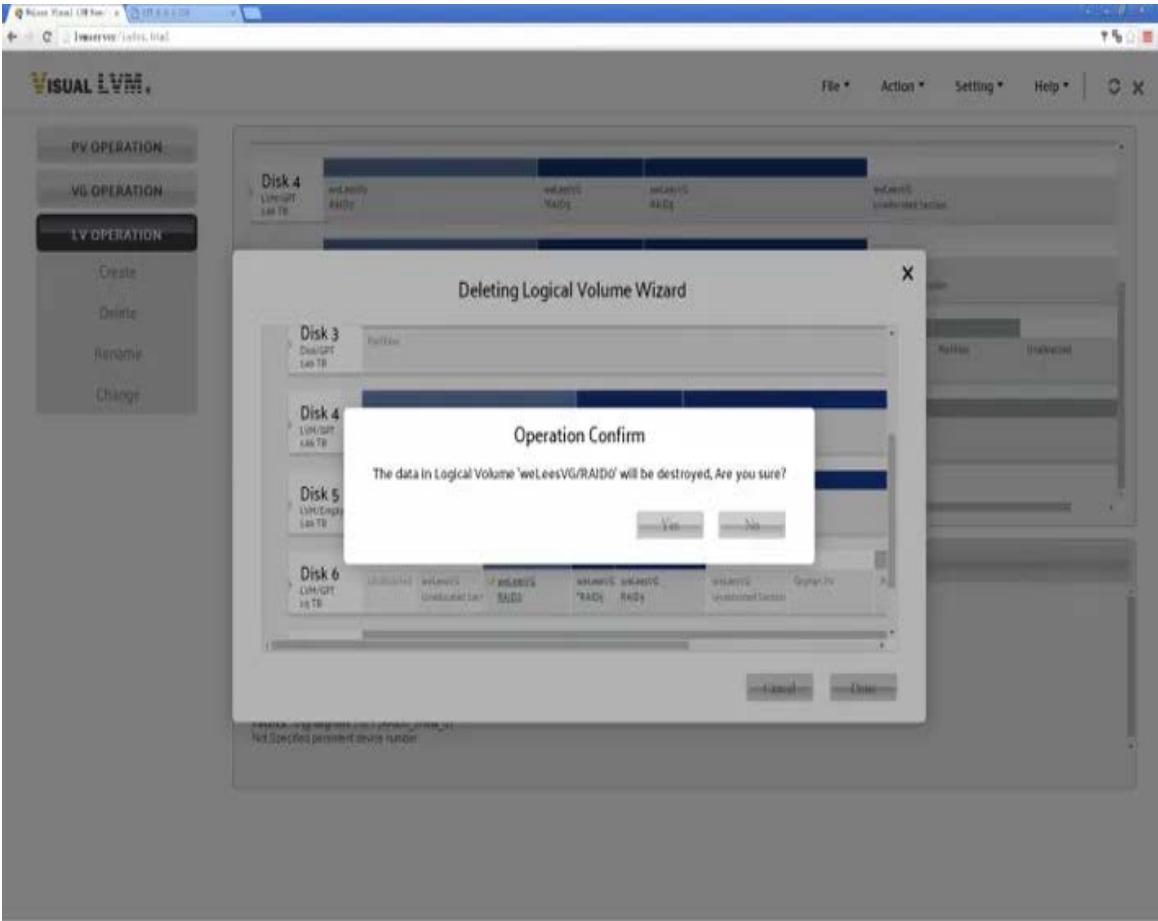
Second, select LV we want to delete.



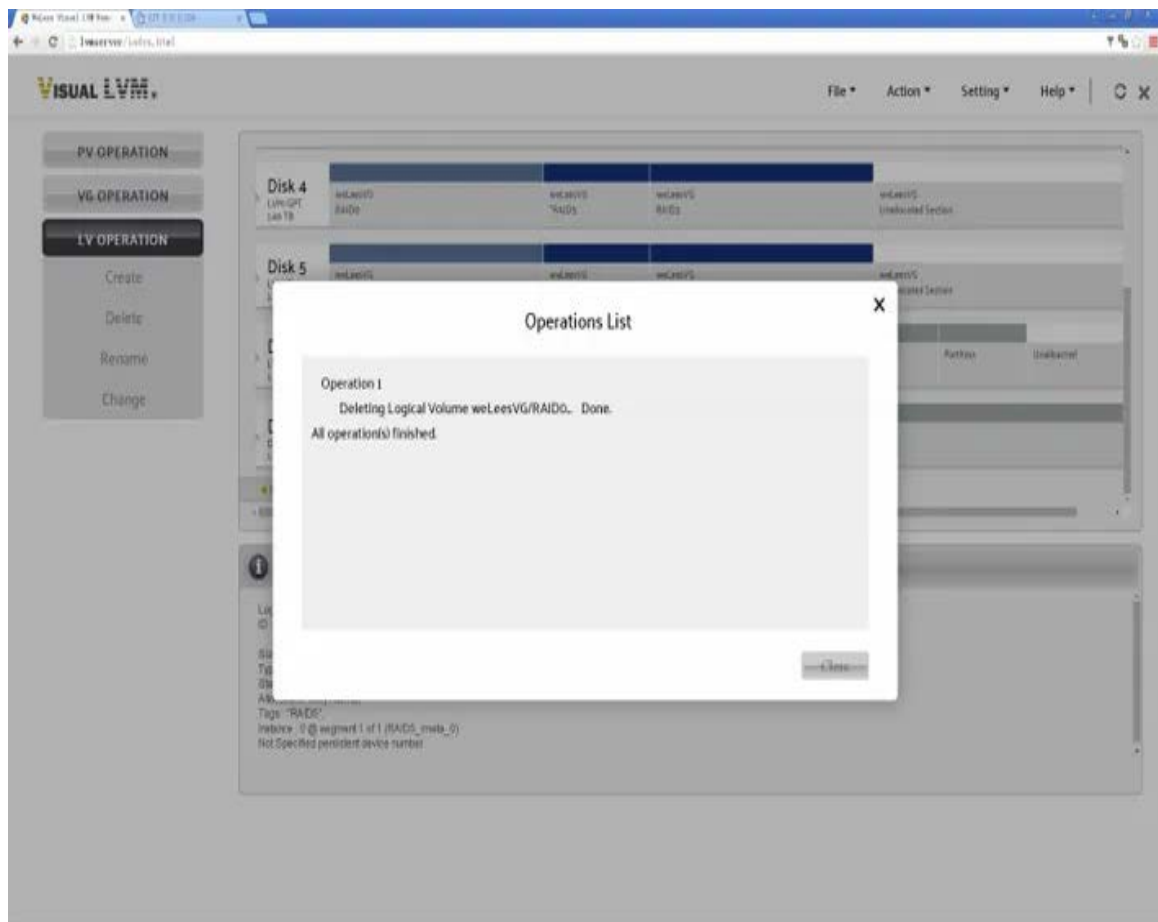


Delete LV dialog, select LV

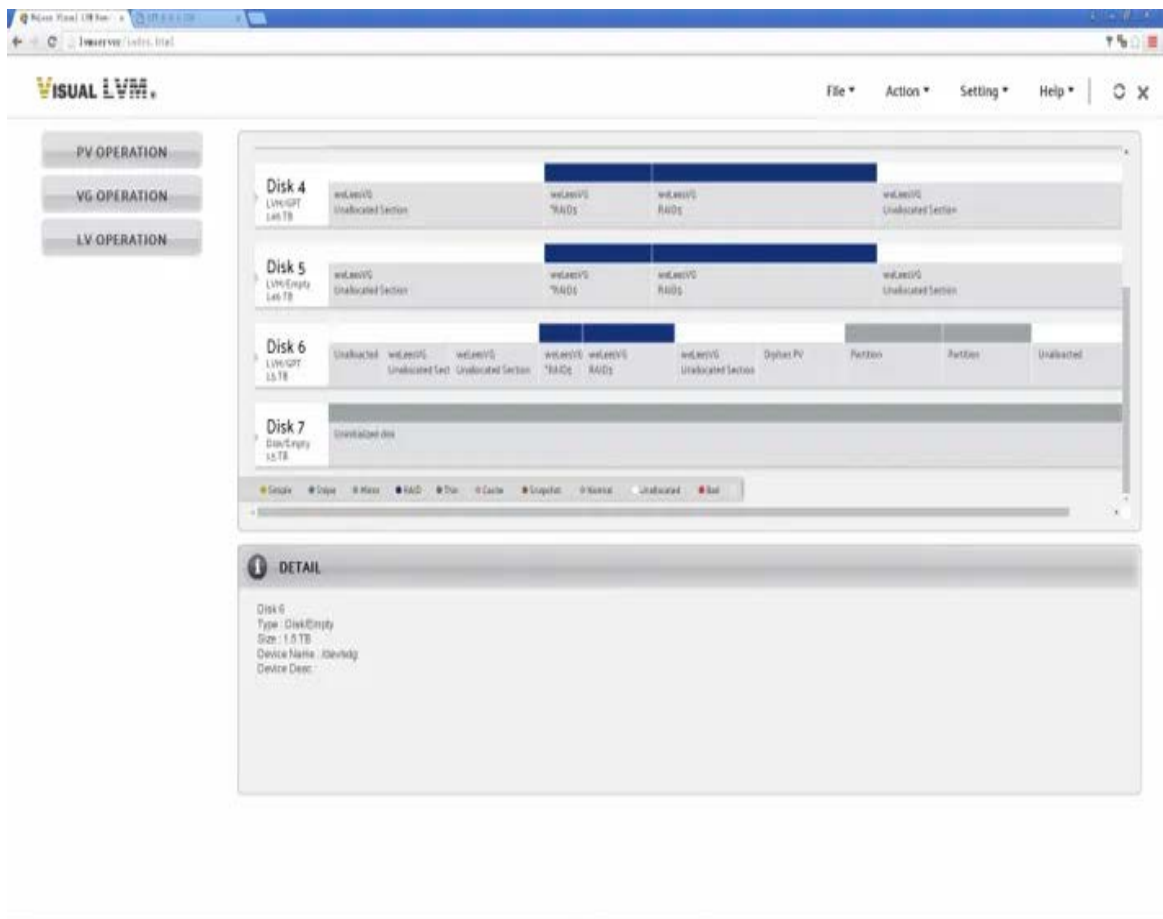
Click Done button, and Visual LVM needs user to confirm



Delete LV confirm



Operation completed, LV was deleted

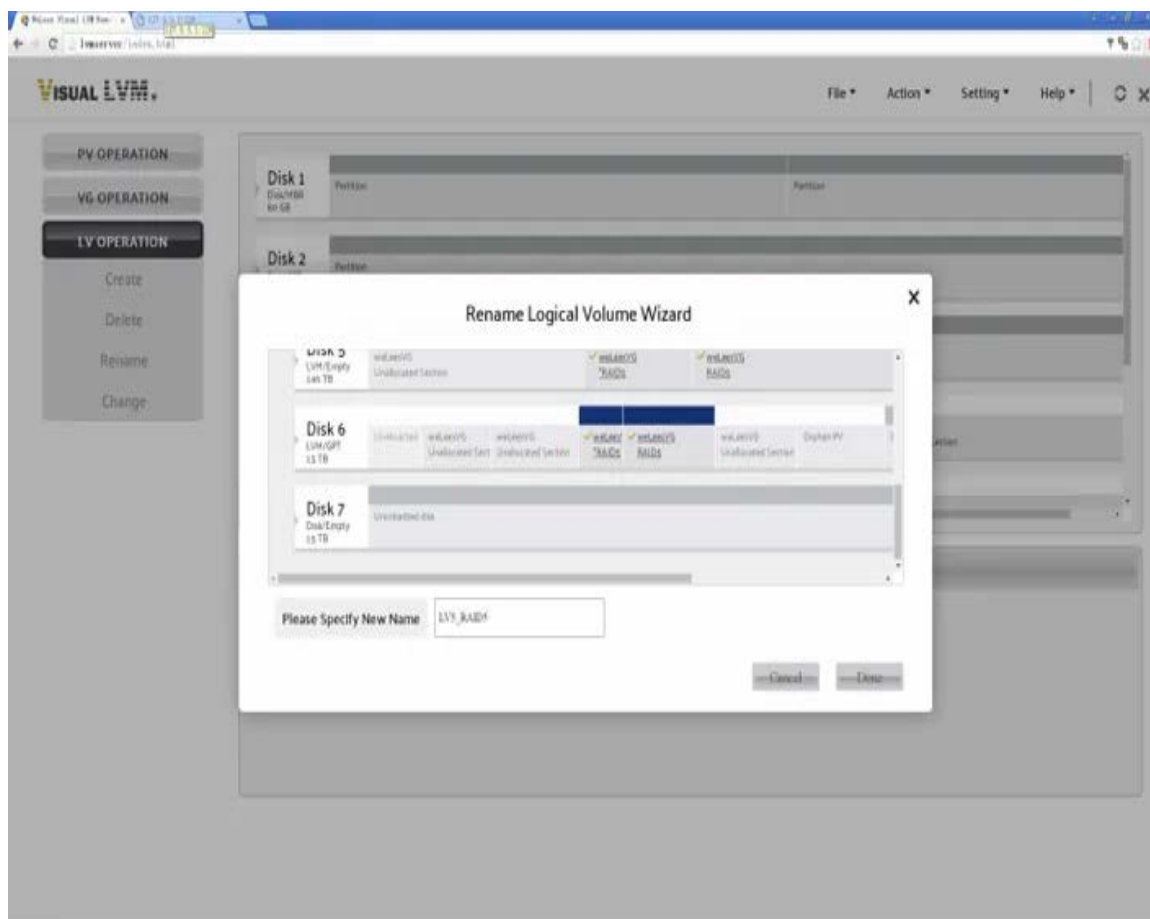


VG with LV "RAID0" deleted

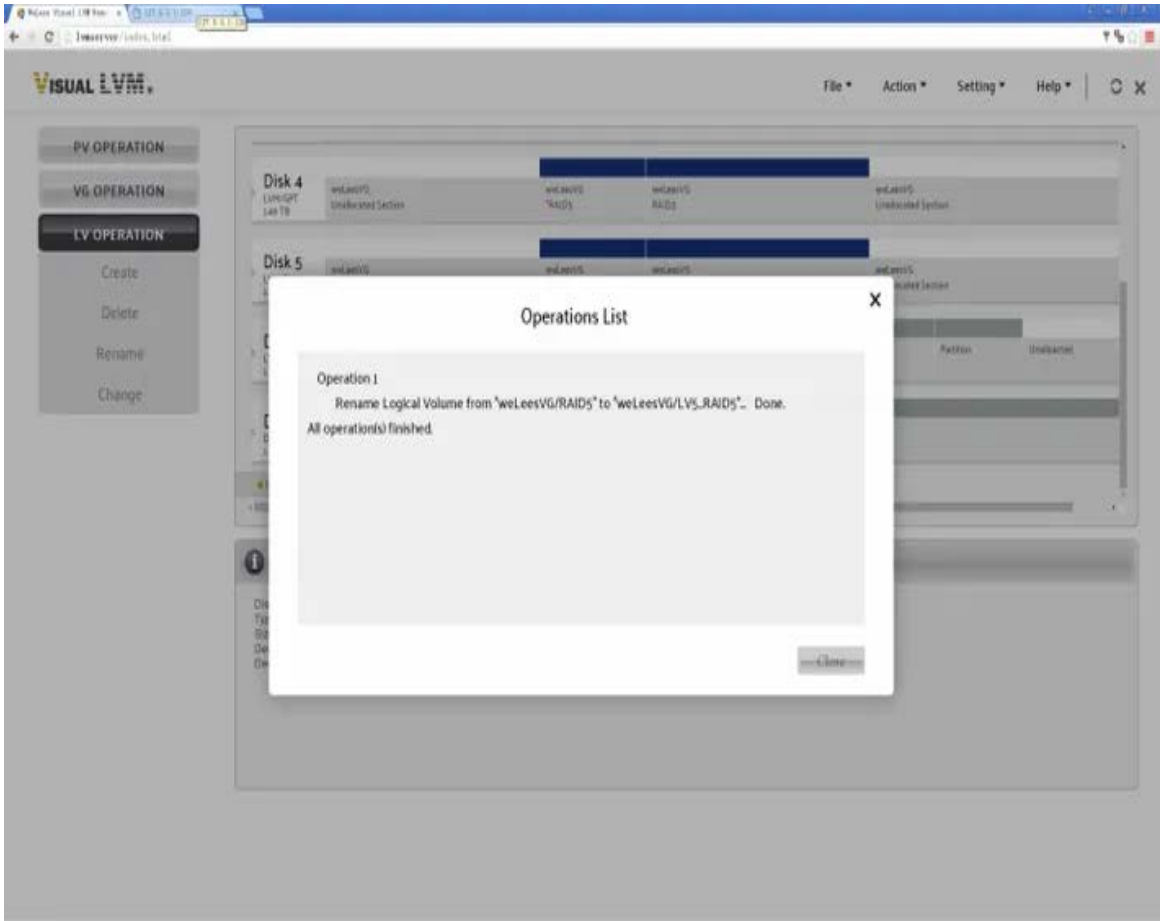
### 3.3. Rename LV

LV The renaming operation is very similar to VG renaming.

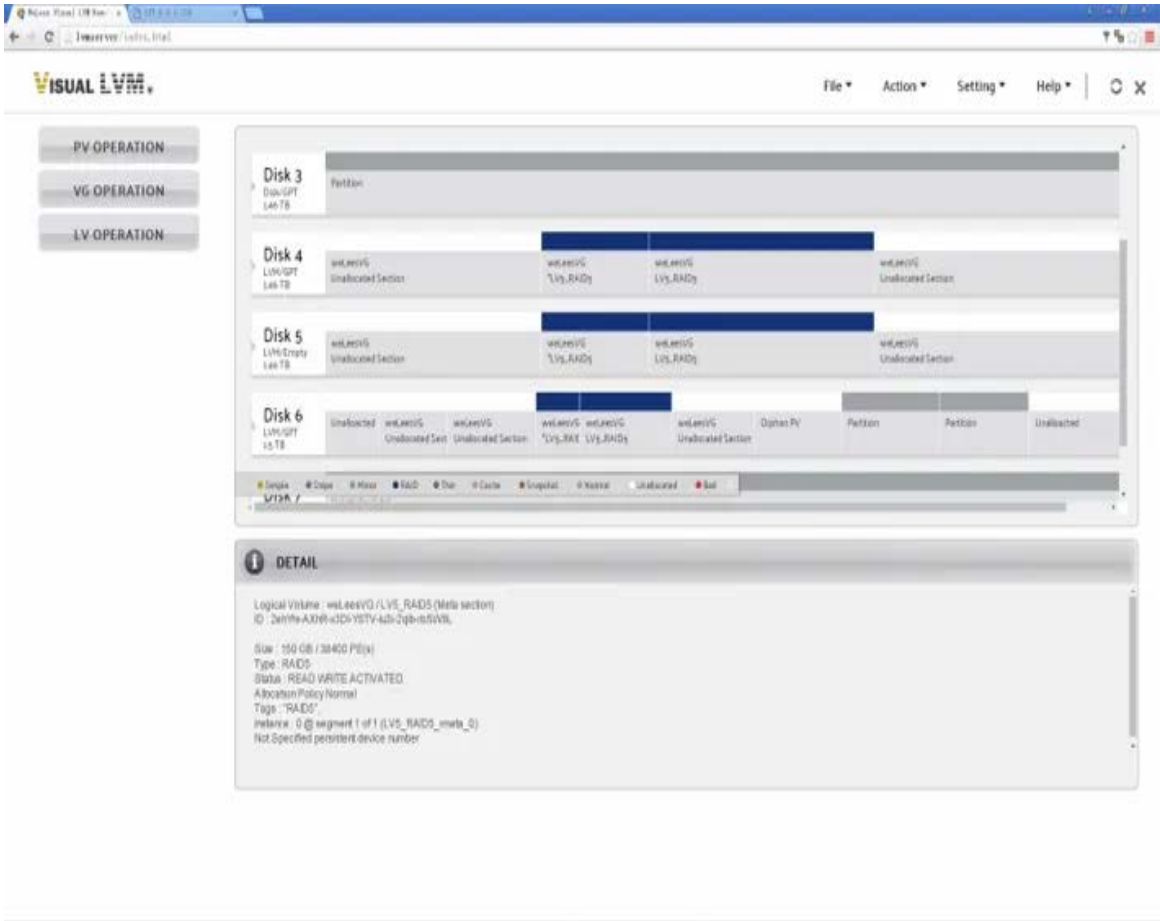
Select LV OPERATION->Rename, and click to select the LV to be renamed, and type the new name. Click Done to perform the action.



Select LV to be renamed and type new name



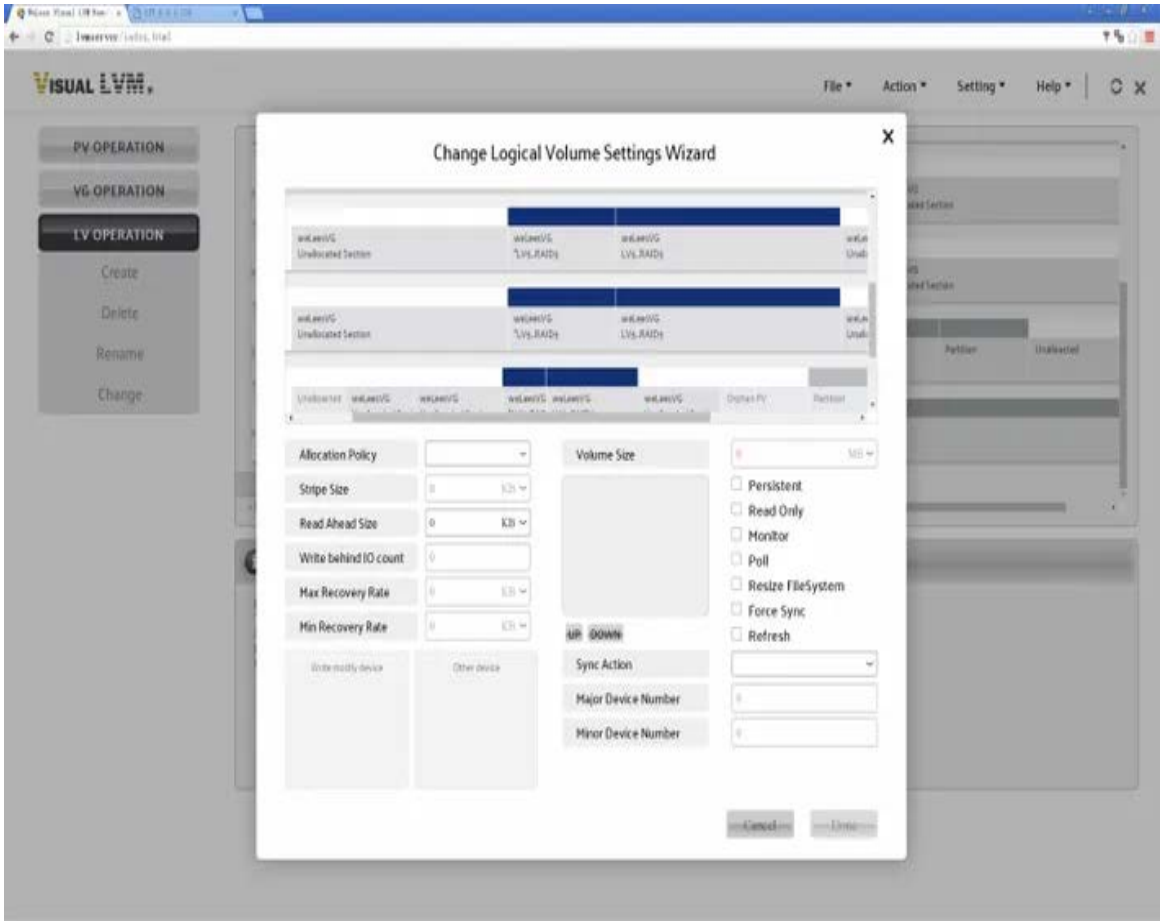
Operation Completed



LV with new name

3.4. Change LV configuration

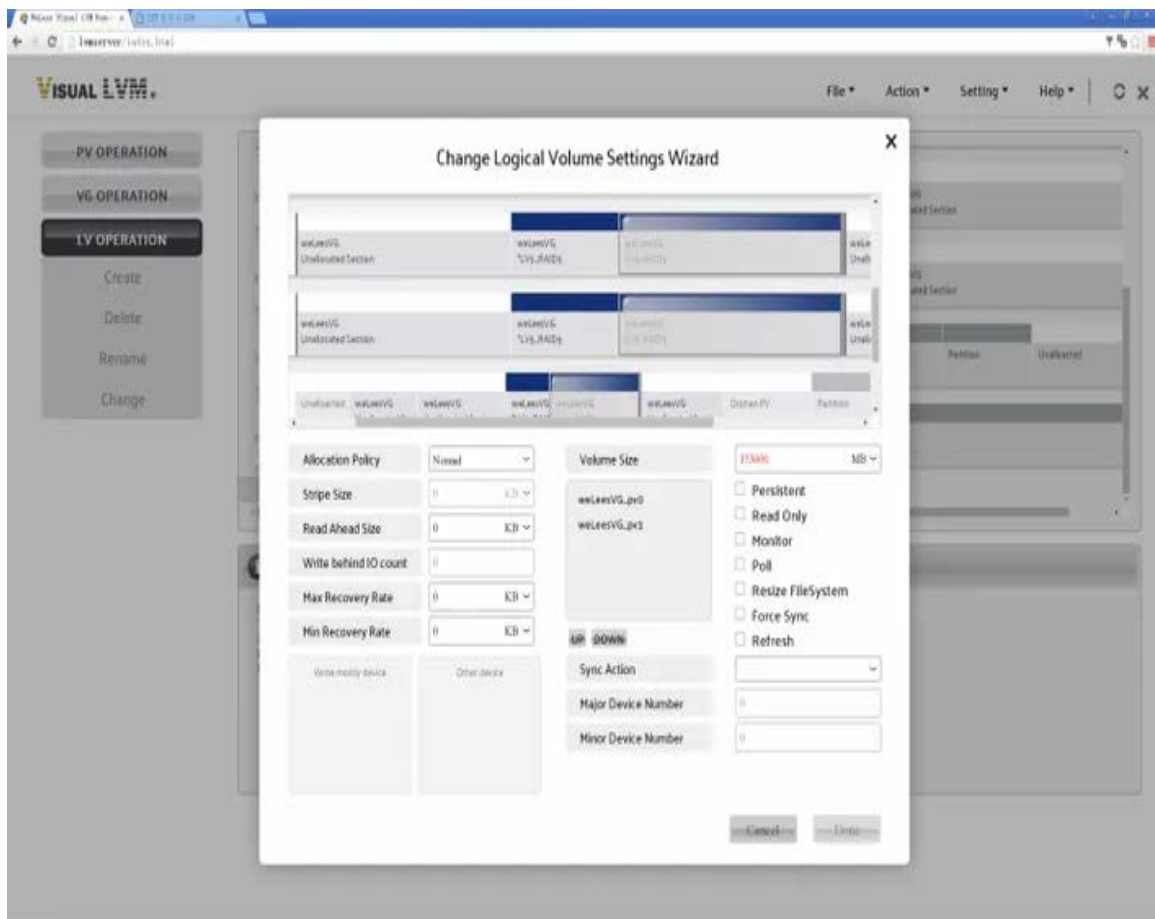
Changing LV configuration is also a collection of many operations. You can change LV allocation rules, stripe sizes, persistent device number, RAID LV prefetch, recovery rate, and synchronization operations.



Change LV configuration dialog

First let's talk about how to resize LV.

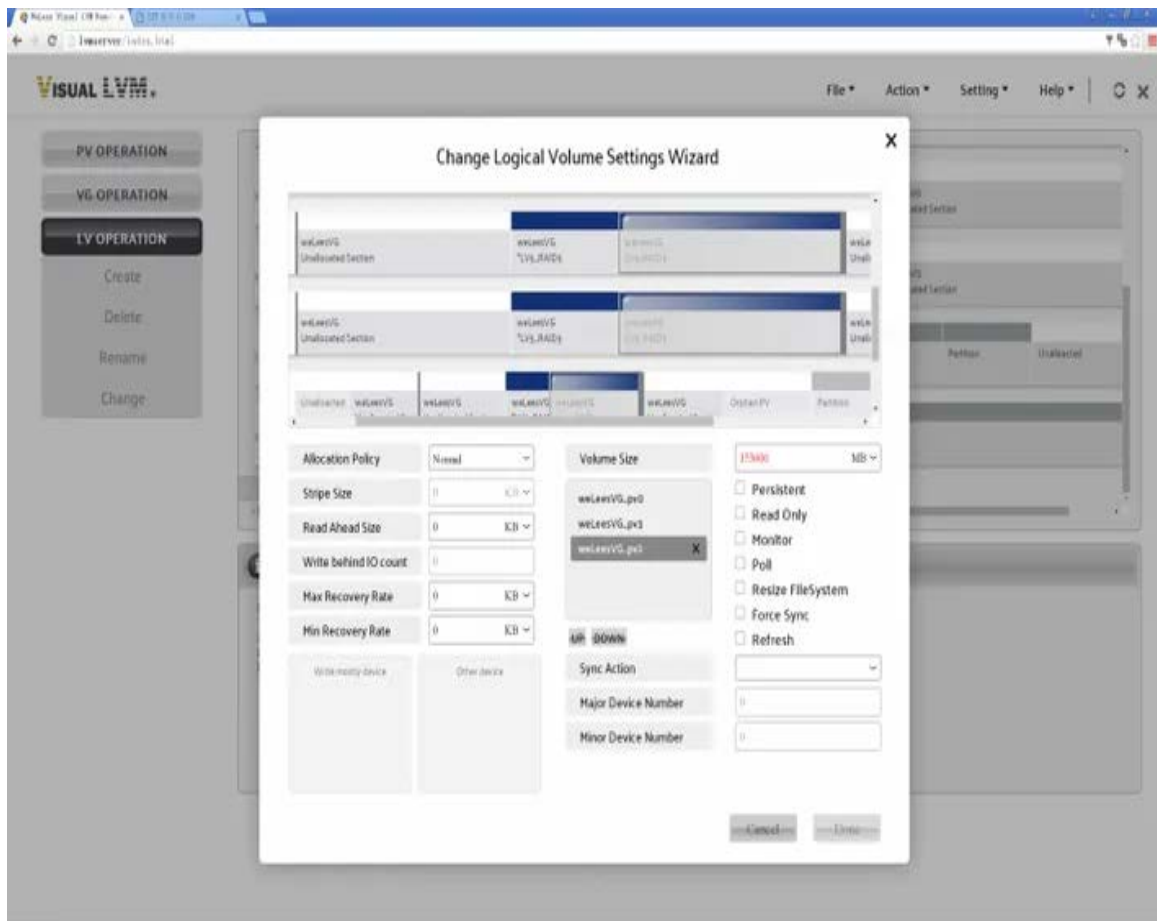




Click the LV to be operated, the resize bar will appear on LV, and the user can drag to reduce the size of the LV. To increase the size, you need to click on the free space of any PV in the same VG to specify the space for expansion.

There are 3 key points

1. For RAID type LV (except RAID0), you need to specify the same number of PV as the original LV to expand the size.
2. For LV composed of multiple segments, the size reduction should be carried out in the order from tail to head. The size of segment near the head cannot be adjusted until the segment behind it is completely reduced.
3. For a RAID type LV, the size of other PVs will change correspondingly by drag the resize bar in any PV to expand.



## Resize LV

## CONCLUSION

We have studied/demonstrated the concepts of LVM and the operation of management. Readers who have read this article can easily be competent in LVM management work even if they have no foundation. Although graphical management operations are not as easy to be run with script as CLI, they are far more user-friendly than CLI operations, which not only enables beginners to quickly and easily master LVM knowledge, but also greatly reduces the work intensity of administrators. In particular, GUI's intuitive representation of the entire storage layout allows administrators to stay away from kinds of complex calculations and transformations, reducing the possibility of errors and improving work efficiency.

Of course, the limitation of old GUI tools requiring graphical module support has limited its use, but as the readers of this article can see, the LVM GUI tool Visual LVM Remote based on B/S architecture has broken through this limitation. Visual LVM Remote does not require the work on machine with graphical desktop module environment and display. It is similar to Telnet tools. As long as there is any device that can be connected to the host through the network, PC, tablet or mobile phone can perform graphical management operations, and even remote management

operations can be performed over the Internet. It's already on the same level of convenience as the CLI.

Hopefully this article will make it easier for you to learn LVM and enjoy its benefits.

 [view as pdf](#) |  [print](#)

Share this page:



## Suggested articles

## 0 Comment(s)

Add comment

Name \*

Email \*



p



I'm not a robot

reCAPTCHA  
[Privacy](#) - [Terms](#)

Submit comment

[Home](#)

[Learn LVM on Linux with graphical tools](#)