



## Kernel Korner - Unionfs: Bringing Filesystems Together

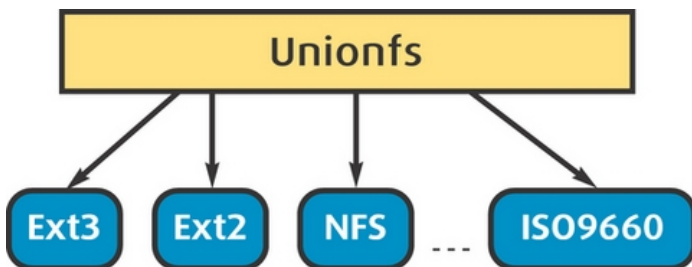
Dec 01, 2004 By [Charles P. Wright \(/user/1001388\)](#) and [Erez Zadok \(/user/801167\)](#)

Like 3 people like this. Be the first of your friends.

in

*Unionfs merges several directories into a single unified view. We describe applications of Unionfs and also interesting implementation aspects.*

For ease of management, it can be useful to keep related but different sets of files in separate locations. Users, however, often prefer to see these related files together. In this situation, unioning allows administrators to keep such files separate physically, but to merge them logically into a single view. A collection of merged directories is called a union, and each physical directory is called a branch. As shown in Figure 1, Unionfs simultaneously layers on top of several filesystems or on different directories within the same filesystem. This layering technique is known as stacking (see the on-line Resources for more on stacking). Unionfs presents a filesystem interface to the kernel, and in turn Unionfs presents itself as the kernel's VFS to the filesystems on which it stacks. Because Unionfs presents a filesystem view to the kernel, it can be employed by any user-level application or from the kernel by the NFS server. Because Unionfs intercepts operations bound for lower-level filesystems, it can modify operations to present the unified view. Unlike earlier stackable filesystems, Unionfs is a true fan-out filesystem; it can access many underlying branches directly.



[\(/files/linuxjournal.com/linuxjournal/articles/077/7714/7714f1.jpg\)](http://files.linuxjournal.com/linuxjournal/articles/077/7714/7714f1.jpg)

Figure 1. A union consists of several underlying branches, which can be of any filesystem type.

### Unionfs Semantics and Usage

In Unionfs, each branch is assigned a precedence. A branch with a higher precedence overrides a branch with a lower precedence. Unionfs operates on directories. If a directory exists in two underlying branches, the contents and attributes of the Unionfs directory are the combination of the two lower directories. Unionfs automatically removes any duplicate directory entries, so users are not confused by duplicated filenames or directories. If a file exists in two branches, the contents and attributes of the Unionfs file are the same as the file in the higher-priority branch, and the file in the lower-priority branch is ignored.

As a concrete example, assume that we unify two directories, /Fruits and /Vegetables:

```
$ ls /Fruits
Apple Tomato
$ ls /Vegetables
Carrots Tomato
$ cat /Fruits/Tomato
I am botanically a fruit.
```

```
$ cat /Vegetables/Tomato
I am horticulturally a vegetable.
```

To use Unionfs, you first need to compile the Unionfs module and load it into the kernel. Next, like any other filesystem, Unionfs is mounted. Unlike other filesystems, Unionfs does not mount on top of a device; it mounts on top of directories that are specified as a mount-time option. To create a union, we mount Unionfs as follows:

```
# mount -t unionfs -o dirs=/Fruits:/Vegetables \
> none /mnt/healthy
```

In this example, the mount option `dirs` tells Unionfs which directories make up the union. Unionfs does not mount any device, so we use `none` as a placeholder. Finally, `/mnt/healthy` is the location of the merged view. Now `/mnt/healthy` contains three files: Apple, Carrots and Tomato. Because we specified `/Fruits` before `/Vegetables`, `/mnt/healthy/Tomato` contains "I am botanically a fruit." If we were to reverse the `dirs=` option, `/mnt/healthy/Tomato` would contain "I am horticulturally a vegetable." (which agrees with the 1893 U.S. Supreme Court ruling on the matter).

This process is recursive. If there were a subdirectory of Fruits named Green that contained a file named Lime and a subdirectory of Vegetables also named Green that contained a file named Lettuce, the result would be:

```
$ ls /mnt/healthy
Apple Carrots Green/ Tomato
$ ls /mnt/healthy/Green
Lime Lettuce
```

Unionfs can be applied in several ways. Simple examples include unifying home directories from multiple servers or merging split ISO images to create a unified view of a distribution. In a similar vein, Unionfs, with copy-on-write semantics, can be used to patch CD-ROMs, for source code management or for snapshotting.

### Unified Home Directories

Often a single client machine mounts home directories from several different NFS servers. Unfortunately, each server has a distinct mountpoint, which is inconvenient for users. It would be ideal if all home directories were available from the same place (`/home` for example). Some automounters use symbolic links to create the illusion of a union. With Unionfs, these links are not necessary. The separate exported directories simply can be unified into a single view. Assume we have two filesystems, one mounted on `/alcid` and the other mounted on `/penguin`. We can unify them into `/home` as follows:

```
# mount -t unionfs -o dirs=/alcid,/penguin \
> none /home
```

---

## Comments

### Comment viewing options

Threaded list - expanded ▼ | Date - newest first ▼ | 50 comments per page ▼ | Save settings  
Select your preferred way to display the comments and click "Save settings" to activate your changes.

["I think UnionFS is great" \(article/7714#comment-324324\)](http://www.linuxjournal.com/article/7714#comment-324324)

Submitted by Thinking (not verified) on Tue, 07/29/2008 - 14:35.

"I think UnionFS is great for Live CD applications(e.g. DSL/Linux - many others). Also, in general UnionFS is great for small simple applications like remembering writes (COWs - copy on writes) to any read-only file system (e.g. CDROM). However, beyond that, and especially the tomato example I think UnionFS is actually far too dangerous for serious computing. Why? Because there is no intelligent way to remember/decide the duplicate file issues. That being said I'm actually very pro-UnionFS; but like seasoning a fine stew - a little seasoning goes a long way, too much seasoning spoils the stew - re: a little UnionFS is great, beyond that it is DANGEROUS".

[adipex phentermine \(http://forums.usfirst.org/member.php?u=12324\)](http://forums.usfirst.org/member.php?u=12324)  
[buy cheap phentermine online \(http://www.forum.psoft.net/member.php?u=21124\)](http://www.forum.psoft.net/member.php?u=21124)  
[buy cheap phentermine \(http://www.mountwashington.org/forums/member.php?u=10194\)](http://www.mountwashington.org/forums/member.php?u=10194)  
[buy phentermine online \(http://www.forum.psoft.net/member.php?u=21125\)](http://www.forum.psoft.net/member.php?u=21125)

#### **UnionFS - great for some apps. very dangerous for others** ([/article/7714#comment-302554](http://article/7714#comment-302554))

Submitted by [mjc777 \(http://www.embeddedrelated.com/blogs-1/nf/Mark\\_Conrad.php\)](http://www.embeddedrelated.com/blogs-1/nf/Mark_Conrad.php) (not verified) on Sat, 10/27/2007 - 13:04.

I think UnionFS is great for Live CD applications(e.g. DSL/Linux - many others). Also, in general UnionFS is great for small simple applications like remembering writes (COWs - copy on writes) to any read-only file system (e.g. CDROM). However, beyond that, and especially the tomato example I think UnionFS is actually far too dangerous for serious computing. Why? Because there is no intelligent way to remember/decide the duplicate file issues. That being said I'm actually very pro-UnionFS; but like seasoning a fine stew - a little seasoning goes a long way, too much seasoning spoils the stew - re: a little UnionFS is great, beyond that it is DANGEROUS (IMHO).

#### **unionfs snapshots of live mounted systems** ([/article/7714#comment-137773](http://article/7714#comment-137773))

Submitted by Cesar (not verified) on Thu, 05/04/2006 - 21:50.

Is it possible to create snapshots of live mounted systems using unionfs? If so how to do it? I would like some help on this issue. I currently use LVM2 snapshots to do consistent backups of live running systems and are having some problems: system hangs, hard locks, etc. If I could do the same as with LVM2 snapshots but with the unionfs it would be great.  
thanks

#### **ISO-File from unionfs** ([/article/7714#comment-55057](http://article/7714#comment-55057))

Submitted by pmt (not verified) on Wed, 06/08/2005 - 07:56.

Is it possible to make a iso-file from unionfs?

I have tried it with mkisofs, but the directory depth is too high.  
What i can do?

#### **re:ISO-File from unionfs** ([/article/7714#comment-128427](http://article/7714#comment-128427))

Submitted by [Jake Miller \(http://www.portalano.net\)](http://www.portalano.net) (not verified) on Tue, 01/31/2006 - 03:53.

Hi Joost,  
Is it the unionfs stuff discussed recently here used for slax?

Unknown:

A little more involved to modify the image, perhaps check the slax site for hints. Note: .iso images are built with mkisofs, not written to like a read/write filesystem. This `_is_` documented, seek ;)

Cheers,

#### **re:ISO-File from unionfs** ([/article/7714#comment-128429](http://article/7714#comment-128429))

Submitted by [Randall M. \(http://www.tuttipagina.org\)](http://www.tuttipagina.org) (not verified) on Tue, 01/31/2006 - 03:56.

Yes, NetBSD 3.0 has support for compressed VND images - they work exactly the same way as e.g. in Knoppix on Linux or Freesbie on FreeBSD. The kernel needs to have support enabled via the VND\_COMPRESSION kernel option (see options(4)). After that, you can create a disk image using `vnd(4)`, and then compress that image with `vndcompress(1)`, put it on the CD, configure it with `vnconfig(8)`'s `-z` option, then mount as usual. You can use UFS or CD9660 as filesystem, your choice, `vnd(4)` won't care.

#### **re:ISO-File from unionfs** ([/article/7714#comment-128428](http://article/7714#comment-128428))

Submitted by [Bill Manson \(http://www.tutticompletti.com\)](http://www.tutticompletti.com) (not verified) on Tue, 01/31/2006 - 03:55.

apparently, unionfs is a standard part of slax. but of course it is only operating when you're booting from the slax cd, not if you're mounting it as an iso. and even then, it only creates the illusion of a modifiable fs, the cd itself isn't changed.

#### **Re:ISO-File from unionfs** ([/article/7714#comment-127103](http://article/7714#comment-127103))

Submitted by Anonymous (not verified) on Tue, 12/06/2005 - 17:34.

Do it with ex2fs . An Image using ex2fs (or other native Linux fs) can be mounted/read ro in Linux without a problem

**[error in code examples?](#)** ([/article/7714#comment-27543](#))

Submitted by Anonymous (not verified) on Sun, 04/03/2005 - 10:37.

In some code examples there is a "," and in some a ":" between the two "dirs=". I get an error when using ",", with ":" it works.

**[NFS exports](#)** ([/article/7714#comment-14669](#))

Submitted by Anonymous (not verified) on Mon, 01/03/2005 - 22:16.

Most problems reported above (compile-time error, Unknown error 524 while doing ls, etc) are fixed in v1.0.4.

However, although the article claims that it is possible to NFS-export and mount a unionfs FS, I get a "getfh failed: Operation not permitted" from mountd when I try to mount it from remote. If I export a parent directory of the unionfs dir, I can mount the parent but cd-ing to the unionfs dir shows an empty directory. Has anyone ever succeeded in NFS-exporting and remote mounting a unionfs directory?

Cheers,  
Gabor

**[Re: NFS exports](#)** ([/article/7714#comment-26431](#))Submitted by [Geoff Mishkin](#) (<http://amsa.dnsalias.org>) (not verified) on Tue, 03/15/2005 - 18:47.

I have a similar problem in version 1.0.9. When I try to mount the unionized directory, I get:

```
mount: amsa.dnsalias.org:/mnt/amsa-e failed, reason given by server: Permission denied
```

However, in the server logs, it would seem that the mount is successful:

```
Mar 15 18:21:12 [rpc.mountd] authenticated mount request from comm700-c0501-dhcp088.bu.edu:707 for /mnt/amsa-e (/mnt/amsa-e)
```

However, the packet trace indicates a return status of `ERR_ACCESS` to the mount request.

Anyone else have a problem like this?

**[The official NEWS file in the](#)** ([/article/7714#comment-38175](#))

Submitted by Anonymous (not verified) on Thu, 05/05/2005 - 13:14.

The official NEWS file in the current download (1.0.11) admits that NFS was broken in 1.0.9 on 2.6 kernels, and claims this to be fixed in 1.0.11.

**[NFS Exports](#)** ([/article/7714#comment-27869](#))

Submitted by Ian Rawlings (not verified) on Mon, 04/04/2005 - 15:26.

Yep, also having exactly the same problem, it appears to be impossible to export a unionfs via NFS. Just to test I wasn't doing anything wrong I have tried exporting the parent directory of a unionfs mountpoint and was able to see the directory and the unionfs mountpoint, but trying to list the contents of the unionfs mountpoint (i.e. the contents of the unionfs itself) lead to a frozen command that eventually freed up with a "permission denied" once I'd restarted the NFS server.

A great shame, UnionFS would be very useful for me and my discless clients, but it appears that it's not compatible with NFS. Note that the initial "authenticated mount request" in the logs means that mountd was happy with the mount request, so it's the NFS daemons/kernel threads that are refusing to co-operate.

**[RE: NFS](#)** ([/article/7714#comment-65751](#))

Submitted by derFlo (not verified) on Wed, 06/15/2005 - 23:45.

official mailinglist:

<http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-June/000610.html>

(<http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-June/000610.html>)

cite:knfsd will not export Unionfs unless you

have an fsid option in /etc/exports

<http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-April/000350.html>

(<http://www.fsl.cs.sunysb.edu/pipermail/unionfs/2005-April/000350.html>)

cite:/mnt/unionfs \*(rw,fsid=42,async)

i dont know what this fsid means (goin to sleep now), but just (touch) created a file thru a nfs mounted directory that showed up in the rw branch of my union.. (debian kernel-image-2.6-11-1k7 , module-assistent,unionfs unionfs-source 1.0.11-1)

**[unionfs for 2.6](#)** ([/article/7714#comment-14611](#))

Submitted by Anonymous (not verified) on Wed, 12/29/2004 - 08:01.

Is unionFS is available for 2.6 kernels?

**[Article and related home page lack download link](#)** ([/article/7714#comment-13464](#))

Submitted by Anonymous (not verified) on Sun, 11/07/2004 - 02:53.

I was quite interested in the unionfs article in the newest issue but can not find any link to download the unionfs software, even from the project's homepage. Please talk with the author to resolve this oversight.



**Software available here** ([/article/7714#comment-13546](#))

Submitted by Anonymous (not verified) on Mon, 11/22/2004 - 10:19.

<ftp://ftp.fsl.cs.sunysb.edu/pub/unionfs> (<ftp://ftp.fsl.cs.sunysb.edu/pub/unionfs>)

Though, I've been trying to get it to compile with 2.4.27/28 and had no luck so far...



**fix f. kernel 2.4.27** ([/article/7714#comment-14848](#))

Submitted by Anonymous (not verified) on Wed, 01/12/2005 - 22:18.

if it's about an error concerning implicit declaration of IS\_IMMUTABLE, try this:

```
--- namei.c.org Thu Jan 13 03:18:06 2005
+++ namei.c Thu Jan 13 03:38:17 2005
@@ -67,7 +67,7 @@
if (IS_APPEND(dir))
return -EPERM;
if (check_sticky(dir, victim->d_inode)||IS_APPEND(victim->d_inode)||
- IS_IMMUTABLE(victim->d_inode))
+ IS_IMMUTABLE_FILE(victim->d_inode)||IS_IMMUTABLE_LINK(victim->d_inode))
return -EPERM;
if (isdir) {
if (!IS_ISDIR(victim->d_inode->i_mode))
```

make CC=gcc-3.0



<http://www.fsl.cs.sunysb.edu/> ([/article/7714#comment-13547](#))

Submitted by Anonymous (not verified) on Mon, 11/22/2004 - 10:27.

<http://www.fsl.cs.sunysb.edu/project-unionfs.html>  
(<http://www.fsl.cs.sunysb.edu/project-unionfs.html>)

Oops. This is the author's page for the above link. Sorry



**/src/unionfs-1.0.3# make all** ([/article/7714#comment-13548](#))

Submitted by Anonymous (not verified) on Mon, 11/22/2004 - 10:58.

```
/src/unionfs-1.0.3# make all
gcc -D__KERNEL__ -DMODULE -DFISTGEN -I. -
I/lib/modules/`uname -r`/build/include -Wall -Wno-unused-
label -Werror -g -O2 -c -o subr.o subr.c
In file included from /lib/modules/2.4.27-1-
k7/build/include/linux/spinlock.h:6,
from /lib/modules/2.4.27-1-k7/build/include/linux/wait.h:16,
from /lib/modules/2.4.27-1-k7/build/include/linux/fs.h:12,
from /lib/modules/2.4.27-1-k7/build/include/linux/capability.h:17,
from /lib/modules/2.4.27-1-k7/build/include/linux/binfmts.h:5,
from /lib/modules/2.4.27-1-k7/build/include/linux/sched.h:9,
from fist.h:32,
from subr.c:22:
/lib/modules/2.4.27-1-k7/build/include/asm/system.h: In function
`__set_64bit_var':
/lib/modules/2.4.27-1-k7/build/include/asm/system.h:190: warning:
dereferencing type-punned pointer will break strict-aliasing rules
/lib/modules/2.4.27-1-k7/build/include/asm/system.h:190: warning:
dereferencing type-punned pointer will break strict-aliasing rules
make: *** [subr.o] Error 1
```

I get the same results with a virgin 2.4.28 source code too.

Anyone?



**Got unionfs working** ([/article/7714#comment-13592](#))

Submitted by Steve (not verified) on Sat, 11/27/2004 - 18:25.

I finally got unionfs working after...

Downgrading from 2.6.9 to 2.4.28  
and  
Downgrading from gcc 3.3.4 to gcc 3.2.3 (just for the build)

I hope this helps.

Steve



**Spoke too soon...** ([/article/7714#comment-13597](#))

Submitted by Anonymous (not verified) on Sun, 11/28/2004 - 12:15.

I've tried searching google, but with no luck. First of I discovered that



the command to mount a unionfs is mount -t unionfs -o dir=dirone:/dirtwo none /combined ... you'll note the : as the article uses a , separator. The other problem I'm having is running 'ls -l' on the /combined directory give me multiple "Unknown error 524" for all subdirectories, thought it seems to be working.

---

**Perfect! It works here too. J** (</article/7714#comment-13596>)

Submitted by Anonymous (not verified) on Sun, 11/28/2004 - 04:12.



Perfect! It works here too. Just incase anyone is interested. I'm using Debian Sarge and just installed the gcc-3.2 compiler. Then changed the symlink gcc to point to gcc-3.2.

Thanks Steve! :-)