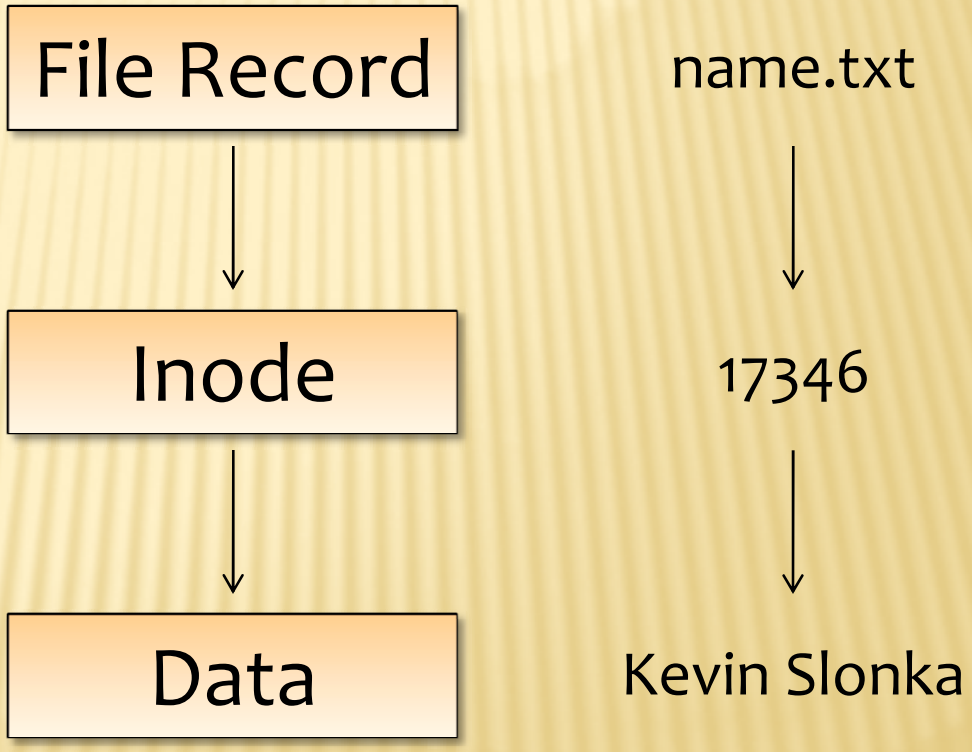
SYMBOLIC AND HARD LINKS EXPLAINED

Kevin J. Slonka
2008-09-18
http://www.kevinslonka.com/

A NORMAL FILE

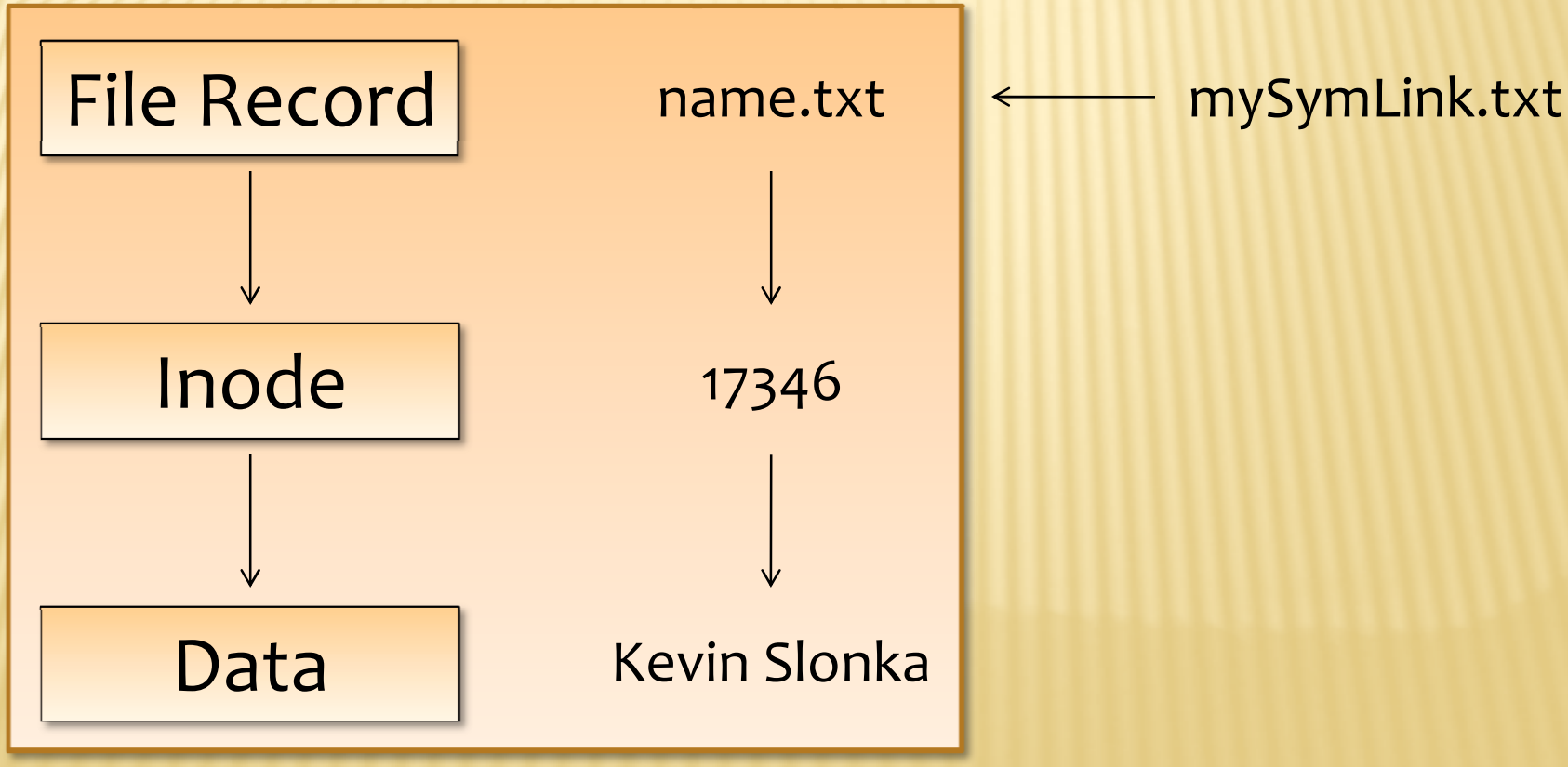
A file record contains the human readable filename and which inode points to the data

The inode contains the UID, GID, permissions, file size, timestamp, and where the data lives on the hard drive



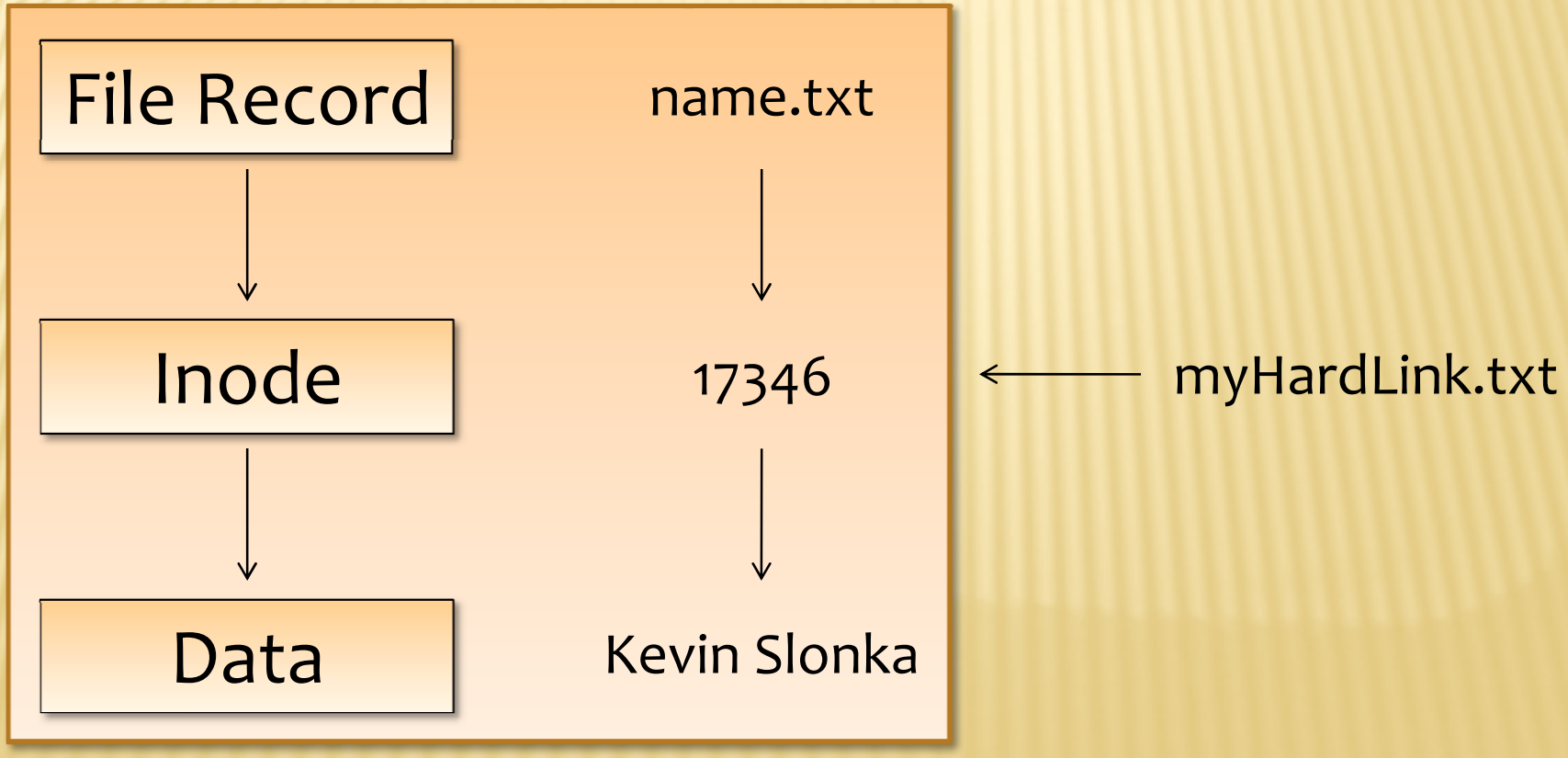
A SYMBOLIC LINK

A symbolic link is a separate ***file record*** that points to another ***file record***



A HARD LINK

A hard link is a separate ***file record*** that points directly to an ***inode***



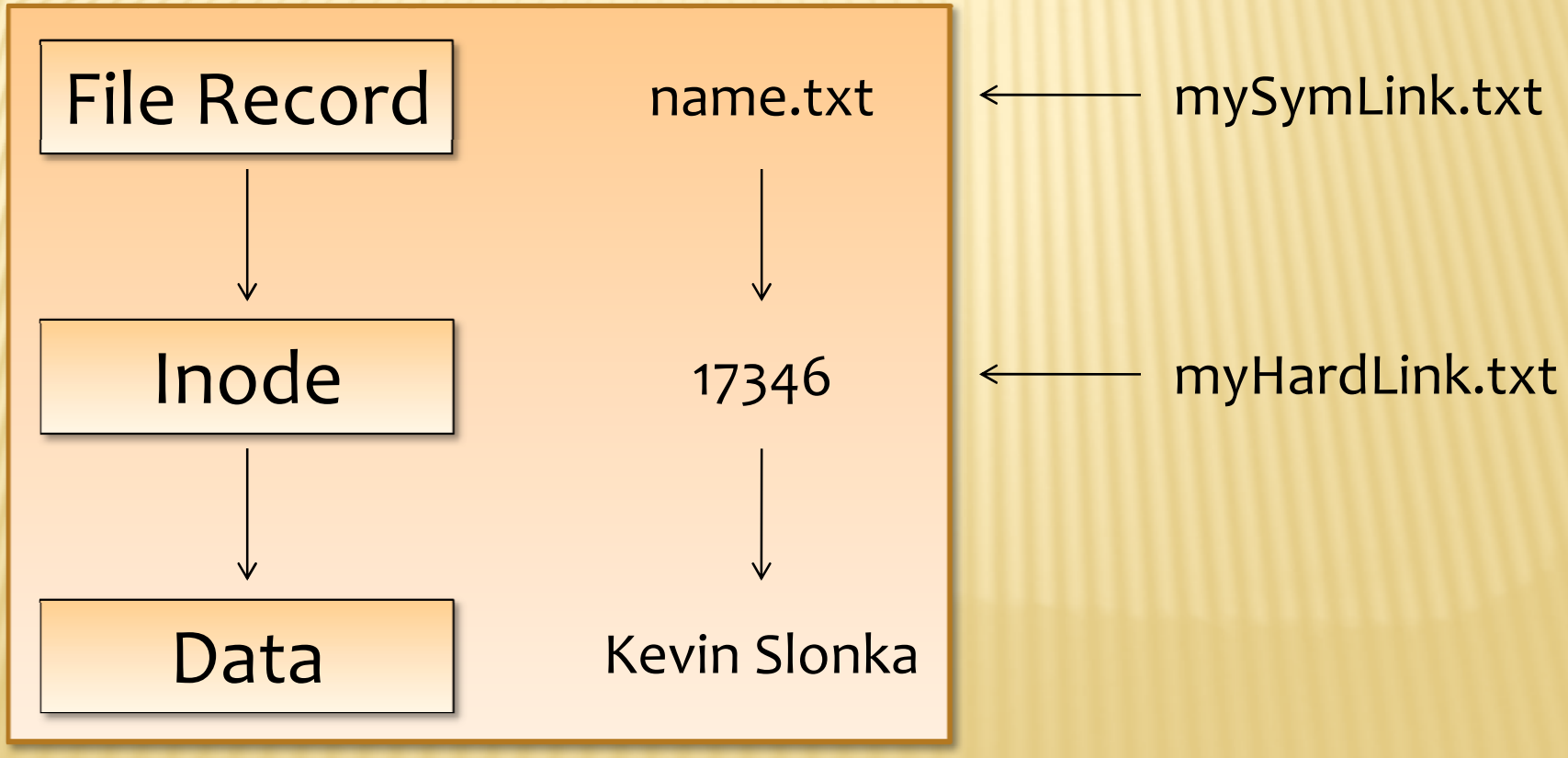
A NOTE ABOUT FILE DELETION

- When you delete a file, you are actually only deleting the ***file record***
- When an ***inode*** has no more ***file records*** associated with it, the ***inode*** and the ***data*** to which it points are no longer accessible*

* Without forensic utilities

THE SITUATION

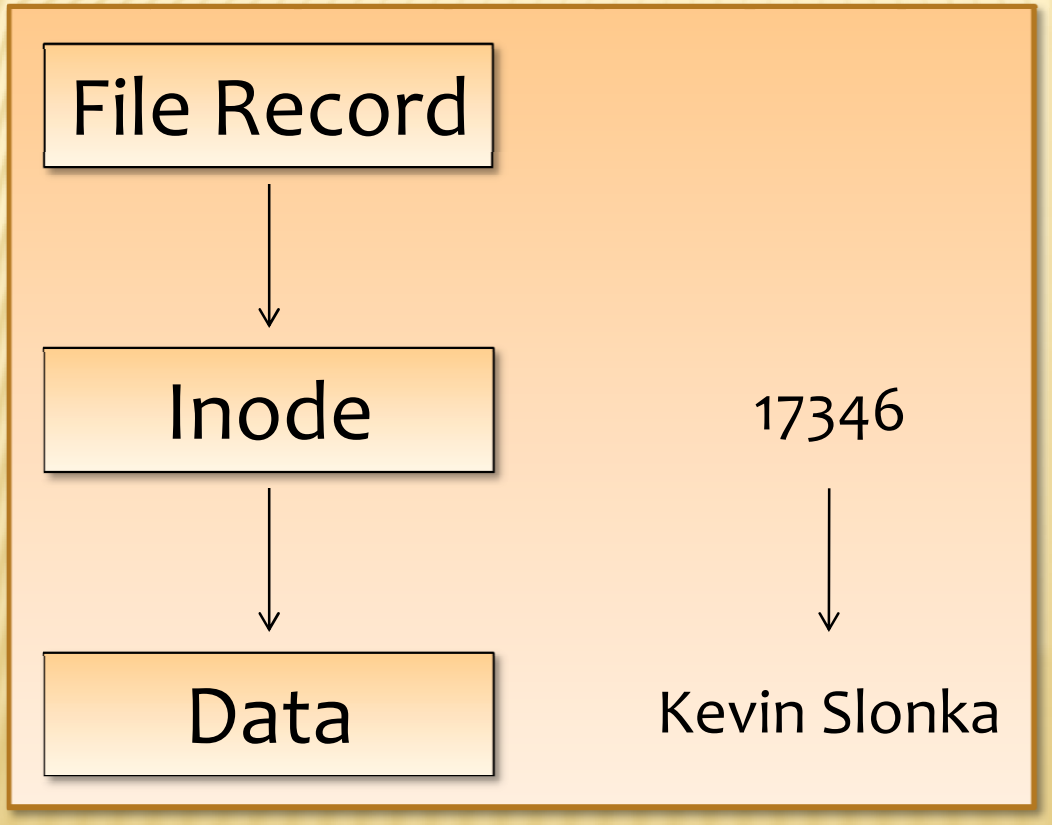
We have a file record, *name.txt*. We created a symbolic link of *name.txt*, ***mySymLink.txt***. We also created a hard link of *name.txt*, ***myHardLink.txt***.



THE UH-OH

Someone deleted our original ***file record***, *name.txt*. What are the effects?

1) *mySymLink.txt* no longer works because the ***file record*** to which it pointed is gone
2) *myHardLink.txt* still works because it pointed to an ***inode***, not a ***file record****



* Note how the ***inode*** and ***data*** are still available, despite the deletion of *name.txt*. This is because there is still a ***file record*** that points to the ***inode*** referenced by *name.txt*

VIEW FROM A SHELL

- Assuming we have our original file and two links, here is how a long listing would appear

```
-rw-r--r--  2 slonkak wheel      13 2008-09-18 21:47 myHardLink.txt
lrwxrwxrwx  1 slonkak wheel       8 2008-09-18 21:47 mySymLink.txt -> name.txt
-rw-r--r--  2 slonkak wheel      13 2008-09-18 21:47 name.txt
```

Symbolic links announce themselves with an arrow pointing to the file to which they are linked

Since symbolic links do not point to an ***inode***, and therefore do not know the size of the ***data*** to which it is pointing, its listed size does not reflect the size of the ***data*** on disk

The second column of a long listing denotes the number of ***file records*** pointing to an ***inode***

Symbolic links have an "l" (lowercase L) in the first slot of the permissions