SYSTEMS GOSSIP MEETUP

# Learning large systems using peer-to-peer gossip

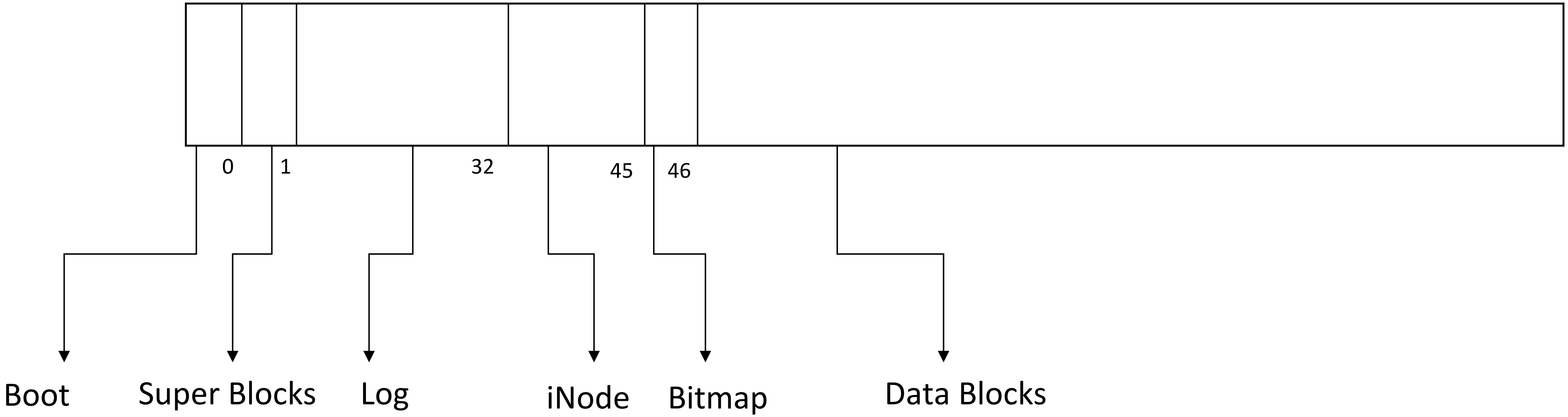
# Policy Against Harassment at ACM Activities

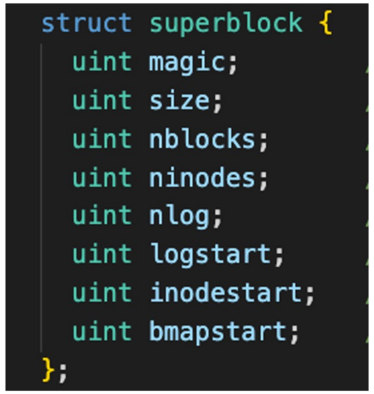
OS Meetup wants to encourage and preserve this open exchange of ideas, which requires an environment that enables all to participate without fear of personal harassment. We define harassment to include specific unacceptable factors and behaviors listed in the ACM's policy against harassment. Unacceptable behavior will not be tolerated.
https://www.acm.org/about-acm/policy-against-harassment

# File System

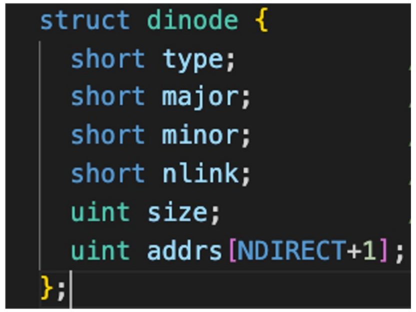
- Recall
  - File System
- File System Fault Tolerance
  - Crash
- Write Ahead Log
  - What is it?
- Write Ahead Log in xv6

# Recall: Disk Layout





```
struct superblock {
  uint magic;
  uint size;
  uint nblocks;
  uint ninodes;
  uint nlog;
  uint logstart;
  uint inodestart;
  uint bmapstart;
};
```



```
struct dinode {
  short type;
  short major;
  short minor;
  short nlink;
  uint size;
  uint addrs[NDIRECT+1];
};
```

# Recall: iNode

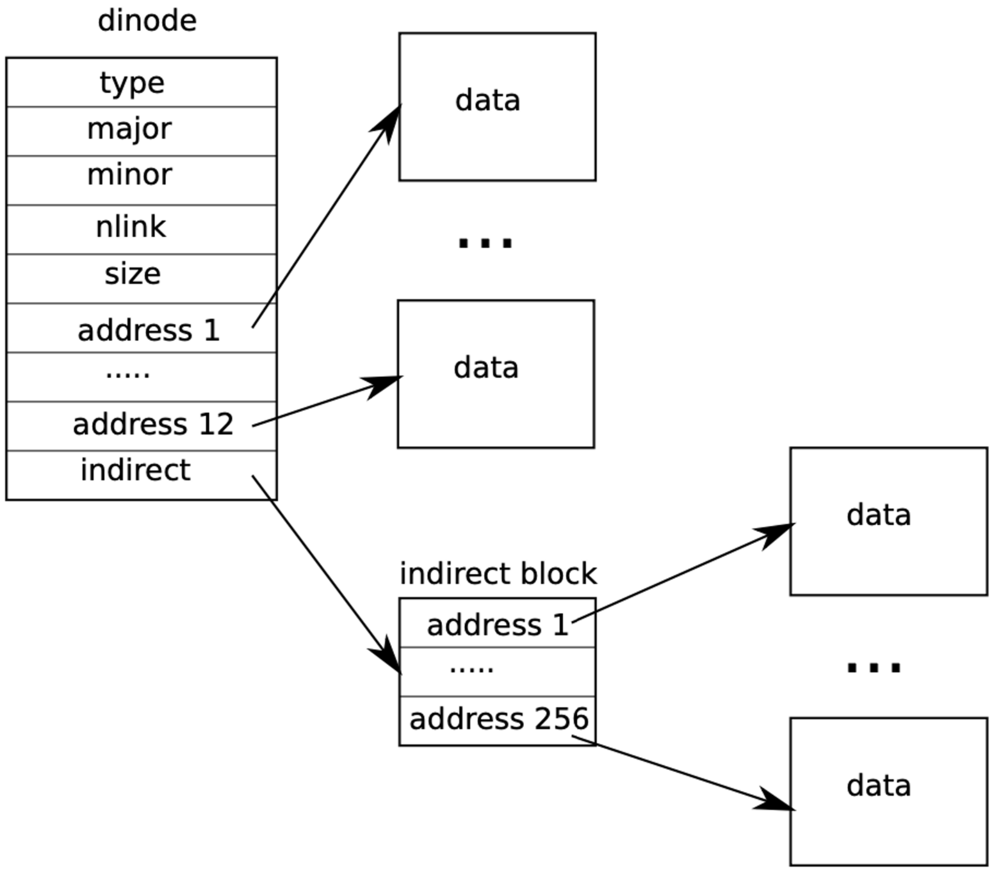
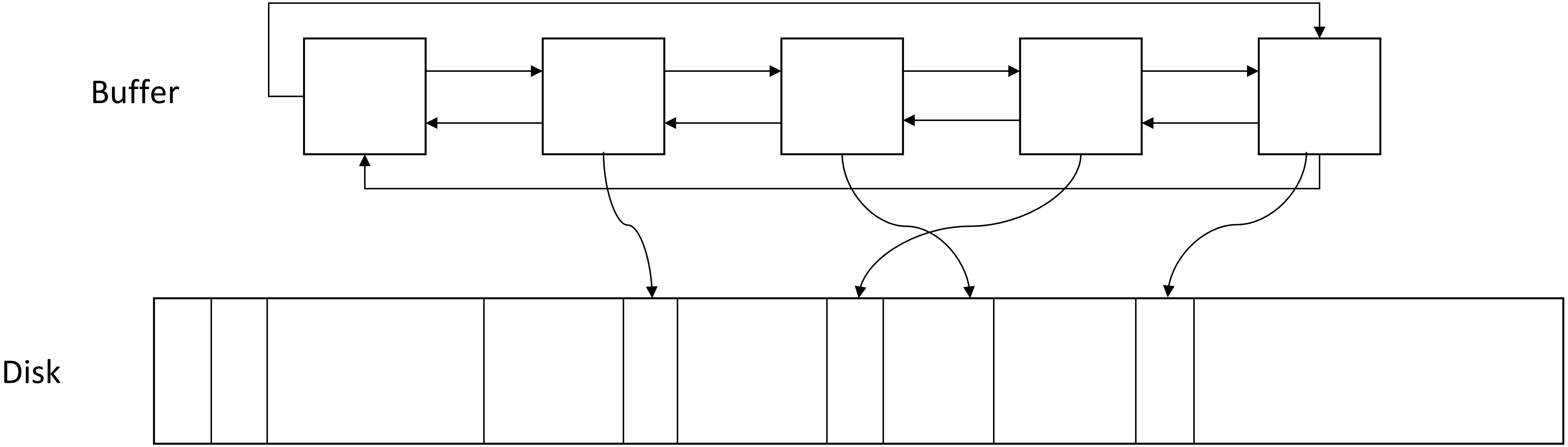


Figure 8.3: The representation of a file on disk.

# Recall: Buffer Cache

- Synchronize access to disk blocks to ensure that only one copy of a block is in memory and that only one kernel thread at a time uses that copy;
- Cache popular blocks so that they don't need to be re-read from the slow disk;



# File System: Crash

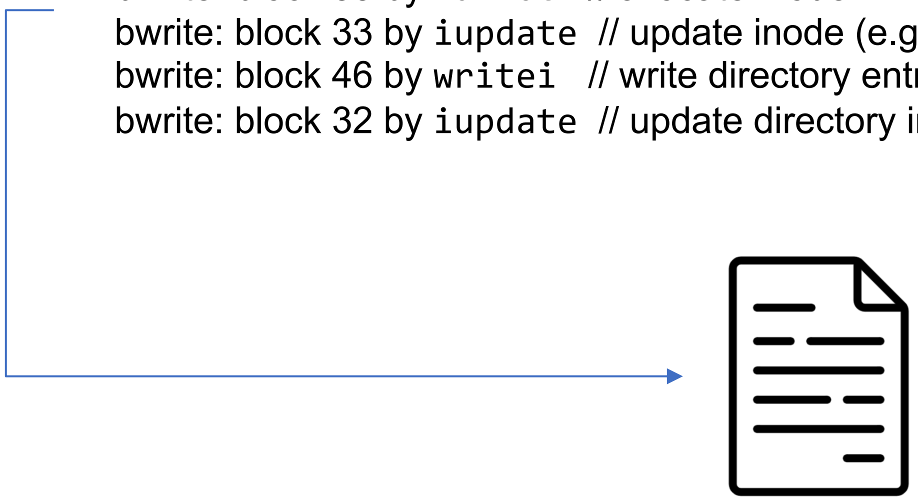
```
$ echo hi > x
   // create trace from last lecture:
   bwrite: block 33 by ialloc   // allocate inode in inode block 33
   bwrite: block 33 by iupdate  // update inode (e.g., set nlink)
   bwrite: block 46 by writei   // write directory entry, adding "x" by dirlink()
   bwrite: block 32 by iupdate  // update directory inode, because inode may have changed
```

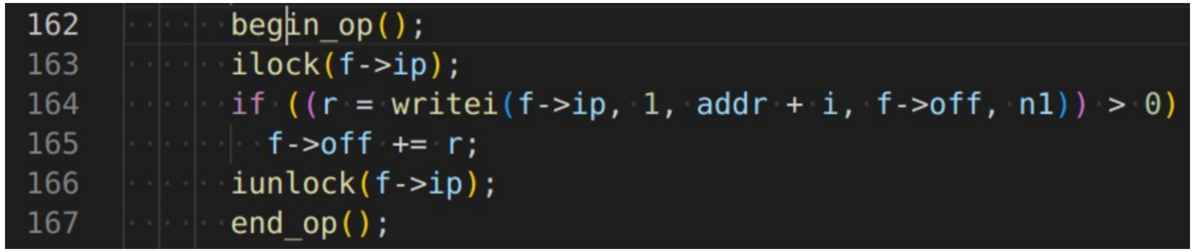
# File System: Crash

```
$ echo hi > x
   // create trace from last lecture:
   bwrite: block 33 by ialloc   // allocate inode in inode block 33
   bwrite: block 33 by iupdate  // update inode (e.g., set nlink)
   bwrite: block 46 by writei   // write directory entry, adding "x" by dirlink()
   bwrite: block 32 by iupdate  // update directory inode, because inode may have changed
```

# File System: Log Crash

```
$ echo hi > x
  // create trace from last lecture:
  bwrite: block 33 by ialloc   // allocate inode in inode block 33
  bwrite: block 33 by iupdate  // update inode (e.g., set nlink)
  bwrite: block 46 by writei   // write directory entry, adding "x" by dirlink()
  bwrite: block 32 by iupdate  // update directory inode, because inode may have changed
```





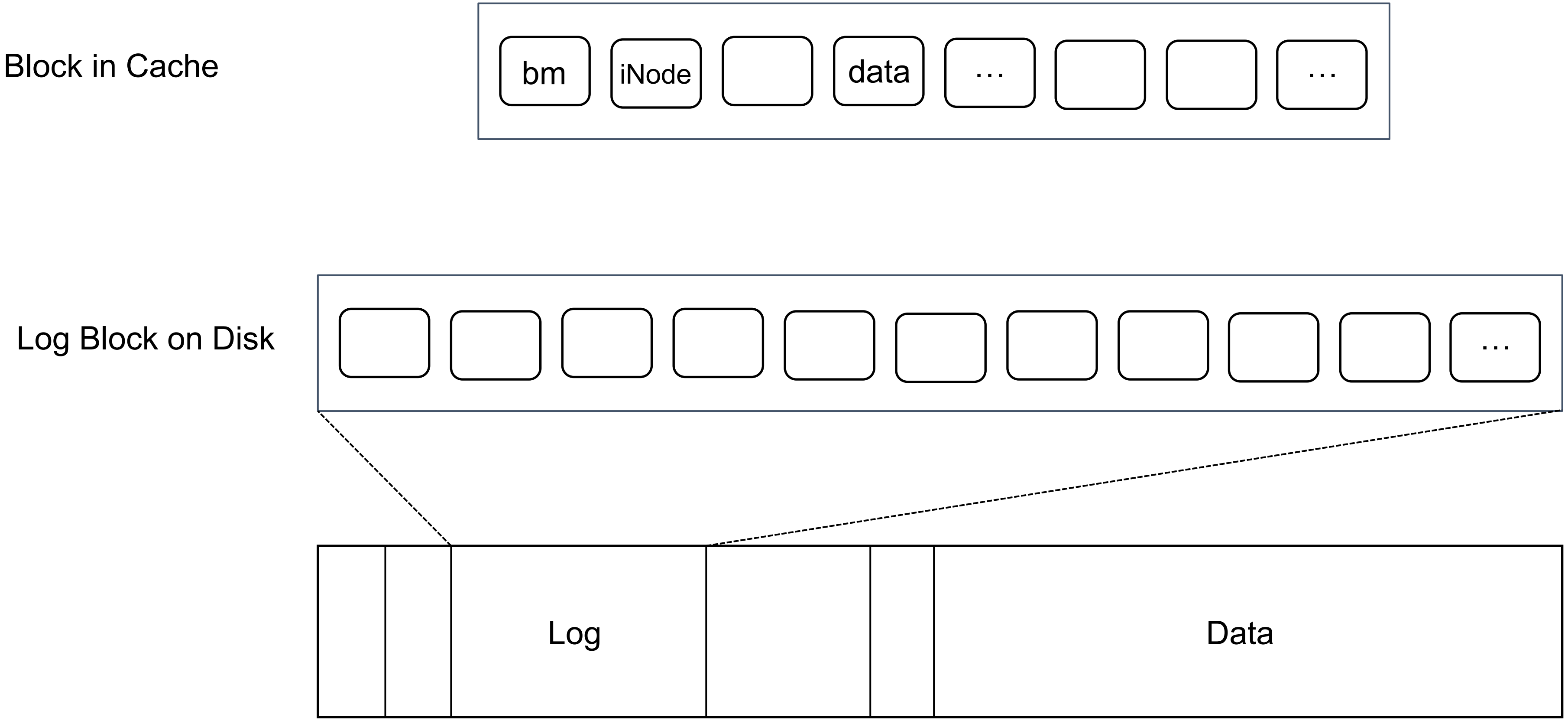
```
    begin_op();
    ilock(f->ip);
    if ((r = writei(f->ip, 1, addr + i, f->off, n1)) > 0)
        f->off += r;
    iunlock(f->ip);
    end_op();
```

# Write Ahead Log

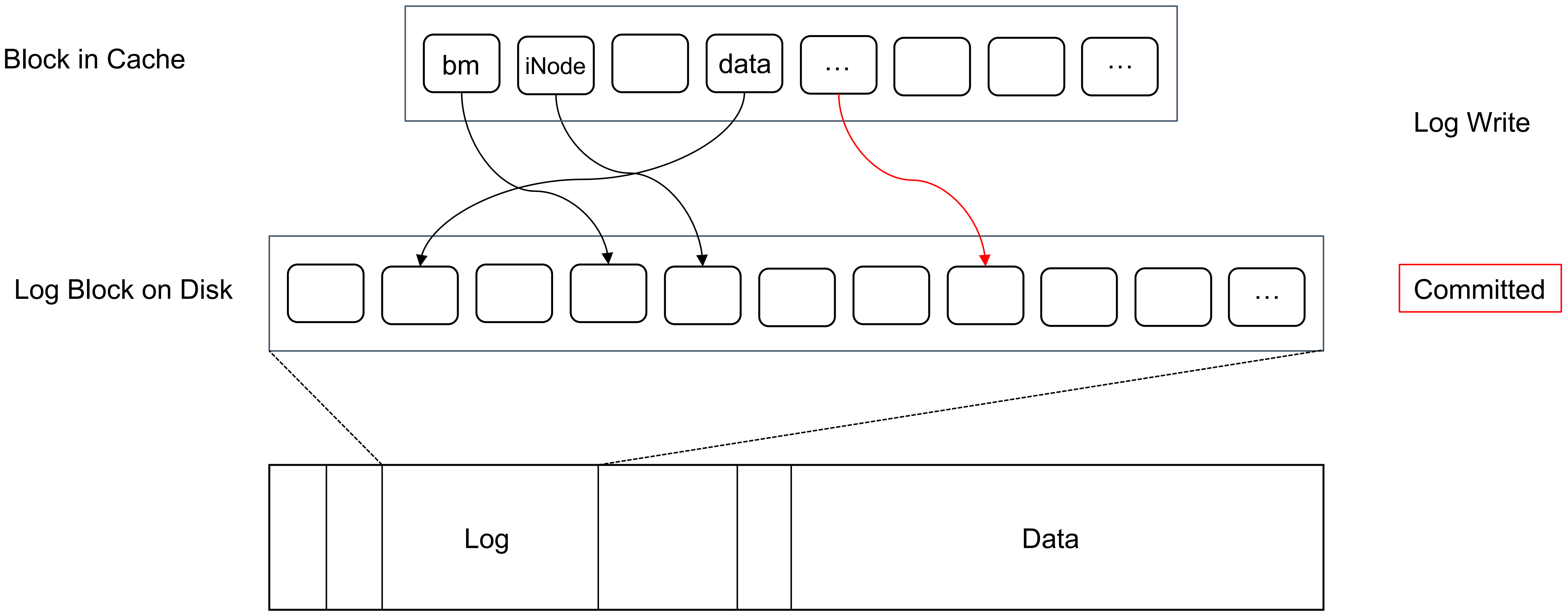
**DBMS** must write to disk the log file records that correspond to changes made to a database object **before** it can flush that object to disk.

**File System** must write to disk the log file records that correspond to changes made to a disk block **before** it can flush that block to disk.

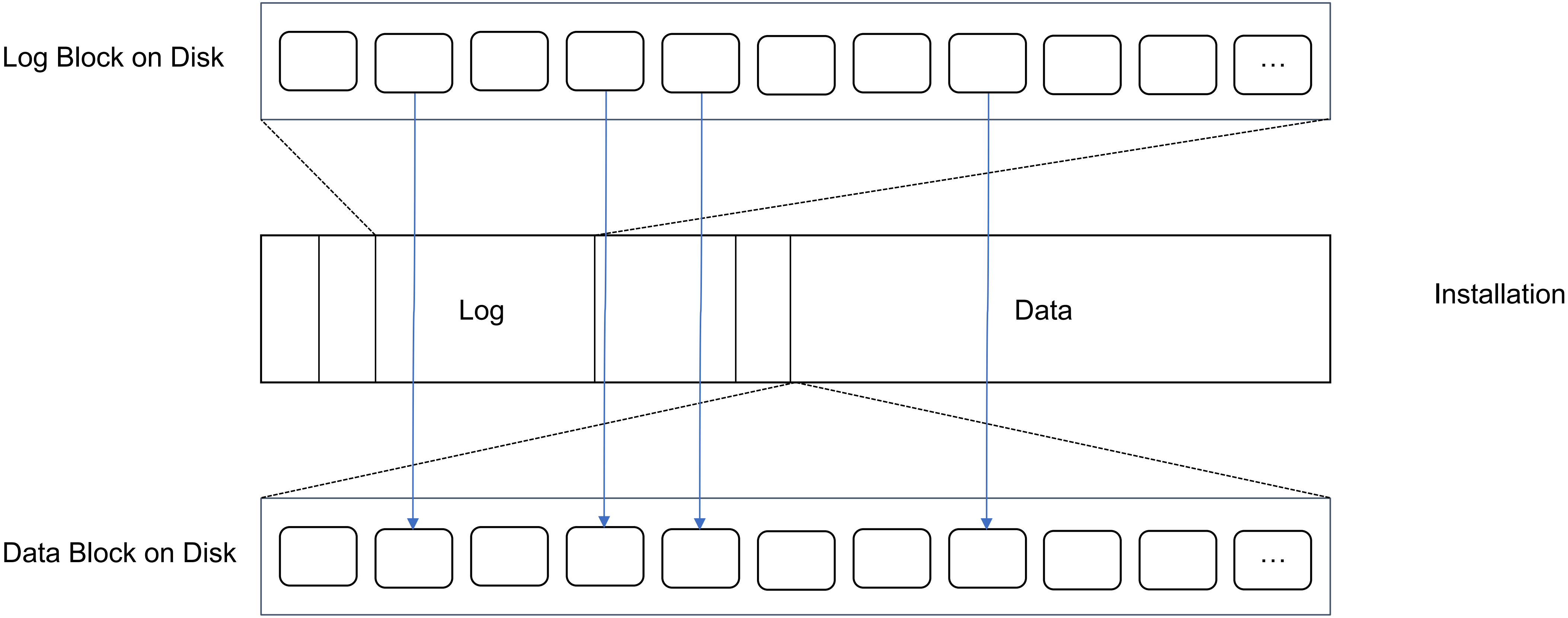
# WAL in xv6



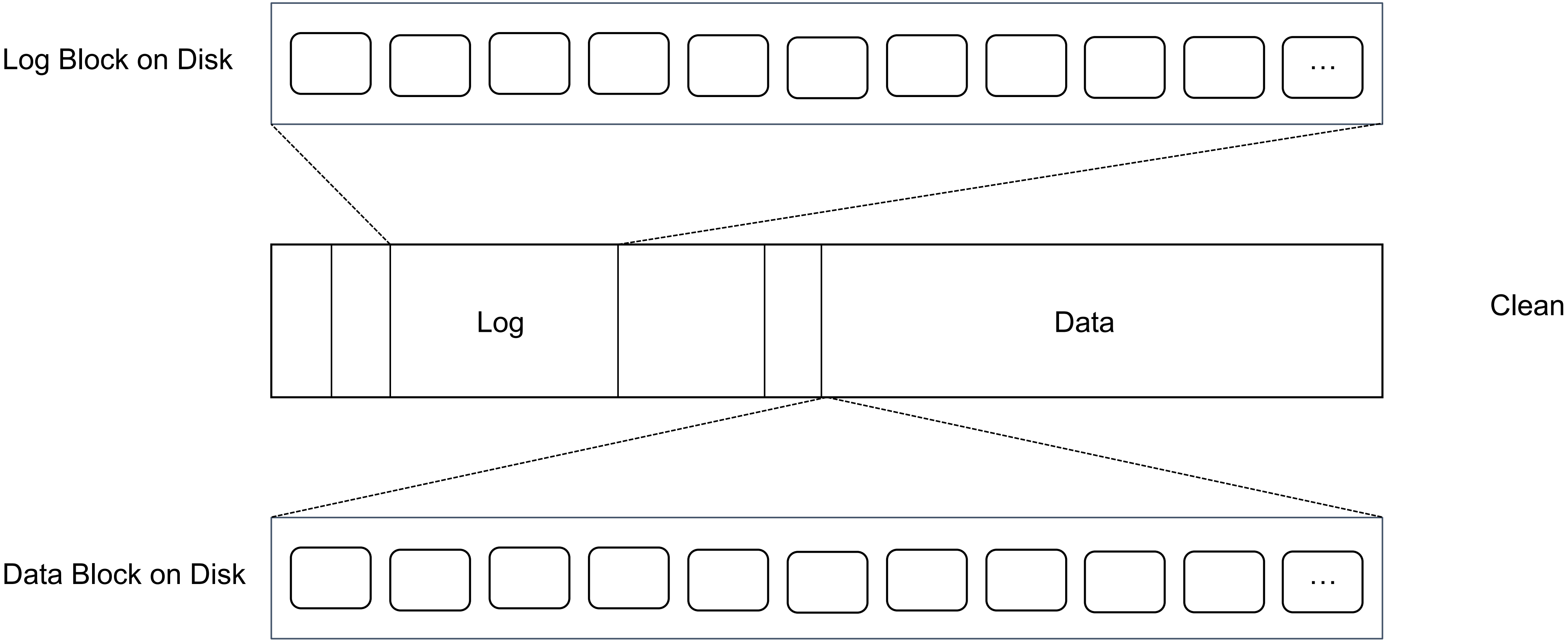
# WAL in xv6


Block in Cache
bm
iNode
data
…
…
Log Write
Log Block on Disk
…
Committed
Log
Data

# WAL in xv6

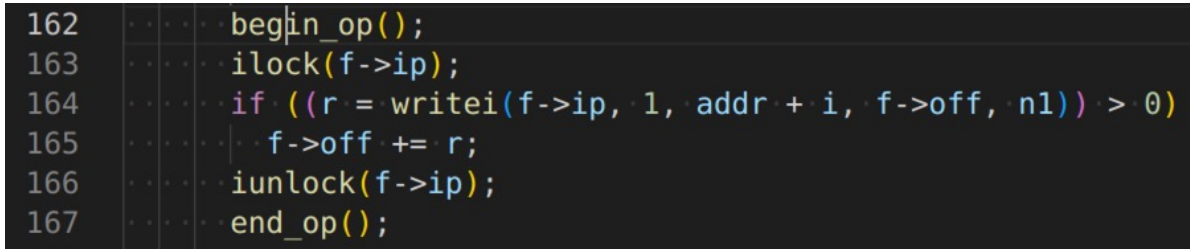


# WAL in xv6



# Open Discussion: Large File

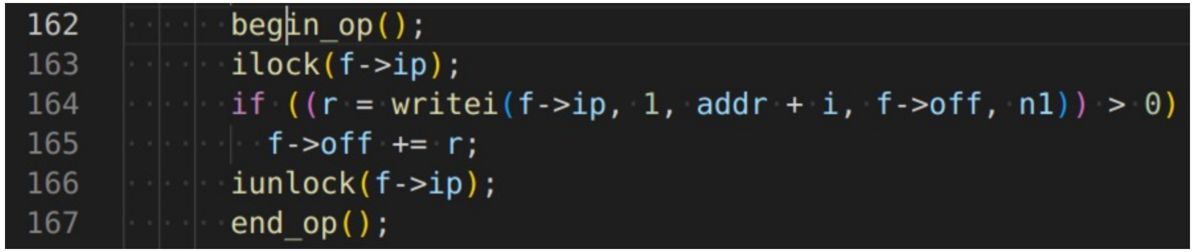
```
$ echo hi > x
   // create trace from last lecture:
   bwrite: block 33 by ialloc   // allocate inode in inode block 33
   bwrite: block 33 by iupdate  // update inode (e.g., set nlink)
   bwrite: block 46 by writei   // write directory entry, adding "x" by dirlink()
   bwrite: block 32 by iupdate  // update directory inode, because inode may have changed
```



```
    begin_op();
    ilock(f->ip);
    if ((r = writei(f->ip, 1, addr + i, f->off, n1)) > 0)
      f->off += r;
    iunlock(f->ip);
    end_op();
```

# Open Discussion: Drawbacks of xv6 Log

```
$ echo hi > x
  // create trace from last lecture:
  bwrite: block 33 by ialloc   // allocate inode in inode block 33
  bwrite: block 33 by iupdate  // update inode (e.g., set nlink)
  bwrite: block 46 by writei   // write directory entry, adding "x" by dirlink()
  bwrite: block 32 by iupdate  // update directory inode, because inode may have changed
```



```c
begin_op();
ilock(f->ip);
if ((r = writei(f->ip, 1, addr + i, f->off, n1)) > 0)
  f->off += r;
iunlock(f->ip);
end_op();
```

# Summary

- ○ Recall
- ○ WAL
- ○ Directory and Path
- ○ Cache
- ○ Next
  - File System Logging