The objective is to make the secondary master's file system exactly the same as the primary so that in the event of a failover, the system will function as it did prior to the failure. **tar** is used in order to preserve meta information (owner, date, permissions, etc) of the files. Thus, first the tar archives are created on the master and copied over to the secondary master (node 0). Then, the backup master file system partitions are mounted on the slave with the / directory mapped to /mnt/root, and the /var and /usr directories becoming /mnt/root/var and /mnt/root/usr. That accomplished, the tar files are unpacked into their file systems. Finally, the file systems are unmounted again so that they will be clean should failover be needed.

In summary:

1. **tar --one-file-system /scratch/root.tar /**
2. **tar --one-file-system /scratch/var.tar /var**
3. **tar --one-file-system /scratch/usr.tar /usr**
4. **cd /scratch**
5. **bpcp root.tar var.tar usr.tar 0:** (this step may take a moment)
6. **bpsh 0 mkdir /mnt/root**
7. **mkdir /mnt/root**
8. **bpsh 0 mount /dev/hda3 /mnt/root**
9. **cd /mnt/root**
10. **bpsh 0 tar xvf /scratch/root.tar**
11. **bpsh 0 mount /dev/hda5 var**
12. **bpsh 0 tar xvf /scratch/var.tar**
13. **bpsh 0 mount /dev/hda6 usr**
14. **bpsh 0 tar xvf /scratch/usr.tar**
15. **bpsh 0 sync**
16. **bpsh 0 umount usr**
17. **bpsh 0 umount var**
18. **cd ..**
19. **bpsh 0 umount root**