



### **AWS08:Kernel Parameters (*kconfig*) defaults & deltas for AWS95 File/Print and Database Server Configurations.**

Written by : Sandhya Vora  
Revised by: Chuq Von Rospach

August, 1993  
May, 1994

This technical note documents the default Kernel parameters settings for the File/Print and Database server Configurations under A/UX version 3.0.1 and 3.1 and how they are different from A/UX version 3.0. It also describes the new configuration parameters introduced in A/UX version 3.0.1 and their default settings.

#### **Kernel Configuration Defaults**

<b>Variable</b>	<b>File/Print</b>	<b>Database</b>	<b>3.0 Defaults</b>
MAXSC	32	32	8
NBLK16	256	256	128
NBLK64	800	800	64
NBLK128	256	256	32
NBLK256	256	256	16
NBLK512	128	128	8
NBLK1024	300	300	8
NBUF	-1	0	0
NCALL	500	150	75
NFILE	7500	1024	256
NINODE	7500	1024	256
NMOUNT	32	32	10
NPBUF	64	64	20
NQUEUE	1200	1200	256
NSTREAM	200	200	32
SBUFSIZE	4096	4096	2048

#### **Kernel Configuration Parameters Introduced in 3.0.1**

<b>Variable</b>	<b>Description</b>	<b>Default</b>
GNOFILE	Global File Table (used by Mac environment)	6144
MSGMAP	Message Map Entries (used by msgop(2))	200
MSGMAX	Maximum Message Size (used by msgop(2))	8192
MSGMNB	Max. Message Queue Size (used by msgop(2))	16384
MSGMNI	Message Queue Identifiers (used by msgop(2))	50
MSGTQL	Number of Message Headers	200
NASIOD	Number of Asynchronous I/O Daemons	4
NASIOQ	Asynchronous I/O Request Queue	128
SEMMAP	Semaphore Map Entries (see semop(2))	50
SEMMNI	Semaphore Identifiers (see semop(2))	50
SEMMNS	Total Number of Semaphores (see semop(2))	300

For more information on kernel parameters, refer to the on-line man page for *kconfig*.

In release 3.1, the behavior of the system has changed when NBUF is set to -1. The new algorithm for determining the size of the buffer cache is:

```
if (v.v_buf == -1) {
    /* if < 20Mb available just allocate 10% for bufs */
    if (availmem <= 5120)
        m = availmem / 10;
    /* else if < 40Mb available, reserve 20Mb for procs */
    else if (availmem < 10240)
        m = availmem - 5120;
    /* else split 50/50 */
    else
        m = availmem / 2;

    /* make sure we've got at least 10% for the bufs */
    if (m < (availmem / 10))
        m = availmem / 10;

    v.v_buf = ptob(m) / v.v_sbufsz;
} else if (v.v_buf == 0)
    v.v_buf = ptob(availmem) / (10 * v.v_sbufsz);
```