NAME

ks_update – dynamically update kstats

SYNOPSIS

#include <sys/types.h>
#include <sys/kstat.h>
#include <sys/ddi.h>
#include <sys/sunddi.h>

int *prefix***_ks_update(kstat_t** **ksp* **, int** *rw***);**

INTERFACE LEVEL

Solaris DDI specific (Solaris DDI)

ARGUMENTS

ksp Pointer to a **kstat**(9S) structure.

rw Read/Write flag. Possible values are

KSTAT_READ Update kstat structure statistics from the driver.

KSTAT_WRITE Update driver statistics from the kstat structure.

DESCRIPTION

The kstat mechanism allows for an optional **ks_update( )** function to update kstat data. This is useful for drivers where the underlying device keeps cheap hardware statistics, but extraction is expensive. Instead of constantly keeping the kstat data section up to date, the driver can supply a **ks_update( )** function which updates the kstat's data section on demand. To take advantage of this feature, set the **ks_update** field before calling **kstat_install**(9F).

The **ks_update( )** function must have the following structure:

```
static int
xx_kstat_update(kstat_t *ksp, int rw)
{
        if (rw == KSTAT_WRITE) {
                /* update the native stats from ksp->ks_data */
                /* return EACCES if you don't support this */
        } else {
                /* update ksp->ks_data from the native stats */
        }
        return (0);
}
```

In general, the **ks_update( )** routine may need to refer to provider-private data; for example, it may need a pointer to the provider's raw statistics. The **ks_private** field is available for this purpose. Its use is entirely at the provider's discretion.

No kstat locking should be done inside the **ks_update( )** routine. The caller will already be holding the kstat's **ks_lock** (to ensure consistent data) and will prevent the kstat from being removed.

RETURN VALUES

ks_update( ) should return

0	for success
EACCES	if **KSTAT_WRITE** is not allowed
EIO	for any other error.

SEE ALSO

kstat_create(9F), **kstat_install**(9F), **kstat**(9S)

Writing Device Drivers