

NAME

mac_rx, **mac_ring_rx** - deliver frames from a driver to the system

SYNOPSIS

```
#include <sys/mac_provider.h>
```

void

```
mac_rx(mac_handle_t mh, mac_resource_handle_t mrh, mblk_t *mp_chain);
```

void

```
mac_rx_ring:(mac_handle_t mh, mac_ring_handle_t mring, mblk_t *mp_chain, uint64_t mr_gen);
```

INTERFACE LEVEL

illumos DDI specific

The **mac_rx_ring**() function point is **Evolving**. API and ABI stability is not guaranteed.

PARAMETERS

<i>mh</i>	The MAC handle obtained from a call to mac_register (9F).
<i>mrh</i>	A reserved parameter that should be passed as NULL.
<i>mring</i>	A pointer to the ring handle that was passed to the driver in the mr_rget (9E) entry point.
<i>mp_chain</i>	A series of one or more mblk (9S) structures chained together by their b_next member.
<i>mr_gen</i>	The generation number for the current ring. The generation comes from the mri_start (9E) entry point.

DESCRIPTION

The **mac_rx**() function is used by device drivers to deliver frames that a device driver has received to the rest of the operating system. This will generally be called at the end of a device driver's interrupt handler after it has converted all of the incoming data into a chain of **mblk**(9S) structures. For a full description of the process that the device driver should take as part of receiving data, see the *Receiving Data* section of **mac**(9E).

Device drivers should ensure that they are not holding any of their own locks when they call the **mac_rx**() function.

Device drivers should not call the **mac_rx**() function after each individual **mblk_t** is assembled. Rather,

the device driver should batch up as many frames as it is willing to process in a given interrupt or are available.

The **mac_rx_ring()** function is similar to the *mac_rx* function; however, it should be called by device drivers that have negotiated the MAC_CAPAB_RINGS capability and indicated that it supports receive groups. Device drivers that have negotiated this capability must not call the **mac_rx()** function, but use the **mac_rx_ring()** function instead. The driver should pass the ring handle in *mring* for the ring in question that it processed. If more than one ring was processed during an interrupt, then the driver must call **mac_ring_rx()** once for each ring and ensure that the *mr_gen* argument matches what was passed to the driver during the *mri_start*(9E) entry point.

When a driver supporting the MAC_CAPAB_RINGS capability is asked to poll via their *mri_poll*(9E) entry point, then the driver should not call the **mac_ring_rx()** function.

CONTEXT

The **mac_rx()** function can be called from **user**, **kernel**, or **interrupt** context.

SEE ALSO

mac(9E), *mac_capab_rings*(9E), *mr_rget*(9E), *mri_poll*(9E), *mri_start*(9E), *mac_register*(9F), *mblk*(9S)