

# Cmnlib Introduction

Charlie Chen



# Purpose

Accton

- **Cmnlib is shorthand for common library. This library provides functionalities that might often be needed by most of programming tasks.**

EdgeCore Networks Confidential

# Outline

Accton

---

- **Memory Management Modules**
- **Data Structure Modules**
- **Conversion Modules**
- **SYSFUN**

EdgeCore Networks Confidential

# Memory Management Modules

Accton

- The primary object of this module types is to provide functionalities related to memory management, including memory allocation and release, memory multiple reference management, etc.
- MM (Memory Management/Memory Monitor) provides memory allocation/free functions with Monitor Information included at the head of each allocated buffer. Therefore the usage of the allocated buffer can be easily traced.
- MM also provides MREF (Memory Reference/Multiple Reference) for a single data block to be accessed by multiple tasks without replicating or reproducing the data block.

# Memory Management Modules

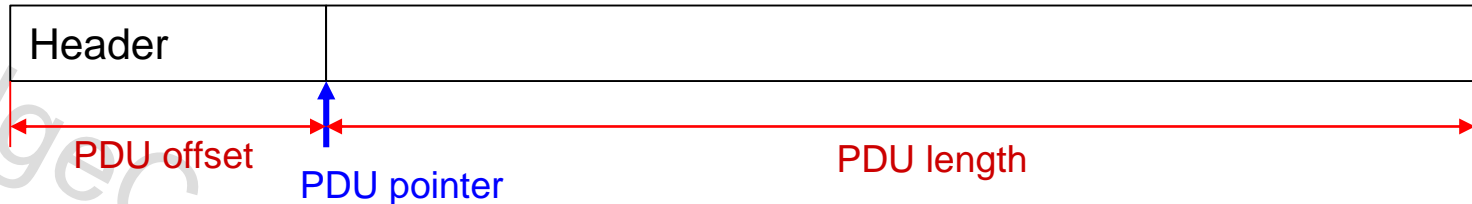
Accton

- **Provide memory allocation, free and add monitor header. We can monitor the memory status from backdoor.**
  - L\_MM\_Malloc – Normal memory allocate
  - L\_MM\_Free – Use L\_MM\_Free for L\_MM\_Malloc

# Memory Management Modules

Accton

- **Memory Reference provides for TX/RX packets.**



- **L\_MM\_Mref\_Construct** – Initiate the mem\_ref information in raw-buffer and set ref\_count as 1 for multiple reference.
- **L\_MM\_Mref\_AddRefCount** – Add ref\_count for multiple reference.
- **L\_MM\_Mref\_Release** – Free the mem\_ref. If reference count is 0 then free the associated buffer.

# Memory Management Modules

Accton

```
/* -----  
* FUNCTION : DEV_NICDRV_DequeueAndProcessPacket  
* -----  
* PURPOSE  : Dequeue frame buffer from WFQ, dispatch to IUC or LAN  
* INPUT    : None  
* OUTPUT   : None  
* RETURN   : TRUE  -- dequeue success  
*           : FALSE -- dequeue fail  
* NOTE     : 1. After dequeue packet, dispatch it to IUC or LAN according to packet  
*           : type  
* -----*/  
static BOOL_T DEV_NICDRV_DequeueAndProcessPacket(void)  
{  
    DataTransferBlock_T *indata_p;  
    L_MM_Mref_Handle_T *mref_handle_p;  
    ControlInfo_T *hdr;  
    WfqMessage_T msg;  
  
    ...  
  
    indata_p = (DataTransferBlock_T *) (msg.data_p);  
    hdr      = &(indata_p->control_info);  
  
    mref_handle_p = L_MM_Mref_Construct(  
        (UI8_T *) (indata_p->buffer), /* packet buffer pointer */  
        SYS_BLD_MAX_LAN_RX_BUF_SIZE_PER_PACKET, /* data block size */  
        0, /* pdu offset, set to 0 as temporary */  
        hdr->size, /* pdu length */  
        L_MM_MREF_FREE_FUN_RX_BUFFER,  
        (void *) ((UI32_T) hdr->cos),  
        NULL);  
  
    ...  
  
    recv_packet_callback[DEV_NICDRV_PROTOCOL_LAN](unit, port, (void *) &dev_nic_mhash, hdr->reason, mref_handle_p);  
  
    return TRUE;  
}
```

# Memory Management Modules

Accton

- **Memory Reference provides for TX/RX packets.**
  - **L\_MM\_AllocateTxBuffer** – Allocate buffer for transmission, and a mem\_ref will be constructed (Don't need to call L\_MM\_Mref\_Construct). The PDU(Protocol Data Unit) offset will be set as default header length (SYS\_ADPT\_TX\_BUFFER\_MAX\_RESERVED\_HEADER\_LEN) to guarantee that lower level module can use this buffer to transmit directly. Callers should write data to buffer from pdu pointer which could be retrieved via L\_MM\_Mref\_GetPdu().



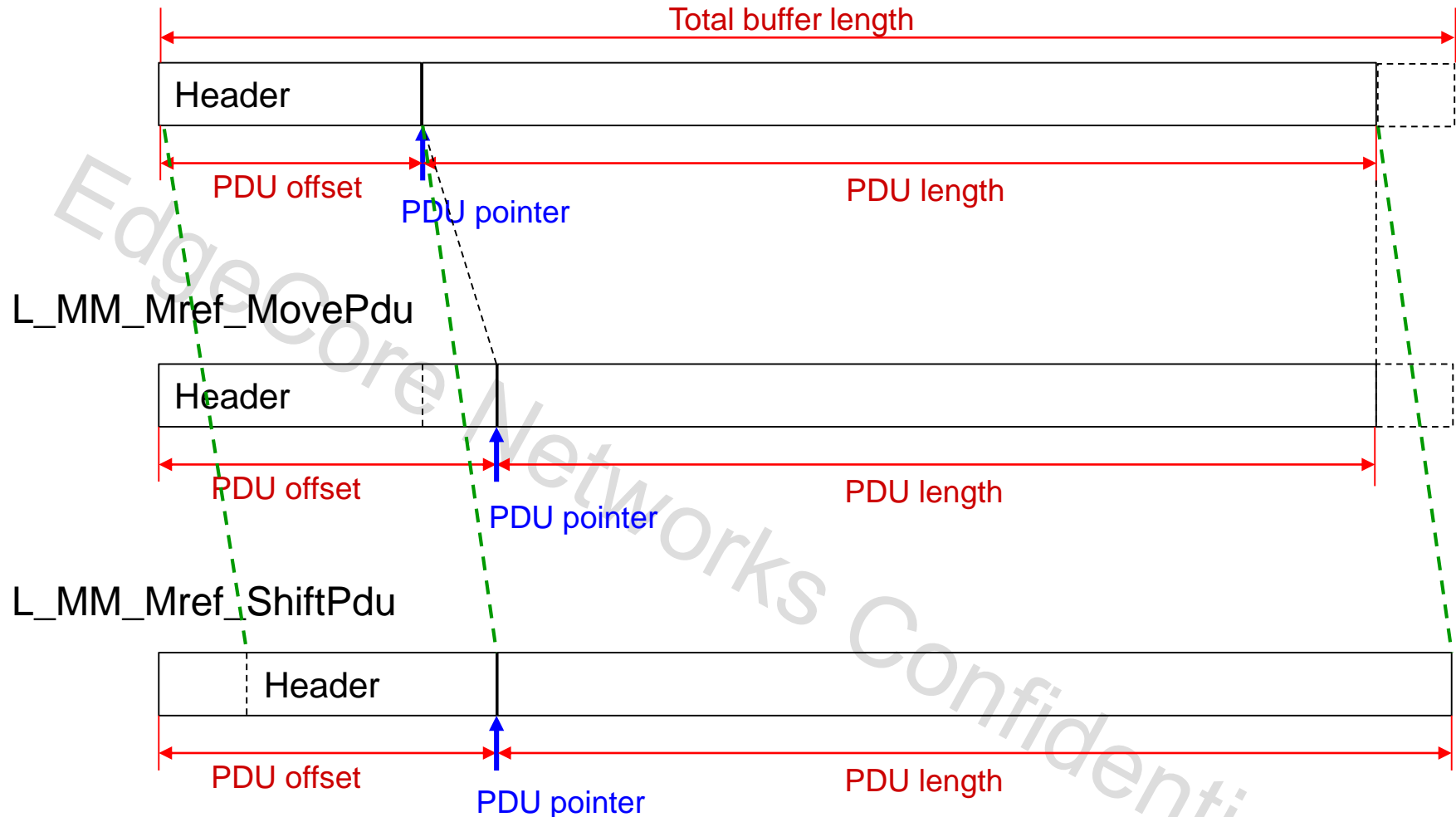
# Memory Management Modules

Accton

- **Memory Reference also provide to get PDU for read or modify packet payloads.**
  - **L\_MM\_Mref\_MovePdu** – Move PDU pointer forward or backward and automatically adjust the PDU length.
  - **L\_MM\_Mref\_ShiftPdu** – Shift the PDU block toward the end of the buffer. (L\_MM\_AllocateTxBuffer will allocate a buffer which the size is more than request. L\_MM\_Mref\_ShiftPdu shall check whether the buffer behind PDU is large enough to do PDU shifting)
  - **L\_MM\_Mref\_SetPduLen** – Set the length of PDU.
  - **L\_MM\_Mref\_GetPdu** – get PDU pointer address and PDU length.

# Memory Management Modules

Accton



# Memory Management Modules

Accton

## ■ Ex: prepare TX packets

- Allocate TX buff

```
mref_handle_p = L_MM_AllocateTxBuffer(data_size, user_id);
```

- write data to buffer from PDU pointer

```
packet_p = L_MM_Mref_GetPdu(mref_handle_p, &pdu_len);
```

- Update PDU length

```
L_MM_Mref_SetPduLen(mref_handle_p, real_data_size);
```

# Data Structure Modules

Accton

- **The modules belongs to this type provides various data structures which are frequently used in programming tasks.** (src/cmnlib/common/datastru)
  - Circular List (l\_cirlst.c /.h)
  - Indexed-Double Linked List (l\_dlist.c /.h)
  - Hash Table (l\_hash.c /.h)
  - Hisam Table (l\_hisam.c /.h)
  - Isam Table (l\_isam.c /.h)
  - Linked List (l\_linklist.c /.h, l\_link\_lst.c /.h)
  - Multiple Priority Queue (l\_mq.c /.h)
  - Message Queue (l\_msg.c /.h)
  - Sorted List (l\_sort\_lst.c /.h)
  - Stack (l\_stack.c /.h)

# Conversion Modules

Accton

- **The operations which convert original data into another form of data will be put in conversion modules.** (src/cmnlb/common/convert)
  - Base64 Encode/Decode (l\_base64.c /.h)
  - MD5 Encode/Decode (l\_md5.c /.h)
  - Need convert by calculation, ex: checksum, CRC (l\_math.c /.h)
  - Convert between string and IPv4 address (l\_inet.c/ .h)

# SYSFUN

Accton

- **Sysfun is a special type in Cmnlib. It defines a set of APIs to provide system functionalities.**

(src/cmnlib/user/kernel/linux)

- System call – Don't use system call provided by OS directly
- Semaphore – SYSFUN\_CreateSem / SYSFUN\_GetSem ...
- Thread – SYSFUN\_SpawnThread
- Periodic timer – SYSFUN\_PeriodicTimer\_Create
- IPC – SYSFUN\_ReceiveMsg / SYSFUN\_SendRequestMsg ...
- ...

# Q&A

Accton

---

EdgeCore Networks Confidential



Thank you!