Reference Platform - Cavium 38XX

- Multicore system-on-chip
  - 600MHz MIPS64 version 2, integer arithmetic
  - 16 cores, 1 MB L2 Cache
  - Extended instruction set
  - 230 Gbps memory bus interface
  - Shared physical memory
    - Shared or distinct per core with TLBs
- In-core coprocessors
  - 3DES/AES
  - MD5/SHA/CRC
- SoC processors
  - Packet Input Processing and Distribution
  - Packet Output Processing
  - Compression/Decompression (DEFLATE)
  - DFA engines with high speed RLDRAM access (up to 1 Gb)
  - RNGs
  - Packet Order Work (POW) processing
  - Free Pool Allocator (FPA)
  - Fetch-and-Add Unit (FAU)
  - Hardware Timers

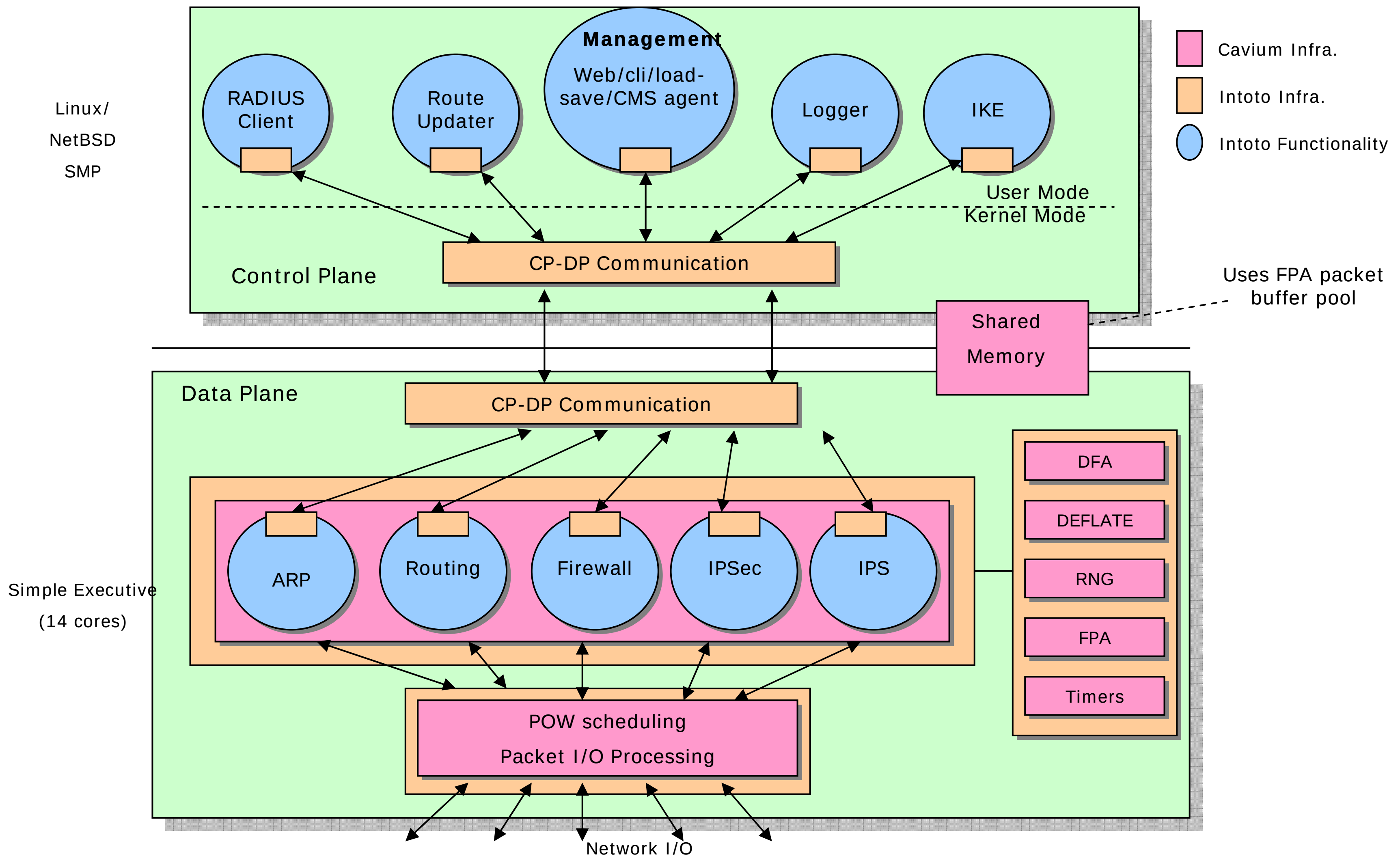
Partitioning Cores

- Cores may be grouped
  - Each group may run a separate OS
    - Linux SMP
    - NetBSD SMP
    - Simple Executive
- iGateway Control Plane
  - NetBSD/Linux
  - 1 or 2 cores (SMP)
  - 64-bit OS
- iGateway Data Plane
  - Simple Executive
  - 64-bit processing

Control Plane / Data Plane
Separation and Features

- Control Plane (CP)
  - Linux/NetBSD SMP
  - Configuration & Management (Web/html, CLI, save/restore)
  - Log client
  - Route updater
  - Central Management System (CMS) agent
  - IKEv1,v2 / PKI
  - OCSP, SCEP
  - RADIUS client, LDAP/SSL client.
- Data Plane (DP)
  - All primary functionality
    - Firewall (including connection establishment)
    - IPSec VPN (Policy-based and Route-based)
    - IPS, P2P, IM with traffic enforcement
    - Virtual Security Gateways (VSG)
    - Virtual routing (table per VSG)
    - ARP, VLAN interface support
  - Symmetric Crypto, Hash, Checksum, DFA pattern match, Zip, RNG

CP / DP Very High Level Schematic



Octeon Capability – Packet processing

- Packet Processing Model (Data Plane)
  - Packet input processor distributes incoming packets
    - Reads network packets from all input network interfaces
    - Types and Tags them (e.g. 5-tuple hash)
    - Inserts them as Work Queue Entries (WQE) into Packet Order Work (POW) unit (prioritized work queues)
    - Can be programmed per port (up to 36 ports)
      - Programmable buffer size, with padding at top and bottom
      - Chaining of buffers for large packet sizes
      - Skip areas and scheduling-control areas for input packet format
      - CRC and IP Checksum verification
      - Tag generation control
        - IP source, destination, protocol type, next_header value,
        - TCP/UDP source, destination ports, VLAN ID
        - Upto 128 bytes in packet (user-defined range)
      - QoS control
  - Work queues are shared across cores
  - All cores can run the same code
  - Cores demand work belonging to specific groups.
  - POW unit schedules work to cores (multiple scheduling schemes)
  - Cores process work and send results
    - Back to work queues
    - Or to packet output processor queues

Octeon capability – QoS

- Inbound QoS Control (which queue to schedule to)
  - Explicit selection of queue
  - Take from VLAN priority field, map to queue
  - Take from IP DiffServ field, map to queue
  - Specify in extended packet headers
  - QoS ‘watchers’ that can inspect protocol types and decide queue
  - All the above may be configured per port (upto 36 ports)
  - A RED variant algorithm supported for dropping packets
- Outbound QoS
  - 128 priority queues for output processing unit
  - Static or weighted priority assignments

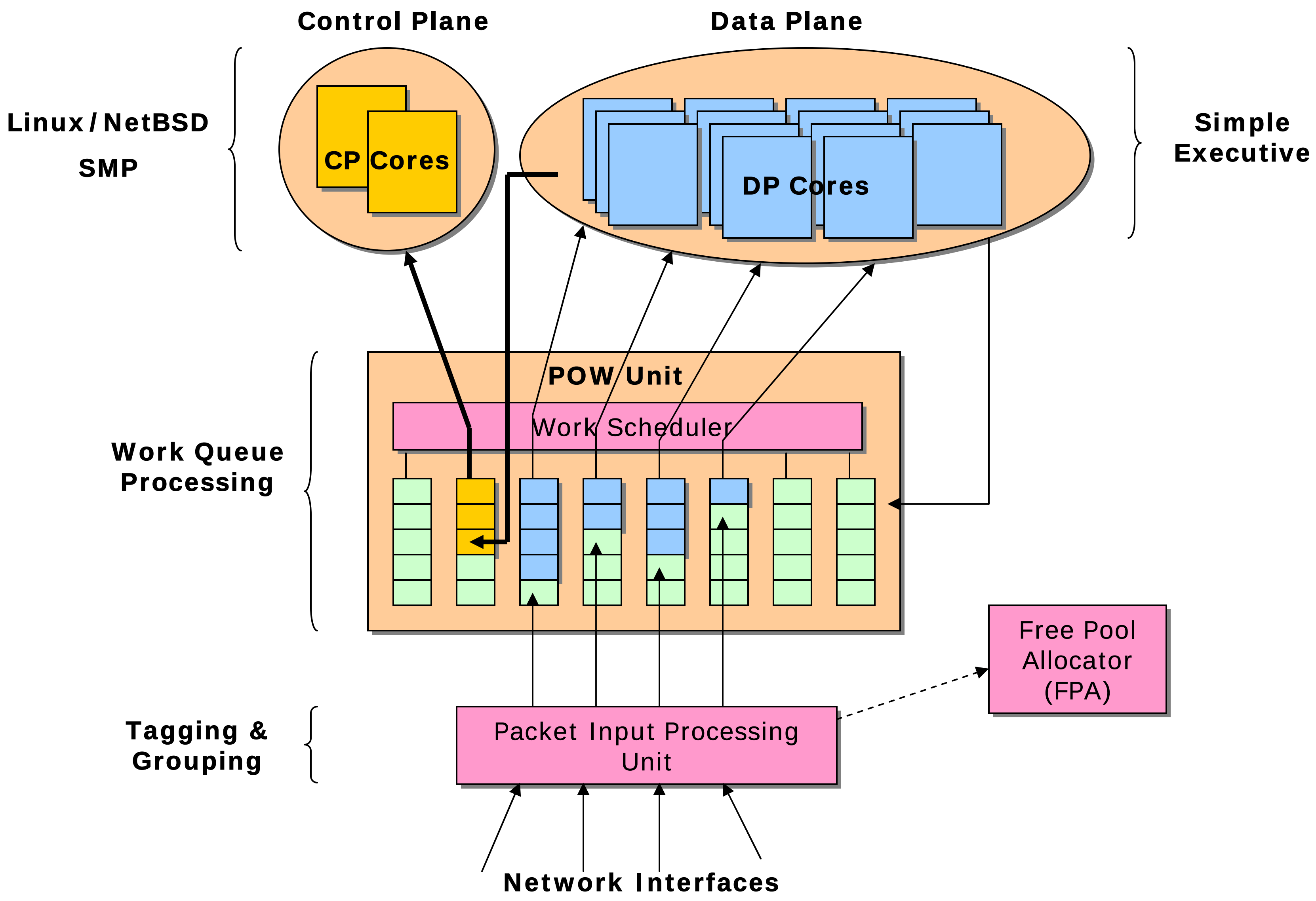
Octeon Capability - POW Work Queues

- POW Work Queues
  - Packet Order Work (POW) hardware unit in SoC
    - Maintains 8 prioritized queues
    - Accepts work queue entries (with tag and destination group id)
    - Schedules work to cores that have workgroup id membership specified in work
    - Supports ordering and atomicity in work scheduling for same tag value
  - Cores
    - Ask for work
    - Receives and processes one piece of work at a time
      - May complete and send result to packet output processor
      - Queue result back into POW for next stage processing
      - Mark work as pending (instead of blocking), to be rescheduled by POW later

Octeon Capability - Work Queue Entry (WQE

- Contents of a WQE
  - Hardware generated checksum
  - Total number of bytes in packet (16 bits)
  - Input physical port (6 bits)
  - Priority of the packet at the time of input (3 bits)
  - Work group id to schedule to (4 bits)
  - Tag Type (ORDERED, ATOMIC, NULL) (3 bits)
  - Tag value (32-bit)
  - Pointer to first segment of the packet
  - 96 bytes of packet data inside the entry
  - Book-keeping pointers and status

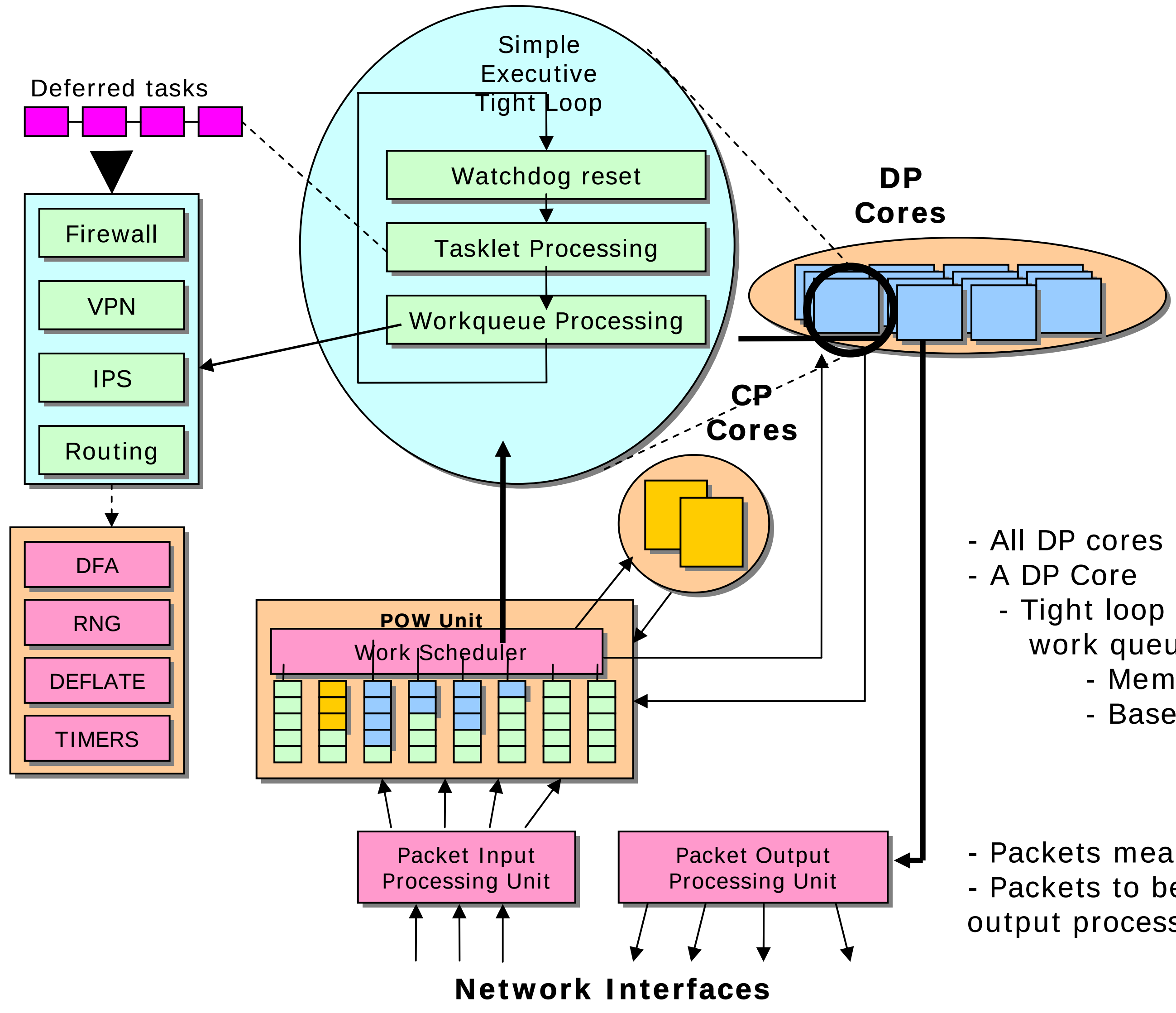
Intoto Usage - Packet Input Processing



Intoto Usage - Packet Input Processing

- Packet Input Processing
  - CP and DP cores have different workgroup ids
  - Input packets for network are tagged and placed into POW unit with workgroup id of the DP
    - 3-tuple hash is used
  - DP cores process packets and apply security services
    - IP reassembly done when needed
      - Uses iGateway memory pool management for reassembly
  - Packets meant for self are sent by DP cores to CP workgroup id
    - Separate priority queue in POW
    - Avoids Data plane congestion from choking control plane
  - POW scheduling is ORDERED for a given tag value
    - Ensures that a 3-tuple stream is processed in sequence

Intoto Usage - Packet Input Processing (DP

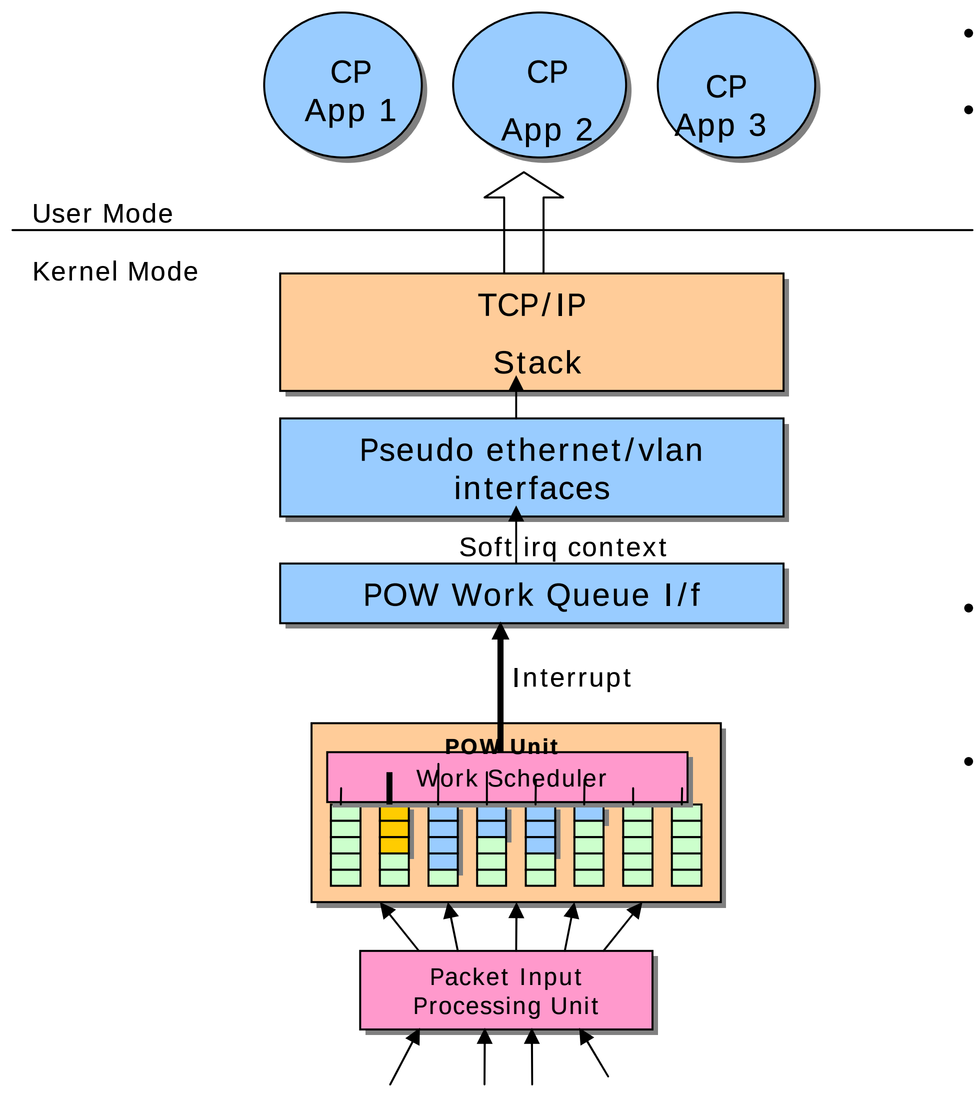


Workgroups

1. DP input packet processing
2. DP to CP packet processing
3. CP to DP packet processing
4. Timer messages group
5. DFA match messages group
6. DP to CP communication
7. CP to DP communication

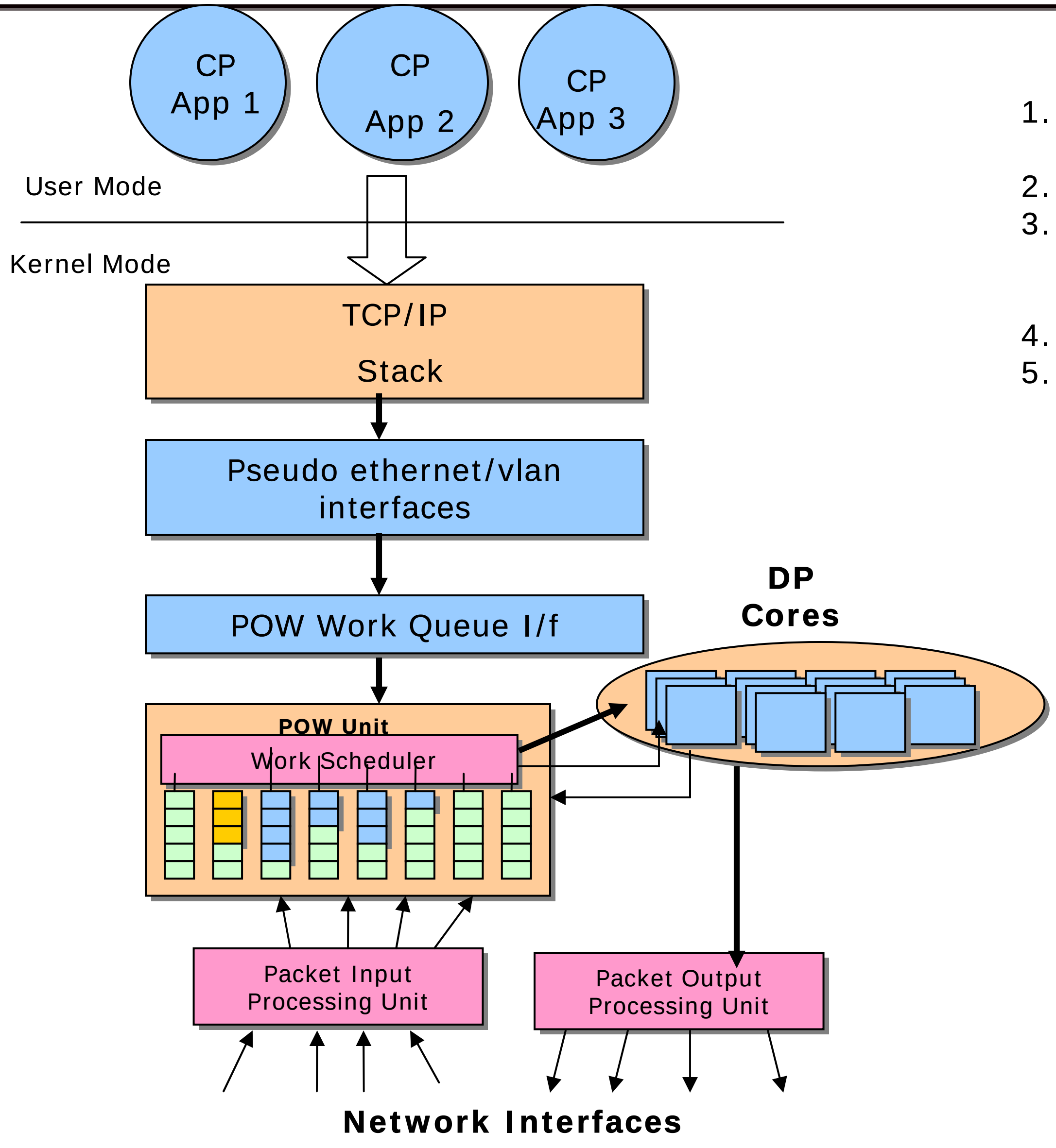
- All DP cores run the same code
- A DP Core
  - Tight loop that continuously processes work queue entries
    - Membership in multiple work groups
    - Based on workgroup
      - Runs firewall/vpn/ips etc.
      - Runs timer processing
      - Runs DFA results
- Packets meant for CP are sent to CP
- Packets to be sent out are sent to packet output processing unit

Packet Input Processing (CP)



- CP asks for work on assigned workgroup ids
- Virtualizes physical interfaces on CP
  - As ethernet devices
  - Packets arrive from DP through POW and interrupt the cores
  - Ethernet driver processes and delivers packets to stack
    - VLAN interfaces supported
  - Egress packets sent to DP through POW
  - Packet allocation/free use FPA (Free Pool Allocator)
  - CP Apps view packets as if arriving on local interfaces
- User Mode
  - Libraries are provided for CP-DP communication
    - Data transfer & events
- IKE
  - IKE processing done in CP

Packet Output Processing (CP)

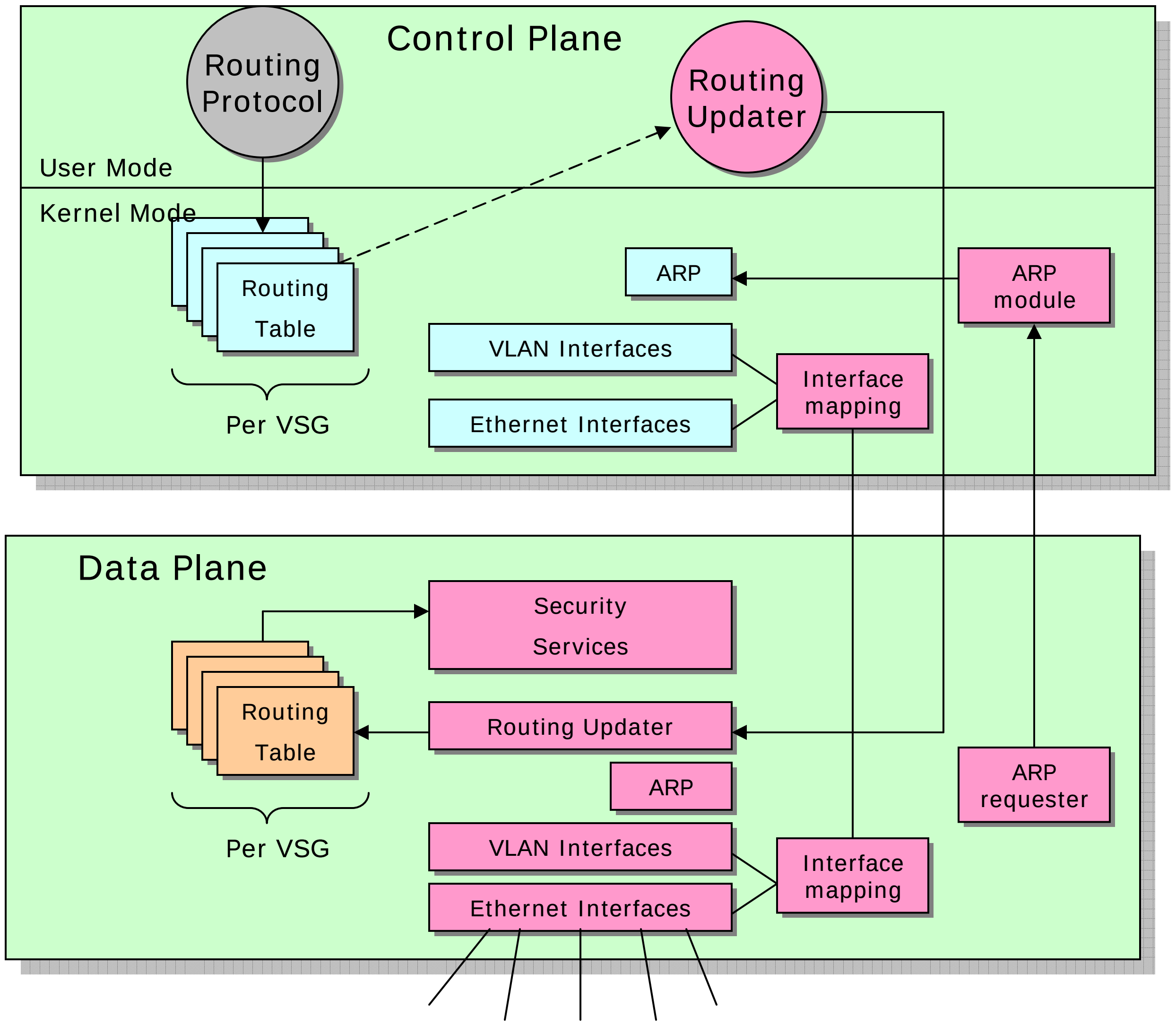


1. Control Plane applications use the local TCP/IP stack
2. Packets hit pseudo Ethernet drivers
3. Sends packets to any DP core using POW work queue entries. All DP cores are members of this workgrp
4. DP Core(s) processes packet
5. If packet needs security processing it is completed
6. If packet needs to be sent out, it is sent to the output processing hardware unit
7. Sends to physical interfaces (PCI/X, SPI, RGMII), up to 36 ports

Intoto Routing & Interfaces Infrastructure

- Data Plane & Control Plane
  - Work together, has counterparts
- Supports multiple interfaces with VLAN support
  - VLAN Tags used to map VLAN interfaces
- Routing tables per VSG (Virtual Security Gateway) is supported
  - Routing protocols run in CP
  - Replicated on DP
- Routing updater
  - Listens to route table changes in CP and publishes to DP
- Interfaces extended to Control Plane
  - All interfaces (including VLAN i/f) appear local to CP
- Interface id mapping between OSes
  - Required for consistency between CP & DP interface management
- ARP cache management
  - DP on cache miss, queues packet, requests CP for ARP update
  - ARP response packets are processed in DP

Interfaces & Routing Support



🟦 OS Feature

Octeon Capability - DFA

- SoC DFA processor
  - Shared by cores
  - Multiple hardware threads (16)
  - Graphs in RLDRAM
  - Submit DFA instructions, ring door bell
  - Gather support
  - Can send results through POW
    - List of all matches
  - Can maintain state across requests
- Cavium SDK
  - Software DFA library (ccsm)
- Usage
  - DFA compiler
  - Load graph into RLDRAM

Intoto Usage - DFA

- RLDRAM partitioned for IPS, P2P & IM
- Hardware DFA graphs
  - Compiled and uploaded (CP->DP)
  - Loaded into RLDRAM
  - DFA matching uses POW messages
    - Separate work-group id
    - Separate FPA pool
- Compile-time fallouts from h/w are sent to Octeon ccsm (software)
  - Cavium Content Search Macros
    - DFA engine for regular expression search
      - Intelligent nodes, reduce graph explosion
- Fallouts from Octeon ccsm are resolved using PCRE
  - Perl-Compatible Regular expression library

Intoto usage of FAU (Fetch-and-Add Unit)

- Fetch-and-Add Unit
  - Maintains a 2KB Register file
  - Useful for sharing atomic counters/pointers and status between cores
    - Supports atomic update, fetch-and-add
    - Useful for general synchronization purposes between cores
    - Accessible from cores and packet output processor
  - Supports 8 bit, 16-bit, 32-bit and 64-bit operations
- Intoto Usage
  - CP-DP communication methods (covered later)
    - Queue pointers
    - Counters
    - Status

Intoto Usage of Zip co-processor

- Zip co-processor
  - Supports DEFLATE – compression and decompression
  - Cores place instruction in memory queue, ring door bell
  - Engines processes one at a time
  - Scatter/gather support
  - History support
  - Max input chunk size 16MB
    - Context can be saved/restored
    - Software provides “history bytes” for continuation
  - Can send completion intimation as POW work queue entry
- iGateway
  - Used in IPSec VPN (IPComp support)
  - Uses work queue entries to report completion
    - Separate work group id
    - Separate FPA pool

Packet allocation and buffer management

- Packet allocation by hardware
  - FPA (h/w free pool allocator) used by input packet processing unit
    - Allocates packet buffer
    - Reads in packet (DMA)
    - Creates work queue entry (WQE) with tag
      - WQE refers to packet buffers (chaining is of buffers possible for large packets)
    - Submits work to POW unit
- Packet buffers
  - iGateway wraps packet buffers internally
    - Appends a header structure
      - Used for maintaining state through packet flow
      - Helps portability of code
  - Hardware allocation adjusted to provide space (head room) for igateway header
    - Mbuf ‘skip’ in Octeon SDK terms
  - Programs packet input processor
    - Support additional header space (before L2) in incoming packets
      - ‘skip 2’ in Octeon terms
    - Support Tag values, tag types, priority queue id and workgroup id in incoming packets
      - Used to create POW work queue entry for packet
    - Useful when integrating with an external packet source (e.g. line card in a chassis) that wants to control some aspects of packet processing
  - Packet sent out by offsetting iGateway header

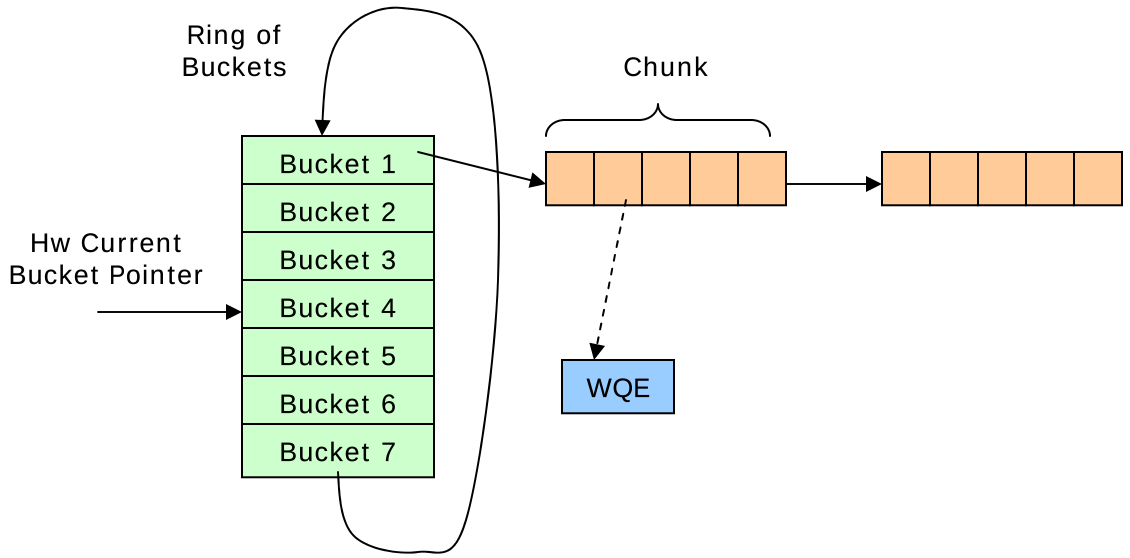
Packet allocation and buffer management

- Freeing packet buffers
  - Packets to control plane
    - FPA allocations are freed by recipient
  - Egressing packets
    - FPA allocations are freed by packet output processing unit
  - Locally generated packets
    - Also use FPA allocations

Octeon capability
Hardware Timers

- Hardware Timer management
  - Multiple rings (up to 16)
  - Each ring has multiple buckets
  - Buckets carry work queue entries
  - Buckets in a ring are traversed by hardware
    - Hw steps from one bucket to next at a specified fixed rate (milliseconds or seconds)
    - All work queue entries found in bucket are submitted by hardware
  - Buckets are chunked to save space
  - Hardware frees chunks to FPA pool

Hardware Timers



Hardware Timers – Intoto Usage

- All timer management uses HW timers
- Bypasses Cavium SDK timer APIs
  - Custom implementation directly with hardware
  - More efficient (preallocation of timer nodes)
- One ring (milliseconds)
- Handles larger durations without needing very long ring
  - Recognizes multiple iterations of the hardware ring pointer to support larger durations
- Separate FPA pool for chunks (freed by hw)
- Timer nodes memory allocation done by iGateway routines (pre-allocated)

Memory (DRAM) Usage

- System memory 2GB (assumed)
  - Boot loader (u-boot) 1MB
  - CP cores – 512MB
  - DP cores – 1.5 GB
  - Code Segment (DP) – 6 to 8 MB
  - Data Segment (DP) – shared – 192MB
  - Heap (DP)
    - Cavium SDK heap is not used
    - iGateway arena-based allocation – 512MB
    - FPA allocations –
      - Packet buffers (2k/buffer)
      - WQE
      - Timers
      - DFA (2) – (2k/buffer – size?)
      - Zlib
      - CP-DP communication
  - Stack (DP) – 1 MB/core
  - RLDRAM – 256 MB
    - P2P 20MB, IPS 40MB

Memory Model

- Cavium 38xxx SDK defaults to independent physical memory per core
  - Data segment gets a separate instance copy for each core
  - TLBs are used to provide same virtual memory address for a global variable on each core
- iGateway DP cores share memory
  - Uses CVMX_SHARED_SEGMENT (shared by cores)
  - All global variables in code are marked compile-time to be placed here
  - Third party libraries – ELF files are modified by an iGateway tool to move globals into the same shared segment

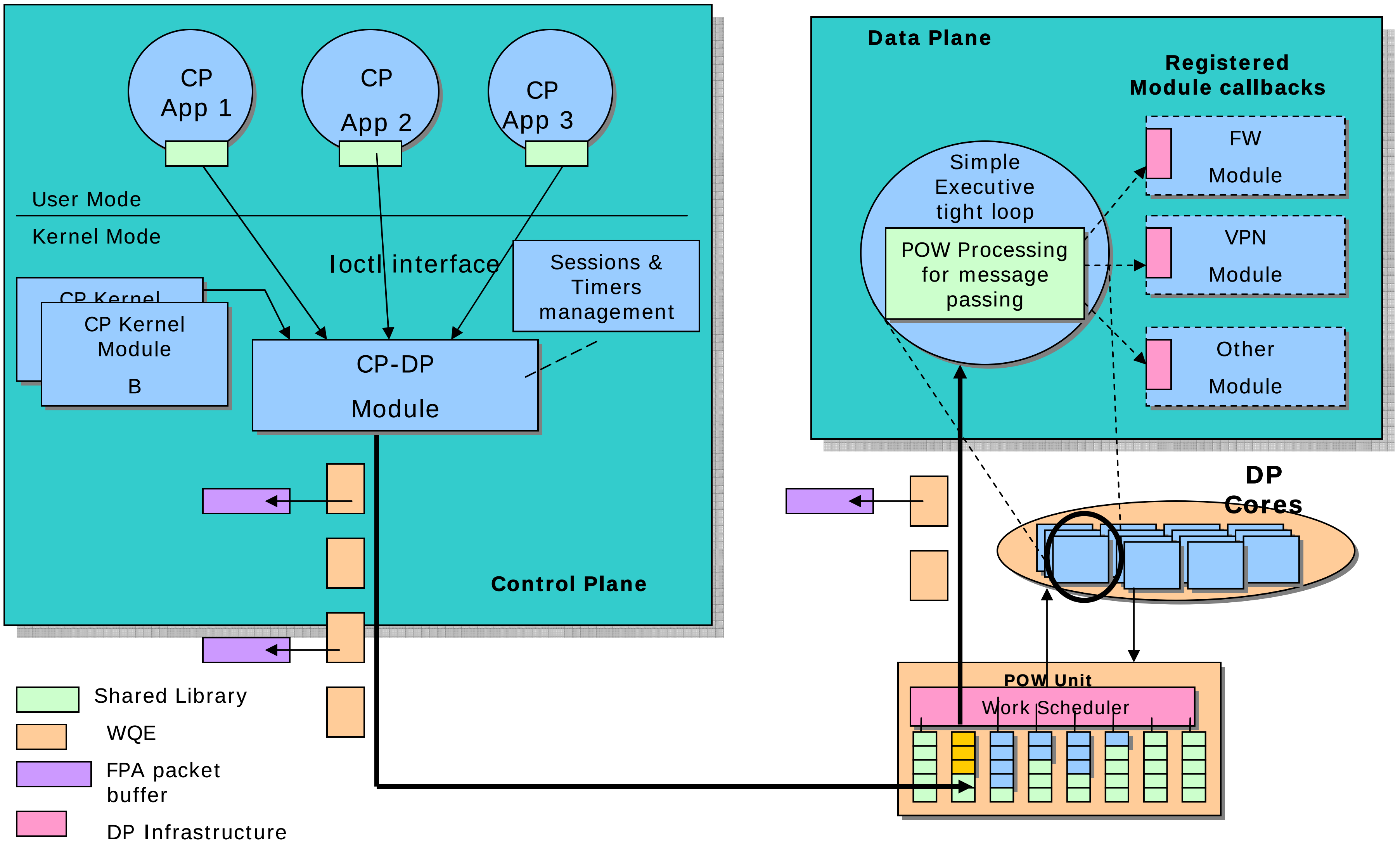
CP-DP Communication

- Intoto developed infrastructure
  - Packet passing infrastructure
  - Message passing infrastructure
  - Ring buffer infrastructure
  - Mailboxes
- Availability
  - Shared library in CP user space
  - Loadable kernel module in CP kernel space
  - Part of simple executive executable in DP space
- Capabilities
  - Synchronous API from CP
    - Needed for commands
  - Asynchronous API from CP with response.
    - Currently not used
  - Asynchronous API from CP with no response
    - Packet transfer
    - ARP resolutions.
    - Route updations.
  - Asynchronous API from DP
    - To send interface events etc..
  - Ring buffer based communication
    - Used to transfer DFA Graph.
    - CLI output.
  - Mail box based communication with FAU.
    - To send log messages from DP to CP.

Message passing infrastructure

- Salient Features
  - Point to point communication between modules
  - Maps to POW work queue entries
    - Upto 84 bytes storage inside work queue entry (per message)
    - Upto 2048 bytes storage for packet buffer (per message)
  - Sender and recipient identified by:
    - <module-id, submodule-id>
  - Sample Usage:
    - Send CLI commands and result code between CP and DP
    - Receive IPSec SAs from IKE; IpSec to IKE messages for connection initiation
- Uses work queue entries that may carry references to buffers
  - Packet buffers allocated by Free Pool Allocator (FPA)
  - Allocated from the same pool as used by packet input processing unit
  - Packet pointers sent between CP-DP are physical addresses
    - Translated to/from virtual address by infrastructure
  - POW Workgroup 11 used for CP to DP communication
  - POW Workgroup 12 used for DP to CP
    - Registered interrupt routine for group id 12 in LKM invokes message handler using softirq

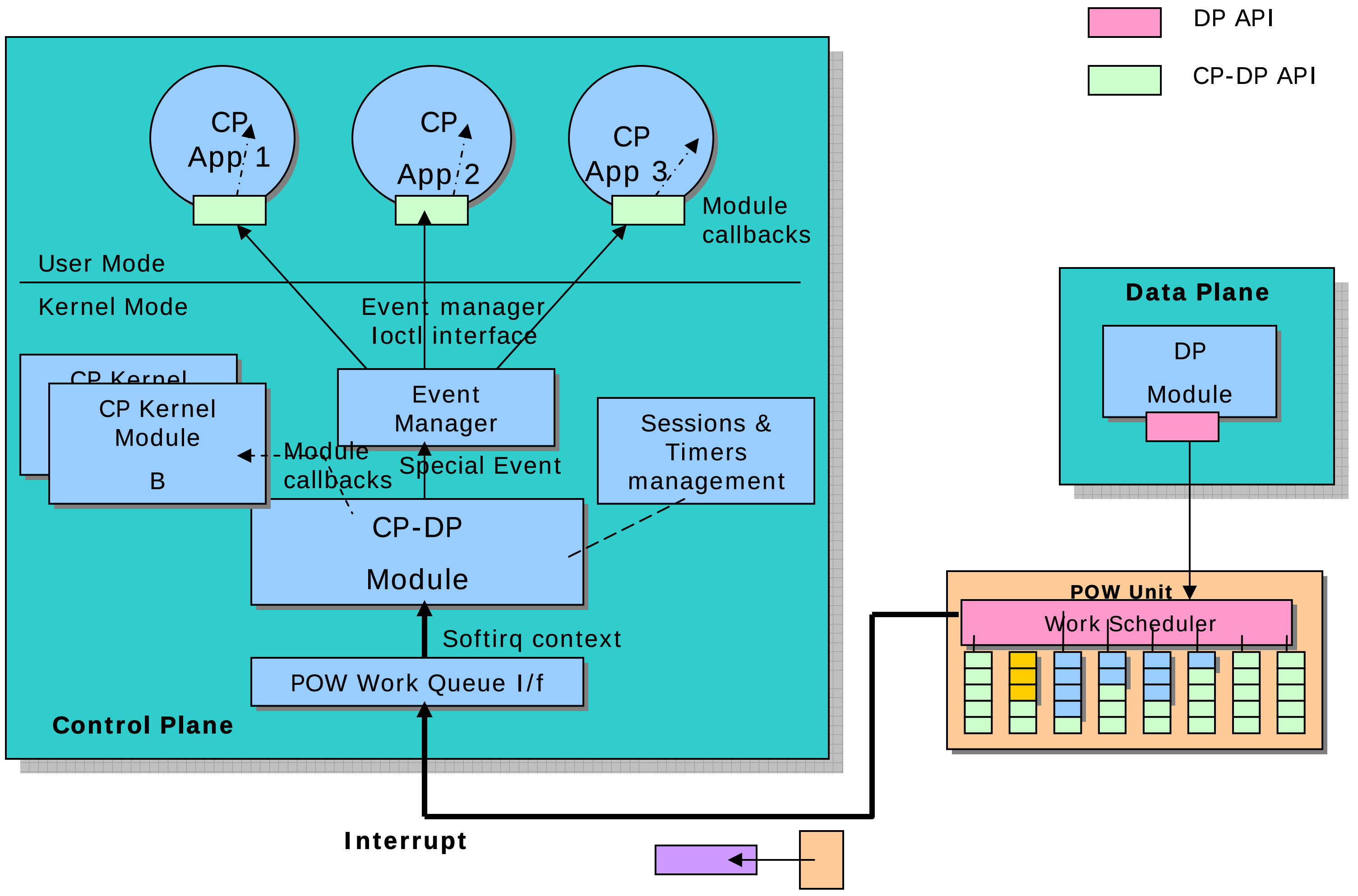
Message passing CP -> DP (Origination or Response, Asynchronous)



Message passing infrastructure

- Asynchronous operation
  - A session-id is created (if sender requests a response)
  - Recipient may respond at a later time using the session id
  - Sender may specify a time out for responses
    - No response, structures/state are freed
  - Modules get call backs for indications, responses and aborted transfers
    - Message pointer
    - Message length
    - Response requested (part of flags)
    - Session-id for response
- Synchronous operation (CP to DP only)
  - Caller waits till timeout period for response
  - DP operations are asynchronous
- No message copying from CP kernel to DP or vice versa

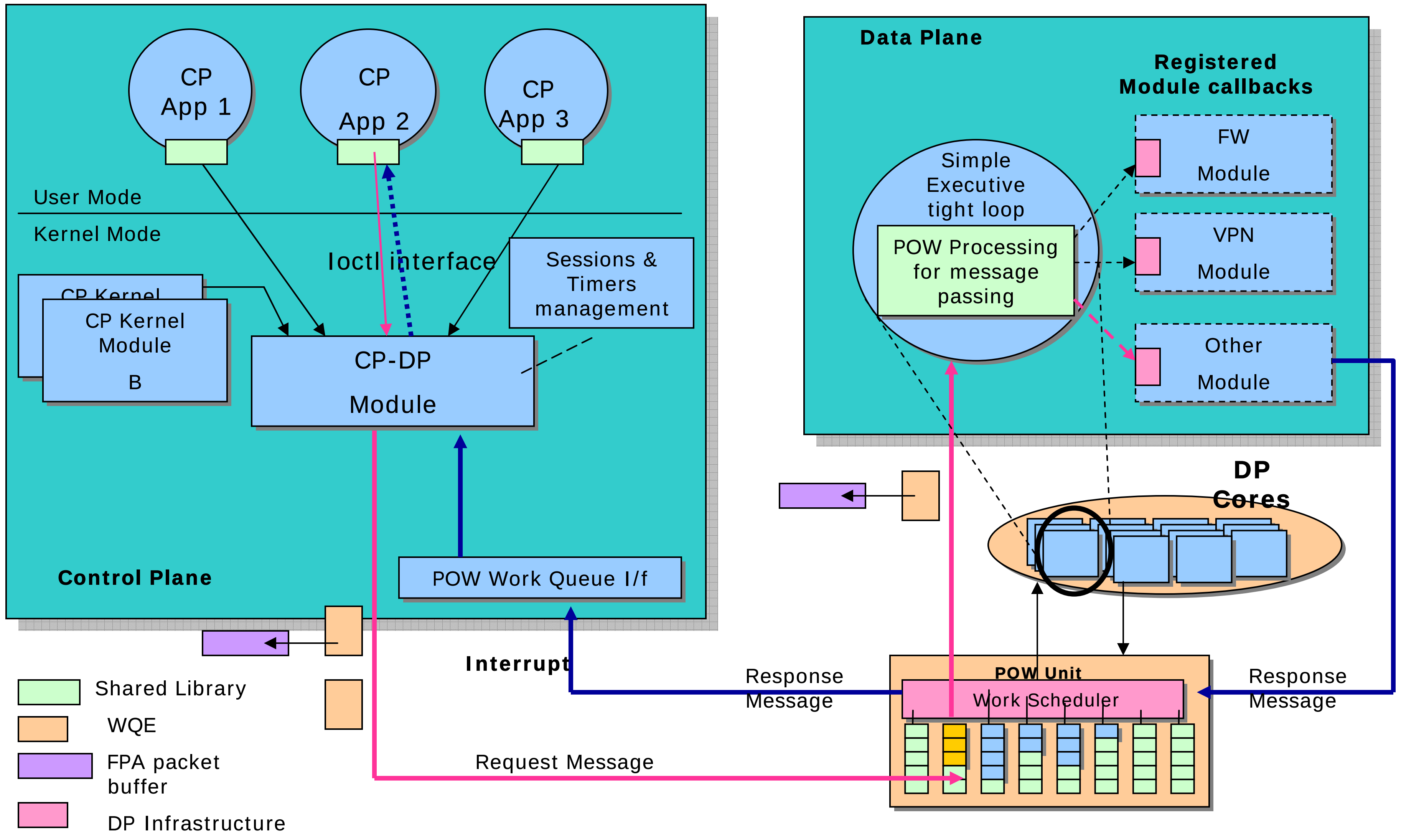
Message Passing DP->CP (Origination or Response, Asynchronous)



Message Passing DP->CP

- DP Cores call API to send message
  - Specifies module-id and submodule-id
  - If a response, session-id is provided
  - Otherwise indicates if a response is solicited
- Message gets sent as WQE to POW
  - DP->CP communication workgroup id is used
  - DP core(s) ask for work on this workgroup id as well
- Work arrival interrupts CP core
  - Interrupt routine raises softirq to process the message
  - Message handled by CP-DP communication module on CP
  - Determines the message is a command/response
  - Passed on to event handler for user-mode
    - Event handler transfers message to application callback routines
  - Callbacks within kernel modules are direct
  - Message is freed after the call
  - No buffer copy from DP to CP

Message passing CP -> DP (Synchronous)



Ring Buffer Infrastructure

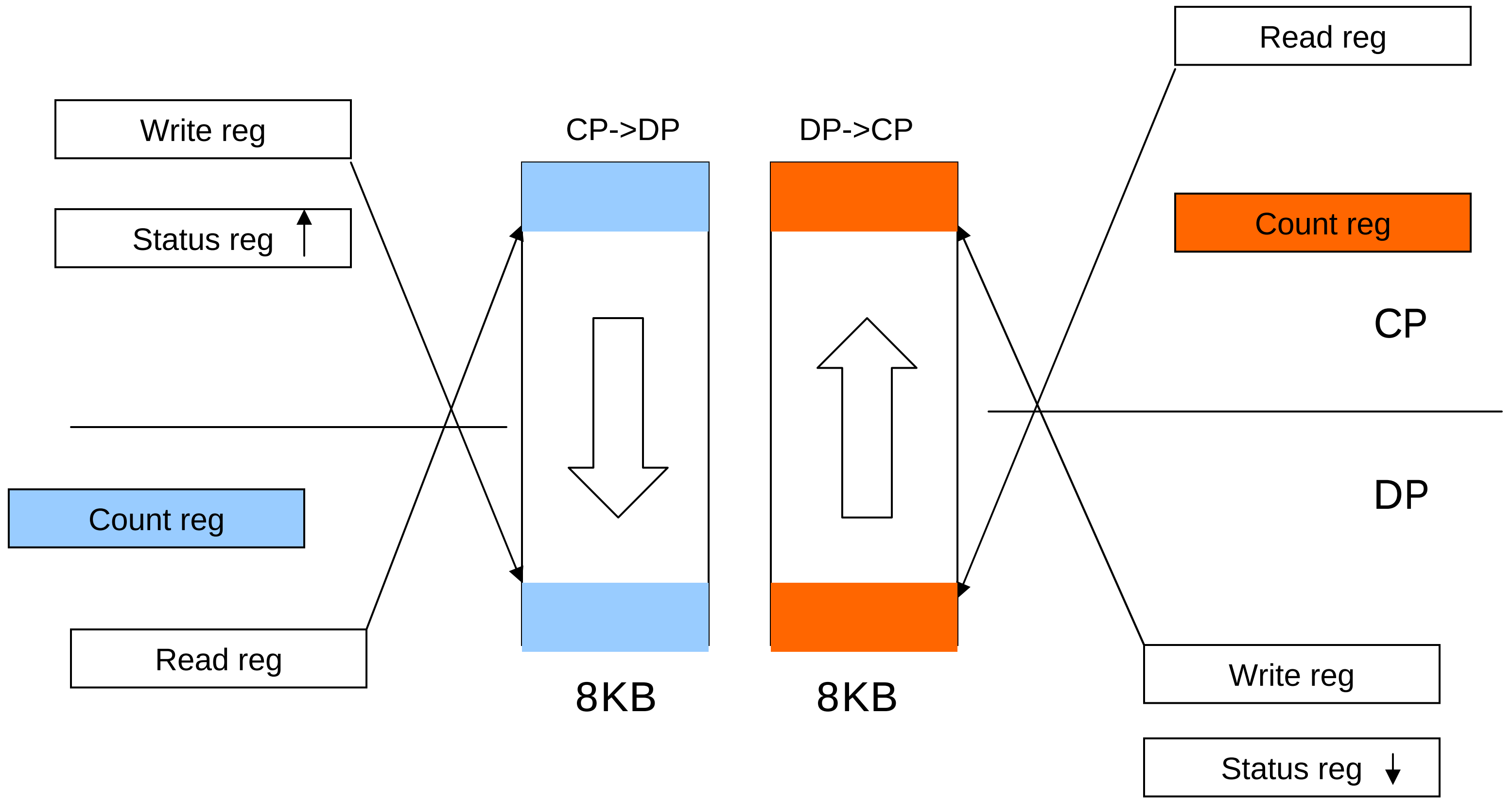
- Producer-Consumer Model
  - One producer, one consumer per ring buffer
  - CP->DP and DP->CP communication supported
- Used to transfer larger amounts of data between CP and DP
  - Message passing method uses the free pool allocator (limited to 2K or to fixed size allocations)
- Used by CP CLI
  - Output of CLI commands could be > 2KB
- Used by Configuration Load/Save engine
- Used for loading DFA graph

Ring Buffer Infrastructure

- Ring Buffers
  - Allocated from bootloader reserved memory
  - Multiple ring buffers may be created
  - Allocated by CP during OS startup
    - 16 kb per queue (DP->CP 8K, CP->DP 8K)
    - Physical address passed to DP via message-passing
  - Head/Tail pointers management
    - Uses FAU registers for:
      - CP Read pointer, DP write pointer, DP count, Status (DP->CP)
      - DP Read pointer, CP write pointer, CP count, Status (CP->DP)

Ring buffer infrastructure – Ring buffer depiction

- A Ring buffer (bi-directional)



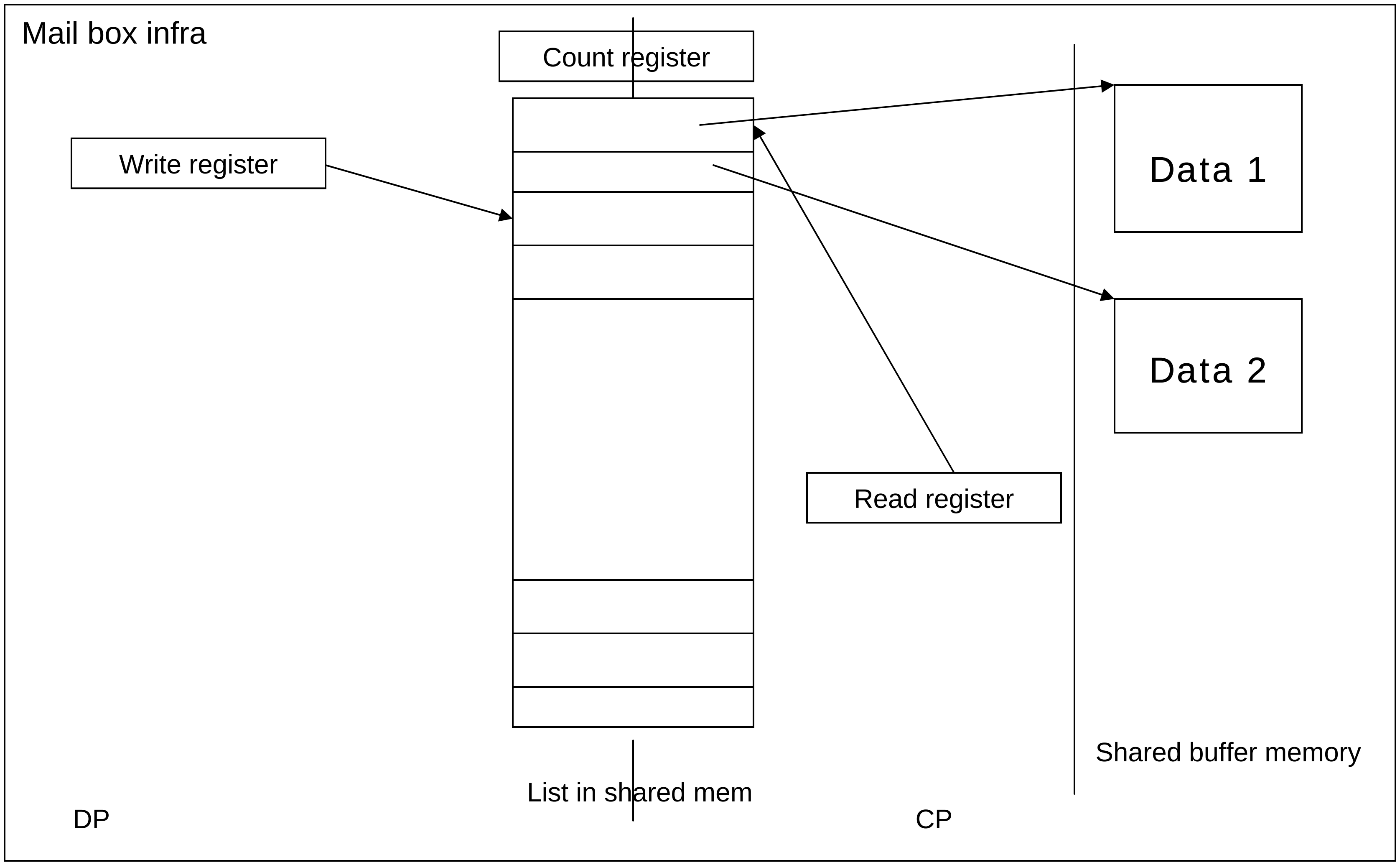
Ring Buffer APIs

- CP API
  - User mode shared library (uses ioctls)
  - Kernel mode API available
  - Creation of a ring buffer done in user mode
    - Returns a handle, used for further calls
  - Handle is passed to DP application (via message-passing mechanism)
- DP API
  - Part of simple executive executable
  - Uses handle received from CP for read/write

Mailbox Infrastructure

- Used for CP to DP communication
  - Passes shared memory pointers
  - Arbitrary size data can be referred
- Mechanism
  - Mailbox: A list (queue) of pointers kept in shared memory
    - Pointers refer to memory allocated from FPA by DP
    - FAU registers for queue management (read/write pointers and count)
  - DP API enqueues an entry
    - Entry contents point to some shared memory area
  - Interrupts the CP core (called a mailbox interrupt)
    - When queue count reaches a threshold
  - CP interrupt handler
    - Wakes up the application
    - Application requests a read (ioctl from user space)
    - Contents of the DP allocated buffer are copied into user-space buffer
    - DP allocated buffer is freed

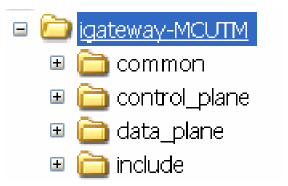
Mail box Infrastructure



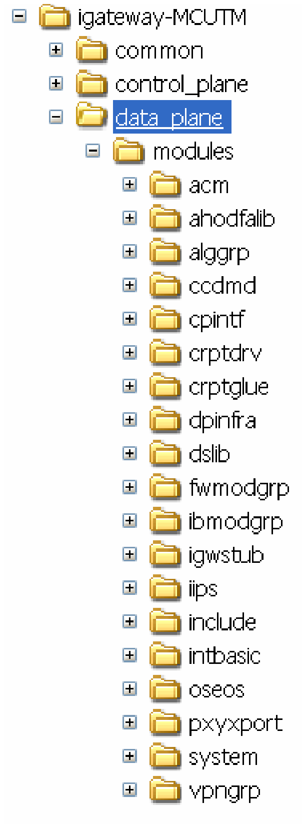
Mailbox Usage

- Message Logging
  - Log messages are sent by DP to CP
    - Logging APIs send logs to mailbox
      - Read by logging daemon on CP

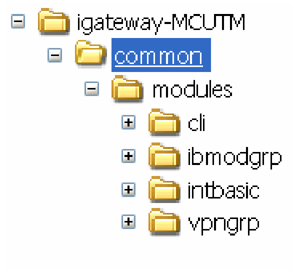
Source Tree Layout



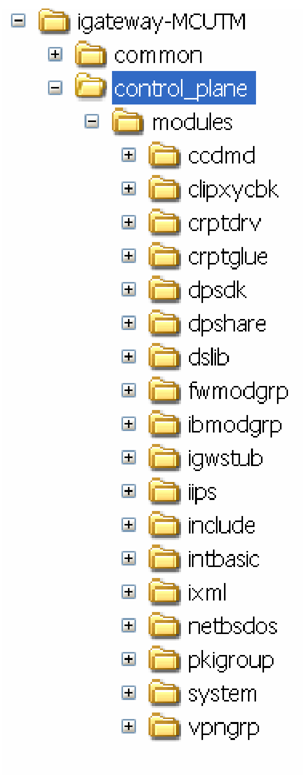
```
⊟ igateway-MCUTM
    ⊞ common
    ⊞ control_plane
    ⊞ data_plane
    ⊞ include
```



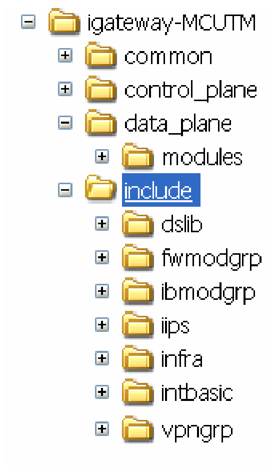
```
⊟ igateway-MCUTM
    ⊞ common
    ⊞ control_plane
    ⊟ data_plane
        ⊟ modules
            ⊞ acm
            ⊞ ahodfalib
            ⊞ alggrp
            ⊞ ccdmd
            ⊞ cpintf
            ⊞ crptdrv
            ⊞ crptglue
            ⊞ dpinfra
            ⊞ dslib
            ⊞ fwmodgrp
            ⊞ ibmodgrp
            ⊞ igwstub
            ⊞ iips
            ⊞ include
            ⊞ intbasic
            ⊞ oseos
            ⊞ pxyxport
            ⊞ system
            ⊞ vpngrp
```



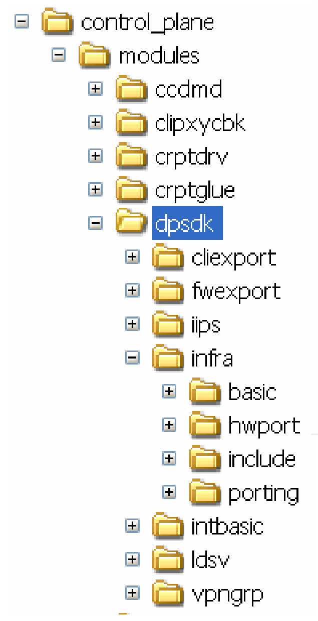
```
⊟ igateway-MCUTM
    ⊟ common
        ⊟ modules
            ⊞ cli
            ⊞ ibmodgrp
            ⊞ intbasic
            ⊞ vpngrp
```



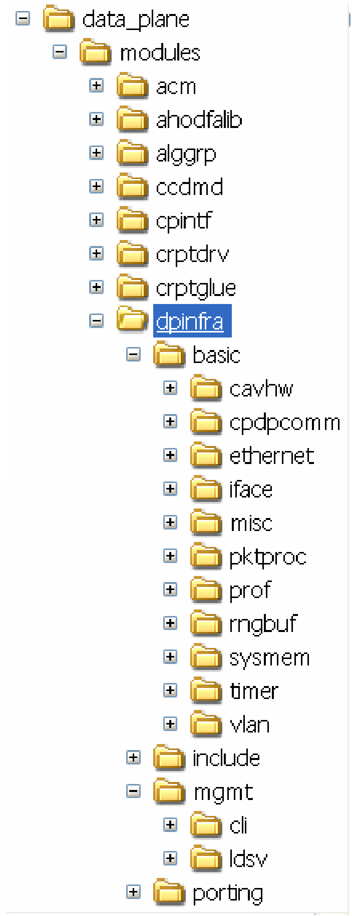
```
⊟ igateway-MCUTM
    ⊞ common
    ⊟ control_plane
        ⊟ modules
            ⊞ ccdmd
            ⊞ clipxycbk
            ⊞ crptdrv
            ⊞ crptglue
            ⊞ dpsdk
            ⊞ dpshare
            ⊞ dslib
            ⊞ fwmodgrp
            ⊞ ibmodgrp
            ⊞ igwstub
            ⊞ iips
            ⊞ include
            ⊞ intbasic
            ⊞ ixml
            ⊞ netbsdos
            ⊞ pkigroup
            ⊞ system
            ⊞ vpngrp
```



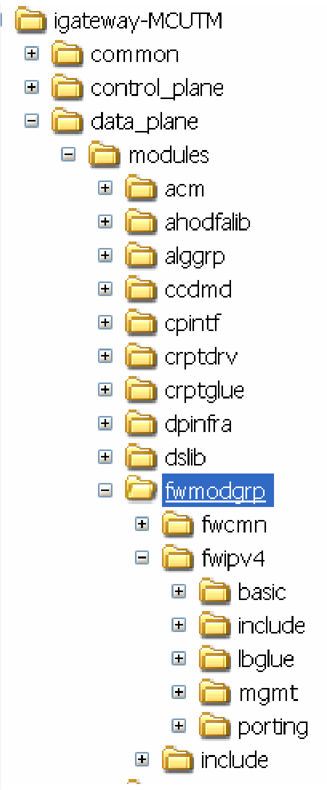
```
⊟ igateway-MCUTM
    ⊞ common
    ⊞ control_plane
    ⊟ data_plane
        ⊞ modules
    ⊟ include
        ⊞ dslib
        ⊞ fwmodgrp
        ⊞ ibmodgrp
        ⊞ iips
        ⊞ infra
        ⊞ intbasic
        ⊞ vpngrp
```



```
⊟ control_plane
    ⊟ modules
        ⊞ ccdmd
        ⊞ clipxycbk
        ⊞ crptdrv
        ⊞ crptglue
        ⊟ dpsdk
            ⊞ cliexport
            ⊞ fwexport
            ⊞ iips
            ⊟ infra
                ⊞ basic
                ⊞ hwport
                ⊞ include
                ⊞ porting
            ⊞ intbasic
            ⊞ ldsv
            ⊞ vpngrp
```



```
⊟ data_plane
    ⊟ modules
        ⊞ acm
        ⊞ ahodfalib
        ⊞ alggrp
        ⊞ ccdmd
        ⊞ cpintf
        ⊞ crptdrv
        ⊞ crptglue
        ⊟ dpinfra
            ⊟ basic
                ⊞ cavhw
                ⊞ cpdpcomm
                ⊞ ethernet
                ⊞ iface
                ⊞ misc
                ⊞ pktproc
                ⊞ prof
                ⊞ rngbuf
                ⊞ sysmem
                ⊞ timer
                ⊞ vlan
            ⊞ include
            ⊟ mgmt
                ⊞ cli
                ⊞ ldsv
            ⊞ porting
```



```
⊟ igateway-MCUTM
    ⊞ common
    ⊞ control_plane
    ⊟ data_plane
        ⊟ modules
            ⊞ acm
            ⊞ ahodfalib
            ⊞ alggrp
            ⊞ ccdmd
            ⊞ cpintf
            ⊞ crptdrv
            ⊞ crptglue
            ⊞ dpinfra
            ⊞ dslib
            ⊟ fwmodgrp
                ⊞ fwcmn
                ⊟ fwipv4
                    ⊞ basic
                    ⊞ include
                    ⊞ lbglue
                    ⊞ mgmt
                    ⊞ porting
                ⊞ include
```

Thank You