I/O request handling

An I/O request of a User VM, which is constructed by the hypervisor, is distributed by the ACRN Hypervisor Service Module to an I/O client corresponding to the address range of the I/O request. Details of I/O request handling are described in the following sections.
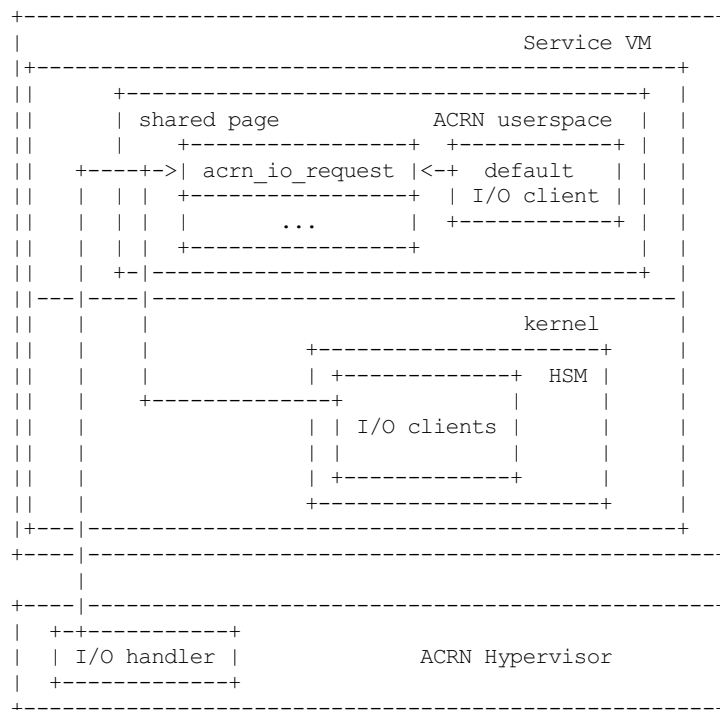
1. I/O request

For each User VM, there is a shared 4-KByte memory region used for I/O requests communication between the hypervisor and Service VM. An I/O request is a 256-byte structure buffer, which is 'struct acrn_io_request', that is filled by an I/O handler of the hypervisor when a trapped I/O access happens in a User VM. ACRN userspace in the Service VM first allocates a 4-KByte page and passes the GPA (Guest Physical Address) of the buffer to the hypervisor. The buffer is used as an array of 16 I/O request slots with each I/O request slot being 256 bytes. This array is indexed by vCPU ID.

2. I/O clients

An I/O client is responsible for handling User VM I/O requests whose accessed GPA falls in a certain range. Multiple I/O clients can be associated with each User VM. There is a special client associated with each User VM, called the default client, that handles all I/O requests that do not fit into the range of any other clients. The ACRN userspace acts as the default client for each User VM.

Below illustration shows the relationship between I/O requests shared buffer, I/O requests and I/O clients.

```
+------------------------------------------------------+
|                                      Service VM       |
|+----------------------------------------------------+ |
||     +-------------------------------------------+  | |
||     | shared page           ACRN userspace  |  | |
||     |     +---------------+  +-----------+  | |  | |
||  +----+->| acrn_io_request |<-+  default  | |  | |
||  |  | | +---------------+  | I/O client | |  | |
||  |  | | |     ...       |  +-----------+  | |  | |
||  |  | | +---------------+  |              | |  | |
||  |  +-|-------------------------------------+  | |
||---|----|--------------------------------------|  | |
||  |  |                             kernel  |  | |
||  |  |         +-------------------+   |  | |
||  |  |         | +-------------+  HSM |   |  | |
||  |    +-------------+          |     |   |  | |
||  |         | | I/O clients |     |   |  | |
||  |         | |             |     |   |  | |
||  |         | | +-------------+   |   |  | |
||  |         +-------------------+   |  | |
|+---|-------------------------------------+  | |
+----|-------------------------------------------+
     |
+----|-------------------------------------------+
|  +-+-----------+                              |
|  | I/O handler |          ACRN Hypervisor     |
|  +-------------+                              |
+------------------------------------------------+
```

3. I/O request state transition

The state transitions of an ACRN I/O request are as follows.

```
FREE -> PENDING -> PROCESSING -> COMPLETE -> FREE -> ...
```

- FREE: this I/O request slot is empty
- PENDING: a valid I/O request is pending in this slot
- PROCESSING: the I/O request is being processed
- COMPLETE: the I/O request has been processed

An I/O request in COMPLETE or FREE state is owned by the hypervisor. HSM and ACRN userspace are in charge of processing the others.

4. Processing flow of I/O requests

a. The I/O handler of the hypervisor will fill an I/O request with PENDING state when a trapped I/O access happens in a User VM.
b. The hypervisor makes an upcall, which is a notification interrupt, to the Service VM.
c. The upcall handler schedules a worker to dispatch I/O requests.

d. The worker looks for the PENDING I/O requests, assigns them to different registered clients based on the address of the I/O accesses, updates their state to PROCESSING, and notifies the corresponding client to handle.

e. The notified client handles the assigned I/O requests.

f. The HSM updates I/O requests states to COMPLETE and notifies the hypervisor of the completion via hypercalls.