Terms
q = queue
sq = subqueue
cl = cacheline
f = fragment size
ps = page size
s = fragments (segments) count

Example parameters
p = 4
cl = 64*b*
ps= 4*Kb*
f = 8*Kb*
s = 8

total_size = (*s* * (*f* + *num_pages*(*ctrl_block_size*))) * *p* * *p* = 1.5*Mb*

Example structure of local data segment

struct collectivies_info

size_t block_size
ptrdiff_t** sbuf
ptrdiff_t** rbuf
ptrdiff_t** sext
ptrdiff_t** rext
size_t** scounts
size_t** rcounts
size_t** sdsize
size_t** rdsize
char* sbuf_contig
char* rbuf_contig
void* memory

for each rank data places in memory

*memory_size = p * (4 * sizeof(ptrdiff_t) + 5 * sizeof(size_t) + 2 * sizeof(char) + sizeof(void))*

memory_map

rtotal_sizes_by_rank - sizeof(size_t) * p
stotal_sizes_by_rank - sizeof(size_t) * p
recv_bytes_by_rank - sizeof(size_t) * p
send_bytes_by_rank - sizeof(size_t) * p
rconvertors_by_rank - sizeof(opal_convertor_t) * p
sconvertors_by_rank - sizeof(opal_convertor_t) * p
recv_buff_ptr_for_rank - sizeof(char *) * p
send_buff_ptr_for_rank - sizeof(char *) * p
rcounts_convertors - sizeof(opal_convertor_t) * p
counts_bytes_received_by_rank - sizeof(size_t) * p
stotal_sizes_by_rank_for_all_ranks - sizeof(size_t) * p * p
reduce temp - *f*
reduce temp - *f*

Example structure of shared memory segment

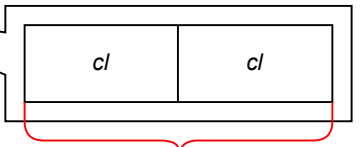
barrier_size = *num_pages*(2 * *cacheline*)

num_pages(2 * *cacheline*) = *ps* for this example

0

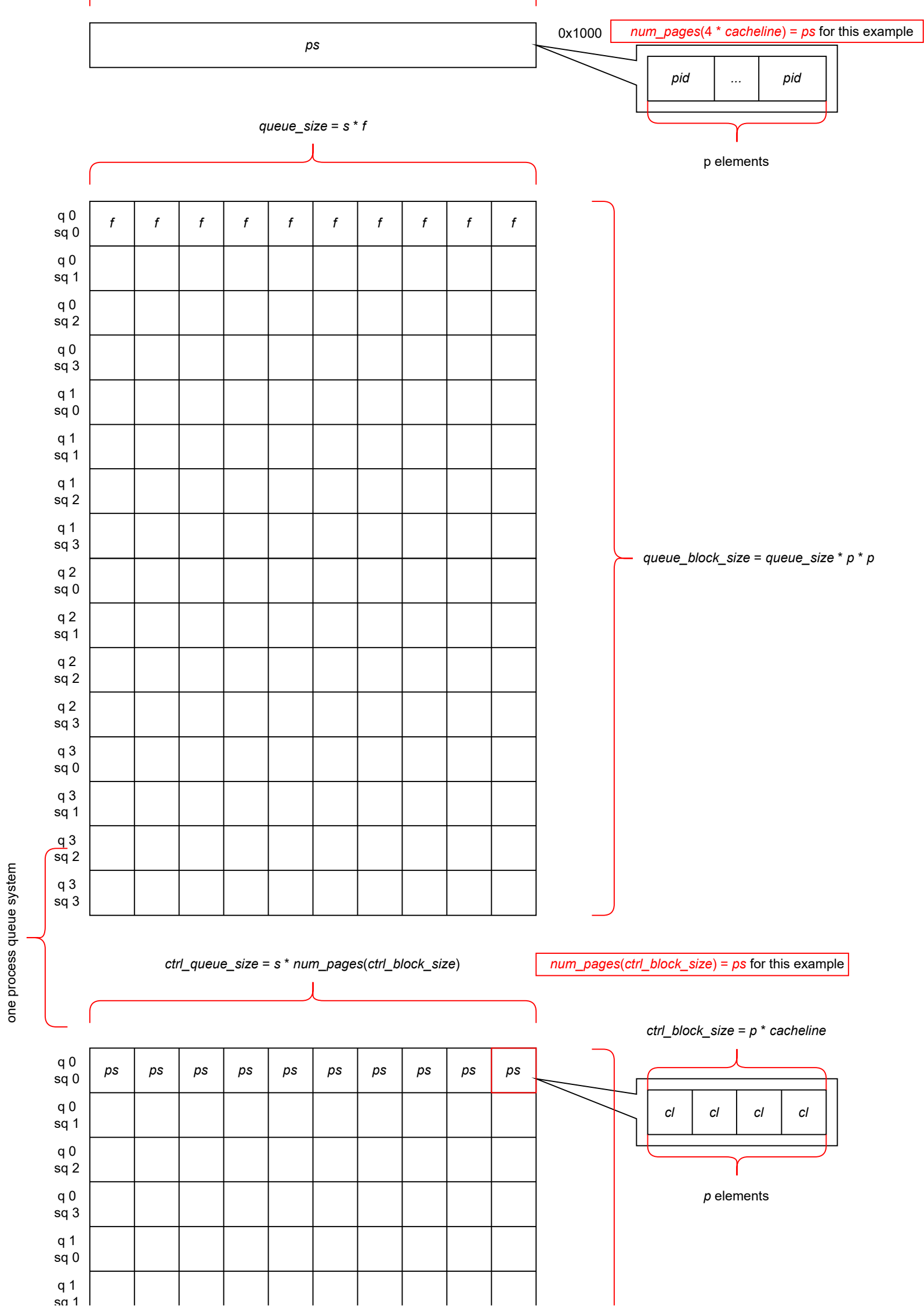
ps

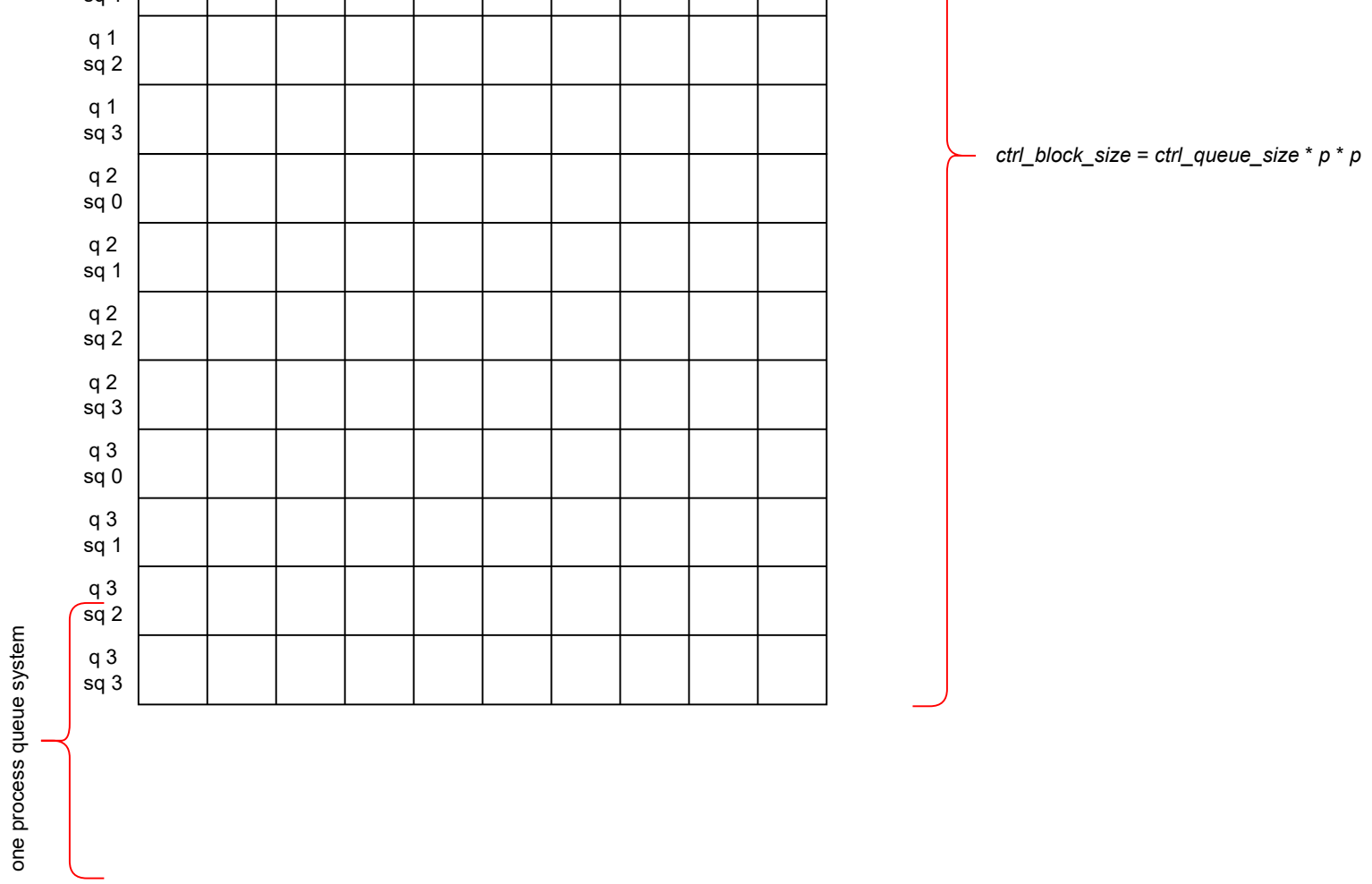
0xFFF



2 elements

pids_size = *num_pages*(p * *cacheline*)


ps
0x1000
num_pages(4 * cacheline) = ps for this example
pid
...
pid
p elements
queue_size = s * f
q 0 sq 0
q 0 sq 1
q 0 sq 2
q 0 sq 3
q 1 sq 0
q 1 sq 1
q 1 sq 2
q 1 sq 3
q 2 sq 0
q 2 sq 1
q 2 sq 2
q 2 sq 3
q 3 sq 0
q 3 sq 1
q 3 sq 2
q 3 sq 3
f
queue_block_size = queue_size * p * p
one process queue system
ctrl_queue_size = s * num_pages(ctrl_block_size)
num_pages(ctrl_block_size) = ps for this example
ctrl_block_size = p * cacheline
ps
cl
cl
cl
cl
p elements


q 1
sq 2
q 1
sq 3
q 2
sq 0
q 2
sq 1
q 2
sq 2
q 2
sq 3
q 3
sq 0
q 3
sq 1
q 3
sq 2
q 3
sq 3
ctrl_block_size = ctrl_queue_size * p * p
one process queue system