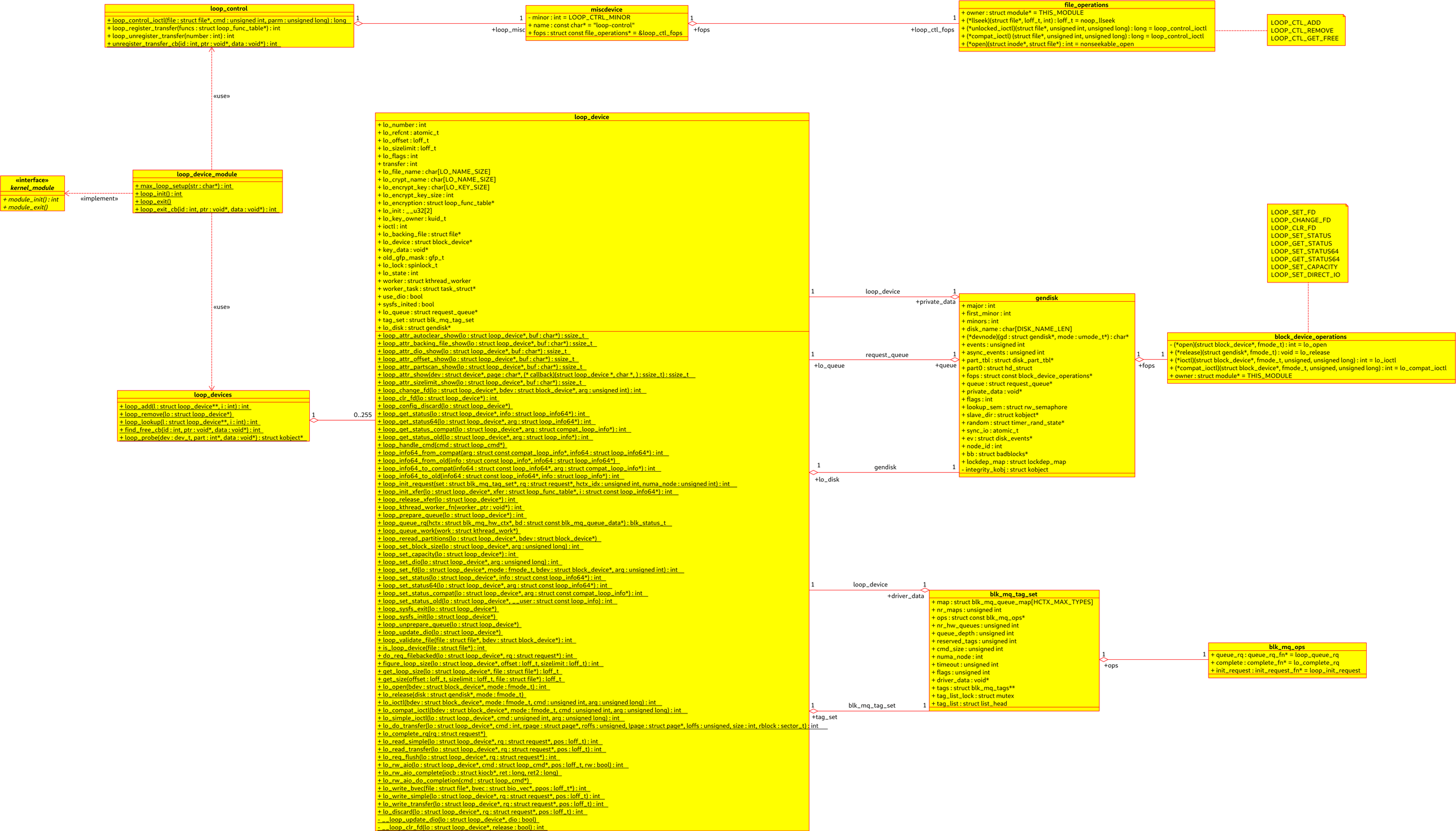
loop_control
+ loop_control_ioctl(file : struct file*, cmd : unsigned int, parm : unsigned long) : long
+ loop_register_transfer(funcs : struct loop_func_table*) : int
+ loop_unregister_transfer(number : int) : int
+ unregister_transfer_cb(id : int, ptr : void*, data : void*) : int
1
1
+loop_misc
miscdevice
- minor : int = LOOP_CTRL_MINOR
+ name : const char* = "loop-control"
+ fops : struct const file_operations* = &loop_ctl_fops
1
+fops
1
+loop_ctl_fops
file_operations
+ owner : struct module* = THIS_MODULE
+ (*llseek)(struct file*, loff_t, int) : loff_t = noop_llseek
+ (*unlocked_ioctl)(struct file*, unsigned int, unsigned long) : long = loop_control_ioctl
+ (*compat_ioctl) (struct file*, unsigned int, unsigned long) : long = loop_control_ioctl
+ (*open)(struct inode*, struct file*) : int = nonseekable_open
LOOP_CTL_ADD
LOOP_CTL_REMOVE
LOOP_CTL_GET_FREE
«use»
«interface»
kernel_module
+ module_init() : int
+ module_exit()
«implement»
loop_device_module
+ max_loop_setup(str : char*) : int
+ loop_init() : int
+ loop_exit()
+ loop_exit_cb(id : int, ptr : void*, data : void*) : int
«use»
loop_devices
+ loop_add(l : struct loop_device**, i : int) : int
+ loop_remove(lo : struct loop_device*)
+ loop_lookup(l : struct loop_device**, i : int) : int
+ find_free_cb(id : int, ptr : void*, data : void*) : int
+ loop_probe(dev : dev_t, part : int*, data : void*) : struct kobject*
1
0..255
loop_device
+ lo_number : int
+ lo_refcnt : atomic_t
+ lo_offset : loff_t
+ lo_sizelimit : loff_t
+ lo_flags : int
+ transfer : int
+ lo_file_name : char[LO_NAME_SIZE]
+ lo_crypt_name : char[LO_NAME_SIZE]
+ lo_encrypt_key : char[LO_KEY_SIZE]
+ lo_encrypt_key_size : int
+ lo_encryption : struct loop_func_table*
+ lo_init : __u32[2]
+ lo_key_owner : kuid_t
+ ioctl : int
+ lo_backing_file : struct file*
+ lo_device : struct block_device*
+ key_data : void*
+ old_gfp_mask : gfp_t
+ lo_lock : spinlock_t
+ lo_state : int
+ worker : struct kthread_worker
+ worker_task : struct task_struct*
+ use_dio : bool
+ sysfs_inited : bool
+ lo_queue : struct request_queue*
+ tag_set : struct blk_mq_tag_set
+ lo_disk : struct gendisk*
+ loop_attr_autoclear_show(lo : struct loop_device*, buf : char*) : ssize_t
+ loop_attr_backing_file_show(lo : struct loop_device*, buf : char*) : ssize_t
+ loop_attr_dio_show(lo : struct loop_device*, buf : char*) : ssize_t
+ loop_attr_offset_show(lo : struct loop_device*, buf : char*) : ssize_t
+ loop_attr_partscan_show(lo : struct loop_device*, buf : char*) : ssize_t
+ loop_attr_show(dev : struct device*, page : char*, (* callback)(struct loop_device *, char *,) : ssize_t) : ssize_t
+ loop_attr_sizelimit_show(lo : struct loop_device*, buf : char*) : ssize_t
+ loop_change_fd(lo : struct loop_device*, bdev : struct block_device*, arg : unsigned int) : int
+ loop_clr_fd(lo : struct loop_device*) : int
+ loop_config_discard(lo : struct loop_device*)
+ loop_get_status(lo : struct loop_device*, info : struct loop_info64*) : int
+ loop_get_status64(lo : struct loop_device*, arg : struct loop_info64*) : int
+ loop_get_status_compat(lo : struct loop_device*, arg : struct compat_loop_info*) : int
+ loop_get_status_old(lo : struct loop_device*, arg : struct loop_info*) : int
+ loop_handle_cmd(cmd : struct loop_cmd*)
+ loop_info64_from_compat(arg : struct const compat_loop_info*, info64 : struct loop_info64*) : int
+ loop_info64_from_old(info : struct const loop_info*, info64 : struct loop_info64*)
+ loop_info64_to_compat(info64 : struct const loop_info64*, arg : struct compat_loop_info*) : int
+ loop_info64_to_old(info64 : struct const loop_info64*, info : struct loop_info*) : int
+ loop_init_request(set : struct blk_mq_tag_set*, rq : struct request*, hctx_idx : unsigned int, numa_node : unsigned int) : int
+ loop_init_xfer(lo : struct loop_device*, xfer : struct loop_func_table*, i : struct const loop_info64*) : int
+ loop_release_xfer(lo : struct loop_device*) : int
+ loop_kthread_worker_fn(worker_ptr : void*) : int
+ loop_prepare_queue(lo : struct loop_device*) : int
+ loop_queue_rq(hctx : struct blk_mq_hw_ctx*, bd : struct const blk_mq_queue_data*) : blk_status_t
+ loop_queue_work(work : struct kthread_work*)
+ loop_reread_partitions(lo : struct loop_device*, bdev : struct block_device*)
+ loop_set_block_size(lo : struct loop_device*, arg : unsigned long) : int
+ loop_set_capacity(lo : struct loop_device*) : int
+ loop_set_dio(lo : struct loop_device*, arg : unsigned long) : int
+ loop_set_fd(lo : struct loop_device*, mode : fmode_t, bdev : struct block_device*, arg : unsigned int) : int
+ loop_set_status(lo : struct loop_device*, info : struct const loop_info64*) : int
+ loop_set_status64(lo : struct loop_device*, arg : struct const loop_info64*) : int
+ loop_set_status_compat(lo : struct loop_device*, arg : struct const compat_loop_info*) : int
+ loop_set_status_old(lo : struct loop_device*, __user : struct const loop_info) : int
+ loop_sysfs_exit(lo : struct loop_device*)
+ loop_sysfs_init(lo : struct loop_device*)
+ loop_unprepare_queue(lo : struct loop_device*)
+ loop_update_dio(lo : struct loop_device*)
+ loop_validate_file(file : struct file*, bdev : struct block_device*) : int
+ is_loop_device(file : struct file*) : int
+ do_req_filebacked(lo : struct loop_device*, rq : struct request*) : int
+ figure_loop_size(lo : struct loop_device*, offset : loff_t, sizelimit : loff_t) : int
+ get_loop_size(lo : struct loop_device*, file : struct file*) : loff_t
+ get_size(offset : loff_t, sizelimit : loff_t, file : struct file*) : loff_t
+ lo_open(bdev : struct block_device*, mode : fmode_t) : int
+ lo_release(disk : struct gendisk*, mode : fmode_t)
+ lo_ioctl(bdev : struct block_device*, mode : fmode_t, cmd : unsigned int, arg : unsigned long) : int
+ lo_compat_ioctl(bdev : struct block_device*, mode : fmode_t, cmd : unsigned int, arg : unsigned long) : int
+ lo_simple_ioctl(lo : struct loop_device*, cmd : unsigned int, arg : unsigned long) : int
+ lo_do_transfer(lo : struct loop_device*, cmd : int, rpage : struct page*, roffs : unsigned, lpage : struct page*, loffs : unsigned, size : int, rblock : sector_t) : int
+ lo_complete_rq(rq : struct request*)
+ lo_read_simple(lo : struct loop_device*, rq : struct request*, pos : loff_t) : int
+ lo_read_transfer(lo : struct loop_device*, rq : struct request*, pos : loff_t) : int
+ lo_req_flush(lo : struct loop_device*, rq : struct request*) : int
+ lo_rw_aio(lo : struct loop_device*, cmd : struct loop_cmd*, pos : loff_t, rw : bool) : int
+ lo_rw_aio_complete(iocb : struct kiocb*, ret : long, ret2 : long)
+ lo_rw_aio_do_completion(cmd : struct loop_cmd*)
+ lo_write_bvec(file : struct file*, bvec : struct bio_vec*, ppos : loff_t*) : int
+ lo_write_simple(lo : struct loop_device*, rq : struct request*, pos : loff_t) : int
+ lo_write_transfer(lo : struct loop_device*, rq : struct request*, pos : loff_t) : int
+ lo_discard(lo : struct loop_device*, rq : struct request*, pos : loff_t) : int
- __loop_update_dio(lo : struct loop_device*, dio : bool)
- __loop_clr_fd(lo : struct loop_device*, release : bool) : int
1
loop_device
1
+private_data
1
request_queue
1
+lo_queue
+queue
gendisk
+ major : int
+ first_minor : int
+ minors : int
+ disk_name : char[DISK_NAME_LEN]
+ (*devnode)(gd : struct gendisk*, mode : umode_t*) : char*
+ events : unsigned int
+ async_events : unsigned int
+ part_tbl : struct disk_part_tbl*
+ part0 : struct hd_struct
+ fops : struct const block_device_operations*
+ queue : struct request_queue*
+ private_data : void*
+ flags : int
+ lookup_sem : struct rw_semaphore
+ slave_dir : struct kobject*
+ random : struct timer_rand_state*
+ sync_io : atomic_t
+ ev : struct disk_events*
+ node_id : int
+ bb : struct badblocks*
+ lockdep_map : struct lockdep_map
- integrity_kobj : struct kobject
1
gendisk
1
+lo_disk
1
+fops
1
LOOP_SET_FD
LOOP_CHANGE_FD
LOOP_CLR_FD
LOOP_SET_STATUS
LOOP_GET_STATUS
LOOP_SET_STATUS64
LOOP_GET_STATUS64
LOOP_SET_CAPACITY
LOOP_SET_DIRECT_IO
block_device_operations
- (*open)(struct block_device*, fmode_t) : int = lo_open
+ (*release)(struct gendisk*, fmode_t) : void = lo_release
+ (*ioctl)(struct block_device*, fmode_t, unsigned, unsigned long) : int = lo_ioctl
+ (*compat_ioctl)(struct block_device*, fmode_t, unsigned, unsigned long) : int = lo_compat_ioctl
+ owner : struct module* = THIS_MODULE
1
loop_device
1
+driver_data
blk_mq_tag_set
+ map : struct blk_mq_queue_map[HCTX_MAX_TYPES]
+ nr_maps : unsigned int
+ ops : struct const blk_mq_ops*
+ nr_hw_queues : unsigned int
+ queue_depth : unsigned int
+ reserved_tags : unsigned int
+ cmd_size : unsigned int
+ numa_node : int
+ timeout : unsigned int
+ flags : unsigned int
+ driver_data : void*
+ tags : struct blk_mq_tags**
+ tag_list_lock : struct mutex
+ tag_list : struct list_head
1
blk_mq_tag_set
1
+tag_set
1
+ops
1
blk_mq_ops
+ queue_rq : queue_rq_fn* = loop_queue_rq
+ complete : complete_fn* = lo_complete_rq
+ init_request : init_request_fn* = loop_init_request