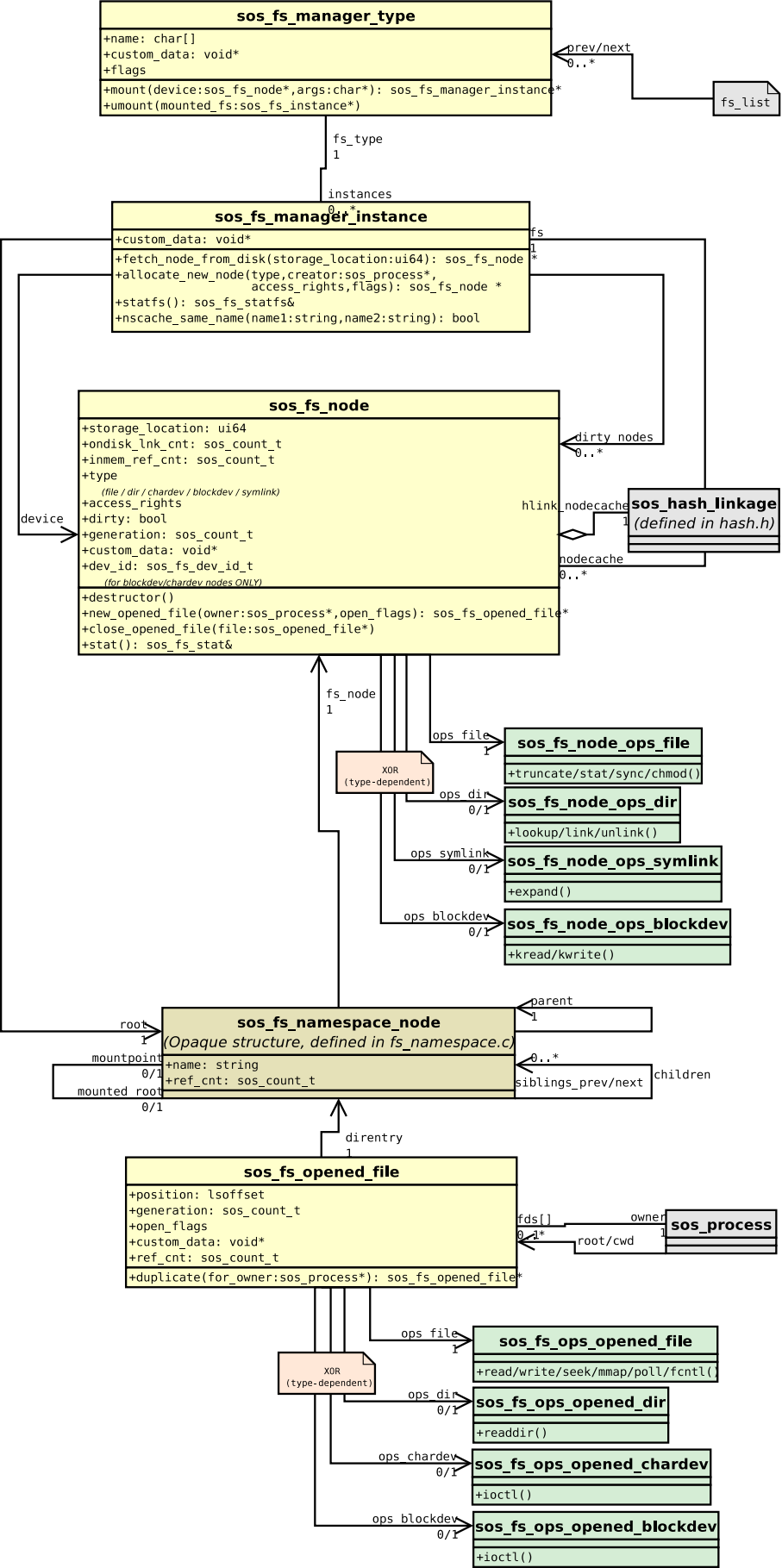
sos_fs_manager_type
+name: char[]
+custom_data: void*
+flags
+mount(device:sos_fs_node*,args:char*): sos_fs_manager_instance*
+umount(mounted_fs:sos_fs_instance*)
prev/next
0..*
fs_list
fs_type
1
instances
0..*
sos_fs_manager_instance
+custom_data: void*
+fetch_node_from_disk(storage_location:ui64): sos_fs_node
+allocate_new_node(type,creator:sos_process*,
access_rights,flags): sos_fs_node *
+statfs(): sos_fs_statfs&
+nscache_same_name(name1:string,name2:string): bool
fs
1
*
sos_fs_node
+storage_location: ui64
+ondisk_lnk_cnt: sos_count_t
+inmem_ref_cnt: sos_count_t
+type
(file / dir / chardev / blockdev / symlink)
+access_rights
+dirty: bool
+generation: sos_count_t
+custom_data: void*
+dev_id: sos_fs_dev_id_t
(for blockdev/chardev nodes ONLY)
+destructor()
+new_opened_file(owner:sos_process*,open_flags): sos_fs_opened_file*
+close_opened_file(file:sos_opened_file*)
+stat(): sos_fs_stat&
dirty_nodes
0..*
device
hlink_nodecache
1
sos_hash_linkage
(defined in hash.h)
nodecache
0..*
fs_node
1
ops_file
1
sos_fs_node_ops_file
+truncate/stat/sync/chmod()
XOR
(type-dependent)
ops_dir
0/1
sos_fs_node_ops_dir
+lookup/link/unlink()
ops_symlink
0/1
sos_fs_node_ops_symlink
+expand()
ops_blockdev
0/1
sos_fs_node_ops_blockdev
+kread/kwrite()
root
1
sos_fs_namespace_node
(Opaque structure, defined in fs_namespace.c)
+name: string
+ref_cnt: sos_count_t
parent
1
0..*
children
siblings_prev/next
mountpoint
0/1
mounted_root
0/1
direntry
1
sos_fs_opened_file
+position: lsoffset
+generation: sos_count_t
+open_flags
+custom_data: void*
+ref_cnt: sos_count_t
+duplicate(for_owner:sos_process*): sos_fs_opened_file*
fds[]
0..*
owner
1
sos_process
root/cwd
ops_file
1
sos_fs_ops_opened_file
+read/write/seek/mmap/poll/fcntl()
XOR
(type-dependent)
ops_dir
0/1
sos_fs_ops_opened_dir
+readdir()
ops_chardev
0/1
sos_fs_ops_opened_chardev
+ioctl()
ops_blockdev
0/1
sos_fs_ops_opened_blockdev
+ioctl()