Errata: Clarification of the Global Lock Example

Olaf Krzikalla
Technische Universität, Dresden, Germany,
olaf.krzikalla@tu-dresden.de

December 14, 2016

Listing 14 (sec. 10.2.3) is confusing

The return value GASPI_ERROR is used to notify the user, that a procedure has terminated due to an error (sec. 3.11). In listing 14 this value is reused in global_try_lock to inform the user, that the lock could not be acquired. This makes it impossible for a caller to distinguish between a true error and a simply failed lock acquisition. In addition it makes the example difficult to understand.

Proposed changes The example should be simplified. The handling of errors and time-outs should be left to the user. Thus use the SUCCESS_OR_DIE macro instead of SUCCESS_OR_RETURN. The time-out should be removed from the function signature of global_try_lock, since time-out handling would complicate the code beyond the scope of the example. Also, introduce some meaningful return values for global_try_lock. The listing below has also removed a superfluous comma in the function declaration. For symmetry reasons we should also change the timeout and error handling of the global_unlock function.
//...
enum try_lock_t { NO_LOCK_ACQUIRED, LOCK_ACQUIRED };

try_lock_t
global_try_lock ( const gaspi_segment_id_t seg,
const gaspi_offset_t off,
const gaspi_rank_t rank_loc
);

gaspi_rank_t iProc;
SUCCESS_OR_DIE (gaspi_proc_rank (&iProc));
gaspi_atomic_value_t old_value;
SUCCESS_OR_DIE (gaspi_atomic_compare_swap (seg
, off,
, rank_loc,
, VAL_UNLOCKED
, iProc
, &old_value
, GASPI_BLOCK
));
return (old_value == VALUE_UNLOCKED) ?
LOCK_ACQUIRED :
NO_LOCK_ACQUIRED;

void
global_unlock ( const gaspi_segment_id_t seg,
const gaspi_offset_t off,
const gaspi_rank_t rank_loc
);

gaspi_rank_t iProc;
SUCCESS_OR_DIE (gaspi_proc_rank (&iProc));
gaspiAtomic_value_t old_value;
SUCCESS_OR_DIE (gaspi_atomic_compare_swap (seg
, off,
, rank_loc,
, iProc
, &old_value
, GASPI_BLOCK
));
assert(old_value == iProc);