

Appendices

Appendix A: Essential Kernel Procedures

This appendix presents the most-used kernel procedures (KPs) and functions in OPNET Modeler. They are grouped by the following areas of functionality:

- Attribute Access
- Distributions
- Dynamic Processes
- Events and Time
- Identification and Discovery
- Interface Control Information (ICIs)
- Interrupt Processing
- Packet Generation and Processing
- Statistic Recording

Attribute Access Get or set attribute values. (Simulation attributes are “global” to the simulation model.)

- *op_ima_obj_attr_get_<type>* → completion code <type> = *color, dbl, int32, objid, str, toggle*
- *op_ima_obj_attr_set_<type>* → completion code <type> = *color, dbl, int32, objid, str, toggle*
- *op_ima_sim_attr_get_<type>* → completion code <type> = *color, dbl, int32, str*

Distributions Load distributions by name; Obtain outcomes from loaded distributions.

- *op_dist_load (dist_name, dist_arg0, dist_arg1)* → distribution handle
- *op_dist_outcome (dist_ptr)* → outcome
- *op_dist_uniform (limit)* → outcome (between 0.0 and limit)

Dynamic Processes Create a new “child” process of a given type; Destroy a process.

- *op_pro_create (model_name, ptc_mem_ptr)* → process handle
- *op_pro_destroy_options (pro_handle, options)* → completion code Identify the current process.
- *op_pro_self ()* → handle for this process Invoke another process (cause it to execute now). As an invoked process, get optional state that is passed.
- *op_pro_invoke (pro_handle, argmem_ptr)* → completion code
- *op_pro_argmem_access()* → argument pointer

Events and Time

Cancel an event.

- *op_ev_cancel (evhandle)* → completion code

Obtain current simulation time.

- *op_sim_time ()* → current simulation time in seconds

Terminate simulation.

- *op_sim_end (line0, line1, line2, line3)* → (no return value)

Identification and Discovery

Find the containing object.

- *op_id_self ()* → object ID of containing object

Find the parent of an object.

- *op_topo_parent (child_objid)* → object ID of parent

Find an object’s descendants in the hierarchy.

- *op_topo_child_count (parent_objid, child_type)* → number of children of specified type

- *op_topo_child* (*parent_objid*, *child_type*, *child_index*) → object ID of the i'th child meeting criteria

Find an object's peers. "objmtype" is one of an enumerated set; "direction" is IN or OUT. Possible use: how many links am I connected to; then, give me the i'th link.

- *op_topo_assoc_count* (*objid*, *direction*, *objmtype*) → number of associations of given direction and type
- *op_topo_assoc* (*objid*, *direction*, *objmtype*, *index*) → object ID of the i'th association meeting the direction and type criteria

Interface Control Information (ICIs)

Create or destroy an ICI.

- *op_ici_create* (*fmt_name*) → new ICI
- *op_ici_destroy* (*iciptr*) → (no return value)

Get or set ICI attribute values.

- *op_ici_attr_get_<type>*, <type> = *dbl*, *int32*, *int64*, *ptr* → completion code
- *op_ici_attr_set_<type>*, <type> = *dbl*, *int32*, *int64*, *ptr* → completion code

Associate an ICI with a particular interrupt.

- *op_ici_install* (*iciptr*) → previously installed ICI

Interrupt Processing Schedule an interrupt for this object or another at a given time. Optionally pass a "code".

- *op_intrpt_schedule_self* (*time*, *code*) → event handle for interrupt
- *op_intrpt_schedule_remote* (*time*, *code*, *mod_objid*) → event handle for interrupt

Obtain various attributes of the current interrupt.

- *op_intrpt_type ()* → type (such as packet arrival, statistic change, self interrupt)
- *op_intrpt_strm ()* → stream for packet arrivals
- *op_intrpt_ici ()* → control information passed with an interrupt (arbitrary structure)

Packet Generation and Processing

Create, copy, or destroy a packet.

- *op_pk_copy (pkptr)* → pointer to new copy of packet
- *op_pk_destroy (pkptr)* → (no return value)

Get or send a packet. (with optional delay)

- *op_pk_send_delayed (pkptr, outstrm_index, delay)* → (no return value)

Get and set named fields of a packet.

- *op_pk_nfd_set <type>, <type> = dbl, info, int32, int64, objid, pkid, pkt, ptr* → completion code
- *op_pk_nfd_get <type>, <type> = dbl, int32, int64, objid, pkid, pkt, ptr* → completion code

Get certain properties of a packet.

- *op_pk_creation_time_get (pkptr)* → simulation time at which packet was created
- *op_pk_total_size_get (pkptr)* → size of packet in bits (sum of field sizes) Insert or remove a packet from a specified subqueue.
- *op_subq_pk_remove (subq_index, pos_index)* → pointer to packet removed from the specified subqueue

Statistic Recording Obtain a handle for a statistic, given its name. Type is Global or Local.

Optionally specify an index when a single statistic name encompasses multiple independent time series.

- *op_stat_reg (stat_name, stat_index, type)* → statistic handle

Write a new value to a particular statistic. (new value is assumed to be recorded at the current time)

- *op_stat_write (stat_handle, value)* → (no return value)