

NAME

`ibnd_discover_fabric`, `ibnd_destroy_fabric`, `ibnd_debug` `ibnd_show_progress` - initialize `ibnetdiscover` library.

SYNOPSIS

```
#include <infiniband/ibnetdisc.h>
```

```
void ibnd_destroy_fabric(ibnd_fabric_t *fabric)
```

```
void ibnd_debug(int i)
```

```
void ibnd_show_progress(int i)
```

```
int ibnd_set_max_smpps_on_wire(int i)
```

DESCRIPTION

`ibnd_discover_fabric()` Discover the fabric connected to the port specified by `ibmad_port`, using a timeout specified. The "from" and "hops" parameters are optional and allow one to scan part of a fabric by specifying a node "from" and a number of hops away from that node to scan, "hops". This gives the user a "sub-fabric" which is "centered" anywhere they chose.

`ibmad_port` must be opened with at least `IB_SMI_CLASS` and `IB_SMI_DIRECT_CLASS` classes for `ibnd_discover_fabric` to work.

`ibnd_destroy_fabric()` free all memory and resources associated with the fabric.

`ibnd_debug()` Set the debug level to be printed as library operations take place.

`ibnd_show_progress()` Indicate that the library should print debug output which shows it's progress through the fabric.

`ibnd_set_max_smpps_on_wire()` Set the number of SMP's which will be issued on the wire simultaneously.

RETURN VALUE

`ibnd_discover_fabric()` return NULL on failure, otherwise a valid `ibnd_fabric_t` object.

`ibnd_destory_fabric()`, `ibnd_debug()` NONE

`ibnd_set_max_smpps_on_wire()` The previous value is returned

EXAMPLES

Discover the entire fabric connected to device mthca0 , port 1.

```
int mgmt_classes[2] = {IB_SMI_CLASS, IB_SMI_DIRECT_CLASS}; struct ibmad_port
*ibmad_port = mad_rpc_open_port(ca, ca_port, mgmt_classes, 2);      ibnd_fabric_t *fabric =
ibnd_discover_fabric(ibmad_port, 100, NULL, 0);      ...      ibnd_destroy_fabric(fabric);
      mad_rpc_close_port(ibmad_port);
```

Discover only a single node and those nodes connected to it.

```
...      str2drpath(&(port_id.drpath), from, 0, 0);      ...
ibnd_discover_fabric(ibmad_port, 100, &port_id, 1);      ...
```

SEE ALSO

libibmad, mad_rpc_open_port

AUTHORS

Ira Weiny <weiny2@llnl.gov>