

**NAME**

`ibv_get_device_list`, `ibv_free_device_list` – get and release list of available RDMA devices

**SYNOPSIS**

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

struct ibv_device **ibv_get_device_list(int *num_devices);

void ibv_free_device_list(struct ibv_device **list);
```

**DESCRIPTION**

**ibv\_get\_device\_list()** returns a NULL-terminated array of RDMA devices currently available. The argument *num\_devices* is optional; if not NULL, it is set to the number of devices returned in the array.

**ibv\_free\_device\_list()** frees the array of devices *list* returned by **ibv\_get\_device\_list()**.

**RETURN VALUE**

**ibv\_get\_device\_list()** returns the array of available RDMA devices, or sets *errno* and returns NULL if the request fails. If no devices are found then *num\_devices* is set to 0, and non-NULL is returned.

**ibv\_free\_device\_list()** returns no value.

**ERRORS****EPERM**

Permission denied.

**ENOSYS**

No kernel support for RDMA.

**ENOMEM**

Insufficient memory to complete the operation.

**NOTES**

Client code should open all the devices it intends to use with **ibv\_open\_device()** before calling **ibv\_free\_device\_list()**. Once it frees the array with **ibv\_free\_device\_list()**, it will be able to use only the open devices; pointers to unopened devices will no longer be valid.

Setting the environment variable **IBV\_SHOW\_WARNINGS** will cause warnings to be emitted to stderr if a kernel verbs device is discovered, but no corresponding userspace driver can be found for it.

**STATIC LINKING**

If **libibverbs** is statically linked to the application then all provider drivers must also be statically linked. The library will not load dynamic providers when static linking is used.

To link the providers set the **RDMA\_STATIC\_PROVIDERS** define to the comma separated list of desired providers when compiling the application. The special keyword 'all' will statically link all supported **libibverbs** providers.

This is intended to be used along with **pkg-config(1)** to setup the proper flags for **libibverbs** linking.

If this is not done then **ibv\_get\_device\_list** will always return an empty list.

Using only dynamic linking for **libibverbs** applications is strongly recommended.

**SEE ALSO**

**ibv\_fork\_init(3)**, **ibv\_get\_device\_guid(3)**, **ibv\_get\_device\_name(3)**, **ibv\_open\_device(3)**

**AUTHOR**

Dotan Barak <dotanba@gmail.com>