

# kernel data structure for DRBD-9

This describes the in kernel data structure for DRBD-9. Starting with Linux v3.14 we are reorganizing DRBD to use this data structure.

## Basic Data Structure

A node has a number of DRBD resources. Each such resource has a number of devices (aka volumes) and connections to other nodes ("peer nodes"). Each DRBD device is represented by a block device locally.

The DRBD objects are interconnected to form a matrix as depicted below; a `drbd_peer_device` object sits at each intersection between a `drbd_device` and a `drbd_connection`:

```
/-----+-----+....+-----\  
| resource | device |    | device |  
+-----+-----+....+-----+  
| connection | peer_device |    | peer_device |  
+-----+-----+....+-----+  
:         :         :    :         :  
:         :         :    :         :  
+-----+-----+....+-----+  
| connection | peer_device |    | peer_device |  
\-----+-----+....+-----/
```

In this table, horizontally, devices can be accessed from resources by their volume number. Likewise, peer\_devices can be accessed from connections by their volume number. Objects in the vertical direction are connected by double linked lists. There are back pointers from peer\_devices to their connections a devices, and from connections and devices to their resource.

All resources are in the `drbd_resources` double-linked list. In addition, all devices can be accessed by their minor device number via the `drbd_devices` idr.

The `drbd_resource`, `drbd_connection`, and `drbd_device` objects are reference counted. The `peer_device` objects only serve to establish the links between devices and connections; their lifetime is determined by the lifetime of the device and connection which they reference.