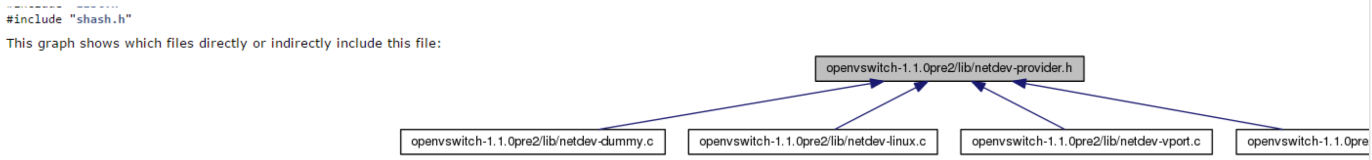
OVS Source Code browser Call graphs

Monday, September 5, 2016 11:13 AM

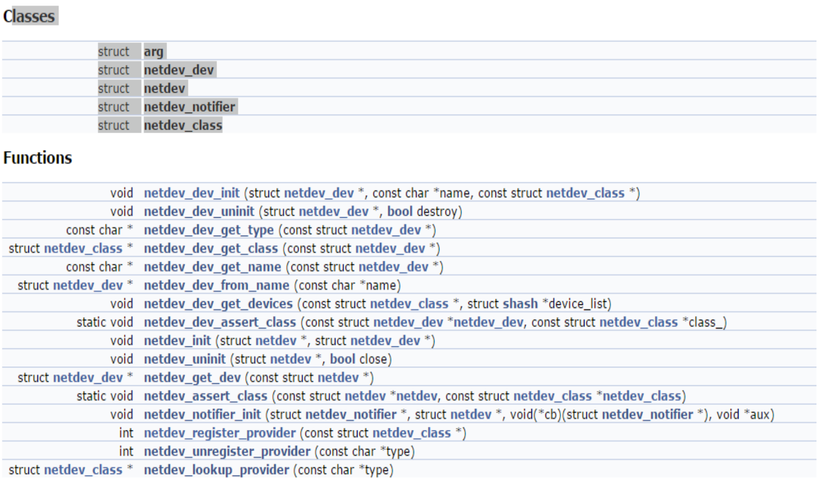
Source:
https://sourcecodebrowser.com/openvswitch/1.1.0~pre2.g2.ea763e0e/netdev-provider_8h.html



```
#include "shash.h"
```

This graph shows which files directly or indirectly include this file:

openvswitch-1.1.0pre2/lib/netdev-provider.h
openvswitch-1.1.0pre2/lib/netdev-dummy.c
openvswitch-1.1.0pre2/lib/netdev-linux.c
openvswitch-1.1.0pre2/lib/netdev-vport.c
openvswitch-1.1.0pre



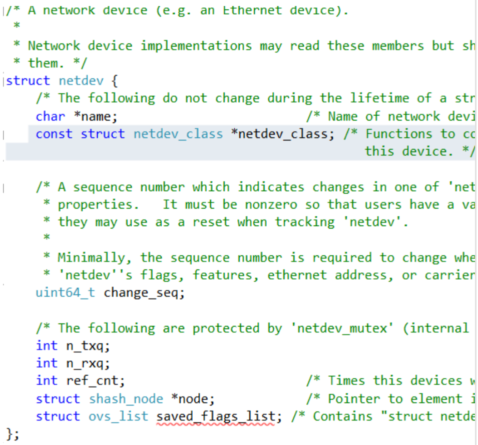
Classes

struct	arg
struct	netdev_dev
struct	netdev
struct	netdev_notifier
struct	netdev_class

Functions

void	netdev_dev_init (struct netdev_dev *, const char *name, const struct netdev_class *)
void	netdev_dev_uninit (struct netdev_dev *, bool destroy)
const char *	netdev_dev_get_type (const struct netdev_dev *)
struct netdev_class *	netdev_dev_get_class (const struct netdev_dev *)
const char *	netdev_dev_get_name (const struct netdev_dev *)
struct netdev_dev *	netdev_dev_from_name (const char *name)
void	netdev_dev_get_devices (const struct netdev_class *, struct shash *device_list)
static void	netdev_dev_assert_class (const struct netdev_dev *netdev_dev, const struct netdev_class *class_)
void	netdev_init (struct netdev *, struct netdev_dev *)
void	netdev_uninit (struct netdev *, bool close)
struct netdev_dev *	netdev_get_dev (const struct netdev *)
static void	netdev_assert_class (const struct netdev *netdev, const struct netdev_class *netdev_class)
void	netdev_notifier_init (struct netdev_notifier *, struct netdev *, void(*cb)(struct netdev_notifier *), void *aux)
int	netdev_register_provider (const struct netdev_class *)
int	netdev_unregister_provider (const char *type)
struct netdev_class *	netdev_lookup_provider (const char *type)

Screen clipping taken: 9/5/2016 5:51 PM

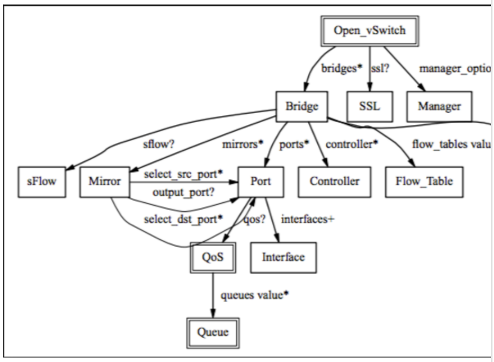


```
/* A network device (e.g. an Ethernet device).
 *
 * Network device implementations may read these members but sh
 * them. */
struct netdev {
    /* The following do not change during the lifetime of a str
    char *name;                                /* Name of network devi
    const struct netdev_class *netdev_class; /* Functions to cc
                                                this device. */

    /* A sequence number which indicates changes in one of 'net
     * properties.   It must be nonzero so that users have a va
     * they may use as a reset when tracking 'netdev'.
     *
     * Minimally, the sequence number is required to change whe
     * 'netdev''s flags, features, ethernet address, or carrier
    uint64_t change_seq;

    /* The following are protected by 'netdev_mutex' (internal
    int n_txq;
    int n_rxq;
    int ref_cnt;                               /* Times this devices w
    struct shash_node *node;                   /* Pointer to element i
    struct ovs_list saved_flags_list; /* Contains "struct netde
};
```

Screen clipping taken: 9/5/2016 7:22 PM



Bridge_reconfigure -> netdev_open->netdev_initialize-> netdev_register_provider

Refer to this link: http://flavioleitner.blogspot.de/2013/06/understanding-big-projects-call-graphs.html

netdev_class has the following function memebers amongst others:

/* Type of netdevs in this class, e.g. "system", "tap", "gre", etc.*/

```
/* ## -------------------- ## */
/* ## netdev_rxq Functions ## */
/* ## -------------------- ## */

/* If a particular netdev class does not support receiving packets, all these
 * function pointers must be NULL. */

   /* Life-cycle functions for a netdev_rxq.  See the large comment above on
    * struct netdev_class. */
   struct netdev_rxq *(*rxq_alloc)(void);
   int (*rxq_construct)(struct netdev_rxq *);
   void (*rxq_destruct)(struct netdev_rxq *);
   void (*rxq_dealloc)(struct netdev_rxq *);

   /* Attempts to receive batch of packets from 'rx' and place array of
    * pointers into '*pkts'. netdev is responsible for allocating buffers.
    * '*cnt' points to packet count for given batch. Once packets are returned
    * to caller, netdev should give up ownership of ofpbuf data.
    *
    * Implementations should allocate buffer with DP_NETDEV_HEADROOM headroom
    * and add a VLAN header which is obtained out-of-band to the packet.
    *
    * Caller is expected to pass array of size MAX_RX_BATCH.
    * This function may be set to null if it would always return EOPNOTSUPP
    * anyhow. */
   int (*rxq_recv)(struct netdev_rxq *rx, struct dp_packet **pkts,
               int *cnt);

   /* Registers with the poll loop to wake up from the next call to
    * poll_block() when a packet is ready to be received with netdev_rxq_recv()
    * on 'rx'. */
   void (*rxq_wait)(struct netdev_rxq *rx);

   /* Discards all packets waiting to be received from 'rx'. */
   int (*rxq_drain)(struct netdev_rxq *rx);
```