

btopush Reference Manual
1.0

Generated by Doxygen 1.4.5

Tue Nov 22 13:40:02 2005

Contents

1	btopush Data Structure Index	1
1.1	btopush Data Structures	1
2	btopush File Index	3
2.1	btopush File List	3
3	btopush Data Structure Documentation	5
3.1	btopush_ctx_t Struct Reference	5
3.2	btopush_dev_t Struct Reference	7
3.3	btopush_file_t Struct Reference	8
4	btopush File Documentation	9
4.1	btopush.c File Reference	9
4.2	btopush.h File Reference	12
4.3	event.c File Reference	19
4.4	get.c File Reference	20
4.5	push.c File Reference	22
4.6	sdp.c File Reference	24

Chapter 1

btopush Data Structure Index

1.1 btopush Data Structures

Here are the data structures with brief descriptions:

btopush_ctx_t	5
btopush_dev_t	7
btopush_file_t	8

Chapter 2

btopush File Index

2.1 btopush File List

Here is a list of all files with brief descriptions:

btopush.c	9
btopush.h	12
event.c	19
get.c	20
push.c	22
sdp.c	24

Chapter 3

btopush Data Structure Documentation

3.1 btopush_ctx_t Struct Reference

```
#include <btopush.h>
```

Data Fields

- `obex_t * handle`
- `bdaddr_t addr`
- `btopush_dev_t dev`
- `btopush_file_t file`
- `int c_state`
- `int req_state`
- `int last_cmd`
- `int last_rsp`

3.1.1 Detailed Description

btopush context structure. Holds local device information and state.

3.1.2 Field Documentation

3.1.2.1 `bdaddr_t btopush_ctx_t::addr`

local device address

3.1.2.2 `int btopush_ctx_t::c_state`

context state

3.1.2.3 btopush_dev_t btopush_ctx_t::dev

associated btopush device

3.1.2.4 btopush_file_t btopush_ctx_t::file

btopush file

3.1.2.5 obex_t* btopush_ctx_t::handle

OBEX handle

3.1.2.6 int btopush_ctx_t::last_cmd

last command executed

3.1.2.7 int btopush_ctx_t::last_rsp

last response

3.1.2.8 int btopush_ctx_t::req_state

request state

The documentation for this struct was generated from the following file:

- **btopush.h**

3.2 btopush_dev_t Struct Reference

```
#include <btopush.h>
```

Data Fields

- `bdaddr_t` **addr**
- `int` **channel**
- `uint32_t` **cid**
- `uint8_t *` **data**
- `size_t` **size**

3.2.1 Detailed Description

Bluetooth device structure. Represents a remote bluetooth enabled device.

3.2.2 Field Documentation

3.2.2.1 `bdaddr_t btopush_dev_t::addr`

bluetooth address of remote device.

3.2.2.2 `int btopush_dev_t::channel`

OPUSH service channel of remote device.

3.2.2.3 `uint32_t btopush_dev_t::cid`

obex connection identifier.

3.2.2.4 `uint8_t* btopush_dev_t::data`

private data buffer.

3.2.2.5 `size_t btopush_dev_t::size`

size of private data.

The documentation for this struct was generated from the following file:

- **btopush.h**

3.3 btopush_file_t Struct Reference

```
#include <btopush.h>
```

Data Fields

- `uint8_t * data`
- `uint8_t * dpos`
- `char * basename`
- `size_t size`
- `int fd`

3.3.1 Detailed Description

File structure.

3.3.2 Field Documentation

3.3.2.1 `char* btopush_file_t::basename`

basename.

3.3.2.2 `uint8_t* btopush_file_t::data`

file data.

3.3.2.3 `uint8_t* btopush_file_t::dpos`

buffer position pointer.

3.3.2.4 `int btopush_file_t::fd`

file descriptor.

3.3.2.5 `size_t btopush_file_t::size`

data size.

The documentation for this struct was generated from the following file:

- `btopush.h`

Chapter 4

btopush File Documentation

4.1 btopush.c File Reference

```
#include "btopush.h"
```

Functions

- void `btopush_init (btopush_ctx_t *ctx)`
- int `btopush_attach_dev (btopush_ctx_t *ctx, btopush_dev_t *dev)`
- int `btopush_set_dev (btopush_ctx_t *ctx, char *address, int channel)`
- int `btopush_open (btopush_ctx_t *ctx)`
- int `btopush_connect (btopush_ctx_t *ctx, char *who)`
- int `btopush_disconnect (btopush_ctx_t *ctx)`
- void `btopush_close (btopush_ctx_t *ctx)`

4.1.1 Function Documentation

4.1.1.1 int `btopush_attach_dev (btopush_ctx_t * ctx, btopush_dev_t * dev)`

Associate a specific device with a btopush context. If a connection is active for this context it should be closed before associating a new device. `btopush_attach_dev` will also try to find the right channel for `dev` if `dev->channel <= 0`. If it fails to do so `BTOPUSH_ERROR` is returned.

Parameters:

ctx btopush context

dev remote bluetooth enabled device. `dev` is not used after the association and can safely be altered.

Returns:

`BTOPUSH_SUCCESS` on success, `BTOPUSH_ERROR` on failure

4.1.1.2 void `btopush_close (btopush_ctx_t * ctx)`

Close device. After closing the device a new device should be associated using `btopush_attach_dev`.

Parameters:

ctx btopush context

4.1.1.3 int btopush_connect (btopush_ctx_t * *ctx*, char * *who*)

Connect to remote device.

Parameters:

ctx btopush context.

who optional descriptor of local device

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.1.1.4 int btopush_disconnect (btopush_ctx_t * *ctx*)

Disconnect remote device.

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.1.1.5 void btopush_init (btopush_ctx_t * *ctx*)

Initializes a btopush context.

Parameters:

ctx btopush context

4.1.1.6 int btopush_open (btopush_ctx_t * *ctx*)

Open connection to remote device.

Parameters:

ctx btopush context.

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.1.1.7 int btopush_set_dev (btopush_ctx_t * *ctx*, char * *address*, int *channel*)

Set the context device to a specific address and channel. If a connection is active for this context it should be closed before associating a new device. *btopush_set_dev* will also try to find the right channel for *address* if *channel* <= 0. If it fails to do so BTOPUSH_ERROR is returned.

Parameters:

ctx btopush context.

address remote bluetooth address.

channel OPUSH channel of remote device.

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2 btopush.h File Reference

```
#include <sys/socket.h>
#include <bluetooth/bluetooth.h>
#include <openobex/obex.h>
```

Data Structures

- struct `btopush_dev_t`
- struct `btopush_file_t`
- struct `btopush_ctx_t`

Defines

- #define `_GNU_SOURCE`
- #define `OBEX_DEFAULT_TIMEOUT` 10
- #define `OBEX_CONNECT_TIMEOUT` 2
- #define `OBEX_DISCONNECT_TIMEOUT` 2
- #define `OBEX_SEND_TIMEOUT` 30
- #define `OBEX_RECV_TIMEOUT` 10
- #define `OBEX_STREAM_CHUNK` 4096
- #define `OBEX_DEVINFO_PATH` "telecom/devinfo.txt"
- #define `OBEX_PBINFO_PATH` "telecom/pb/info.log"
- #define `OBEX_CALINFO_PATH` "telecom/cal/info.log"
- #define `BTOPUSH_MAX_DEV` 20
- #define `BTOPUSH_ERROR` 0
- #define `BTOPUSH_SUCCESS` 1

Typedefs

- typedef `btopush_dev_t` `btopush_dev_t`
- typedef `btopush_file_t` `btopush_file_t`
- typedef `btopush_ctx_t` `btopush_ctx_t`

Enumerations

- enum {
`BTOPUSH_STATE_UNDEF` = 0, `BTOPUSH_STATE_CLOSED`,
`BTOPUSH_STATE_DISCON`, `BTOPUSH_STATE_OPEN`,
`BTOPUSH_STATE_CONN` }
- enum {
`BTOPUSH_REQS_NONE` = 0, `BTOPUSH_REQS_BUSY`, `BTOPUSH_REQS_DONE`, `BTOPUSH_REQS_ERROR`,
`BTOPUSH_REQS_TIMEOUT` }

Functions

- void **catch_obex_event** (obex_t *handle, obex_object_t *obj, int mode, int event, int obex_cmd, int obex_rsp)
- int **wait_for_response** (obex_t *handle, int timeout)
- int **btopush_inq_objpush** (btopush_dev_t *dev)
- int **btopush_get_channel** (bdaddr_t *src, bdaddr_t *dst)
- int **btopush_attach_dev** (btopush_ctx_t *ctx, btopush_dev_t *dev)
- int **btopush_set_dev** (btopush_ctx_t *ctx, char *address, int channel)
- void **btopush_init** (btopush_ctx_t *ctx)
- int **btopush_open** (btopush_ctx_t *ctx)
- void **btopush_close** (btopush_ctx_t *ctx)
- int **btopush_connect** (btopush_ctx_t *ctx, char *who)
- int **btopush_disconnect** (btopush_ctx_t *ctx)
- int **btopush_open_file** (btopush_ctx_t *ctx, char *file)
- int **btopush_close_file** (btopush_ctx_t *ctx)
- int **btopush_push_stream** (btopush_ctx_t *ctx)
- void **btopush_feed_stream** (obex_t *handle, obex_object_t *object)
- int **btopush_get** (btopush_ctx_t *ctx, char *name)
- int **btopush_get_devinfo** (btopush_ctx_t *ctx)
- int **btopush_get_pbinf** (btopush_ctx_t *ctx)
- int **btopush_get_calinf** (btopush_ctx_t *ctx)

4.2.1 Define Documentation

4.2.1.1 `#define _GNU_SOURCE`

4.2.1.2 `#define BTOPUSH_ERROR 0`

4.2.1.3 `#define BTOPUSH_MAX_DEV 20`

4.2.1.4 `#define BTOPUSH_SUCCESS 1`

4.2.1.5 `#define OBEX_CALINFO_PATH "telecom/cal/info.log"`

4.2.1.6 `#define OBEX_CONNECT_TIMEOUT 2`

4.2.1.7 `#define OBEX_DEFAULT_TIMEOUT 10`

4.2.1.8 `#define OBEX_DEVINFO_PATH "telecom/devinfo.txt"`

4.2.1.9 `#define OBEX_DISCONNECT_TIMEOUT 2`

4.2.1.10 `#define OBEX_PBINFO_PATH "telecom/pb/info.log"`

4.2.1.11 `#define OBEX_RECV_TIMEOUT 10`

4.2.1.12 `#define OBEX_SEND_TIMEOUT 30`

4.2.1.13 `#define OBEX_STREAM_CHUNK 4096`

4.2.2 Typedef Documentation

4.2.2.1 `typedef struct btopush_ctx_t btopush_ctx_t`

btopush context structure. Holds local device information and state.

4.2.2.2 `typedef struct btopush_dev_t btopush_dev_t`

Bluetooth device structure. Represents a remote bluetooth enabled device.

4.2.2.3 `typedef struct btopush_file_t btopush_file_t`

File structure.

4.2.3 Enumeration Type Documentation

4.2.3.1 anonymous enum

Enumerator:

BTOPUSH_STATE_UNDEF

BTOPUSH_STATE_CLOSED

BTOPUSH_STATE_DISCON

BTOPUSH_STATE_OPEN
BTOPUSH_STATE_CONN

4.2.3.2 anonymous enum

Enumerator:

BTOPUSH_REQS_NONE
BTOPUSH_REQS_BUSY
BTOPUSH_REQS_DONE
BTOPUSH_REQS_ERROR
BTOPUSH_REQS_TIMEOUT

4.2.4 Function Documentation

4.2.4.1 int btopush_attach_dev (btopush_ctx_t * *ctx*, btopush_dev_t * *dev*)

Associate a specific device with a btopush context. If a connection is active for this context it should be closed before associating a new device. *btopush_attach_dev* will also try to find the right channel for *dev* if *dev->channel* ≤ 0 . If it fails to do so BTOPUSH_ERROR is returned.

Parameters:

ctx btopush context

dev remote bluetooth enabled device. *dev* is not used after the association and can safely be altered.

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure

4.2.4.2 void btopush_close (btopush_ctx_t * *ctx*)

Close device. After closing the device a new device should be associated using *btopush_attach_dev*.

Parameters:

ctx btopush context

4.2.4.3 int btopush_close_file (btopush_ctx_t * *ctx*)

close file associated to btopush context.

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.4 int btopush_connect (btopush_ctx_t * *ctx*, char * *who*)

Connect to remote device.

Parameters:

ctx btopush context.

who optional descriptor of local device

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.5 int btopush_disconnect (btopush_ctx_t * *ctx*)

Disconnect remote device.

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.6 void btopush_feed_stream (obex_t * *handle*, obex_object_t * *object*)

Feed next chunk of data to stream.

Parameters:

handle OBEX handle

object command object

4.2.4.7 int btopush_get (btopush_ctx_t * *ctx*, char * *name*)

OBEX get. Get a file from a remote device.

Parameters:

ctx btopush context

name filename to get

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.8 int btopush_get_calinfo (btopush_ctx_t * *ctx*)

Get calendar info if possible. This function tries to get OBEX_CALINFO_PATH: telecom/cal/info.log

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.9 int btopush_get_channel (bdaddr_t * src, bdaddr_t * dst)

Gets OPUSH channel

Parameters:

src source address

dst destination address

Returns:

BTOPUSH_ERROR on failure, channel number otherwise.

4.2.4.10 int btopush_get_devinfo (btopush_ctx_t * ctx)

Get device info if possible. This function tries to get OBEX_DEVINFO_PATH: telecom/devinfo.txt

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.11 int btopush_get_pbinfo (btopush_ctx_t * ctx)

Get phonebook info if possible. This function tries to get OBEX_PBINFORM_PATH: telecom/pb/info.log

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.12 void btopush_init (btopush_ctx_t * ctx)

Initializes a btopush context.

Parameters:

ctx btopush context

4.2.4.13 int btopush_inq_objpush (btopush_dev_t * dev)

Bluetooth address inquiry *btopush_inq_objpush* checks for OPUSH capability and channel.

Parameters:

dev array of **btopush_dev_t**(p.7) devices. Must be minimal **BTOPUSH_MAX_DEV** large.

Returns:

BTOPUSH_ERROR on failure, the amount of found OPUSH capable devices otherwise.

4.2.4.14 int btopush_open (btopush_ctx_t * ctx)

Open connection to remote device.

Parameters:

ctx btopush context.

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.15 int btopush_open_file (btopush_ctx_t * ctx, char * file)

open file and associate it to btopush context.

Parameters:

ctx btopush context

file filename

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.16 int btopush_push_stream (btopush_ctx_t * ctx)

start pushing data to associated device

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.17 int btopush_set_dev (btopush_ctx_t * ctx, char * address, int channel)

Set the context device to a specific address and channel. If a connection is active for this context it should be closed before associating a new device. *btopush_set_dev* will also try to find the right channel for *address* if *channel* <= 0. If it fails to do so BTOPUSH_ERROR is returned.

Parameters:

ctx btopush context.

address remote bluetooth address.

channel OPUSH channel of remote device.

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.2.4.18 void catch_obex_event (obex_t * handle, obex_object_t * obj, int mode, int event, int obex_cmd, int obex_rsp)**4.2.4.19 int wait_for_response (obex_t * handle, int timeout)**

4.3 event.c File Reference

```
#include "btopush.h"  
#include <stdio.h>
```

Functions

- static void **handle_finished_request** (obex_t *handle, obex_object_t *object, int obex_cmd, int obex_rsp)
- void **catch_obex_event** (obex_t *handle, obex_object_t *obj, int mode, int event, int obex_cmd, int obex_rsp)
- int **wait_for_response** (obex_t *handle, int timeout)

4.3.1 Function Documentation

4.3.1.1 void **catch_obex_event** (obex_t * *handle*, obex_object_t * *obj*, int *mode*, int *event*, int *obex_cmd*, int *obex_rsp*)

4.3.1.2 void **handle_finished_request** (obex_t * *handle*, obex_object_t * *object*, int *obex_cmd*, int *obex_rsp*) [static]

4.3.1.3 int **wait_for_response** (obex_t * *handle*, int *timeout*)

4.4 get.c File Reference

```
#include "btopush.h"
```

Functions

- `int btopush_get (btopush_ctx_t *ctx, char *name)`
- `int btopush_get_devinfo (btopush_ctx_t *ctx)`
- `int btopush_get_pbinfo (btopush_ctx_t *ctx)`
- `int btopush_get_calinfo (btopush_ctx_t *ctx)`

4.4.1 Function Documentation

4.4.1.1 `int btopush_get (btopush_ctx_t * ctx, char * name)`

OBEX get. Get a file from a remote device.

Parameters:

ctx btopush context
name filename to get

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.4.1.2 `int btopush_get_calinfo (btopush_ctx_t * ctx)`

Get calendar info if possible. This function tries to get OBEX_CALINFO_PATH: telecom/cal/info.log

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.4.1.3 `int btopush_get_devinfo (btopush_ctx_t * ctx)`

Get device info if possible. This function tries to get OBEX_DEVINFO_PATH: telecom/devinfo.txt

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.4.1.4 int btopush_get_pbinfo (btopush_ctx_t * *ctx*)

Get phonebook info if possible. This function tries to get OBEX_PBINFO_PATH: telecom/pb/info.log

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.5 push.c File Reference

```
#include "btopush.h"
#include <sys/mman.h>
#include <sys/stat.h>
#include <sys/ioctl.h>
#include <sys/types.h>
#include <fcntl.h>
#include <stdlib.h>
#include <string.h>
```

Functions

- `int btopush_open_file (btopush_ctx_t *ctx, char *file)`
- `int btopush_close_file (btopush_ctx_t *ctx)`
- `int btopush_push_stream (btopush_ctx_t *ctx)`
- `void btopush_feed_stream (obex_t *handle, obex_object_t *object)`

4.5.1 Function Documentation

4.5.1.1 `int btopush_close_file (btopush_ctx_t * ctx)`

close file associated to btopush context.

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.5.1.2 `void btopush_feed_stream (obex_t * handle, obex_object_t * object)`

Feed next chunk of data to stream.

Parameters:

handle OBEX handle

object command object

4.5.1.3 `int btopush_open_file (btopush_ctx_t * ctx, char * file)`

open file and associate it to btopush context.

Parameters:

ctx btopush context

file filename

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.5.1.4 int btopush_push_stream (btopush_ctx_t * *ctx*)

start pushing data to associated device

Parameters:

ctx btopush context

Returns:

BTOPUSH_SUCCESS on success, BTOPUSH_ERROR on failure.

4.6 sdp.c File Reference

```
#include "btopush.h"
#include <stdio.h>
#include <bluetooth/bluetooth.h>
#include <bluetooth/rfcomm.h>
#include <bluetooth/sdp.h>
#include <bluetooth/sdp_lib.h>
```

Functions

- int `btopush_inq_objpush` (`btopush_dev_t *dev`)
- int `btopush_get_channel` (`bdaddr_t *src`, `bdaddr_t *dst`)

4.6.1 Function Documentation

4.6.1.1 int `btopush_get_channel` (`bdaddr_t *src`, `bdaddr_t *dst`)

Gets OPUSH channel

Parameters:

- src* source address
- dst* destination address

Returns:

BTOPUSH_ERROR on failure, channel number otherwise.

4.6.1.2 int `btopush_inq_objpush` (`btopush_dev_t *dev`)

Bluetooth address inquiry `btopush_inq_objpush` checks for OPUSH capability and channel.

Parameters:

- dev* array of `btopush_dev_t`(p.7) devices. Must be minimal `BTOPUSH_MAX_DEV` large.

Returns:

BTOPUSH_ERROR on failure, the amount of found OPUSH capable devices otherwise.

Index

- `_GNU_SOURCE`
 - `btotush.h`, 14
- `addr`
 - `btotush_ctx_t`, 5
 - `btotush_dev_t`, 7
- `basename`
 - `btotush_file_t`, 8
- `btotush.c`, 9
 - `btotush_attach_dev`, 9
 - `btotush_close`, 9
 - `btotush_connect`, 10
 - `btotush_disconnect`, 10
 - `btotush_init`, 10
 - `btotush_open`, 10
 - `btotush_set_dev`, 10
- `btotush.h`, 12
 - `_GNU_SOURCE`, 14
 - `btotush_attach_dev`, 15
 - `btotush_close`, 15
 - `btotush_close_file`, 15
 - `btotush_connect`, 15
 - `btotush_ctx_t`, 14
 - `btotush_dev_t`, 14
 - `btotush_disconnect`, 16
 - `BTOTUSH_ERROR`, 14
 - `btotush_feed_stream`, 16
 - `btotush_file_t`, 14
 - `btotush_get`, 16
 - `btotush_get_calinfo`, 16
 - `btotush_get_channel`, 16
 - `btotush_get_devinfo`, 17
 - `btotush_get_pbinfio`, 17
 - `btotush_init`, 17
 - `btotush_inq_objpush`, 17
 - `BTOTUSH_MAX_DEV`, 14
 - `btotush_open`, 17
 - `btotush_open_file`, 18
 - `btotush_push_stream`, 18
 - `BTOTUSH_REQS_BUSY`, 15
 - `BTOTUSH_REQS_DONE`, 15
 - `BTOTUSH_REQS_ERROR`, 15
 - `BTOTUSH_REQS_NONE`, 15
 - `BTOTUSH_REQS_TIMEOUT`, 15
 - `btotush_set_dev`, 18
 - `BTOTUSH_STATE_CLOSED`, 14
 - `BTOTUSH_STATE_CONN`, 15
 - `BTOTUSH_STATE_DISCON`, 14
 - `BTOTUSH_STATE_OPEN`, 14
 - `BTOTUSH_STATE_UNDEF`, 14
 - `BTOTUSH_SUCCESS`, 14
 - `catch_obex_event`, 18
 - `OBEX_CALINFO_PATH`, 14
 - `OBEX_CONNECT_TIMEOUT`, 14
 - `OBEX_DEFAULT_TIMEOUT`, 14
 - `OBEX_DEVINFO_PATH`, 14
 - `OBEX_DISCONNECT_TIMEOUT`, 14
 - `OBEX_PBINFIO_PATH`, 14
 - `OBEX_RECV_TIMEOUT`, 14
 - `OBEX_SEND_TIMEOUT`, 14
 - `OBEX_STREAM_CHUNK`, 14
 - `wait_for_response`, 18
- `btotush_attach_dev`
 - `btotush.c`, 9
 - `btotush.h`, 15
- `btotush_close`
 - `btotush.c`, 9
 - `btotush.h`, 15
- `btotush_close_file`
 - `btotush.h`, 15
 - `push.c`, 22
- `btotush_connect`
 - `btotush.c`, 10
 - `btotush.h`, 15
- `btotush_ctx_t`, 5
 - `addr`, 5
 - `btotush.h`, 14
 - `c_state`, 5
 - `dev`, 5
 - `file`, 6
 - `handle`, 6
 - `last_cmd`, 6
 - `last_rsp`, 6
 - `req_state`, 6
- `btotush_dev_t`, 7
 - `addr`, 7
 - `btotush.h`, 14
 - `channel`, 7
 - `cid`, 7

- data, 7
- size, 7
- btopush_disconnect
 - btopush.c, 10
 - btopush.h, 16
- BTOPUSH_ERROR
 - btopush.h, 14
- btopush_feed_stream
 - btopush.h, 16
 - push.c, 22
- btopush_file_t, 8
 - basename, 8
 - btopush.h, 14
 - data, 8
 - dpos, 8
 - fd, 8
 - size, 8
- btopush_get
 - btopush.h, 16
 - get.c, 20
- btopush_get_calinfo
 - btopush.h, 16
 - get.c, 20
- btopush_get_channel
 - btopush.h, 16
 - sdp.c, 24
- btopush_get_devinfo
 - btopush.h, 17
 - get.c, 20
- btopush_get_pbinfo
 - btopush.h, 17
 - get.c, 20
- btopush_init
 - btopush.c, 10
 - btopush.h, 17
- btopush_inq_objpush
 - btopush.h, 17
 - sdp.c, 24
- BTOPUSH_MAX_DEV
 - btopush.h, 14
- btopush_open
 - btopush.c, 10
 - btopush.h, 17
- btopush_open_file
 - btopush.h, 18
 - push.c, 22
- btopush_push_stream
 - btopush.h, 18
 - push.c, 22
- BTOPUSH_REQS_BUSY
 - btopush.h, 15
- BTOPUSH_REQS_DONE
 - btopush.h, 15
- BTOPUSH_REQS_ERROR
 - btopush.h, 15
- BTOPUSH_REQS_NONE
 - btopush.h, 15
- BTOPUSH_REQS_TIMEOUT
 - btopush.h, 15
- btopush_set_dev
 - btopush.c, 10
 - btopush.h, 18
- BTOPUSH_STATE_CLOSED
 - btopush.h, 14
- BTOPUSH_STATE_CONN
 - btopush.h, 15
- BTOPUSH_STATE_DISCON
 - btopush.h, 14
- BTOPUSH_STATE_OPEN
 - btopush.h, 14
- BTOPUSH_STATE_UNDEF
 - btopush.h, 14
- BTOPUSH_SUCCESS
 - btopush.h, 14
- c_state
 - btopush_ctx_t, 5
- catch_obex_event
 - btopush.h, 18
 - event.c, 19
- channel
 - btopush_dev_t, 7
- cid
 - btopush_dev_t, 7
- data
 - btopush_dev_t, 7
 - btopush_file_t, 8
- dev
 - btopush_ctx_t, 5
- dpos
 - btopush_file_t, 8
- event.c, 19
 - catch_obex_event, 19
 - handle_finished_request, 19
 - wait_for_response, 19
- fd
 - btopush_file_t, 8
- file
 - btopush_ctx_t, 6
- get.c, 20
 - btopush_get, 20
 - btopush_get_calinfo, 20
 - btopush_get_devinfo, 20
 - btopush_get_pbinfo, 20

handle
 btopush_ctx_t, 6
handle_finished_request
 event.c, 19

last_cmd
 btopush_ctx_t, 6
last_rsp
 btopush_ctx_t, 6

OBEX_CALINFO_PATH
 btopush.h, 14
OBEX_CONNECT_TIMEOUT
 btopush.h, 14
OBEX_DEFAULT_TIMEOUT
 btopush.h, 14
OBEX_DEVINFO_PATH
 btopush.h, 14
OBEX_DISCONNECT_TIMEOUT
 btopush.h, 14
OBEX_PBINFO_PATH
 btopush.h, 14
OBEX_RECV_TIMEOUT
 btopush.h, 14
OBEX_SEND_TIMEOUT
 btopush.h, 14
OBEX_STREAM_CHUNK
 btopush.h, 14

push.c, 22
 btopush_close_file, 22
 btopush_feed_stream, 22
 btopush_open_file, 22
 btopush_push_stream, 22

req_state
 btopush_ctx_t, 6

sdp.c, 24
 btopush_get_channel, 24
 btopush_inq_objpush, 24

size
 btopush_dev_t, 7
 btopush_file_t, 8

wait_for_response
 btopush.h, 18
 event.c, 19