

TPRETURN(3)

| REVISION HISTORY | | | |
|------------------|------|-------------|------|
| NUMBER | DATE | DESCRIPTION | NAME |
| | | | |

Contents

| | | |
|----------|---------------------|----------|
| 1 | SYNOPSIS | 1 |
| 2 | DESCRIPTION | 2 |
| 3 | RETURN VALUE | 3 |
| 4 | ERRORS | 4 |
| 5 | EXAMPLE | 5 |
| 6 | BUGS | 6 |
| 7 | SEE ALSO | 7 |
| 8 | COPYING | 8 |

Chapter 1

SYNOPSIS

```
#include <atmi.h>
```

```
void tpreturn(int rval, long rcode, char *data, long len, long flags);
```

```
Link with -latmisrv|-latmisrvnomain|-latmisrvinteg -latmi -lubf -lnstd -lpthread -lrt -lm
```

Chapter 2

DESCRIPTION

Function is used to terminate service processing and return control to caller. The return buffer *data* must be auto buffer or allocated by **tpalloc()**. In either case the buffer's are made free by Enduro/X. I.e. The original request buffer is made free (if available) and the return buffer is made free too, if different than request buffer.

Normally the auto-buffer shall not be made free by user, as that will cause the corruption at the stage of the **tpreturn**. But auto-buffer free by user can be suitable in case if doing call state transfer over the non XATMI sub-system, then call context restore in other OS process with **tpsrvsetctxdata(3)** must be done with **TPNOAUTBUF** flag specified, and the original service which received request prior doing **tpcontinue(3)** shall make the auto buffer free **tpfree(3)**. If the **TPNOAUTBUF** was specified at state restore, then **tpreturn()** command will not try to free the auto buffer.

Note that if with call buffer user internally makes calls which changes the buffer type, then original call buffer is made free and the changed type buffer now is user allocated. Thus to deallocated the buffer either user does free of it, or performs **tpreturn/tpforward** with it.

Len is needed in case if buffer type is not self describing. Internally function uses long jump to get back to internal message polling mechanisms. **tpreturn()** is used for normal service calls and for conversational service calls. **tpdiscon()** shall not be used by service program, but use **tpreturn()** instead. *rcode* field is used by user program, to return back the specific value to caller.

rval values:

TPSUCCESS service terminated with success.

TPFAIL service terminated with failure. In case of global transaction, this return value will mark it as abort only.

flags values:

TPSOFTTIMEOUT soft timeout occurred. This can be used to simulate **TPETIME** error for caller. In this case *rcode* is set to **TPETIME**, that is later used at client side for **tpreturn(3)** to respond with XATMI error.

TPSOFTERR software generate XATMI error code. In this case *rcode* indicates the error code to be returned to service caller.

This function is available only for XATMI servers.

Chapter 3

RETURN VALUE

Function is marked as void and it have no return values.

Chapter 4

ERRORS

No errors available (but some details might be logged in trace files).

Chapter 5

EXAMPLE

See `atmitest/test001_basiccall/atmisv1.c` for sample code.

Chapter 6

BUGS

Report bugs to support@mavimax.com

Chapter 7

SEE ALSO

tpforward(3) **tpcall(3)**

Chapter 8

COPYING

© Mavimax, Ltd