

**TPGETRPLY(3)**

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

# Contents

<a href="#">1</a>	<a href="#">SYNOPSIS</a>	<a href="#">1</a>
<a href="#">2</a>	<a href="#">DESCRIPTION</a>	<a href="#">2</a>
<a href="#">3</a>	<a href="#">RETURN VALUE</a>	<a href="#">3</a>
<a href="#">4</a>	<a href="#">ERRORS</a>	<a href="#">4</a>
<a href="#">5</a>	<a href="#">EXAMPLE</a>	<a href="#">5</a>
<a href="#">6</a>	<a href="#">BUGS</a>	<a href="#">6</a>
<a href="#">7</a>	<a href="#">SEE ALSO</a>	<a href="#">7</a>
<a href="#">8</a>	<a href="#">COPYING</a>	<a href="#">8</a>

## Chapter 1

# SYNOPSIS

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

```
int tpgetrply(int *cd, char **data, long *len, long flags);
```

For XATMI client link with *-latmiclt -latmi -lubf -lnstd -lpthread -lrt -lm*

For XATMI server link with *-latmisrvl -latmisrvnomainl -latmisrvinteg -latmi -lubf -lnstd -lpthread -lrt -lm*

---

## Chapter 2

# DESCRIPTION

Get the response from asynchronous service call. The *cd* is pointer to call descriptor to which value might be set, in that case only matched reply is returned, other messages are dropped. *data* and *len* are field used for storing the reply messages. Fields must be allocated by **tpalloc()**. In case if **TPGETANY** flag is used, then any first response which is received is returned to user application.

### Valid flags

**TPGETANY** Return first response message received. The connection identifier is stored into *cd*.

**TPNOBLOCK** Do not block for waiting on response message. Instead function returns with *cd* equals to 0. And function return result is SUCCEED (0).

**TPSIGRSTRT** Restart the system call in progress if interrupted by signal handler. This affects only underlaying *mq\_\** function calls.

**TPNOTIME** Ignore timeout setting (**NDRX\_TOUT** env variable). Wait for reply for indefinitely.

**TPNOCHANGE** Do not allow to change the reply buffer type. If flag is set and different buffer type is received than original, then error **TPEINVAL** is returned.

**TPNOABORT** Do not abort global transaction (if one in progress), even if service failed.

---

## Chapter 3

# RETURN VALUE

On success, **tpgetrply()** returns 0; on error, -1 is returned, with **tperrno** set to indicate the error.

## Chapter 4

# ERRORS

Note that **tpsterror()** for last calls returns generic error message plus custom message with debug info from last call.

**TPEBLOCK TPENOBLOCK** was specified in flags and no message is in queue.

**TPEINVAL** Invalid parameter is given to function. Particularly pointer parameters are NULL.

**TPEBADDESC** *cd* parameter points to not issued or canceled call descriptor and **TPGETANY** flag was not specified.

**TPETIME** Service did not reply in given time (*NDRX\_TOUT* time).

**TPESVCFAIL** Service returned *TPFAIL*. This is application level failure.

**TPESVCERR** System level service failure. Server died during the message presence in service queue.

**TPESYSTEM** System failure occurred during serving. See logs i.e. user log, or debugs for more info.

**TPEOS** System failure occurred during serving. See logs i.e. user log, or debugs for more info.

---

## Chapter 5

# EXAMPLE

See `atmitest/test018_tpacalltout/atmict18.c` for sample code.



## Chapter 6

# BUGS

Report bugs to [support@mavimax.com](mailto:support@mavimax.com)

## Chapter 7

## SEE ALSO

**tpcall(3)** **tpacall(3)**

## **Chapter 8**

# **COPYING**

© Mavimax, Ltd