

# TPCOMMIT(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 tpcommit (long flags);
```

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

For XATMI server link with *-latmisrv|-latmisrvnomain|-latmisrvinteg -latmi -lubf -lnstd -lpthread -lrt -lm*

---

## Chapter 2

# DESCRIPTION

Function does commit the global transaction. Transaction must not be marked as abort only (e.g. in case if **tpcall()** failed). After issuing the command, **tmsrv(8)** will do the commit procedure, prepare phase first, log the results of prepare to persistent storage, and then do the actual commit. The commit control value value is set either by **tpscmt(3)** or **tx\_set\_commit\_return(3)** additionally it can be overridden by the **TPTXCOMMITDLOG** flag.

By default tpccommit performs full commit.

### Valid flags

**TPTXCOMMITDLOG** If set, tpccommit will return when decision for commit is logged (i.e. first phase of 2 phase commit is done - resource managers are in prepared state). Thus transaction will be completed by **tmsrv** background thread, which runs periodically by **-s** (command line argument) seconds.

## Chapter 3

# RETURN VALUE

On success, **tpcommit()** return zero; on error, -1 is returned, with **tperrno** set to indicate the error.

## Chapter 4

# ERRORS

Note that **tpstrerror()** returns generic error message plus custom message with debug info from last function call.

**TPEINVAL** flags was not 0 or was not **TPTXCOMMITDLOG**.

**TPETIME** Transaction manager (**tmsrv(8)**) did not respond in configured time-out time. The state of transaction is unknown.

**TPEABORT** Global transaction was marked for abort and was aborted, or prepare state failed for some of the resource managers and transaction was aborted.

**TPEHAZARD** The state of transaction is not fully know. It can be that it is partially committed and partially aborted.

**TPEHEURISTIC** The state of transaction is not full known. The transaction heuristically completed.

**TPESVCERR** Failed to call transaction manager, with service error. The state of transaction is unknown.

**TPEPROTO** XA subsystem was not initialized (did not call **tpopen()**), no global transaction started or caller is not initiator of transaction.

**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/test021_xafull/atmict21.c` for sample code.

## Chapter 6

# BUGS

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

## Chapter 7

## SEE ALSO

**tpbegin(3) tpabort(3) tpopen(3) tpscmt(3)**

## Chapter 8

# COPYING

© Mavimax, Ltd