

# TX\_SET\_TRANSACTION\_CONTROL(3)

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

# Contents

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

## Chapter 1

# SYNOPSIS

```
#include <tx.h>
```

```
int tx_set_transaction_control(TRANSACTION_CONTROL control);
```

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

Change the logic of how **tx\_commit(3)** or **tx\_rollback(3)** shall behave.

Setting affects current transaction and is used during the phase when commit or abort is performed. The setting value is stored in thread local stored for particular control thread.

Parameter *control* accepts following values:

- **TX\_UNCHAINED** this is **default** value. Meaning that when commit or rollback has been performed, the current control thread will not be in transaction, i.e. thread have left any global transaction.
  - **TX\_CHAINED** means that when transaction is committed or rolled back, a new transaction is started. The start of new transaction is performed only those cases when there is proper context for it. For example is commit or abort encouraged protocol error, the new transaction is not started. The commit/rollback error and new transaction state is returned combined error code. If error is received during starting next transaction, the error code is corrected (summed) with **TX\_NO\_BEGIN** error code.
-

## Chapter 3

# RETURN VALUE

On success, `tx_set_transaction_control()` return **TX\_OK**; on error, error code is returned

## Chapter 4

# ERRORS

**TX\_EINVAL** The value of *when\_return* is not **TX\_UNCHAINED** nor **TX\_CHAINED**.

**TX\_PROTOCOL\_ERROR** Transaction manager is not open by **tx\_open(3)**

## Chapter 5

# EXAMPLE

See `atmitest/test021_xafull/atmict21tx.c` for sample code.



## Chapter 6

# BUGS

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

## Chapter 7

## SEE ALSO

`tx_open(3)` `tx_close(3)` `tx_begin(3)` `tx_commit(3)` `tx_rollback(3)` `tx_info(3)` `tx_set_transaction_timeout(3)` `tx_set_commit_return(3)`

## **Chapter 8**

# **COPYING**

© Mavimax, Ltd