

**TPFREECTXT(3)**

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

# Contents

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

## Chapter 1

# SYNOPSIS

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

```
void tpfreetxt(TPCONTEXT_T context);
```

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

---

## Chapter 2

# DESCRIPTION

Function frees up the context data which was returned by **tpgetctxt(3)** function call. Function does not do the **tpterm(3)**. Thus to remove context data fully, firstly thread running with context must call **tpterm(3)**, then **tpgetctxt(3)** and then call **tpfreectxt()**.

This function uses underlying thread local storage infrastructure which is provided separately for each of the major Enduro/X libraries - libnstd (Standard library), libufb (UBF buffer library) and libatmi (ATMI library). If operations at library levels are required, then following functions can be used:

1. `ndrx_nstd_tls_new()`, `ndrx_ufb_tls_new()`, `ndrx_atmi_tls_new()` - allocate TLS data for library.
2. `ndrx_nstd_tls_get()`, `ndrx_ufb_tls_get()`, `ndrx_atmi_tls_get()` - get the TLS data for library (currently associated with thread).
3. `ndrx_nstd_tls_set()`, `ndrx_ufb_tls_set()`, `ndrx_atmi_tls_set()` - set the thread local data from saved pointer.
4. `ndrx_nstd_tls_free()`, `ndrx_ufb_tls_free()`, `ndrx_atmi_tls_free()` - free the thread local data.

Function call run in NULL context. If current TLS context equals to *context* then current thread is set to NULL context.

## Chapter 3

# RETURN VALUE

N/A

# Chapter 4

# ERRORS

N/A

## Chapter 5

# EXAMPLE

See `atmitest/test016_contextsw/atmict16.c` for sample code.



## Chapter 6

# BUGS

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

## Chapter 7

## SEE ALSO

**`tpgetctx(3)` `tpsetctx(3)` `tpsrvsetctxdata(3)` `tpsrvgetctxdata(3)` `tpcontinue(3)` `tpinit(3)`**

## Chapter 8

# COPYING

© Mavimax, Ltd