

TPFREECTXT(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 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), libubf (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_ubf_tls_new()`, `ndrx_atmi_tls_new()` - allocate TLS data for library.
2. `ndrx_nstd_tls_get()`, `ndrx_ubf_tls_get()`, `ndrx_atmi_tls_get()` - get the TLS data for library (currently associated with thread).
3. `ndrx_nstd_tls_set()`, `ndrx_ubf_tls_set()`, `ndrx_atmi_tls_set()` - set the thread local data from saved pointer.
4. `ndrx_nstd_tls_free()`, `ndrx_ubf_tls_free()`, `ndrx_atmi_tls_free()` - free the thread local data.

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 madars.vitolins@gmail.com

Chapter 7

SEE ALSO

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

Chapter 8

COPYING

© Mavimax, Ltd