TPLOGSETREQFILE(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>

int tplogsetreqfile(char **_data_, char *_filename_, char *_filesvc_);

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 enables request logging with advanced configuration. With this function you may do following:

- Take the log file name from UBF buffer (sent in *data* field) and use it as request logging facility output;
- Initialise the UBF buffer with request logging field name from *filename* parameter and user *filename* as output for logging;
- If field is not present in *data* and not present in *filename*, then call *filesvc* (if set, not NULL and not EOS), XATMI service with *data* buffer, receive the value in UBF buffer and use it as logging output. It is assumed that target *filesvc* service will call this function on incoming buffer with *filename* set. So basically concept is that you may develop in system special XATMI server which classify the incoming requests and assigns them corresponding request logging files.

All function parameters are conditional. If every field will be NULL or EOS, then function will return error. The order of the parameter usage is following:

1. If *data* is present and it is UBF buffer, then it tries to use field name from UBF buffer (field **EX_NREQLOGFILE**);
2. If EX_NREQLOGFILE is not in the UBF buffer, then use *filename*;
3. If *filename* is not set, then try to invoke *filesvc*;
4. If *filesvc* is not set, then fail with error;
5. If *data* is not UBF, then try to setup from filename;
6. If *data* buffer is not present, then try to setup from *filename*;
7. If *filename* is not set, then fail with error.

Chapter 3

RETURN VALUE

On success, **tplogsetreqfile()** 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 Missing file name in parameters (invalid parameter combination).

TPENOENT No service (*filesvc* parameter) advertised in system.

TPETIME Service did not reply in given time (*NDRX_TOUT*).

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/test031_logging/atmiclt31.c** for sample code.

Chapter 6

BUGS

Report bugs to madars.vitolins@gmail.com

Chapter 7

SEE ALSO

tploggetreqfile(3) tploggetbufreqfile(3) tplogconfig(3) tplogdump(3) tplogdumpdiff(3) tplog(3) tplogsetreqfile_direct(3) ex_devguide(guides)

Chapter 8

COPYING