TPALLOC(3)

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

SYNOPSIS

#include <atmi.h>

char *tpalloc(char **type*, char **subtype*, long *size*);

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 allocates XATMI buffer by given *type* and *subtype*. The buffer size is indicated by *size* argument, if it set to 0 and buffer type is *UBF*, then system allocates 1024 for new buffer. Currently subtype is not used, and it can be leaved NULL or empty string. If function succeeds, then Enduro/X registers buffer in internal registry. If user allocates the buffer, it is up to user to free it with **tpfree()**, only exception is with auto buffers which are allocated at point when when service receives request, and Enduro/X will free the buffer after doing **tpreturn()** or **tpforward()**.

Valid type

UBF Unified Buffer Format. Which is type key-value storage with extensive API. See **ubf.h** for list of functions.

UBF32 The same as **UBF**.

FML Compatibility mode with Tuxedo. Same as **UBF**.

FML32 Compatibility mode with Tuxedo. Same as **UBF**.

STRING String buffer. It can store any ASCII character except 0x00, which will indicate the end of the data i.e. zero terminated string.

JSON This is also string format, can hold ASCII chars, 0x00 is used for string termination. The difference between this format and **STRING**, is that JSON buffer can be used for converting it automatically to **UBF** and vice versa, either by service request or by issuing **tpjsontoubf()**, **tpubftojson()**.

CARRAY Byte array buffer. Can hold 0x00 byte.

VIEW This is view buffer, see **viewfile(5)** for view descriptor format. Basically with view the C structure is being defined (generated) which included in the code by header file. For **VIEW** buffer *subtype* must be specified. The views must be configured by setting up *VIEWDIR* and *VIEWFILES* environment variables. *VIEWFILES* must contain the list of compiled *object-files*. If attempt to allocate view that is not defined or the environment is not configured, **tpalloc** will return **TPENOENT** error. When allocating the view, the size must be specified how much space shall be allocated. How ever, it must not be less that sizeof(<generated view struct>). If size is less than 1024, then 1024 bytes are allocated. If 1024 or given size is less than structure size, then warning is printed in ULOG. If size is smaller than struct size, then any operations with structure (like **tpcall(3)** or **Bvstof(3)**) can cause segmentation fault.

Chapter 3

RETURN VALUE

On success, **tpalloc()** 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 Invalid type parameters are given to function, possibly NULL. This error also can be generated in case if the environment is not configured for Enduro/X properly.

TPEOTYPE Invalid type specified to function. VIEW sub-type not found or environment is not configured.

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. In case of insufficient memory this error will be generated too.

Chapter 5

EXAMPLE

See **atmitest/test001_basiccall/atmiclt1.c** for sample code.

Chapter 6

BUGS

Report bugs to support@mavimax.com

Chapter 7

SEE ALSO

tpfree(3) viewfile(5) tprealloc(3)

Chapter 8

COPYING