

**TPIMPORT(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>5</b>
<b>4</b>	<b>ERRORS</b>	<b>6</b>
<b>5</b>	<b>EXAMPLE</b>	<b>7</b>
<b>6</b>	<b>BUGS</b>	<b>8</b>
<b>7</b>	<b>SEE ALSO</b>	<b>9</b>
<b>8</b>	<b>COPYING</b>	<b>10</b>

## Chapter 1

# SYNOPSIS

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

```
int tpimport(char *istr, long ilen, char **obuf, long *olen, long flags);
```

For XATMI client link with *-latmiclt -latmi -lview -lnstd -lpthread -lrt -lm*

For XATMI server link with *-latmisrvl -latmisrvnomainl -latmisrvinteg -latmi -lview -lnstd -lpthread -lrt -lm*

---

## Chapter 2

# DESCRIPTION

Function converts JSON formatted text buffer into typed message buffer.

*istr* is JSON formatted buffer to import. If *ilen* is 0, *istr* is assumed to point to a null-terminated string. *obuf* is output buffer should be allocated with **tpalloc()** call. If the allocated buffer size is shorter then received one, then Enduro/X will automatically realloc the buffer to new size. If *olen* not NULL there will be stored imported buffer length.

### Valid flags

If **TPEX\_STRING** flag is set then input JSON formatted buffer is in base64.

If **TPEX\_NOCHANGE** then do not allow to change the reply buffer type. If flag is set and different buffer type is received than original, then error **TPEINVAL** is returned.

### JSON tags in import file:

- **buftype** - determine which type of buffer will be imported (STRING, UBF, VIEW, CARRAY, JSON)
- **version** - version of JSON text buffer to import
- **subtype** - contains name of VIEW buffer, used only to import VIEW buffer
- **data** - contains JSON formatted data

For example following JSON formatted text

```
{
  "buftype": "UBF",
  "version": 1,
  "data": {
    {
      "T_SHORT_FLD": 1765,
      "T_LONG_FLD": [3333111, 2],
      "T_CHAR_FLD": "A",
      "T_FLOAT_FLD": 1,
      "T_DOUBLE_FLD": [1111.220000, 333, 444],
      "T_STRING_FLD": "HELLO WORLD",
      "T_CARRAY_FLD": "AAECA0hFTExPIEJJTkFSWQQFAA=="
    }
  }
}
```

Will be imported to following UBF buffer:

T_SHORT_FLD	1765
T_LONG_FLD	3333111
T_LONG_FLD	2
T_CHAR_FLD	A

```

T_FLOAT_FLD      1.00000
T_DOUBLE_FLD     1111.220000
T_DOUBLE_FLD     333.000000
T_DOUBLE_FLD     444.000000
T_STRING_FLD     HELLO WORLD
T_CARRAY_FLD     \00\01\02\03HELLO BINARY\04\05\00

```

For example following JSON formatted text

```

{
  "buftype\" : \"VIEW\",
  "version\" : 1,
  "subtype\" : \"MYVIEW56\",
  "data\" :
  {
    "MYVIEW56\" :
    {
      "tshort1\" : 1,
      "tlong1\" : 2,
      "tchar1\" : \"A\",
      "tfloat1\" : 1,
      "tdouble1\" : 21,
      "tstring1\" : \"ABC\",
      "tcarray1\" : \"SEVMTE8AAAAAAA==\"
    }
  }
}

```

Will be imported into following VIEW:

```

VIEW MYVIEW56
#type  cname      fbname          count  flag    size    null
short   tshort1    -              1      -      -      -
long    tlong1     -              1      -      -      -
char    tchar1     -              1      -      -      -
float   tfloat1    -              1      -      -      -
double  tdouble1   -              1      -      -      -
string  tstring1   -              1      -      15     -
carray  tcarray1   -              1      -      10     -
END

```

For example following JSON formatted text

```

{
  "buftype\" : \"STRING\",
  "version\" : 1,
  "data\" : \"HELLO WORLD\"
}

```

Will be imported into following STRING buffer

```
HELLO WORLD
```

For example following JSON formatted text

```

{
  "buftype\" : \"CARRAY\",
  "version\" : 1,
  "data\" : \"SEVMTE8gV09STEQgQ0FSUkFZ\"
}

```

Will be imported into following CARRAY buffer

```
0000 48 45 4c 4c 4f 20 57 4f 52 4c 44 20 43 41 52 52  HELLO WORLD CARR
0010 41 59                                             AY
```

For example following JSON formatted text

```
{
  "buftype": "JSON",
  "version": 1,
  "data": {
    "T_SHORT_FLD": 1765,
    "T_LONG_FLD": [3333111, 2],
    "T_CHAR_FLD": "A",
    "T_FLOAT_FLD": 1,
    "T_DOUBLE_FLD": [1111.220000, 333, 444],
    "T_STRING_FLD": "HELLO WORLD",
    "T_CARRAY_FLD": "AAECA0hFTExPIEJJTkFSWQQFAA=="
  }
}
```

Will be imported into following JSON buffer

```
{
  "T_SHORT_FLD": 1765,
  "T_LONG_FLD": [3333111, 2],
  "T_CHAR_FLD": "A",
  "T_FLOAT_FLD": 1,
  "T_DOUBLE_FLD": [1111.220000, 333, 444],
  "T_STRING_FLD": "HELLO WORLD",
  "T_CARRAY_FLD": "AAECA0hFTExPIEJJTkFSWQQFAA=="
}
```

## Chapter 3

# RETURN VALUE

On success, **tpacall()** return call descriptor (>0); 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 parameter is given to function. Either *istr* is NULL or flags does not allow to change the value.

**TPEINVAL** Invalid JSON or invalid base64 encoding, *istr* is NULL or *obuf* is NULL.

**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

Sample code see under:

- `atmitest/test056_tpimpexp/atmict56_ubf.c` - import UBF buffer
  - `atmitest/test056_tpimpexp/atmict56_view.c` - import VIEW buffer
  - `atmitest/test056_tpimpexp/atmict56_carray.c` - import CARRAY buffer
  - `atmitest/test056_tpimpexp/atmict56_json.c` - import JSON buffer
  - `atmitest/test056_tpimpexp/atmict56_string.c` - import STRING buffer
-

## Chapter 6

# BUGS

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

## Chapter 7

## SEE ALSO

**tpexport(3)**

## **Chapter 8**

# **COPYING**

© Mavimax, Ltd