

*AHELP for CIAO 3.4*

## xpagetofile

Context: xpa*Jump to: [Description](#) [CHANGES IN CIAO 3.2](#) [See Also](#)*

## Synopsis

Retrieve data from one or more XPA servers and write the results to a file.

## Syntax

```

XPAGetToFile( XPA_Type xpaHdl, String_Type dest, String_Type cmd )
XPAGetToFile( XPA_Type xpaHdl, String_Type dest, String_Type cmd, fp )
XPAGetToFile( XPA_Type xpaHdl, String_Type dest, String_Type cmd, fp,
Integer_Type max_rec )
XPAGetToFile( XPA_Type xpaHdl, String_Type dest, String_Type cmd, fp,
Integer_Type max_rec, String_Type mode )

```

fp can be one of String\_Type, File\_Type, or FD\_Type

Return values: (Array\_Type names, Array\_Type messages)

## Description

The XPAGetToFile() function is similar to XPAGet() except that the returned data is stored in a file rather than as a S-Lang variable. This can be useful if a large amount of binary data is returned, in which case converting to a string format – as done by XPAGet() – is not very useful. The XPAGetB() function may also be of use in this situation.

The file handle to which the data is to be written is given by the fp variable. The default value (i.e. when it is not supplied) is to use stdout. If the argument is given then it can either be a string (in which case it represents the name of the file that will be created), a File\_Type variable – as created by fopen() and popen() – or a FD\_Type variable – as created by open().

The max\_rec parameter allows you to limit the number of servers that are contacted, and the mode parameter is currently unused.

The return values contain the name and message sent back by each server that processed the command. The length of either array gives the number of servers contacted.

## CHANGES IN CIAO 3.2

The return value of XPAGetToFile() has changed in CIAO 3.2. See the "Backwards Compatability" section below for a way of loading the module so that the old behavior is retained. The routine no longer returns an integer but two arrays.

## Backwards Compatability

The old behavior can be retained by declaring the variable

```
_CIAO3_XPA_COMPAT_
```

in the Global namespace prior to the first loading of the module. As an example, after:

```
if (0 != _featurep("xpa"))  
  error("The XPA module has already been loaded");  
public variable _CIAO3_XPA_COMPAT_  
require("xpa");
```

then the CIAO 3.1 versions of the commands will be used. The first statement, fearuting the `_featurep()` function, is not necessary, but added as a precaution to check that the XPA module has not previously been loaded.

## See Also

*modules*

[xpa](#)

*xpa*

[slxpa\\_version](#), [slxpa\\_erno](#), [xpa\\_maxhosts](#), [xpa\\_version](#), [xpaaccess](#), [xpaclose](#), [xpaget](#), [xpagetb](#),  
[xpaopen](#), [xpaset](#)

---

The Chandra X-Ray Center (CXC) is operated for NASA by the Smithsonian Astrophysical Observatory.  
60 Garden Street, Cambridge, MA 02138 USA.  
Smithsonian Institution, Copyright © 1998–2006. All rights reserved.

URL:  
<http://cxc.harvard.edu/ciao3.4/xpagetofile.html>  
Last modified: December 2006