

*AHELP for CIAO 3.4*

## ds9

Context: [modules](#)

*Jump to:* [Description](#) [Examples](#) [WORLD COORDINATE SYSTEMS CHANGES IN CIAO 3.2](#) [See Also](#)

## Synopsis

A simple S–Lang interface to ds9

## Description

The ds9 package provides a set of S–Lang functions for interacting with ds9 which are based on the XPA module (see "ahelp xpa").

The functions can be loaded using either of the following statements:

```
require ("ds9");
() = evalfile ("ds9");
```

## Functions provided by the module

The following functions are provided by the package; use "ahelp <function>" to get a detailed description of a function:

- ds9\_launch
- ds9\_view
- ds9\_quit
- ds9\_clear
- ds9\_center
- ds9\_get\_cmap and ds9\_set\_cmap
- ds9\_get\_coords
- ds9\_get\_crosshair and ds9\_put\_crosshair
- ds9\_get\_array and ds9\_put\_array
- ds9\_get\_file and ds9\_put\_file
- ds9\_get\_regions and ds9\_put\_regions
- ds9\_get\_scale and ds9\_put\_scale
- ds9\_put\_wcs, ds9\_put\_wcs\_keys, and ds9\_put\_wcs\_struct
- ds9\_pan
- ds9\_get\_zoom and ds9\_set\_zoom

## Example 1

```
chips> require ("ds9")
chips> ds9_view ("acisf03662N001_evt2.fits")
```

Here we load in the ds9 package and then use it to start up a ds9 – if one is not running – and get it to view a file. The ds9\_put\_file() command can be used to change the file being displayed:

```
chips> ds9_put_file ("expmap.fits")
```

## Example 2

```
chips> () = ds9_launch
chips> ds9_put_array ([1:512*512])
```

Here we use the package to send an array (the numbers 1 to 512\*512) to ds9 for display. We can also read the data from ds9, which provides the possibility for manipulating the ds9 display:

```
chips> img = ds9_get_array
chips> img
Integer_Type[512,512]
chips> ds9_put_array (sqrt (img))
```

## WORLD COORDINATE SYSTEMS

The routines default to treating images as arrays of raw pixel values. It is also possible to attach coordinate systems to these arrays to define the mapping between pixel number and the desired physical coordinate system.

### From a text file

If the text file image.wcs contains

```
CRPIX1  256
CRVAL1  512
CDELT1  2
CTYPE1  X
CRPIX2  256
CRVAL2  512
CDELT2  2
CTYPE2  Y
```

then it can be used by saying

```
chips> ds9_view ([1:512*512])
chips> ds9_put_wcs_keys ("image.wcs")
```

### As a string array

The numeric values for the transform can also be supplied as an eight–element string–array. For example:

```
chips> wcs = String_Type [8]
chips> wcs[[0:3]] = ["CRPIX1 256", "CRVAL1 512", "CDELTA1 2", "CTYPE1 X"]
chips> wcs[[4:7]] = ["CRPIX2 256", "CRVAL2 512", "CDELTA2 2", "CTYPE2 Y"]
chips> ds9_put_wcs_keys ("image.wcs")
```

## Numeric arrays

The WCS values can also be set using numeric arrays:

```
chips> crpix = [256,256]
chips> crval = [512,512]
chips> cdelt = [2,2]
chips> ds9_put_wcs (crpix, crval, cdelt)
```

## As a structure

The WCS values can also be specified using a S-Lang struct:

```
chips> s = struct { crpix, crval, cdelt, ctype, cunit }
chips> s.crval = [256,256]
chips> s.crval = [512,512]
chips> s.cdelt = [2,2]
chips> ds9_put_wcs_struct (s)
```

## Removing the WCS

The coordinate transformation can be cleared by saying:

```
chips> ds9_put_wcs_keys ("")
```

## CHANGES IN CIAO 3.2

This package is new to CIAO 3.2. It can only be used when the CIAO 3 compatibility mode of the XPA module is not in use (see "[ahelp xpa](#)" for more information on this mode). This means that this package will not work correctly in Sherpa in CIAO 3.2.

## See Also

*ds9*

[ds9 center](#), [ds9 clear](#), [ds9 get array](#), [ds9 get cmap](#), [ds9 get coords](#), [ds9 get crosshair](#), [ds9 get file](#), [ds9 get regions](#), [ds9 get scale](#), [ds9 get zoom](#), [ds9 launch](#), [ds9 pan](#), [ds9 put array](#), [ds9 put crosshair](#), [ds9 put file](#), [ds9 put regions](#), [ds9 put wcs](#), [ds9 put wcs keys](#), [ds9 put wcs struct](#), [ds9 quit](#), [ds9 set cmap](#), [ds9 set scale](#), [ds9 set zoom](#), [ds9 view](#)

