Contextual list of Ahelp files for CIAO 3.4

ahelp caldb calibration chandra chips concept dm ds9 group gui guide modules paramio pixlib proposaltools region sherpa slang slangrtl stackio tools varmm varmmrl xpa

Context	Topic	Summary
ahelp	ahelp	Access the CIAO help documentation.
caldb	calCreateInfo	Create a CALDB structure for use in other CALDB calls.
	calFindFile	Query the Calibration Database for a file
	calGetData	Returns the code name stored in the CALDB structure.
	calGetDate	Returns the observation start date stored in the CALDB structure.
	calGetDetector	Returns the name of the detector stored in the CALDB structure.
	calGetError	Returns the severity of the last CALDB error.
	calGetFilter	Returns the name of the filter stored in the CALDB structure.
	calGetInstrument	Returns the name of the instrument stored in the CALDB structure.
	calGetQuery	Returns the value of the query expression stored in the CALDB structure.
	calGetTelescope	Returns the name of the telescope stored in the CALDB structure.
	calGetTime	Returns the observation start time stored in the CALDB structure.
	calPrintInfo	Display the contents of the CALDB structure
	calSetData	Sets the data product code name for the next CALDB query.
	calSetDate	Sets the date for the next CALDB query.
	calSetDetector	Sets the name of the detector in the CALDB structure.
	calSetExpression	Sets the boundary condition(s) for the CALDB query.
	calSetFilter	Sets the filter to be used in a CALDB query.
	calSetInstrument	Sets the name of the instrument in the CALDB structure.
	calSetTelescope	Sets the name of the telescope in the CALDB structure.
	calSetTime	Sets the time for the next CALDB query.

calibration	ardlib	Analysis Reference Data Library
	caldb	CALibration DataBase (CALDB)

chandra	coords	Coordinate systems used in Chandra analysis
	eventdef	Column definition in Chandra event files
	guide	Grating User Interactive Data Extension (GUIDE)
	isis	Interactive Spectral Interpretation System (ISIS)
	level	Describes the amount of processing performed to create a given product.
	mtl	A description of the Mission Time Line (MTL)
	pileup	An overview of pileup in the Chandra ACIS detector.
	times	"Times" used in Chandra datasets and Chandra data analysis

chips	axes	Specifies the appearance of the bounding box surrounding a drawing area. The command AXIS is equivalent.
	batch	Turns on/off use of a plotting device.
	browse	Launches the CIAO file–browsing GUI, Prism. The command PRISM is equivalent.
	chips_auto_redraw	Set the automatic redrawing mode.
	chips_clear	Removes all plotting objects, creating a new blank drawing area.
	chips_color_name	Converts a color number to a string.
	chips_color_value	Converts the name of a color to its numeric value.
	chips_eval	Call ChIPS commands from S–Lang
	chips_get_pane	Get the current pane/drawing area.
	chips_get_xrange	Get the upper and/or lower x–axis limits of the plot
	chips_get_xscale	Get the x–axis scale
	chips_get_yrange	Get the upper and/or lower y–axis limits of the plot
	chips_get_yscale	Get the y–axis scale
	chips_get_zrange	Get the upper and/or lower z–axis limits of the plot
	chips_get_zscale	Get the z–axis scale
	chips_label	Adds a label to a drawing area.
	chips_line	Adds a line to a drawing area.
	chips_pickpoints	Read 1 or more cursor positions from ChIPS.
	chips_redraw	Redraws all plotting objects.
	chips_set_pane	Set the current pane/drawing area.
	chips_set_xrange	Set the upper and/or lower x–axis limits of the plot
	chips_set_xscale	Set the x–axis scale
	chips_set_yrange	Set the upper and/or lower y–axis limits of the plot
	chips_set_yscale	Set the y–axis scale
	chips_set_zrange	Set the upper and/or lower z–axis limits of the plot
	chips_set_zscale	Set the z–axis scale

chips_split	Creates multiple drawing areas.
chips_version	Report the version of ChIPS as a number or string.
chips	Introduction to ChIPS, CIAO's plotting package.
clear	Removes all plotting objects and creates a new blank drawing area.
colorsys	Sets the color system for hardcopy plots.
contour	Adds a contour plot to a drawing area.
curve	Plots a curve or change its attributes (PLOT is equivalent).
c	Defines the current curve.
delete	Deletes plotting objects (drawing area, curves, lines and labels) from a plot.
display	Causes the specified 2–D data to be displayed, using the default imaging display tool (i.e. ds9)
drawarea	Specifies the location for a new drawing area.
d	Defines the current drawing area.
errs	Specifies the appearance of errorbars.
exit	Terminates the program. The command QUIT is equivalent.
font	Changes the default global font.
grids	Adds grid lines to a drawing area and/or changes grid attributes.
info	Prints summary information about the current plotting objects.
label	Adds a label to a drawing area and/or changes label attributes.
levels	Sets the contour levels in a contour plot.
limits	Sets the ranges to be displayed in the axes.
linear	Changes one or more axes to linear scale.
line	Adds a line to a drawing area and/or changes line attributes.
list	Lists the contents of datafiles using the CIAO tool dmlist.
ln	Defines the current line.
location	Specifies the location for an existing drawing area.
log	Changes one or more axes to logarithmic scale.
l	Defines the current label.
pack	Renumbers plotting objects after a delete operation.
pagesize	Sets the page dimensions for hardcopy plots.
pickpoints	Starts the interactive cursor mode, which reads out the cursor position.
print	Creates a hardcopy of the current plot.
redo	Redoes the most recently undone plotting command.
redraw	Redraws the plot or sets the redraw mode
relativesize	Alters the relative size of drawing areas. The command RELSIZE is equivalent.
restore	Restores a plot with all attributes.
skip	Skips the designated number of lines from subsequent input files.

	split	Creates multiple drawing areas and/or arranges their locations.
	store	Saves a plot with all attributes.
	surface	Adds a surface plot to a drawing area.
	symbol	Specifies the appearance of symbols in a curve.
	ticks	Specifies the spacing of tick marks.
	tickvals	Controls the appearance of numerical tick mark labels.
	title	Adds a title and/or changes the title attribute.
	type	Reports the data that are associated with a curve.
	undo	Undoes the most recently executed plotting command.
	verbose	Regulates reporting of error and informational messages.
	viewpoint	Controls the rotation angle of a surface plot.
	xlabel	Adds a label along the X−axis of a drawing area and/or changes X−axis label attributes.
	ylabel	Adds a label along the Y−axis of a drawing area and/or changes Y−axis label attributes.
	zlabel	Adds a label along the Z−axis of a drawing area and/or changes Z−axis label attributes.
concept	autoname	Autonaming is a feature available in various tools for automatically naming an output file based upon the name of an input file.
	ciao	Chandra Interactive Analysis of Observations
	configure	Configuration and customization of CIAO
	merging_rules	A description of the merging rules used when combining header information.
	parameter	Describes the parameter interface used by CIAO.
	session	A session is a conceptual notion of how various CIAO applications interact with each other to form an integrated data analysis environment.
	stack	How to set a parameter to more than one value using a stack.
	subspace	Describes the filtering applied to a file
dm	dmbinning	The CIAO binning syntax
	dmcols	Selecting columns in a table
	dmfiltering	The CIAO filtering syntax
	dmimages	Images in CIAO: logical and physical coordinate systems
	dmimfiltering	The CIAO filtering syntax for images
	dmintro	The Data Model library underlies most of the CXC tools.
	dmopt	Controlling data model internal options
	dmregions	CIAO region filtering syntax
	dmsyntax	The Data Model syntax for filtering and binning files.
	dm	The CIAO Data Model

ds9	ds9_center	Center image at position
	ds9_clear	Erase a DS9 frame
	ds9_get_array	Retrieve displayed image
	ds9_get_cmap	Retrieve colormap from DS9
	ds9_get_coords	Retrieve position of next mouseclick within any frame
	ds9_get_crosshair	Retrieve position of crosshair cursor
	ds9_get_file	Retrieve name of file being displayed
	ds9_get_regions	Retrieve descriptions of regions applied to a displayed image
	ds9_get_scale	Retrieve image scale
	ds9_get_zoom	Retrieve zoom level
	ds9_launch	Establish connection to a DS9 process
	ds9_pan	Shift image position
	ds9_put_array	Visualize an image pixel array
	ds9_put_crosshair	Set position of crosshair cursor
	ds9_put_file	Load FITS file
	ds9_put_regions	Request that region descriptions be applied to the displayed image
	ds9_put_wcs_keys	Apply WCS to displayed image, using pre–formatted FITS keywords
	ds9_put_wcs_struct	Apply WCS to displayed image, using structure field values
	ds9_put_wcs	Apply WCS to displayed image, using raw numeric or string values
	ds9_quit	Terminate a DS9 process
	ds9_set_cmap	Change colormap of DS9
	ds9_set_scale	Change image scale
	ds9_set_zoom	Zoom in or out
	ds9_view	Launch DS9 with file or image pixel array
group	grpAdaptiveSnr	Adaptively group an array by signal to noise.
	grpAdaptive	Group an array by the number of counts per group using an adaptive scheme.
	grpBinFile	Group an array using the grouping applied to another dataset.
	grpBinWidth	Group an array into a set of equal–width groups (by group width).
	grpBin	Group an array using low and high boundaries.
	grpGetChansPerGroup	Calculate the number of channels (elements) in each group.
	grpGetGroupSum	Apply the supplied grouping to an array.
	grpGetGrpNum	Calculate the group number for each element in the array.
	grpMaxSlope	Group an array so that its absolute gradient is above a user–defined limit.
	grpMinSlope	Group an array so that its absolute gradient is below a

		user–defined limit.
	grpNumBins	Group an array into a set of equal–width groups (by number of groups).
	grpNumCounts	Group an array by the number of counts per group.
	grpSnr	Group an array by signal to noise.

gui	analysis–menu	The "Analysis" menu in CIAO GUIs allows users to run command–line tools.
	ciao.par	Parameter file to configure options for CIAO applications.
	ciaoshmem	Provides information on and configuration of CIAO sessions.
	filtwin	GUI to perform interactive filtering of data
	firstlook	GUI to allow an efficient means of accessing Chandra data products
	gui	Graphical User Interface (GUI)
	peg	Parameter Editor Gui (for CXCDS parameter interface)
	prism	Format–independent file browsing GUI
	taskmonitor	GUI to run a task as a background process and display the output

guide	describe	Describe is a GUIDE command that prints out detailed information about either one or two energy levels and, if two levels are input, any atomic transitions between them.
	identify	List emission lines near given wavelength.
	ionbal	Calculate collisional ionization equilibrium ion balance.
	mdl2latex	Output latex table of fitted emission lines and fluxes.
	strong	List "strong" emission lines at a given temperature, within specified wavelength bounds.

modules	caldb	The S–Lang interface to the CXC CALDB library
	ds9	A simple S–Lang interface to ds9
	group	The S–Lang interface to the CXC grouping library
	paramio	The S–Lang interface to the CXC parameter system
	pixlib	The S–Lang interface to the CXC pixlib library
	region	The S–Lang interface to the CXC region library
	stackio	The S–Lang interface to the CXC stack library
	varmmrl	The Varmm readline module for S–Lang
	varmm	The Variable, Math and Macro S–Lang library
	xpa	The S–Lang interface to the XPA library.

paramio	paccess	Returns the path to the specified parameter file.
	paramclose	Close a parameter file opened by paramopen.
	paramopen	Open a parameter file.

	pgets	Read/write individual parameter values from S–Lang.
	pget	Get a parameter value within S–Lang.
	plist_names	List parameter names for a single tool from S–Lang.
	pquery	Query a parameter value from S–Lang.
	pset	Set a parameter value within S–Lang.
	punlearn	Restore the system defaults for a parameter file from S–Lang.

pixlib	pix_apply_aspect	Convert from FPC to Sky tangent coordinates by applying an aspect solution.
	pix_chip_to_fpc	Convert from the Chip to Focal Plane coordinate (FPC) system.
	pix_chip_to_gdp	Convert from the Chip to Grating Dispersion Plane (GDP) coordinate system.
	pix_chip_to_tdet	Convert from the Chip to Tiled Detector (TDET) coordinate system.
	pix_close_pixlib	Convert between different Chandra coordinate systems in S–Lang
	pix_deapply_aspect	Convert from Sky tangent coordinates to FPC by reversing the aspect solution.
	pix_disp_config	Display the current settings of the pixlib module.
	pix_dmTanPixToWorld	Convert from FPC to World (celestial) coordinates by applying an aspect solution.
	pix_dmTanWorldToPix	Convert from World (celestial) to FPC coordinates by applying an aspect solution.
	pix_fpc_to_chip	Convert from the Focal Plane (FPC) to Chip coordinate system.
	pix_fpc_to_gdp	Convert from the Focal Plane (FPC) to Grating Dispersion Plane (GDP) coordinate system.
	pix_fpc_to_msc	Convert from the Focal Plane (FPC) to Mirror Spherical (MSC) coordinate system.
	pix_gac_to_gdp	Convert from the Grating diffracted Angular coordinates (GAC) to Grating Dispersion Plane (GDP) coordinate system.
	pix_gdp_to_gac	Convert from the Grating Dispersion Plane (GDP) to Grating diffracted Angular coordinates (GAC) system.
	pix_get_energy	Return the photon energy corresponding to a given grating dispersion (GAC) value.
	pix_get_flength	Return the value of the telescope focal length used by the pixlib module.
	pix_get_grating_angle	Return the grating angle of the currently–selected grating arm.
	pix_get_grating_period	Return the grating period of the currently–selected grating arm.
	pix_get_grating_wavelength	Return the photon wavelength corresponding to a given grating dispersion (GAC) value.
	pix_get_rowland	Return the value of the Rowland Circle diameter used by the

		pixlib module.
	pix_init_pixlib	Initialize the pixlib library.
	pix_set_aimpoint	Set the aim point to be used by the pixlib module.
	pix_set_detector	Set the detector to use in the pixlib routines.
	pix_set_fpsys	Set the focal plane system for the pixlib module.
	pix_set_gdpsys	Set the grating dispersion plane (GDP) system for the pixlib module.
	pix_set_grating	Set the grating arm and order to use in the pixlib module.
	pix_set_gzo	Set the grating zero−order position for the pixlib module.
	pix_set_simoffset	Set the stage position for the pixlib module.
	pix_set_tdetsys	Set the tiled detector plane system for the pixlib module.
	pix_tdet_to_chip	Convert from the Tiled Detector (TDET) to Chip coordinate system.

proposaltools	colden	Interactive program to evaluate the neutral hydrogen column density at a given direction on the sky
	dates	DATES is an interactive calendar and time conversion tool.
	obsvis	Observation Visualizer
	pimms	PIMMS (Portable, Interactive Multi−Mission Simulator) converts source fluxes and count rates between missions.
	precess	Interactive astronomical coordinate conversion program that provides precession of equatorial coordinates and conversion between equatorial, ecliptic, galactic, and supergalactic coordinates
	prop−coords	Defines available coordinate systems for Chandra Proposal Tools
	prop−time	Defines available calendars and timescales for Chandra Proposal Tools
	prop−tools	Tools that aid in the preparation and submission of Chandra proposals

region	regArea	Calculate the area enclosed by a region.
	regExtent	Calculate the bounding box of a region.
	regInsideRegion	Is a point (or set of points) inside a region?
	regParse	Parse a region for use by the routines in the region library
	regPrintRegion	Print out details about a region.
	regRegionString	Print out details about a region.

sherpa	analysis	Specifies whether to analyze datasets in energy, wavelength, or channel space.
	atten	Attenuation by ISM. Integration OFF.
	autoest	Automatic estimation of initial parameter values in Sherpa models

background	Defines a model expression to be used for the background. The command BG is an abbreviated equivalent.
back	Inputs the contents of one or more background data files.
bayes	A Bayesian maximum likelihood function.
bbodyfreq	Blackbody as a function of frequency. Integration ON.
bbody	Blackbody as a function of energy. Integration ON.
berrors	Defines an expression to be used to specify the statistical errors for background data. The commands BACKERRORS and BSTATERRORS are equivalent.
beta1d	1–D surface brightness beta–model. Integration OFF
beta2d	2–D Lorentzian with varying power law. Integration OFF. The LORPOW2D model is equivalent.
box1d	1–D box function. Integration OFF.
box2d	2–D box function. Integration OFF.
bpl1d	Broken power law function. Integration ON.
bsyserrors	Defines an expression or file to be used to specify the systematic errors for background data.
bye	Terminates the Sherpa program. Other equivalent termination commands are EXIT and QUIT.
calc_kcorr	Calculate the k correction for a spectral model, redshift, and energy range.
cash	A maximum likelihood function.
cast	[REMOVED AS OF CIAO 3.0.2] Casts a dataset to single or double precision.
chicvar	Chi–square statistic with constant variance computed from the counts data.
chidvar	Chi–square statistic with variance computed from the data.
chigehrels	Chi–square statistic with the Gehrels variance function.
chimvar	Chi–square statistic with variance computed from model amplitudes.
chiprimini	Chi–square statistic with Primini variance function.
chisquare	Chi–square statistic.
close	Closes the image display window.
compute_errors	Module function to estimate errors for an array of data
compute_statistic	Module function to estimate a statistic given arrays of data, model amplitudes, and errors, etc.
const1d	1–D constant amplitude model. Integration ON.
const2d	2–D constant amplitude model. Integration ON.
coord	Specifies the coordinate system for use in fits to 2–D images.
cos	Cosine function. Integration OFF.
covariance	Computes covariance matrices, and provides an estimate of confidence intervals for selected thawed parameters.
cplot	Causes the specified 2–D data to be displayed, with a contour plot, via ChIPS.

createparamset	To be especially efficient, one may establish and assign a name to a model component, as well as set model parameters and their ranges, all at one time using a single command:
create_model	Module functions to instantiate new source or instrument models.
create	Establishes a model component and its parameters, for use in the current Sherpa session.
cstat	A maximum likelihood function.
dataspace	Creates a data grid on which models may be evaluated.
data	Inputs the contents of one or more source data files.
dcounts	Calculates the sum of observed counts data for source or background datasets.
delta1d	1–D delta function. Integration ON.
delta2d	2–D delta function. Integration ON.
dered	Dereddening function. Integration OFF.
devaucouleurs	DeVaucouleurs profile. Integration OFF.
dollarsign	Must precede any Unix command issued within Sherpa or ChIPS.
echo	Turns on/off reporting of user keystrokes.
edge	Photoabsorption edge model. Integration OFF.
eflux	Calculates the unconvolved energy flux for source or background datasets.
eqwidth	Computes the equivalent width of an emission or absorption line in source or background data.
erase	Erases user inputs and settings, and/or model components.
erfc	A 1–D complementary error function. Integration OFF.
erf	A 1–D error function. Integration OFF.
errors	Defines an expression to be used to specify the statistical errors for source data.
fakeit	Creates a simulated 1–D dataset.
farf2d	A 2–D file–based ancillary response model. Alternate names include FEXPMAP and FEXPMAP2D.
farf	A 1–D file–based ancillary response model.
feffile	Specifies the Fits Embedded Function (FEF) file whose contents will be displayed with FEFPLOT.
fefplot	Plots an instrument response stored in a Fits Embedded Function (FEF) file that is read in via FEFFILE.
fit	Initiates fitting (optimization). The command [B]RUN is equivalent.
flux	Calculates the unconvolved photon flux for source or background datasets.
fpsf1d	A 1–D file–based PSF instrument model.
fpsf	A 2–D file–based PSF instrument model.
freeze	Prohibits model parameter(s) from varying.

frmf	A 1–D file–based response matrix model.
ftest	Computes significance using the F test.
gauss1d	1–D unnormalized Gaussian function. Integration ON.
gauss2d	2–D unnormalized Gaussian function. Integration OFF.
getx	Assigns x–axis values taken from a plot to model parameters.
gety	Assigns y–axis values taken from a plot to model parameters.
get_analysis	Module function to retrieve the current analysis setting.
get_arf_axes	Module functions to retrieve the energy/wavelength grid of an ARF associated with source and background data
get_axes	Module functions to get the energy/wavelength/channel grid of source and background datasets.
get_coord	Module function to retrieve the current coordinate setting for 2–D image data.
get_data	Module functions to get the amplitudes of source and background datasets.
get_dcounts_sum	Calculates the sum of observed counts in source and background datasets using module functions in Sherpa.
get_defined_models	Module functions that return lists of defined Sherpa source and instrument models
get_dir	Module function to retrieve the current path.
get_eflux	Calculate the energy flux (unconvolved) for source or background datasets using module functions in Sherpa.
get_energy_axes	Module functions to retrieve the energy grids of source and background datasets.
get_eqwidth	Module functions that compute the equivalent width of an emission or absorption line in source or background data.
get_errors	Module functions to get the error estimates of source and background datasets.
get_filename	Module functions to retrieve filenames associated with a dataset.
get_filter_expr	Module function to retrieve the strings describing filters applied to source and background datasets.
get_filter	Module functions to get the filter arrays associated with source and background datasets.
get_fit	Module functions to get information about the quality of a fit.
get_flux2d	Module functions for computing fluxes/summing counts in 2–D images.
get_fluxed_spectrum	Retrieve a fluxed spectrum (counts divided by ARF) using module functions in Sherpa.
get_flux_str	Retrieves a default structure for use with get_pflux(), etc.
get_ftest	Module function that returns the statistical significance computed with the F test
get_groups	Module functions for retrieving a grouping or quality array from source and background files.

get_lfactorial	Module function to compute the natural logarithm of the factorial of the input quantity
get_mcounts_sum	Module functions for computing the sum of convolved model counts in source and background datasets.
get_metadata	Module functions to retrieve metadata associated with source and background datasets.
get_method_expr	Module function to retrieve the name of the current optimization method.
get_models	Module functions that return lists of available source and instrument models
get_model_params	Access to the default model and instrument parameters of Sherpa from S–Lang.
get_num_par	Module functions that report the total number of parameters for all defined models, including instrument models
get_paramestint	Module functions to retrieve the value and statistic arrays from the most recent run of a parameter estimation method
get_paramestlim	Module functions to determine confidence intervals, and retrieve the parameter bounds.
get_paramestreg	Module functions to retrieve the value and statistic arrays from the most recent run of a parameter estimation method
get_paramest	Module functions to retrieve the output of parameter estimation methods.
get_par	Module function for getting model parameter values, etc.
get_pflux	get_flux(), get_pflux(), and get_bpflux() calculate the photon flux (unconvolved) for source and background datasets using the module functions in Sherpa.
get_photon_axes	Module functions to retrieve photon–space grids over which models are evaluated.
get_photon_energy_axes	Module functions to retrieve photon–space energy grids over which models are evaluated.
get_photon_wave_axes	Module functions to retrieve photon–space wavelength grids over which models are evaluated.
get_qvalue	Module function that returns the statistical significance computed as a q–value.
get_raw_axes	Module functions to retrieve the raw channel number grids of source and background datasets.
get_record	Module function to return a record of model parameter values at the end of each iteration of the fitting process.
get_source_components	Return, as an array, the names of the model components in the source expression of a dataset.
get_source	Module functions to retrieve predicted source and background model photon amplitudes.
get_stackexpr	Module functions to retrieve a model stack expression.
get_statistic	Module functions to get the current value of the statistic comparing source and background data and model values.

get_stats	Module functions to retrieve predicted model counts, statistics, residuals, sigma residuals, and ratios from source and background datasets.
get_stat_expr	Module function to retrieve the name of the current statistic.
get_syserrors	Module functions to get the systematic error estimates of source and background datasets.
get_verbose	Module function to display Sherpa's current verbosity setting.
get_wave_axes	Module functions to retrieve the wavelength grids of source and background datasets.
get_weights	Module functions to get the statistical weights of source and background datasets.
get	Summary of Sherpa/S–Lang module functions that retrieve settings or data.
goodness	Reports information on the goodness–of–fit.
grid–powell	A grid search utilizing the Powell method at each grid point.
gridmodel	N–D user–specified amplitude model. Integration OFF.
grid	A grid search of parameter space, with no minimization.
groupByCounts	Group a dataset by number of counts or signal–to–noise within Sherpa.
group	Causes Sherpa to apply a read–in bin grouping scheme to source or background data.
guess	Estimates initial parameter values and ranges, based on input data.
hubble	Hubble–Reynolds profile. Integration OFF. The REYNOLDS model is equivalent.
ignore	Specifies a data portion to be excluded, for 1– or 2–D data.
image	Causes the specified 2–D data to be displayed, via ds9.
instrument	Defines an expression to be used for modeling the instrument in source or background data analysis. The command RESPONSE is equivalent.
integrate	Controls the integration of model components.
interval–projection	Plots the fit statistic as a function of parameter value, using the PROJECTION algorithm. The commands INT–PROJ and INTPROJ are abbreviated equivalents.
interval–uncertainty	Plots the fit statistic as a function of parameter value, using the UNCERTAINTY algorithm. The commands INT–UNC and INTUNC are abbreviated equivalents.
is_paramset	Module functions to determine if whether the specified parameter(s) are thawed or frozen.
is_subtracted	Module function to determine if whether the specified dataset is background–subtracted.
is	Summary of Sherpa/S–Lang module query functions.
jdpileup	A CCD pileup model developed by John Davis of MIT.
jointmode	Joint–mode models define the function argument, on which a model expression is evaluated, to be a particular data axis.

journal	Turns on/off the writing of all commands to a file.
kernel	Defines an expression to be used as a kernel in 1D source model convolution.
levenberg–marquardt	The Levenberg–Marquardt optimization method.
linebroad	Line broadening profile. Integration ON.
lineid	Sets a user–defined line identification (or general model description) for the model in the MDL output.
linkparam	Parameter expressions are used to link model component parameters:
list_paramest	Module functions listing the current and default values of the parameters used to configure each Sherpa parameter estimation method.
list_par	Module function that lists information about all defined model parameters.
list	Summary of Sherpa/S–Lang module list functions.
load_arf	Module function to load data from an ARF file into Sherpa
load_ascii	Module function to load ASCII data into Sherpa
load_backset	Module function to load background data into Sherpa
load_back_from	Module function to load background from a PHA file into Sherpa
load_dataset	Module function to load data into Sherpa
load_fitsbin	Module functions to load source and background data from a FITS binary table into Sherpa
load_image	Module functions to load source and background data from a FITS image into Sherpa
load_inst_from	Module function to load data from ARF and RMF files into Sherpa source or background instrument models.
load_inst	Module function to load data from ARF and RMF files into Sherpa. load_instrument is an alternate name.
load_pha2	Module function to load source data from PHA Type II files into Sherpa
load_pha	Module function to load source and background data from PHA Type I files into Sherpa
load_rmf	Module function to load data from an RMF file into Sherpa
load	Summary of Sherpa/S–Lang module functions that load data into Sherpa
lorentz1d	1–D normalized Lorentzian function. Integration ON.
lorentz2d	2–D unnormalized Lorentzian function. Integration OFF.
lplot	Causes the specified 1–D data to be displayed, via ChIPS.
mcounts	Calculates the sum of convolved model amplitudes for source or background datasets.
mdl	A Model Descriptor List (MDL) file stores a collection of datasets and model(s) in a structured format that can be read or written by Sherpa.

method	Specifies the optimization method. The command SEARCHMETHOD is equivalent.
mlr	Computes significance using the Maximum Likelihood Ratio test.
modelexpr	Model components can be used to create model expressions, in conjunction with the SOURCE and BACKGROUND commands (which then are used to fit the data), or with model stacks:
modelstack	A model stack assigns an arbitrary name to a model expression, for subsequent use in parameter expressions and/or nested models.
models	Summary of Available Models
monte−lm	A Monte Carlo search utilizing the Powell method at each selected point.
monte−powell	A Monte Carlo search utilizing the Powell method at each selected point.
montecarlo	A Monte Carlo search of parameter space.
nbeta	1−D normalized beta function. Integration ON.
nestedmodel	Nested models define the function argument on which a model expression is evaluated to be another model expression. A nested model, <nested_model>, appears as an element of a model expression, <modelExpr>.
ngauss1d	1−D normalized Gaussian function. Integration ON.
noise	Defines an expression to be used for modeling detector noise in source and background datasets.
notice	Specifies a data portion to be included, for 1− or 2−D data.
numbersign	Denotes a comment. The symbol % also denotes a comment.
open	Opens the image display window.
oplot	Causes multiple data curves to be displayed in the same drawing area, via ChIPS.
paramest	An interactive interface to the parameter estimation routines in Sherpa.
paramprompt	Turns on/off prompting for model parameter values.
paramset	The model components that have been established in the current Sherpa session, and their parameter information, may be listed with the command SHOW MODELS. Values for these established model component parameters may be set individually using one of the following command syntax options:
pileup	Defines the pileup model expression to be used for fitting a 1D dataset.
plotx	[REMOVED AS OF CIAO 3.0.2] Sets the unit type for the x−axis of a plot.
ploty	Sets the unit type for the y−axis of a plot.
plot_eprof	Display a radial profile (elliptical annuli) of 2D data, fit, and residuals.

plot_rprof	Display a radial profile (circular annuli) of 2D data, fit, and residuals.
poisson	Poisson function. Integration OFF.
polynom1d	1−D polynomial function. Integration ON.
polynom2d	2−D polynomial function. Integration ON.
powell	The Powell optimization method.
powlaw1d	1−D power law. Integration ON.
projection	Estimates confidence intervals for selected thawed parameters.
prompt	Changes the Sherpa prompt.
ptsrc1d	A 1−D file−based point−source fitting model.
ptsrc2d	A 2−D file−based point−source fitting model.
read	Inputs the contents of one or more files.
record	Controls output of parameters values and statistics to an ASCII file.
region−projection	Creates a contour plot of confidence regions using the PROJECTION algorithm. The commands REG−PROJ and REGPROJ are abbreviated equivalents.
region−uncertainty	Creates a contour plot of confidence regions using the UNCERTAINTY algorithm. The commands REG−UNC and REGUNC are abbreviated equivalents.
rename	Changes the name that has been given to a model component by the user.
reset	Restores settings and/or parameter values.
restore_paramest	Module functions to restore the default values of the parameters used to configure each Sherpa parameter estimation method.
rsp2d	A 2−D instrument model utilizing an exposure map and point−spread function.
rsp	A 1−D instrument model.
run_fit	Module function to fit datasets, and retrieve information about the final fit.
run_paramestint	Module functions to display statistics as a function of parameter value, and to retrieve the value and statistic arrays
run_paramestlim	Module functions to determine confidence intervals, and retrieve the parameter bounds.
run_paramestreg	Module functions to display contours of statistics as a function of parameter values, and to retrieve arrays of values and statistics
run_paramest	Module functions to run parameter estimation routines and retrieve information.
run	Summary of Sherpa/S−Lang module run functions.
save_state	The save_state() function
save	Saves information to an ASCII file.
schechter	Schechter function. Integration OFF.

setback	Sets attributes of a background dataset.
setdata	Sets attributes of a source dataset.
setplot	An alternative interface to the Sherpa plot customisation variables
set_analysis	Module function to set the units for 1–D spectral analysis.
set_axes	Module functions for creating new source or background dataspaces.
set_backscale	Module functions for setting source and background extraction region areas.
set_coord	Module function to set the coordinate system for 2–D image analysis.
set_dataspace	An alternative form of DATASPACE and set_axes().
set_data	Set source and background data using the S–lang module function in Sherpa.
set_dir	Module function to change directories.
set_erroff	Switch off the drawing of error bars in Sherpa plots
set_erron	Switch on the drawing of error bars in Sherpa plots
set_errors	Module functions for assigning source and background error estimates.
set_exptime	Module functions for setting source and background exposure times.
set_filter	Module functions for assigning source and background filters from an array.
set_groups	Module functions for grouping and setting quality to source and background files from an array.
set_ignore2d	Module functions for ignoring source and background regions in 2–D datasets.
set_ignore_all	Module functions for ignoring all source and background data.
set_ignore_bad	Module functions for ignoring bad (quality > 0) channels.
set_ignore	Module functions for ignoring source and background filter ranges in 1–D datasets.
set_lin	Switch plotting axis to linear scale in Sherpa plots
set_log	Switch axis to log in Sherpa plots
set_notice2d	Module functions for noticing source and background regions in 2–D datasets.
set_notice_all	Module functions for noticing all source and background data.
set_notice	Module functions for noticing source and background filter ranges in 1–D datasets.
set_paramset	Module functions to thaw or freeze the specified parameter(s).
set_par	Module function for setting model parameter values, etc.
set_stackexpr	Module functions to set a model stack expression.
set_subtract	Module functions to subtract background from a source dataset, or to undo subtraction.
set_syserrors	

	Module functions for assigning source and background systematic error estimates.
set_verbose	Module function to reset Sherpa's verbosity.
set_weights	Module functions for assigning source and background statistical weights.
set	Summary of Sherpa/S–Lang module functions that change settings or data.
sherpa–module	Accessing data with the Sherpa/S–Lang module functions.
sherpa–plot–hooks	Customizing Sherpa plots using the prefunc and postfunc fields of Sherpa State Objects
sherpa.cov	Configure COVARIANCE in Sherpa.
sherpa.dataplot	Configure appearance of Sherpa plots.
sherpa.fitplot	Configure appearance of Sherpa plots.
sherpa.intproj	Configure INTERVAL–PROJECTION in Sherpa.
sherpa.intunc	Configure INTERVAL–UNCERTAINTY in Sherpa.
sherpa.multiplot	Configure appearance of Sherpa plots.
sherpa.output	A Description of the sherpa.output State Object
sherpa.plot	Configure appearance of Sherpa plots.
sherpa.proj	Configure PROJECTION in Sherpa.
sherpa.regproj	Configure REGION–PROJECTION in Sherpa.
sherpa.regunc	Configure REGION–UNCERTAINTY in Sherpa.
sherpa.resplot	Configure appearance of Sherpa plots.
sherpa.unc	Configure UNCERTAINTY in Sherpa.
sherpa_eval	Call Sherpa commands from S–Lang
sherpa_plotfns	Plot–related functions for Sherpa.
sherpa_utils	Provides a number of S–Lang functions useful for Sherpa.
sherpa	Command summary of Sherpa, CIAO's modeling and fitting engine.
shexp10	Exponential function, base 10. Integration OFF.
shexp	Exponential function. Integration OFF.
shlog10	Logarithm function, base 10. Integration OFF.
shloge	Natural logarithm function. Integration OFF.
show	Reports current status.
sigma–rejection	The SIGMA–REJECTION optimization method for fits to 1–D data. Alternate names are SIG–REJ and SR.
simplex	A simplex optimization method.
simspec	Create and fit a simulated PHA spectrum.
simul–ann–1	A simulated annealing search, with one parameter varied at each step.
simul–ann–2	A simulated annealing search, with all parameters varied at each step.
simul–pow–1	A combination of SIMUL–ANN–1 with POWELL.
simul–pow–2	A combination of SIMUL–ANN–2 with POWELL.

sin	Sine function. Integration OFF.
source	Defines the source model expression to be used for fitting a dataset. The command SRC is an abbreviated equivalent.
splot	Causes the specified 2–D data to be displayed, with a surface plot, via ChIPS.
sqrt	Square root function. Integration OFF.
staterrors	Defines an expression or file to be used to specify the statistical errors for source data.
statistic	Specifies the fitting statistic.
stephi1d	1–D step function. Integration OFF. The command HIGHPASS is equivalent.
steplo1d	1–D step function. Integration OFF. The command LOWPASS is equivalent.
subtract	Performs background subtraction.
syserrors	Defines an expression or file to be used to specify the systematic errors for source data.
tan	Tangent function. Integration OFF.
thaw	Allows model parameter(s) to vary.
tpsf1d	A 1–D TCD–model–based instrument model.
tpsf	A 2–D TCD–model–based instrument model.
truncate	Resets negative model amplitudes to zero.
uncertainty	Estimates confidence intervals for selected thawed parameters.
ungroup	Causes Sherpa to undo a grouping scheme that had been applied to source or background data.
unlink	Removes a link between model parameters.
unsubtract	Undoes background subtraction.
usermethod	A user–defined method.
usermodel	User implemented model. Integration OFF.
userstat	User implemented statistic.
use	Calls and executes a Sherpa script.
version	Reports the Sherpa version that is in use.
write	Causes the specified information to be written to the screen or to a file.
xsabsori	Ionized absorber. XSpec model.
xsacisabs	Decay in the ACIS quantum efficiency. XSpec model.
xsapec	APEC thermal plasma model. XSpec model.
xsbapec	APEC thermal plasma model with velocity broadening as a free parameter. XSpec model.
xsbbodyrad	Blackbody spectrum with norm proportional to surface area. XSpec model.
xsbbody	Blackbody spectrum. XSpec model.
xsbexrav	E–folded broken power law reflected from neutral matter. XSpec model.

xsbexriv	E–folded broken power law reflected from ionized matter. XSpec model.
xsbknpower	Broken power law. XSpec model.
xsbmc	Comptonization by relativistically moving matter. XSpec model.
xsbremss	Thermal bremsstrahlung. XSpec model.
xsbvapec	APEC thermal plasma model with variable abundances and velocity broadening as a free parameter. XSpec model.
xsc6mekl	6th–order Chebyshev polynomial DEM using mekal. XSpec model.
xsc6pmekl	Exponential of 6th–order Chebyshev polyn. DEM using mekal. XSpec model.
xsc6pvmkl	Variable abundance version of c6pmekl. XSpec model.
xsc6vmekl	Variable abundance version of c6mekl. XSpec model.
xscabs	Compton scattering (non–relativistic). XSpec model.
xscemekl	Multi–temperature mekal. XSpec model.
xscevmkl	Multi–temperature vmeka. XSpec model.
xscflow	Cooling flow model. XSpec model.
xscompbb	Comptonized blackbody spectrum after Nishimura et al. (1986). XSpec model.
xscompls	Comptonization spectrum after Lamb and Sanford (1979). XSpec model.
xscompst	Comptonization spectrum after Sunyaev and Titarchuk (1980). XSpec model.
xscomptt	Comptonization spectrum after Titarchuk (1994). XSpec model.
xsconstant	Energy–independent multiplicative factor. XSpec model.
xscutoffpl	Power law with high energy exponential cutoff. XSpec model.
xscyclabs	Cyclotron absorption line. XSpec model.
xsdiskbb	Multiple blackbody disk model. XSpec model.
xsdiskline	Line emission from relativistic accretion disk. XSpec model.
xsdiskm	Disk model with gas pressure viscosity. XSpec model.
xsdisko	Modified blackbody disk model. XSpec model.
xsdiskpn	Accretion disk around a black hole. XSpec model.
xsdisk	Disk model. XSpec model.
xsdust	Dust scattering out of the beam. XSpec model.
xsedge	Absorption edge. XSpec model.
xsequil	Equilibrium ionization collisional plasma model from Borkowski. XSpec model.
xsexpabs	Low–energy exponential cutoff. XSpec model.
xsexpdec	An exponential decay. XSpec model.
xsexpfac	Exponential factor. XSpec model.
xsgabs	A multiplicative gaussian absorption line. XSpec model.

xsgaussian	Simple gaussian line profile. XSpec model.
xsgnei	Generalized single ionization NEI plasma model. XSpec model.
xsgrad	GR accretion disk around a black hole. XSpec model.
xsgrbm	Gamma–ray burst model. XSpec model.
xshighecut	High energy cutoff. XSpec model.
xshrefl	Simple reflection model good up to 15 keV. XSpec model.
xslaor	Line from accretion disk around a black hole. XSpec model.
xslorentz	Lorentzian line profile. XSpec model.
xsmekal	Mewe–Kaastra–Liedahl thermal plasma (1995). XSpec model.
xsmeka	Mewe–Gronenschild–Kaastra thermal plasma (1992). XSpec model.
xsmkcflow	Cooling flow model based on mekal. XSpec model.
xsnei	Simple nonequilibrium ionization plasma model. XSpec model.
xsnotch	Notch line absorption. XSpec model.
xsnpshock	Plane–parallel shock with ion and electron temperatures. XSpec model.
xsnsa	Spectra in the X–ray range (0.05–10 keV) emitted from a hydrogen atmosphere of a neutron star. XSpec model.
xsnteea	Pair plasma model. XSpec model.
xspcfabs	Partial covering fraction absorption. XSpec model.
xspecabundan	Performs the XSPEC command abund.
xspecxsect	Performs the XSPEC command xsect.
xspegpwrlw	Power law with pegged normalization. XSpec model.
xspexrav	Exponentially cutoff power law reflected from neutral matter. XSpec model.
xspexriv	Exponentially cutoff power law reflected from ionized matter. XSpec model.
xsphabs	Photo–electric absorption. XSpec model.
xsplabs	Absorption model with power law dependence on energy. XSpec model.
xsplcabs	Cutoff power law observed through dense, cold matter. XSpec model.
xsposm	Positronium continuum. XSpec model.
xspowerlaw	Simple photon power law. XSpec model.
xspshock	Constant temperature, plane–parallel shock plasma model. XSpec model.
xspwab	An extension of partial covering fraction absorption into a power–law distribution of covering fraction. XSpec model.
xsraymond	Raymond–Smith thermal plasma. XSpec model.
xsredden	IR/optical/UV extinction from Cardelli et al. (1989). XSpec model.

xsredge	Recombination edge. XSpec model.
xsrefsch	E–folded power law reflected from an ionized relativistic disk. XSpec model.
xssedov	Sedov model with electron and ion temperatures. XSpec model.
xssmedge	Smoothed absorption edge. XSpec model.
xsspline	Spline multiplicative factor. XSpec model.
xssrcut	Synchrotron radiation from cutoff electron distribution. XSpec model.
xssresc	Synchrotron radiation from escape–limited electron distribution. XSpec model.
xsssssice	Einstein Observatory SSS ice absorption. XSpec model.
xsstep	Step function convolved with gaussian. XSpec model.
xstbabs	Calculates the absorption of X–rays by the ISM. XSpec model.
xstbgrain	Calculates the absorption of X–rays by the ISM with variable hydrogen to H2 ratio and grain parameters. XSpec model.
xstbvarabs	Calculates the absorption of X–rays by the ISM, allowing user to vary all abundances, depletion factors, and grain properties. XSpec model.
xsuvred	UV reddening. XSpec model.
xsvapec	APEC thermal plasma model with variable abundances. XSpec model.
xsvarabs	Photoelectric absorption with variable abundances. XSpec model.
xsvbremss	Thermal bremsstrahlung spectrum with variable H/He. XSpec model.
xsvequil	Ionization equilibrium collisional plasma model with variable abundances. XSpec model.
xsvgnei	Non–equilibrium ionization collisional plasma model with variable abundances. XSpec model.
xsvmcflow	Cooling flow model based on vmekal. XSpec model.
xsvmekal	M–K–L thermal plasma with variable abundances. XSpec model.
xsvmeka	M–G–K thermal plasma with variable abundances. XSpec model.
xsvnei	Non–equilibrium ionization collisional plasma model with variable abundances. XSpec model.
xsvnpshock	Plane–parallel shock plasma model with separate ion and electron temperatures and variable abundances. XSpec model.
xsvphabs	Photoelectric absorption with variable abundances. XSpec model.
xsvpshock	Constant temperature, plane–parallel shock plasma model with variable abundances. XSpec model.
xsvraymond	

		Raymond–Smith thermal plasma with variable abundances. XSpec model.
	xsvsedov	Sedov model with separate ion and electron temperatures and variable abundances. XSpec model.
	xswabs	Photoelectric absorption (Morrison and McCammon). XSpec model.
	xswndabs	Photoelectric absorption with low energy window. XSpec model.
	xsxion	Reflected spectra of a photo–ionized accretion disk or ring. XSpec model.
	xszbbody	Redshifted blackbody. XSpec model.
	xszbremss	Redshifted thermal bremsstrahlung. XSpec model.
	xszedge	Redshifted absorption edge. XSpec model.
	xszgauss	Redshifted gaussian. XSpec model.
	xszhighect	Redshifted high energy cutoff. XSpec model.
	xszpcfabs	Redshifted partial covering absorption. XSpec model.
	xszphabs	Redshifted photoelectric absorption. XSpec model.
	xszpowerlw	Redshifted power law. XSpec model.
	xsztbabs	Calculates the absorption of X–rays by the ISM for modeling redshifted absorption. Does not include a dust component. XSpec model.
	xszvarabs	Redshifted photoelectric absorption with variable abundances. XSpec model.
	xszvfeabs	Redshifted absorption with variable iron abundance. XSpec model.
	xszvphabs	Redshifted photoelectric absorption with variable abundances. XSpec model.
	xszwabs	Redshifted "Wisconsin absorption." XSpec model.
	xszwndabs	Redshifted photoelectric absorption with low energy window. XSpec model.
	xs	XSpec model functions.
slang	math	Mathematical operations in S–Lang.
	overview	Overview of the S–Lang programing language
	slang	The S–Lang programing language
	slsh	Evaluate and run S–Lang code.
	tips	S–Lang tips and example functions
	usermodel	Creating Sherpa Usermodels with S–Lang
	variables	Variables in S–Lang
slangrtl	abs	Compute the absolute value of a number
	acosh	Compute the inverse cosh of an number
	acos	Compute the arc–cosine of an number

append_to_slang_load_path	Append a directory to the load–path
array_info	Returns information about an array
array_map	Apply a function to each element of an array
array_sort	Sort an array
array_to_bstring	Convert an array to a binary string
asinh	Compute the inverse–sinh of an number
asin	Compute the arc–sine of an number
assoc_delete_key	Delete a key from an Associative Array
assoc_get_keys	Return all the key names of an Associative Array
assoc_get_values	Return all the values of an Associative Array
assoc_key_exists	Check to see whether a key exists in an Associative Array
atanh	Compute the inverse–tanh of an number
atan	Compute the arc–tangent of an number
atof	Convert a string to a double precision number
autoload	Load a function from a file
bstring_to_array	Convert a binary string to an array of characters
bstrlen	Get the length of a binary string
byte_compile_file	Compile a file to byte–code for faster loading.
char	Convert an ascii value into a string
chdir	Change the current working directory.
chmod	Change the mode of a file
chown	Change the owner of a file
clearerr	Clear the error of a file stream
close	Close an open file descriptor
Conj	Compute the complex conjugate of a number
cosh	Compute the hyperbolic cosine of an number
cos	Compute the cosine of an number
create_delimited_string	Concatenate strings using a delimiter
ctime	Convert a calendar time to a string
cumsum	Compute the cumulative sum of an array
current_namespace	Get the name of the current namespace
define_case	Define upper–lower case conversion.
double	Convert an object to double precision
dup_fd	Duplicate a file descriptor
dup	Duplicate the value at the top of the stack
errno_string	Return a string describing an errno.
errno	Error code set by system functions.
error	Generate an error condition
evalfile	Interpret a file containing S–Lang code.
eval	Interpret a string as S–Lang code
exch	Exchange two items on the stack

exp	Compute the exponential of an number
extract_element	Extract the nth element of a string with delimiters
fclose	Close a file
fdopen	Convert a FD_Type file descriptor to a stdio File_Type object
feof	Get the end–of–file status
ferror	Determine the error status of an open file descriptor
fflush	Flush an output stream
fgetslines	Read all the lines from an open file
fgets	Read a line from a file.
fileno	Convert a stdio File_Type object to a FD_Type descriptor
fopen	Open a file
fprintf	Create and write a formatted string to a file
fputs	Write a string to an open stream
fread	Read binary data from a file
fseek	Reposition a stream
ftell	Obtain the current position in an open stream
fwrite	Write binary data to a file
getcwd	Get the current working directory
getegid	Get the effective group id
getenv	Get the value of an environment variable
geteuid	Get the effective user–id of the current process
getgid	Get the group id
getpid	Get the current process id
getppid	Get the parent process id
getuid	Get the user–id of the current process
get_doc_string_from_file	Read documentation from a file
get_import_module_path	Get the search path for dynamically loadable objects
get_slang_load_path	Get the value of the interpreter's load–path
get_struct_field_names	Retrieve the field names associated with a structure
get_struct_field	Get the value associated with a structure field
gmtime	Break down a time in seconds to GMT timezone
Imag	Compute the imaginary part of a number
implements	Name a private namespace
import	Dynamically link to a specified module
init_char_array	Initialize an array of characters
integer	Convert a string to an integer
int	Typecast an object to an integer
isatty	Determine if an open file descriptor refers to a terminal
isdigit	Tests for a decimal digit character
is_defined	Indicate whether a variable or function defined.
is_list_element	Test whether a delimited string contains a specific element

is_struct_type	Determine whether or not an object is a structure
is_substr	Test for a specified substring within a string.
kill	Send a signal to a process
length	Get the length of an object
listdir	Get a list of the files in a directory
localtime	Break down a time in seconds to local timezone
log10	Compute the base–10 logarithm of an number
log	Compute the logarithm of an number
lseek	Reposition a file descriptor's file pointer
lstat_file	Get information about a symbolic link
make_printable_string	Format a string suitable for parsing
max	Get the maximum value of an array
message	Print a string onto the message device
min	Get the minimum value of an array
mkdir	Create a new directory
mkfifo	Create a named pipe
mul2	Multiply a number by 2
open	Open a file
pack	Pack objects into a binary string
pad_pack_format	Add padding to a pack format
path_basename	Get the basename part of a pathname
path_concat	Combine elements of a pathname
path_dirname	Get the directory name part of a pathname
path_extname	Return the extension part of a pathname
path_get_delimiter	Get the value of a search–path delimiter
path_is_absolute	Determine whether or not a pathname is absolute
path_sans_extname	Strip the extension from a pathname
pclose	Close an object opened with popen
polynom	Evaluate a polynomial
popen	Open a process
pop	Discard an item from the stack
prepend_to_slang_load_path	Prepend a directory to the load–path
printf	Create and write a formatted string to stdout
provide	Declare that a specified feature is available
putenv	Add or change an environment variable
readlink	Get the value of a symbolic link
read	Read from an open file descriptor
Real	Compute the real part of a number
remove	Delete a file
rename	Rename a file
require	Make sure a feature is present, and load it if not

reshape	Reshape an array
rmdir	Remove a directory
setgid	Set the group–id of the current process
setpgid	Set the process group–id
setuid	Set the user–id of the current process
set_float_format	Set the format for printing floating point values.
set_import_module_path	Set the search path for dynamically loadable objects
set_slang_load_path	Set the value of the interpreter's load–path
set_struct_fields	Set the fields of a structure
set_struct_field	Set the value associated with a structure field
shift	Shift the elements of a 1–d array
sign	Compute the sign of a number
sinh	Compute the hyperbolic sine of an number
sin	Compute the sine of an number
sizeof_pack	Compute the size implied by a pack format string
slangrtl	S–Lang Run Time Library
sleep	Pause for a specified number of seconds
sprintf	Format objects into a string
sqrt	Compute the square root of an number
sqr	Compute the square of a number
sscanf	Parse a formatted string
stat_file	Get information about a file
stat_is	Parse the st_mode field of a stat structure
strcat	Concatenate strings
strchopr	Chop or split a string into substrings.
strchop	Chop or split a string into substrings.
strcmp	Compare two strings
strcompress	Remove excess whitespace characters from a string
string_match_nth	Get the result of the last call to string_match
string_match	Match a string against a regular expression
string	Convert an object to a string representation.
strjoin	Concatenate elements of a string array
strlen	Compute the length of a string
strlow	Convert a string to lowercase
strncmp	Compare the first few characters of two strings
strreplace	Replace one or more substrings
strsub	Replace a character with another in a string.
strtok	Extract tokens from a string
strtrans	Replace characters in a string
strtrim_beg	Remove leading whitespace from a string
strtrim_end	Remove trailing whitespace from a string

strtrim	Remove whitespace from the ends of a string
strup	Convert a string to uppercase
str_delete_chars	Delete characters from a string
str_quote_string	Escape characters in a string.
str_replace	Replace a substring of a string
str_uncomment_string	Remove comments from a string
substr	Extract a substring from a string
sum	Sum over the elements of an array
system	Execute a shell command
tanh	Compute the hyperbolic tangent of an number
tan	Compute the tangent of an number
tic	Start timing
times	Get process times
time	Return the current data and time as a string
toc	Get elapsed CPU time
tolower	Convert a character to lowercase.
toupper	Convert a character to uppercase.
transpose	Transpose an array
typecast	Convert an object from one data type to another.
typeof	Get the data type of an object.
umask	Set the file creation mask
uname	Get the system name
unpack	Unpack Objects from a Binary String
usage	Generate a usage error
use_namespace	Change to another namespace
verror	Generate an error condition
vmessage	Print a formatted string onto the message device
where	Get indices where an integer array is non−zero
write	Write to an open file descriptor
_apropos	Generate a list of functions and variables
_auto_declare	Set automatic variable declaration mode
_clear_error	Clear an error condition
_debug_info	Configure debugging information
_featurep	Test whether or not a feature is present
_function_name	Returns the name of the currently executing function
_get_namespaces	Returns a list of namespace names
_isnull	Check array for NULL elements
_NARGS	The number of parameters passed to a function
_pop_n	Remove objects from the stack
_print_stack	print the values on the stack.
_push_struct_field_values	Push the values of a structure's fields onto the stack

	reshape	Copy an array to a new shape
	slangtrace	Turn function tracing on or off.
	slang_doc_dir	Installed documentation directory
	slang_guess_type	Guess the data type that a string represents.
	slang_version_string	The S–Lang library version number as a string
	slang_version	The S–Lang library version number
	stkdepth	Get the number of objects currently on the stack.
	stk_reverse	Reverse the order of the objects on the stack.
	stk_roll	Roll items on the stack
	time	Get the current time in seconds
	traceback	Generate a traceback upon error
	trace_function	Set the function to trace
	typeof	Get the data type of an object
	class_id	Return the class–id of a specified type
	class_type	Return the class–type of a specified type
	eqs	Test for equality between two objects
	get_defined_symbols	Get the symbols defined by the preprocessor
	get_reference	Get a reference to a global object
	is_initialized	Determine whether or not a variable has a value
	pop_args	Remove n function arguments from the stack
	push_args	Remove n function arguments onto the stack
	uninitialize	Uninitialize a variable
stackio	stk_append	Add an item (or items) to a stack.
	stk_build	Build a stack from text input (file name or regular expression).
	stk_change_current	Change the current element of a stack.
	stk_change_num	Change the given element of a stack.
	stk_close	Close a stack.
	stk_count	Returns the number of elements in a stack.
	stk_current	Returns the current position of the stack.
	stk_delete_current	Deletes the current element from the stack.
	stk_delete_num	Deletes the specified element from the stack.
	stk_disp	Display the contents of a stack.
	stk_expand_n	Create a numbered stack of N elements.
	stk_read_next	Returns the next element from a stack.
	stk_read_num	Returns the selected element from a stack.
	stk_rewind	Rewind the stack to its starting position.
	stk_set_current	Sets the current position of the stack.
tools	acisreadcorr	Flag and relocate out–of–time CCD events
	acisspec	(1) Extract source and/or background ACIS imaging spectra

	for both point–like and extended sources; (2) coadd ACIS spectra; (3) build weighted ARFs and RMFs.
acis_bkgrnd_lookup	Find the matching ACIS "blank–sky" dataset for an observation
acis_build_badpix	Create an observation–specific bad–pixel file
acis_classify_hotpix	Determine whether a suspicious pixel is hot or affected by an afterglow.
acis_detect_afterglow	Identify and flag cosmic ray afterglows (L1)
acis_expmap	Generate ACIS exposure maps for celldetect recursive blocking
acis_fef_lookup	Find the FITS Embedded Function file for use by mkrmf
acis_find_hotpix	Identify pixels that have an unusually large (or small) number of events.
acis_process_events	Produce or update TIME, coordinates, PI, GRADE, and STATUS information in ACIS event files
acis_run_hotpix	Identify and flag "hot" pixels and cosmic–ray "afterglows."
acis_set_ardlib	Sets the bad pixel paremeters of the ardlib file to the given file
aconvolve	Convolve an N–dimensional image with a kernel
acrosscorr	Cross–correlate two N–dimensional images or autocorrelate an image
add_grating_orders	Add positive and negative diffraction orders of a grating PHA spectra and the corresponding ARFs
add_grating_spectra	Add two, source and background, grating PHA spectra, averages the corresponding ARFs, and group the coadded spectrum.
analyze_ltcrv	analyze_ltcrv.sl – An algorithm for cleaning lightcurves
apowerspectrum	Compute the power spectrum of an N–dimensional input array, or from two columns (independent/dependent variable) in an input file
arestore	restore image resolution using deconvolution techniques
ascii2fits	Convert simple ASCII files to FITS format files
asphist	Bin the aspect solution into a 3D histogram of duration vs pointing offset and roll offset.
axbary	Apply barycentric corrections to event times, GTIs, and header values.
celldetect	Use a "sliding cell" to search for sources
chart_spectrum	chart_spectrum.sl – Create a source spectrum for input to ChaRT
color_image	Run dmimg2jpg to make a color JPEG image
csmooth	Adaptively smooth a 2–D image
destreak	Remove streak events from ACIS data
dither_region	Compute fraction of region area that covers chips
dmappend	Append multiple blocks/extensions to an existing output file.
dmarfadd	Add multiple ARF files together, weighting by exposure.

dmcontour	Make contour regions from a 2–D image
dmcoords	Convert between Chandra coordinate systems
dmcopy	Filter and bin tables and images.
dmdiff	Compare values in two FITS files.
dmextract	Make a histogram table file (e.g. PHA file, lightcurve file) from a table column. Generate count histogram on supplied regions for a spatial table or image file.
dmfilth	Replace pixel values in source regions of an image with values interpolated from surrounding background regions
dmgroupreg	Translates DS9 regions and groups to CIAO format.
dmgroup	Group a specified column in a table with various options
dmgti	Create a Good Time Interval from limits placed on time–based files (event and Mission Time Line files).
dmhedit	Edit datamodel file headers
dmhistory	Extracts the tool history from the file header
dmimg2jpg	Make a color JPEG image (or EPS) from three image files
dmimgcalc	Perform arithmetic on images
dmimghist	Create a histogram of the pixel values in a 2–D image
dmimgpick	Maps image values to rows in a table
dmimgthresh	Set low (or high) intensity or exposure regions of an image to a constant value.
dmjoin	Join two files whose sampling is different.
dmkeypar	Retrieve information about a keyword from an input file
dmlist	List contents or structure of a file.
dmmakepar	Write header keywords to a parameter file
dmmakereg	Convert a region string to a CXC FITS region file
dmmerge	Merge two or more compatible tables (e.g. event files) into one.
dmpaste	Add new columns to a table.
dmreadpar	Create header keywords from a parameter file
dmregrid	Rebin a stack of 2–dimensional images.
dmsort	Sort a table.
dmstat	Compute statistics for images and columns in tables.
dmtcalc	Modify and create columns in a table
dmtype2split	Create a type 1 output file for specified rows of a type 2 file.
fullgarf	Create a grating arf for a particular order and grating for a given observation.
get_sky_limits	Find the region covered by an image in sky coordinates
get_src_region	Outputs regions that have counts higher than background
hrc_build_badpix	Create observation–specific bad pixel file for HRC
hrc_dtfstats	Calculate statistics from hrc_calc_dead_time and a GTI
hrc_process_events	Correct HRC event positions, times, PHA, etc.
lc_clean	

	lc_clean.sl – Clean a lightcurve to match the ACIS "blank–sky" datasets
lightcurve	Bin source and background events by specified time intervals
merge_all	Combine any number of observations. If desired, create corresponding exposure maps and exposure–corrected images.
mkacisrmf	Generate an RMF for Chandra imaging data
mkarf	Generate an ARF for Chandra imaging data (and grating 0–th order)
mkbgreg	Creates a stack of background regions for a given source list
mkexpmap	Generate a Chandra imaging exposure map (effective area vs. sky position)
mkgarf	Generate a Chandra Grating ARF for one detector element.
mkgrmf	Generate an RMF for Chandra grating data
mkinstmap	Generate a Chandra instrument map (effective area vs. detector position)
mkoif	Create an OIF (observation index file) for a dataset
mkpsf	Extract a point spread function (PSF) from the PSF library.
mkrmf	Generate an RMF for Chandra imaging data
mksubbgreg	Subtracts source regions from background regions
mkwarf	Generate a weighted ARF for Chandra ACIS data
monitor_photom	Generate a photometric lightcurve from ACA monitor data
mtl_build_gti	Create Good Time Interval from Mission Time Line and limits
paccess	Gives the location of a tool's parameter file
pdump	Dumps the values of the parameters, with one parameter per line.
pget	Get parameter values
pline	Display the parameter values, with all parameters on a single line
plist	List the contents of a parameter file.
pquery	Query whether a parameter exists and, if so, get and return its value.
pset	Set parameter values on the command line
psextract	Extract source and background ACIS spectra for point–like sources and build associated ARFs and RMFs.
psf_project_ray	Project rays generated by ChaRT onto a semi–infinite detector plane.
punlearn	Restore system default parameter values
quizcaldb	Find calibration files in the CALDB
regions	regions.sl – Display the FEF regions covered by a source
reproject_aspect	Modify aspect solution file RA, Dec, and roll or updfile WCS to minimize position differences between two source lists.
reproject_events	Regrid an event file (or stack) to a common tangent point
reproject_image_grid	Projects image from one WCS to another

Contextual list of Ahelp files for CIAO 3.4

reproject_image	Projects image from one WCS to another
rmfimg	Create a simple image of an RMF or ARF file
show_tgain_corr	Display the time–dependent gain correction for a given chip location.
show_wgt	show_wgt.sl – Examine the weights file created by mkwarf
skyfov	Make a field–of–view region in sky coordinates
specextract	Extract source and background ACIS spectra for point–like and extended sources and build associated WMAPs, ARFs and RMFs.
spectrum	spectrum.sl – Calculate spectral weights for creating an instrument map
sso_freeze	Reproject sky x and y coordinates into a Solar System Object centered reference frame.
sstats	sstats.sl– Calculate statistics of images
stk_build	stk_build is used to build a stack from a user–defined string. The output is to stdout, stderr, or a file.
stk_count	stk_count is used to count the number of items in a stack. The output is to the stk_count parameter file.
stk_read_num	stk_read_num is used to read one element of a user–defined stack. The output is to the outelement field of the stk_read_num parameter file.
stk_where	stk_where finds the location of an item in a stack
syntax	Syntax used for writing mathematical operations in dmtcalc, dmimgcalc, and dmgti
tgdetect	Detect and centroid the zero order image in a spatial sub–region of a grating event list.
tgextract	Bin event list grating wavelengths column into a one–dimensional counts histogram, by source, grating part, and diffraction order.
tgidselectsrc	Filter an input source list based upon a SNR threshold.
tgmatchsrc	Match sources between two source lists by their 2–D separation files.
tg_bkg	Creates a PHA background file for use in XSPEC
tg_create_mask	Create a region file to define spectrum sky boundaries
tg_osort_img	Create an image that shows the density of events in different orders
tg_resolve_events	Assign grating events to spectral orders; use detector energy resolution for order separation, if available.
tg_scale_reg	Create a region file that labels distances along the grating arm
vtpdetect	Voronoi Tessellation and Percolation (VTP) source detection
wavdetect	Mexican–Hat Wavelet source detection (wtransform+wrecon)
wcs_match	Create a frame transformation to minimize the aspect difference between data from the same sky region.
wcs_update	Modify aspect solution file RA, Dec, and roll or infile WCS based on a transformation matrix.

	wrecon	Combine wavelet transform coefficients into sources
	wtransform	Obtain wavelet transform coefficients for poisson image data

varmm	apropos	S–Lang function to find defined symbols (functions and variables)
	clearstack	S–Lang function to clear the stack.
	dup_struct	S–Lang function to duplicate a structured variable.
	fits_bitpix	Converts S–Lang variables and data types to FITS BITPIX values.
	is_struct_defined	S–Lang function to see if a structure or field in a structure is defined
	print	S–Lang functions to print variables, arrays and structures
	readarf	S–Lang function to read an Ancillary Response File (ARF)
	readascii	S–Lang function to read in an ASCII data file.
	readbintab	S–Lang function to read a FITS binary table.
	readfile	S–Lang functions to read a data file into a S–Lang variable.
	readimage	S–Lang function to read a FITS image.
	readpha	S–Lang function to read a spectrum in PHA format (both type I and II)
	readrdb	S–Lang function to read a file in RDB format.
	readrmf	S–Lang function to read a Redistribution Matrix File (RMF)
	reverse	S–Lang function to reverse a 1D array
	set_state_defaults	Restore the default values of state (configuration) variables
	set_state	Set multiple fields of a state (configuration) variable
	username	Returns the login name of the owner of the current process.
	writeascii	S–Lang function to create an ASCII output file from S–Lang arrays
	writefits	S–Lang function to create a FITS output file.
	__exit	S–Lang function to abort applications.

varmmrl	varmm_rl_blink	Sets the "blink" mode of the ChIPS and Sherpa command line.
	varmm_rl_editmode	Set the editing mode ("emacs" or "vi") of the ChIPS and Sherpa command line.

xpa	slxpa_errno	This variable stores the error status for the XPA commands.
	xpaaccess	Find out how many XPA access points are available that match a given name.
	xpaclose	Close a persistent XPA connection created by XPAOpen().
	xpagetb	Retrieve binary data from one or more XPA servers.
	xpagettofile	Retrieve data from one or more XPA servers and write the results to a file.
	xpaget	Retrieve data from one or more XPA servers.

xpaopen	Create a persistent connection for XPA calls.
xpaset	Send a command or data to one or more XPA servers.
xpa_maxhosts	This variable stores the maximum number of hosts to be contacted in a XPA call.
xpa_version	The version of the XPA library being used by the S−Lang module.
slxpa_version	The version of the S−Lang XPA module.

URL:
http://cxc.harvard.edu/ciao3.4/index_context.html
Last modified: 23 March 2007

Contextual list of Ahelp files for CIAO 3.4