



AHELP for CIAO 3.4

source

 Context: [sherpa](#)

Jump to: [Description](#) [Examples](#) [Bugs](#) [See Also](#)

Synopsis

Defines the source model expression to be used for fitting a dataset. The command SRC is an abbreviated equivalent.

Syntax

```
sherpa> SOURCE [<dataset range> | ALLSETS] = <modelExpr>
```

<dataset range> = # (or more generally #:#,#:#, etc.) such that # specifies a dataset number and #:# represents an inclusive range of datasets; one may specify multiple inclusive ranges by separating them with commas. The default dataset is dataset 1.

Description

The model expression, <modelExpr>, is an algebraic combination of one or more of the following elements:

```
{<sherpa_modelname> | <sherpa_modelname>[<modelname> | <modelname> | <model_stack> | <nested_model>}
```

along with numerical values. The following operators are recognized: + - * / () { }. See the CREATE command for further information.

Note that:

- The documentation on Sherpa Models contains a summary list, and descriptions, of the models that are available within Sherpa, which include models from XSPEC, v. 11.3.
- By default, if the model expression includes a model component that has not previously been established, Sherpa will prompt for the initial parameter values for that model component. This prompting can be turned off using the PARAMPROMPT OFF command.

To reset a source model stack, issue the command:

```
sherpa> SOURCE [<dataset range> | ALLSETS] =
```

Example 1

Establish a model component, and define it as the source model to be used for fitting a specific dataset:

```
sherpa> SOURCE 2 = GAUSS
GAUSS.fwhm parameter value [10]
GAUSS.pos parameter value [0]
GAUSS.ampl parameter value [1]
sherpa>
```

This command defines the Sherpa model GAUSS as the source model to be used for fitting dataset number 2. Note that the user accepted the given initial values for all of the parameters, using the <RETURN> key.

Example 2

Establish a model component and assign it a name; define the model component as the source model to be used for fitting a specific dataset:

```
sherpa> PARAMPROMPT OFF
Model parameter prompting is off
sherpa> GAUSS[modelb]
sherpa> SOURCE 2 = modelb
```

In the first command, the name modelb is given to the Sherpa model component GAUSS (see the CREATE command for more information on the model language syntax). The second command defines this model as the source model to be used for fitting dataset number 2.

Example 3

Establish model components, and assign them names; create a source model expression to be used for fitting a specific dataset:

```
sherpa> POW[modelc]
sherpa> GAUSS[modelf]
sherpa> SOURCE 1 = modelc + modelf
```

The last command in this series assigns the model expression modelc + modelf as the source model to be used for fitting dataset number 1. The following commands each assign various other model expressions to source models for dataset numbers 2, 3, and 4 respectively:

```
sherpa> SOURCE 2 = 10*(modelc + modelf)
sherpa> SOURCE 3 = (modelc - modelf)/2
sherpa> SOURCE 4 = 0.5*modelc + 0.7*modelf
```

See the CREATE command for further information about creating model expressions.

Example 4

Establish a model component, assign it a name, and define it as the source model to be used for fitting:

```
sherpa> ERASE ALL
sherpa> SOURCE 1 = POW[modelc]
```

In this single command, the name modelc is assigned to the Sherpa model component POW, and then this model is defined as the source model to be used for fitting dataset number 1.

Example 5

Establish model components, assign them names, and use them to define a source model expression to be used for fitting:

```
sherpa> ERASE ALL
sherpa> SOURCE 2 = GAUSS[modelb] + POW[modelc]
```

In this single command, the following is performed: the name modelb is assigned to the Sherpa model component GAUSS; the name modelc is assigned to the Sherpa model component POW; a model expression, which here is the sum of these two models components, is defined as the source model to be used for fitting dataset number 2.

Example 6

Establish a model component, assign it a name, define the parameters, and define it as the source model to be used for fitting:

```
sherpa> ERASE ALL
sherpa> SOURCE = POLY[modela](3.0:1.0:4.0)
```

With this single command, the name modela is assigned to the Sherpa model component POLY, the value of 3.0 is given to the model's first parameter (in this case parameter c0), the minimum of 1.0 is set for this parameter, the maximum of 4.0 is set for this parameter, and then this model is defined as the source model to be used for fitting dataset number 1.

Example 7

Establish multiple model components, assign them names, and use them in a source model expression definition:

```
sherpa> SOURCE 2 = GAUSS[modelb](3:2.5:4.403, 1:-10:10, 1:-3.5:3.5)
+ POW[modelc]
```

With this single command, the following is performed: the name modelb is assigned to the Sherpa model component GAUSS; various parameter values and ranges are set for the parameters of modelb; the name modelc is assigned to the Sherpa model component POW; a model expression, which is the sum of these two models, is defined as the source model to be used for fitting dataset number 2.

Example 8

Establish continuum and line model stacks, and combine these stacks into a source model expression definition:

```
sherpa> PARAMPROMPT OFF
sherpa> CONT = POWLAW1D[modeld]
sherpa> ELINE = NGAUSS[modele]
sherpa> SOURCE = CONT + ELINE
```

In the second and third commands, the names modeld and modele are assigned to the Sherpa model components POWLAW1D and NGAUSS respectively; these model components are then assigned to the user-defined model stacks CONT and ELINE. These model stacks are then assigned to the source model expression.

Bugs

See the [Sherpa bug pages](#) online for an up-to-date listing of known bugs.

See Also

sherpa

[autoest](#), [background](#), [create](#), [create_model](#), [createparamset](#), [fit](#), [freeze](#), [get_defined_models](#),
[get_model_params](#), [get_models](#), [get_num_par](#), [get_par](#), [get_stackexpr](#), [getx](#), [gety](#), [guess](#), [instrument](#),
[integrate](#), [is_paramset](#), [jointmode](#), [kernel](#), [lineid](#), [linkparam](#), [mdl](#), [modeexpr](#), [modelstack](#),
[nestedmodel](#), [noise](#), [paramprompt](#), [paramset](#), [pileup](#), [rename](#), [run_fit](#), [set_par](#), [set_paramset](#),
[set_stackexpr](#), [thaw](#), [truncate](#), [unlink](#)

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/source.html>
Last modified: December 2006