Kst2 command line syntax

Kst 2 will have an entirely new command line syntax, which, while reminiscent of the kst 1 syntax, will be much more powerful.

There are two modes:

Load a kst file:

```
kst [OPTIONS] kstfile
```

`[OPTIONS]` will override the datasource parameters for *all* data sources in the kst file:

```
-F <datasource>
-f <startframe>
-n <numframes>
-s <frames per sample>
-a                          (apply averaging filter: requires -s)
```

Read a data file:

```
kst datasource OPTIONS [datasource OPTIONS []]
```

`OPTIONS` are read and interpreted in order. Except for data object options, all are applied to all future data objects, unless later overridden.

File Options:

```
-f <startframe>             default: 0
-n <numframes>              default: to end of file ("eof")
-s <frames per sample>      default: 0 (read every sample)
-a                          apply averaging filter: requires -s
```

Position:

```
-P <plot name>:             Place curves in one plot.
-A                          Place future curves in individual plots.
```

Appearance

```
-d:                         use points
-l:                         use lines (default)
-b:                         use bargraph
```

Data Object Modifiers

```
-x <field>:                 X axis vector (curves).  Default INDEX
-e <field>:                 Y error flags (curves).  Default none.

-r <rate>:                  sample rate (spectra & spectograms).
```

Data Objects:

```
-y <field>                  plot an XY curve of field.
```

```
-p <field>                  plot the spectrum of field.
-h <field>                  plot a histogram of field.
-z <field>                  plot an image of matrix field.
```

Examples:

Data sources and fields

Plot all data in column 2 from *data.dat*.

```
kst data.dat -y 2
```

Same as above, except only read 20 lines, starting at line 10.

```
kst data.dat -f 10 -n 20 -y 2
```

...also read col 1. One plot per curve.

```
kst data.dat -f 10 -n 20 -y 1 -y 2
```

...instead read col 1 from data2.dat

```
kst data.dat -f 10 -n 20 -y 2 data2.dat -y 1
```

...instead read 40 lines starting at 30 in data2.dat

```
kst data.dat -f 10 -n 20 -y 2 data2.dat -f 30 -n 40 -y 1
```

Specify the X vector and error bars.

Plot x = col 1 and Y = col 2 and error flags = col 3 from data.dat

```
kst data.dat -x 1 -e 3 -y 2
```

Get the X vector from data1.dat, and the Y vector from data2.dat.

```
kst data1.dat -x 1 data2.dat -y 1
```

Placement:

Plot column 2 and column 3 in plot P1 and column 4 in plot P2

```
kst data.dat -P P1 -y 2 -y 3 -P P2 -y 4
```