

Chukwa Adaptors List

In Chukwa, data is collected by adaptors. Adaptors run inside the agent process, and can be dynamically started and stopped.

(mention that commands can go in `initial_adaptors` or else at control port)

The general syntax of the add command is as follows:

```
add [name =] <adaptor_class_name> <datatype> <adaptor specific params> <initial offset>.
```

The `name` field lets you give this adaptor a meaningful name, in order to stop it later. If you do not specify a name, Chukwa will autogenerate one by hashing the class name and parameters. The adaptor class name specifies what adaptor to run. The `datatype` specifies how to process the collected data. The initial offset should normally be zero; specifying a nonzero initial offset lets you do advanced things like only processing a portion of a file.

The adaptors built into Chukwa are as follows:

FileAdaptor: Push a whole file, as one Chunk, then exit. Requires file name as parameter.

Example: add `FileTailer FooData` /tmp/foo 0
This pushes file /tmp/foo, labelling the data as FooData.

filetailer.FileTailingAdaptor:

Takes a file name as mandatory parameter.
Pushes the file as flat bytes, ignoring the content. Useful for streaming binary data.

Example: add `filetailer.FileTailingAdaptor BarData` /foo/bar 0

filetailer.CharFileTailingAdaptorUTF8

The same, but breaking chunks only at carriage returns. Useful for most ASCII log file formats. **filetailer.CharFileTailingAdaptorUTF8NewLineEscaped**
The same, but breaking chunks only at non-escaped carriage returns. Useful for pushing Chukwa-formatted log files, where exception stack traces stay in a single chunk. **DirTailingAdaptor**

Takes a directory path and a second adaptor name as mandatory parameters; repeatedly scans that directory and all subdirectories, and starts the indicated adaptor running on each file.

Example: add `DirTailingAdaptor logs` /var/log/ `filetailer.CharFileTailingAdaptorUTF8` 0

ExecAdaptor Takes a frequency (in milliseconds) as optional parameter, and then program name as mandatory parameter. Runs that program repeatedly at a rate specified by frequency.

Example:

```
add ExecAdaptor Df 60000 /bin/df -x nfs -x none 0
```

Runs `df` every minute, labelling output as `Df`.