

Writing Readers

A reader is sort of a generator and a serializer in one package. Therefore, it is not possible to modify or transform the data later on the way to the browser. Consequently, a reader does not deal with SAX but reads or creates an arbitrary format and returns a stream of binary data.

The output stream is set through the SitemapOutputComponent interface's method `setOutputStream()`. The `generate()` method triggers the actual reading.

Other than that, a reader is a regular SitemapModelComponent and gets its parameters passed through the `setup()` method. Like any other SitemapModelComponent, it may but does not need to implement CacheableProcessingComponent to enable caching.

To write your own reader, you should inherit from one of the existing readers like the ServiceableReader or the RessourceReader or even the AbstractReader.

All that needs to be done to write a custom reader based on the AbstractReader class, is to override the `generate()` method and dump whatever data this reader deals with to the OutputStream `this.out`.

Of course, to write a good reader, the implementation should handle exceptions gracefully, like unexpected closing of the output stream when the user instructs the browser to stop loading for example and should try to work with the caching subsystem in order to provide good performance.