# The Parser Model

The parser model here follows the model in section [8.2.1](http://www.w3.org/TR/2012/CR-html5-20121217/syntax.html#parsing) of the HTML5 specification, though we do not assume a networking layer.

[ InputStream ] // Generic support for reading input.

||

[ Scanner ] // Breaks down the stream into characters.

||

[ Tokenizer ] // Groups characters into syntactic

||

[ Tree Builder ] // Organizes units into a tree of objects

||

[ DOM Document ] // The final state of the parsed document.

## InputStream

This is an interface with at least two concrete implementations:

* StringInputStream: Reads an HTML5 string.
* FileInputStream: Reads an HTML5 file.

## Scanner

This is a mechanical piece of the parser.

## Tokenizer

This follows section 8.4 of the HTML5 spec. It is (roughly) a recursive descent parser. (Though there are plenty of optimizations that are less than purely functional.

## EventHandler and DOMTree

EventHandler is the interface for tree builders. Since not all implementations will necessarily build trees, we've chosen a more generic name.

The event handler emits tokens during tokenization.

The DOMTree is an event handler that builds a DOM tree. The output of the DOMTree builder is a DOMDocument.

## DOMDocument

PHP has a DOMDocument class built-in (technically, it's part of libxml.) We use that, thus rendering the output of this process compatible with SimpleXML, QueryPath, and many other XML/HTML processing tools.

For cases where the HTML5 is a fragment of a HTML5 document a DOMDocumentFragment is returned instead. This is another built-in class.