# Architecture

Created: 16 May, 2018 – Modified: 16 May, 2018

## Components

* Analyzer
* Renderer
* Parser
* Administrator
* Interpreter
* Curator
* Assistants

### Analyzer

The analyzer evaluates requests submitted to the server. These requests can be:

* Form submissions
* URLs
* Ajax

#### Sends To

* **Renderer**

#### Sends To and Receives From

* **Administrators**
* **Assistants**
* **Parser**

#### Receives From

* **Client**

### Renderer

The renderer renders dynamic output that is sent to the client. The render can be static (rendered once and written to a file, database, or other) or dynamic (rendered every time). This includes:

* HTML
* Forms
* Scripts
* Styling

#### Sends to

* **Client**

#### Sends To and Receives From

* **Assistants**

#### Receives from

* **Analyzer**

### Parser

The parser analyzes form data that is submitted to the server.

#### Sends To and Receives From

* **Administrators**
* **Analyzer**
* **Assistants**

### Administrators

Administrators manage the specific areas much like how a company is structured. For example, one department in responsible for shipping while another department is responsible for selling. The shipping department may have sub-departments such handling, packaging, labeling, etc. Programmatically, each department is its own class. If a class should not be instantiated, then it should be abstract.

#### Sends To and Receives From

* **Analyzer**
* **Assistants**
* **Curator**
* **Interpreter**
* **Parser**

### Interpreter

The interpreter provides a medium for administrators to talk to one another. For example, the Ecommerce department uses the URL to communicate departments, products, customers, etc. specifics. It records this URL as the 'department'. The Pages department uses the URL to communicate which page. It records this URL as 'directory'. These aliases represent the same thing: the URL.

#### Sends To and Receives From

* **Administrators**
* **Assistants**

### Curator

The curator manages the data on the server. This could be but is not limited to:

* Databases
* Files

#### Sends To and Receives From

* **Administrators**
* **Assistants**
* **Server**

### Assistants

Assistants take care of the non-specific details as well as overall operation of the architecture.

#### Sends To and Receives From

* **All**