

ADOBE EXPERIENCE MANAGER

Components

1. What are components?

- Modular units which realize specific functionality to present your content on your website.
- Re-usable.
- Developed as self-contained units within one folder of the repository.
- Have no hidden configuration files.
- Can contain other components.
- Can run anywhere within any AEM system.
- Have a standardized user interface.
- Have edit behavior that can be configured.
- Use dialog boxes that are built using sub-elements based on Granite UI components.
- Are developed using [HTL](#) (recommended) or JSP.
- Can be developed to create customized components that extend the default functionality.

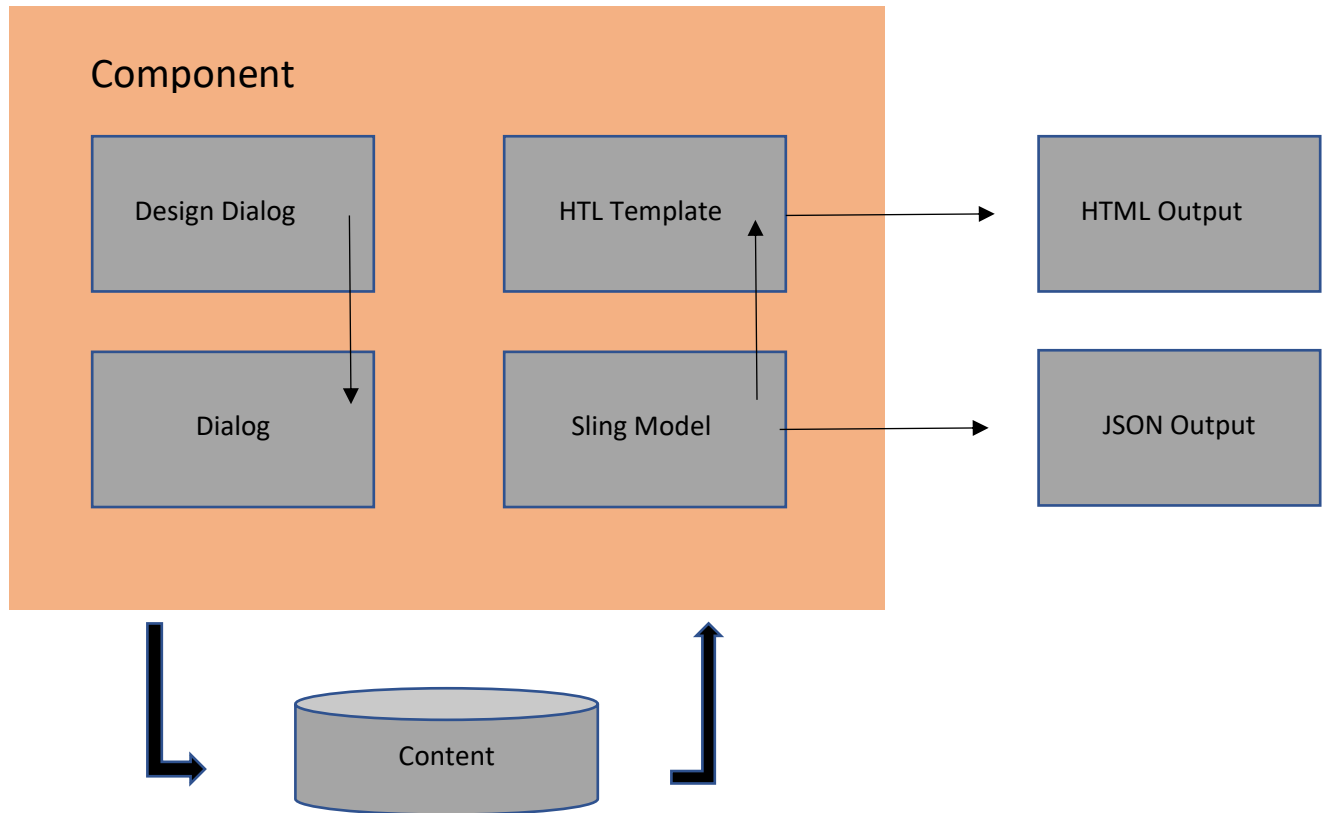
Because components are modular, you can:

- Develop a new component on your local instance.
- Deploy it to your test environment.
- Deploy it to your live authoring environment, where they allow authors and/or administrators to add and configure content.
- Deploy it to your live publish environment(s), where they are used to render content for visitors to your website. Certain components, for example, for Communities, also accept input from your users.

Each AEM component:

- Is a resource type.
- Is a collection of scripts that completely realize a specific function.
- Can function in *isolation*, meaning either within AEM or a portal.

2. Components – Architecture



Let's break this into two parts

The frontend, which includes:

- JSS and CSS assist in styling the elements, and
- HTML Template Language (HTL) combines HTML elements with the backend content.

The backend, which comprises of:

- View (logic) that helps create the front-end content.
- Content Model, which is essential for defining the component's content.
- Edit Dialog and Content Configuration that developers and designers use to decide what content to display, editable elements, and the available options for editing.

3. Component's structure

The structure of an AEM component is powerful and flexible, the main considerations are:

- [Resource Type](#)
- [Component Definition](#)
- [Properties and Child Nodes of a Component](#)
- [Dialogs](#)
- [Design Dialogs](#)

Refer the links given above and ignore everything related to Classic UI.

4. **Sling Model**

This [link](#) will help you to understand the sling models.

5. **HTL**

This [link](#) will give you an overview of HTL.

6. **[Creating and Configuring a Dialog](#)**

This section has the following sub sections, refer to the links to have an understanding of these.

- [Coral UI and Granite UI](#)
- [Creating a new dialog.](#)
- [Creating a new field.](#)

What did we learn?

- What are components?
- Architecture of components.
- The basic structure of a component.

- Sling Model.
- HTL
- Creating and configuring a dialog.