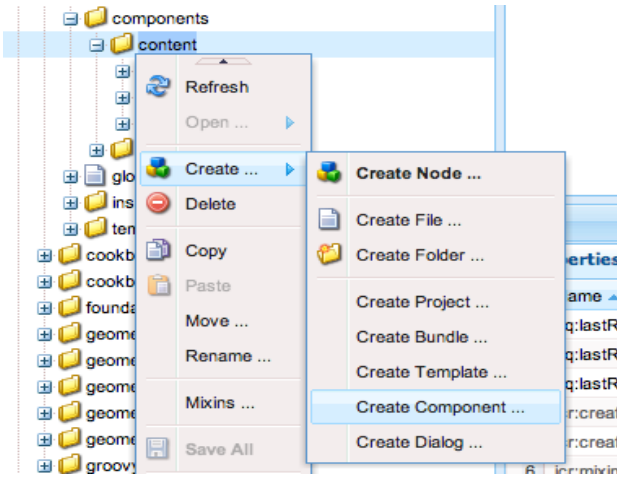
AEM Component Development

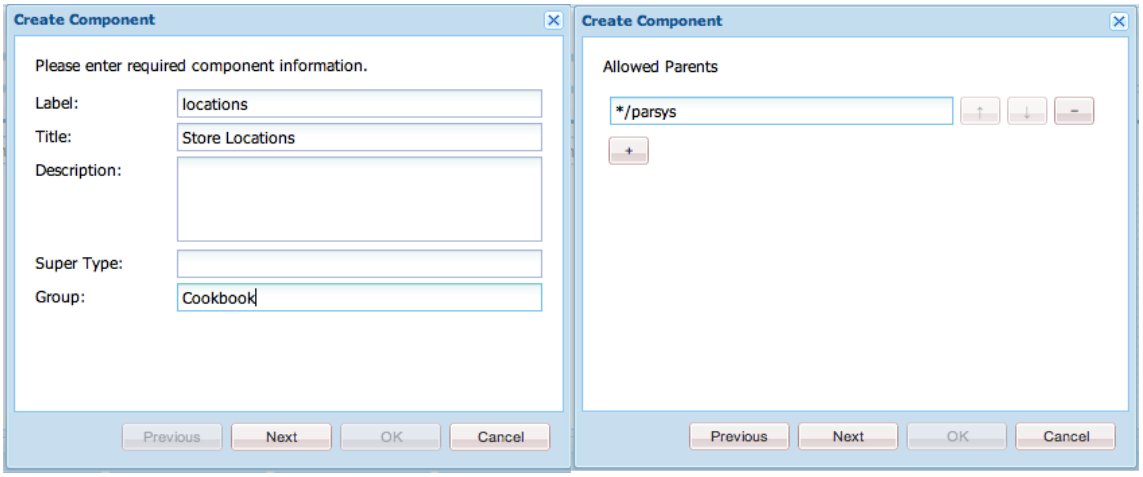
The purpose of this document to is to provide detailed steps on creating an AEM component. The document covers component creation, enabling the component on a page, creating the dialog and creating clientlibs for the component.

CRXDE Lite Pro Tip: *Save often! Aven though you have created an item using one of the wizards inside CRXDE Lite, it does not mean that item is saved. Until you click save your component or dialog will not be available inside the AEM authoring environment.*

Creating Components

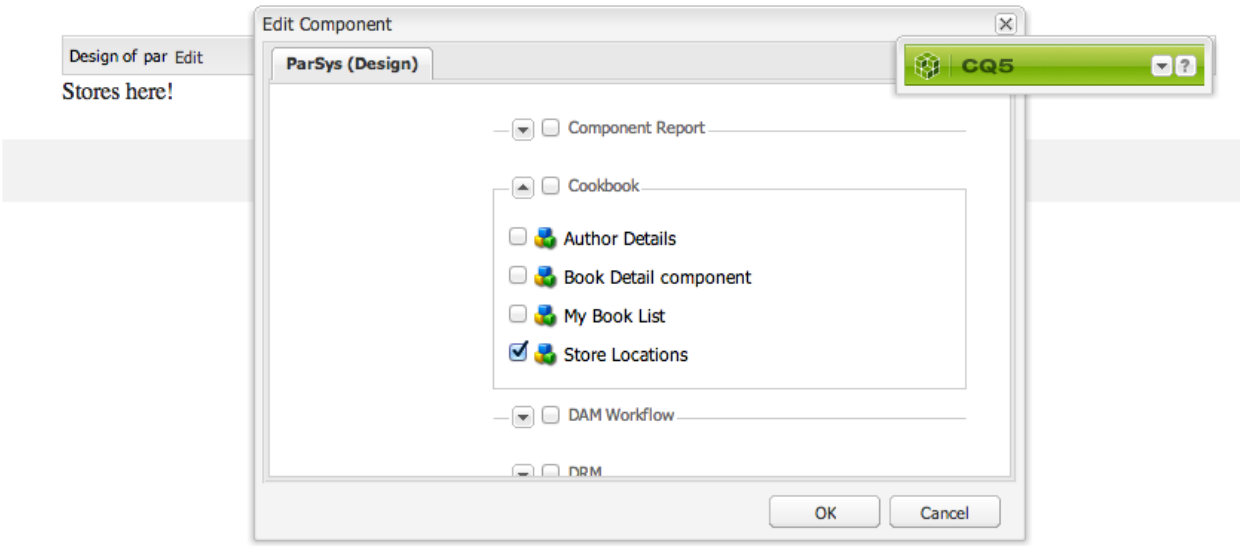
To create a component you will want to use CRXDE Lite to utilize their wizards for creating a component. On the first panel you should enter a label, title and group. The label is the node name for the component, the title is the component name that will show in the sidekick and the group will be the component grouping the component will be included with inside the sidekick. The only additional element needed will be the allowedParents value which should be */parsys for any component that can be graded onto a page from the sidekick. All other items can be defaulted.



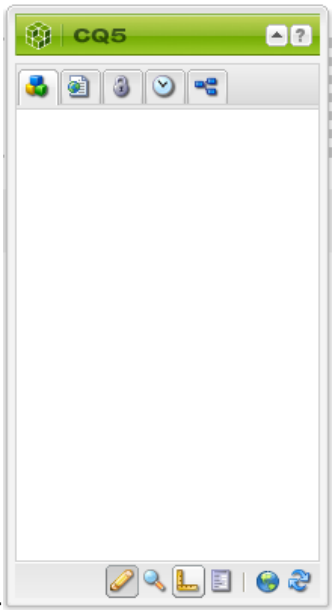


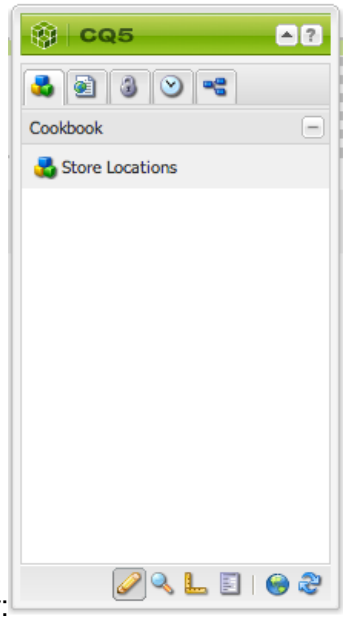
Adding the component to the sidekick

To enable the component to be available on your page you need to add the component in design mode. Design mode is accessed via the sidekick and will display reload the page with an Edit link. Click Edit then find the component group for your component. Then click the checkbox next to the component group to add all components for that group or check the individual component and click OK.



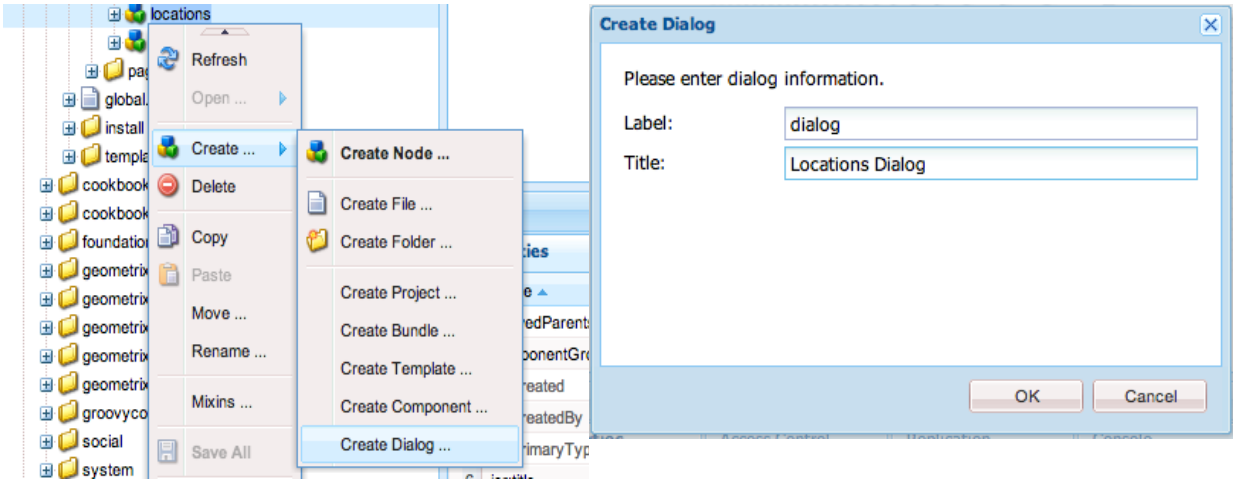
Now the dialog should list the components you have enabled for the page like the image below:

Before: 

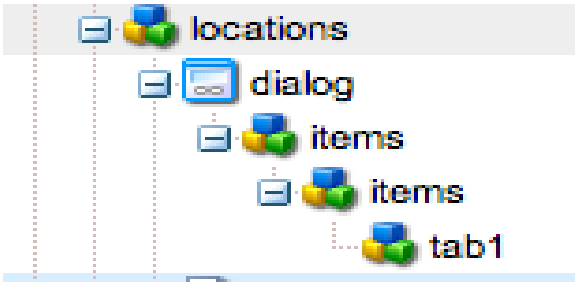
After: 

Creating Dialogs

To create a dialog we will use the wizard inside CRXDE Lite. The wizard will simply ask for a Title for the dialog which you can enter but is not shown inside author mode. The wizard will generate a default node structure that will create a simple dialog with a single tab panel with no items.

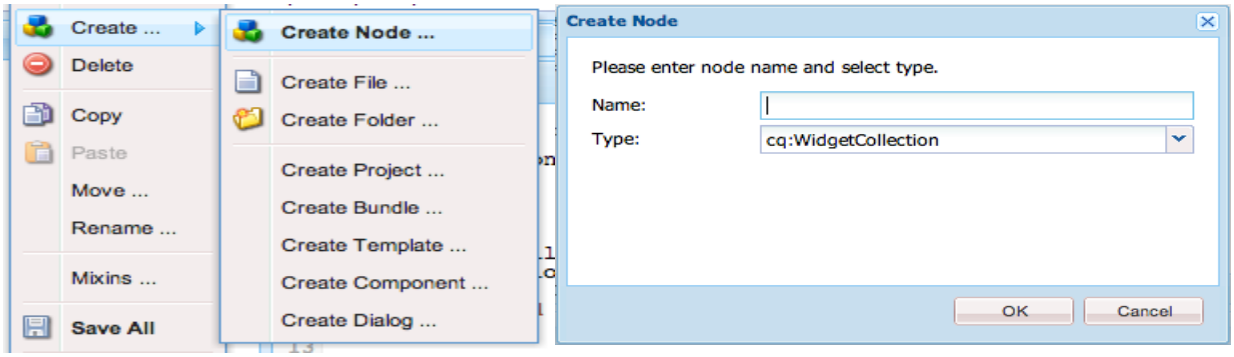


Default dialog node structure:



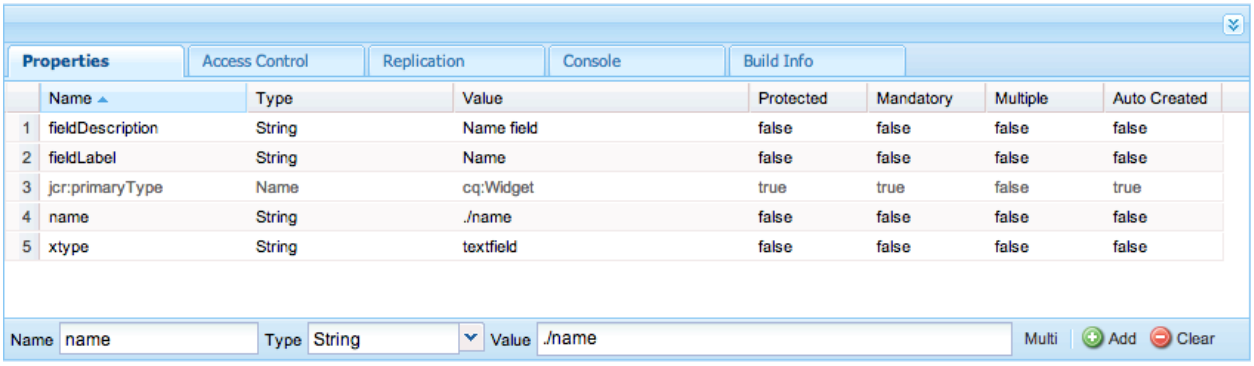
Customizing the Dialog

To add items to the dialog we will need to add a cq:WidgetCollection node the tab1 node created by the dialog wizard. We will use the wizard to create a new node and give the name it 'items' and select the type cq:WidgetCollection. This node will contain all of the items we will display on tab1. Now create a node under items with a name 'name' and type cq:Widget.



Now that you have created the node 'name' we need to provide a fieldLabel, fieldDescription, name and xtype. To accomplish this we will add these properties using the data entry fields

inside the property view. To add fieldLabel, enter fieldLabel in the name field, select String as the type and enter 'Name' in the value field then click Add. You will need to do this for all the properties needed. Two key properties are name and xtype: name sets the property that will store the value from the dialog and xtype sets the type of widget will display in the dialog.



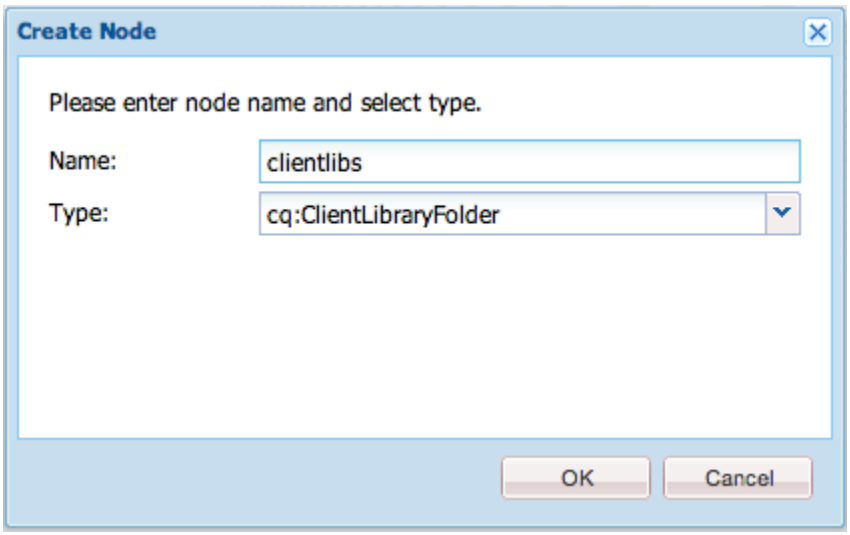
After saving your changes, you can add the component to the page and open the dialog by double clicking on the component. The dialog should now look like the image below.

Xtypes

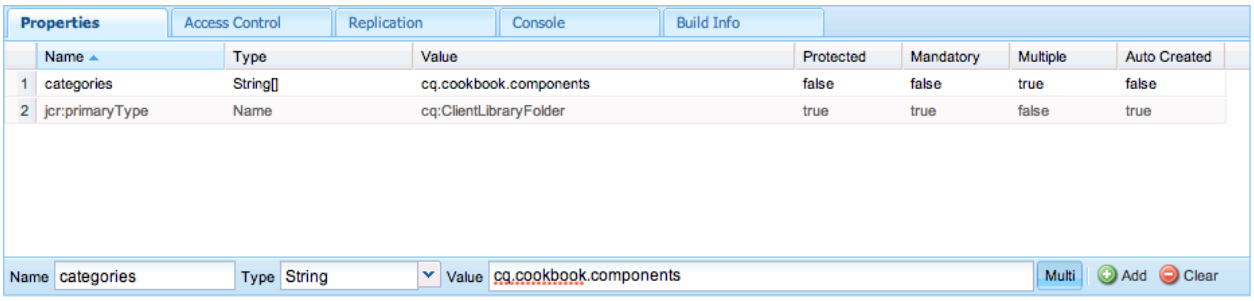
AEM utilizes xtypes to provide custom widgets inside a component's dialog. To learn more about xtypes checkout the link below that shows all of the available xtypes:
http://dev.day.com/docs/en/cq/current/developing/widgets/xtypes.html

Creating Clientlibs

To create a clientlibs node we will use the wizard inside CRXDE Lite. The new node will reside inside the component node structure. The name will be 'clientlibs' and the type will be cq:ClientLibraryFolder.



Once the clientlibs folder is created we need to add a categories property. This property name will be categories, type will be String and value will be the category group name for you clientlibs. This property is actually a String[] so we need to make sure we select the Multi option before clicking save.



Clientlibs Categories

There are a couple ways to enable clientlibs within your application. For this training we have utilized the embedding of clientlibs inside your design's clientlibs. A clientlib folder can have a property called embed that allows you to embed clientlibs from components. The categories property is used to reference a specific clientlibs for embedding. Please see the link below for more information on clientlibs:

http://experiencedelivers.adobe.com/cemblog/en/experiencedelivers/2012/12/clientlibs-explained-by-example.html

Adding CS and JS

The clientlibs folder will be utilized to contain the component specific CSS and JavaScript. Under the clientlibs folder, you need to create a css and js folders and create a css.txt and js.txt file. You will then place your css files inside the css folder and the javascript inside the js folder. To enable the individual files to be read by AEM you need to edit the cs.txt and js.txt to include

the individual js and css files.