* ‘*global*’ folder under /conf holds the default templates as a fall back
* To create editable templates using CRX ,

/conf

<your-folder-name> [sling:Folder]

settings [sling:Folder]

wcm [cq:Page]

templates [cq:Page]

policies [cq:Page]

* Other way is via configuration browser

**ACL’s & Permissions**

* *Template-authors –* only those users who can create templates should be part of this group

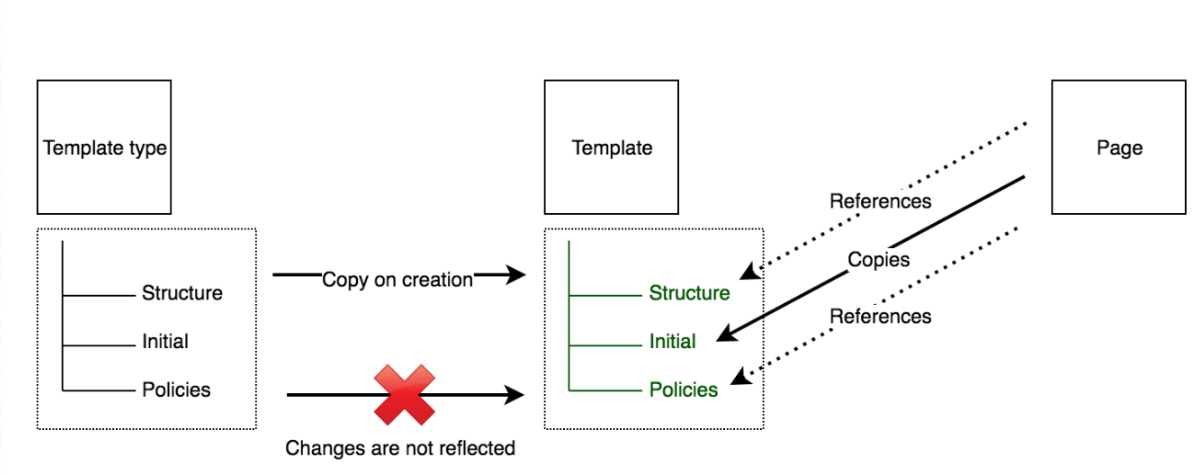
|  |  |  |
| --- | --- | --- |
|  | Template Authors | Content Authors |
| /conf/project-name/settings/wcm/templates | Read, write , replicate | replicate |
| /conf/project-name/settings/wcm/policies | Read, write , replicate | Replicate |
| /conf/project-name/settings/wcm/templateTypes | Read | read |

**Structure**

* Changes made to structure will be immediately available on all the pages created using this template
* Components added here cannot be moved or removed
* Merged with *initial* when pages are created
* Once a component is unlocked, the property *editable* is set to true on *structure/jcr:content/root/component/*
* Once a component having content is unlocked , a copy of it will be moved under *initial*

**Initial**

* has a *jcr:content* node which will be copied to all the pages to be created from this template
* this is merged with structure for creation of pages
* Any existing pages will not be updated if the initial content is changed after creation.



Main points from above diagram

* Template types are just to create the mandatory content to be available under templates
* When a new template (A) is created from a templateType , [Structure,Initial,Policies] of template type are entirely copied to template
* when changes are made to templateType, these won’t be reflected to the templates already created previously
* Upon page creation from template (A), [Structure,Initial,Policies] of template are accumulated on page. Initial is copied and rest all are referenced. So changes made to structure & policies remain dynamic

NOTE

* In static templates, we will have to hard code the clientlibs into the page component where as in editable templates, we can apply clientlibs dynamically using the template editor
* We can preconfigure the components (which can be used in a particular parsys) using template editor where as in static templates its not possible (design mode 🡪select the allowed components only at the time of authoring).
* Template types and policies are inherited across all folders according to the following order of precedence:
  + The current folder.
  + Parent(s) of the current folder.
  + /conf/global
  + /apps
  + /libs

When rendering a page:

* **Templates**:
  + The cq:template property of its jcr:content node will be referenced to access the template that corresponds to that page.
* **Components**:
  + The page component will merge the structure/jcr:content tree of the template with the jcr:content tree of the page.
  + The page component will only allow the author to edit the nodes of the template structure that have been flagged as editable (as well as any children).
  + When rendering a component on a page, the relative path of that component will be taken from the jcr:content node; the same path under the policies/jcr:content node of the template will then be searched.
    - The cq:policy property of this node points to the actual content policy (i.e. it holds the design configuration for that component).
    - This allows you to have multiple templates that re-use the same content policy configurations.

# Creating a new template Type

<http://www.sgaemsolutions.com/2017/09/dynamic-editable-templates-in-aem-63.html>

Useful Links

<https://blog.3sharecorp.com/creating-editable-templates-adobe-experience-manager>

<https://aemhq.com/posts/what-s-the-point-of-using-a-template-type/>