Activation/Deactivation

# Overview:

As all Windows Phone applications, the story creator app must support activation/deactivation. When the user reactivates or re-launches the application, we should go back to the home page. On the home page, there should be an option to let the user continue the previous story.

# Goals:

* Persistent the story in isolated storage.
* Restore chosen media.
* Restore media data such as transition duration, photo duration, transition and name.

# None Goals:

* When reactivated, come back to the previous page.

# Design:

There’re several options to store/restore data. Each option has advantages and disadvantages:

## Persistent the thumbnail and resized images:

We can persistent the thumbnail and resized images as well as other information. This allows us to restore the thumbnail without querying the media library again. But this approach requires more storage, and it takes time to store the data.

## Do not persistent the thumbnail and resized images:

If we don’t persistent the thumbnail and resized images, we must query the media library to get the thumbnail during activation. This may or may not require more time compared to restoring thumbnail from isolated storage. But even if it takes a little more time, the tradeoff is accepted. The advantage is smaller disk space consumption, and during deactivation, it takes less time since the thumbnails do not need to be persisted.

If we don’t persistent the resized image, we must resize it again during the preview stage or the uploading stage. This takes more time compared to reading the resized images from isolated storage. But the activation/deactivation time can be reduced.

So for now, we do not persistent the thumbnail and the resized images. In the future, we may reconsider the design.

## Persistent the transition:

We use the transition’s name to uniquely identify a transition. So we need to persistent the name of the transition.

The code to persistent transition is best to be placed in the transition classes themselves. Thus we need to construct an abstract base class that contains some common persistence logic, and allow child classes to override it. Refer to the transition library design spec for more information.