

# Listview and Scrollviewer Tombstoning helper

This article provides a library which saves the scroll position of [ListView](#) and [ScrollViewer](#) controls during tombstoning.

## Introduction



One of the Windows Phone design guidelines is that page states should be saved when a running application is tasked-away from (for example by selecting the Windows button) so that it can be restored to the same state when it is restarted. This gives users the impression that the application is simply being returned to activity, rather than fully restarted.

Apps that are tasked away from are initially made *dormant* (ie still in memory but not active) and if you switch back to them nothing will have changed. However in low memory an app may be *tombstoned* (removed from memory). In order to restore a tombstoned application you need to restore it from settings you have previously saved to memory.

This article provides a utility class (StateManager) that handles saving and restoring the scroll bar states of [ListView](#)/[ScrollViewer](#) controls, and demonstrates its usage.

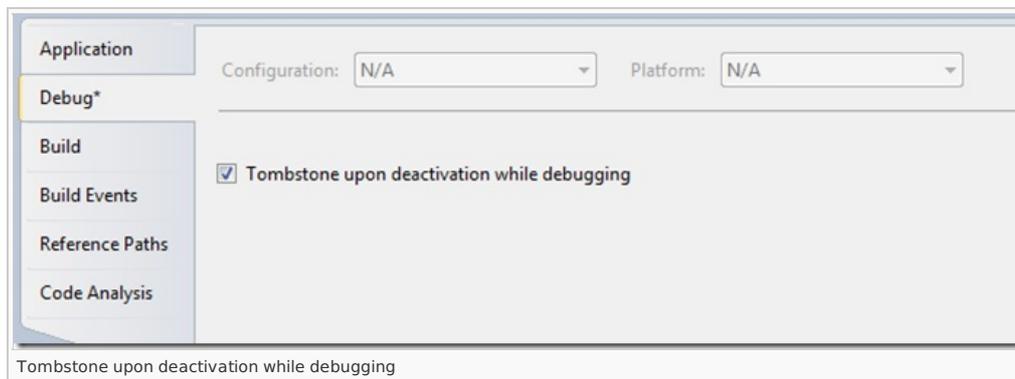
## Problem

Before explaining how the utility class works, let's see what happen when the scroll position is not saved:

1. Create a "Windows Phone Databound Application".
2. Run the app then scroll the list to the last item.
3. Press the Start button (to task away from the application)
4. Press the Back button (to return to the application)

The state is preserved, because the application was *dormant*.

Now, stop debugging the application and go to the properties of the project. In the Debug section, check the option "Tombstone upon deactivation while debugging" to force the app to tombstone if tasked away from when debugging.



Execute the last 3 steps and you'll see that the list scroll position is not saved.

## Solution

To fix this issue, insert the `StateManager` class (provided at the end of the article) in your project and call `StateManager.SaveScrollViewerOffset()` and `StateManager.RestoreScrollViewer` on the list that you want to save/restore scroll position. The best places to insert those calls are in the page methods `OnNavigatedTo()` and `OnNavigatedFrom()`.

```
protected override void OnNavigatedTo(System.Windows.Navigation.NavigationEventArgs e)
{
    base.OnNavigatedTo(e);
    StateManager.RestoreScrollViewerOffset(MainListBox);
}

protected override void OnNavigatedFrom(System.Windows.Navigation.NavigationEventArgs e)
{
    base.OnNavigatedFrom(e);
    StateManager.SaveScrollViewerOffset(MainListBox);
}
```

**Download the sample project:** <File:SaveScrollViewerApp.zip>

