

# Application Life Cycle - Windows Phone

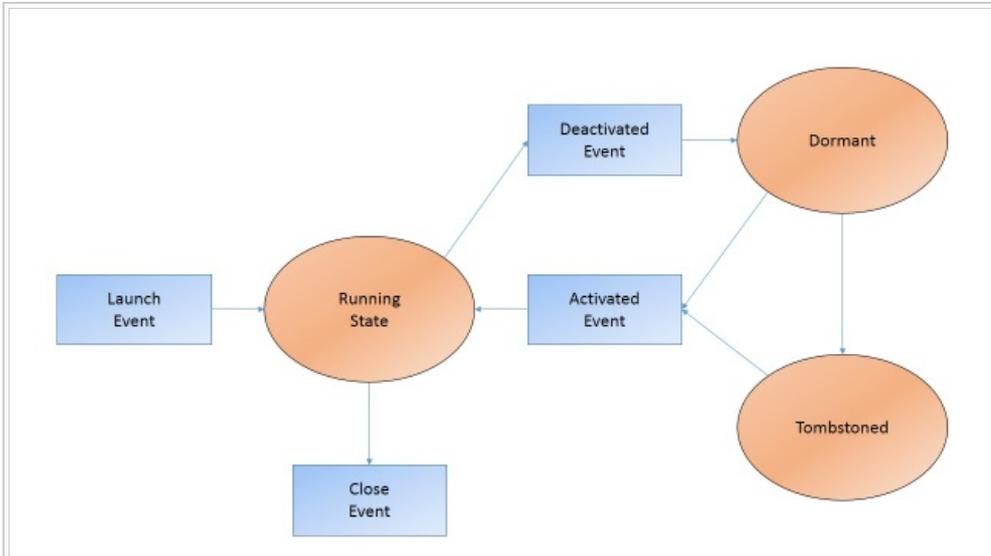
This article provides an overview of both the normal Windows Phone Application Life Cycle and "exceptions" (like fast resume and running the app under the lock screen).

## Standard app lifecycle



18 Sep  
2011

An overview of the standard Windows Phone 8 (and 7.5) app lifecycle is given below. Detailed information on the application lifecycle is available at [App activation and deactivation for Windows Phone \(MSDN\)](#).



Run-state lifecycle in Windows Phone 7.5 and Windows Phone 8

At launch an app enters the running state and will remain there until the app gets:

- closed down by the user - either by pressing the back button or through the Task Switcher (available since Windows Phone 8 Update 3), or
- deactivated when the application launches a Chooser (for example, using CameraCaptureTask), or
- deactivated when the lock screen engages, or
- deactivated if the user navigates to another app (for example, by pressing the Start button)

A dormant app is still resident in memory and may be quickly restarted. If the system runs out of available memory (as it is used by other apps) a dormant app may be tombstoned. This stores the app's navigation information (Back-Stack), closes it, and removes it from memory. It is up to the developer to save data to allow for rehydration of a tombstoned app when the user returns to it.

The user can return to a dormant or tombstoned app using the back button or through the Task Switcher. By default, if the user launches the app again from a Live Tile, the app list or any system integration point (for example, a Lens), a new app instance is opened. In this case the dormant/tombstoned app instance is terminated.

## Background services

There are some use cases that are not possible as part of the normal application lifecycle - specifically those that require operations to continue while the app is behind the lock screen or "in the background". Many of these cases can be addressed using dedicated background services:

- Audio playback (see [How to play background audio for Windows Phone \(MSDN\)](#))
- File Transfer (see [Background file transfers for Windows Phone \(MSDN\)](#))
- Background agents (see [Background agents for Windows Phone \(MSDN\)](#))
- Location tracking (see [Running location-tracking apps in the background for Windows Phone 8 \(MSDN\)](#))

## Exceptions

There are further use cases that cannot be addressed when an app is deactivated in either the standard application lifestyle or by using background services. Therefore the normal cycle can be modified as outlined below.

## Running under the lock screen

It is possible to keep the current application active when the lock screen is engaged. The article [Run Windows Phone application under lock screen](#) and [Idle detection for Windows Phone \(MSDN\)](#) explains how this is done, and also how to optimize that app when running in that state to conserve battery power.

## Resuming app from live tile instead of restarting

The default behaviour on launching an app from a Live Tile, the app list or system integration points is to restart the app with a new application instance. On Windows Phone 8 this behaviour can be modified so that the OS will instead resume an already running instance of the app. This is much faster than creating a new instance, and gives the user the experience of resuming rather than starting the app.

Both approaches are outlined in [Fast app resume for Windows Phone 8 \(MSDN\)](#).

## Tracking user movement in the background

On Windows Phone 8 it is possible to have a **single app** continue to run completely in the background. This is primarily intended to support use cases like location tracking (e.g. Nokia's Here Drive).

See [How to run location-tracking apps in the background for Windows Phone 8 \(MSDN\)](#) for information on how to enable and optimize for this state in an app.