

Archived:Handling Camera resource on Symbian

 Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template `{{ReviewForRemovalFromArchive|user=~~~~~|write your reason here}}`.

Overview

This how-to describes how to reserve and release the camera resource when the application is activated (either brought to the foreground or started) or inactivated (either sent to the background or exited).

In order to handle the camera resource correctly when the focus is gained or lost, the application must be able to react to foreground events. This code snippet uses `MCoeForegroundObserver` to implement this functionality, but you may want to see [Getting notifications of focus change and launch of other apps using Symbian C++](#) for another possibility.

This snippet can be self-signed.

Preconditions

To use this code snippet, the application needs to provide implementation for using the camera (for example, [S60 Platform: Camera Example with Autofocus](#)).

MMP file

The following capabilities and libraries are required:

```
CAPABILITY UserEnvironment
LIBRARY ecam.lib
```

Header file

Inherit your class from `MCoeForegroundObserver` and override the `HandleGainingForeground` and `HandleLosingForeground` functions to be able to react to foreground events and handle the camera resource.

```
#include <ECam.h>           // link against ecam.lib
#include <ccamautofocus.h> // only needed if autofocus extension is meant to be
                           // used; link against CamAutoFocus.lib

...

/**
 * From MCoeForegroundObserver
 */
virtual void HandleGainingForeground();
virtual void HandleLosingForeground();

...

CCamera* iCamera;

CCamAutoFocus* iAutoFocus; // optional
```

Source file

To listen for the foreground events, make the class observe changes in them:

```
iEikonEnv->AddForegroundObserverL( *this );
```

Reserving the camera resource

```
// Gets called when the application is brought to the foreground
void CYourClass::HandleGainingForeground()
{
    iCamera->Reserve(); // Asynchronous. Calls MCameraObserver::ReserveComplete
                       // when the request completes.
}
```

Releasing the camera resource

```
// Gets called when the application is sent to the background
void CYourClass::HandleLosingForeground()
{
    // Bring the AF subsystem to idle state, in case it is used
    TRAPD( err, iAutoFocus->ResetToIdleL() );
    if ( !err )
    {
        iAutoFocus->Close();
    }

    // Release the camera
    iCamera->Release();
}
```