



ID	CS000821	Creation date	January 29, 2008
Platform	S60 3rd Edition S60 3rd Edition FP1	Tested on devices	Nokia E90 Communicator
Category	Symbian C++	Subcategory	Imaging

Keywords (APIs, classes, methods, functions): MCoeForegroundObserver, HandleGainingForeground, HandleLosingForeground

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 [TSS000754 - Getting notifications of focus change and launching of other applications](#) 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 )
```

CS000821_-_Handling_Camera_resource

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