


[Knowledge Base Home](#)

This known issue was originally created by **Alammi** and has been verified by Forum Nokia.

ID	KIS000850	Creation date	March 25, 2008
Platform	S60 3rd Edition, Feature Pack 1	Devices	Nokia N95, Nokia E90 Communicator, Nokia 6110 Navigator
Category	Symbian C++	Subcategory	Location-based Services

Keywords (APIs, classes, methods, functions):

Overview

Trying to use `RPositioner::NotifyPositionUpdate()` to receive GPS position fails after a certain number of updates have been made, if update options have been omitted.

Description

Approximately after 4-5 minutes (255 update cycles), `RPositioner::NotifyPositionUpdate()` completes with error `KErrInUse (-14)`. This happens when the client has not set valid update options with `RPositioner::SetUpdateOptions()`.

This is not the defined return value for this API. After returning `KErrInUse` the `RPositioner` has "died" and always returns `KErrInUse`.

Note that the problem does not appear when using a Bluetooth GPS through the **Location Acquisition API**, or on the emulator.

How to reproduce

Header file:

```
class CMyPositioner : public CActive
{
    ...

    RPositionServer iPositionServer;
```

KIS000850_-_RPositioner::NotifyPositionUpdate_return_KErrInUse_after_few_minutes

```
RPositioner iPositioner;  
TPositionSatelliteInfo iSatelliteInfo;  
};
```

Implementation:

```
_LIT(KRequestor, "MyPositioner");  
  
void CMyPositioner::ConstructL()  
{  
    ...  
    User::LeaveIfError( iPositionServer.Connect() );  
    User::LeaveIfError( iPositioner.Open(iPositionServer) );  
    User::LeaveIfError( iPositioner.SetRequestor( CRequestor::ERequestorService,  
                                                CRequestor::EFormatApplication,  
                                                KRequestor ) );  
  
    CActiveScheduler::Add( this );  
}  
  
void CMyPositioner::StartL()  
{  
    ...  
    iPositioner.NotifyPositionUpdate( iSatelliteInfo, iStatus );  
    SetActive();  
}  
  
void CMyPositioner::RunL()  
{  
    if (iStatus == KErrNone)  
    {  
        // Got iSatelliteInfo successfully, handle it  
        HandleSatelliteInfo();  
  
        // Request next update  
        iPositioner.NotifyPositionUpdate(iSatelliteInfo, iStatus);  
        SetActive();  
    }  
    else  
    {  
        // Fails with KErrInUse (-14) after 255 cycles  
        _LIT( KDebugMessage, "NotifyPositionUpdate() failed: %d" );  
        RDebug::Print( KDebugMessage, iStatus.Int() );  
    }  
}
```

Solution

Set valid update options with `RPositioner::SetUpdateOptions()` prior to calling `RPositioner::NotifyPositionUpdate()`:

```
iPositioner.SetUpdateOptions(  
    TPositionUpdateOptions( aInterval,
```

KIS000850_-_RPositioner::NotifyPositionUpdate_return_KErrInUse_after_few_minutes

```
        TTimeIntervalMicroSeconds(0), /* timeout */
        TTimeIntervalMicroSeconds(0), /* max age */
        aAcceptPartialUpdates));

iPositioner.NotifyPositionUpdate( iSatelliteInfo, iStatus );
SetActive();
...
```

Note that the update interval (aInterval) value must be non-zero.