Knowledge Base Home

ID	KIS000588	**Creation date**	February 22, 2007 (Last updated: November 1, 2007)
Platform	S60 3rd Edition, S60 3rd Edition, FP1	**Devices**	
Category	Symbian C++	**Subcategory**	Base & System, Application Framework

Keywords (APIs, classes, methods, functions):

Description

When letting the system find and launch the handler application for a file with a recognized MIME type, the handler receives a call to its `CAknDocument::OpenFileL()` implementation. However, if the handler application is already running in the background, the `OpenFileL()` call for the document class is not received.

Solution

In addition to `CEikDocument::OpenFileL()`, data handler applications in S60 must also provide an implementation for `CAknAppUi::OpenFileL()`.

A common way to solve this is to redirect the call to document class in `CAknAppUi::OpenFileL()`:

```
void CHandlerAppUi::OpenFileL( const TDesC& aFileName )
  {
  CHandlerDocument* doc = static_cast<CHandlerDocument*>( Document() );
  doc->OpenFileL( ETrue, aFileName, iEikonEnv->FsSession() );
  }
```

If the handler application is running in the background, the calling application must call `TApaTask::SwitchOpenFile()`:

```
TBool CCallingAppUi::RefreshDocumentFileL( const TUid& aUid,
                                           const TDesC& aFileName )
  {
  TApaTaskList taskList( iCoeEnv->WsSession() );
```

```
  // Find handler application by its UID
  TApaTask task = ( taskList.FindApp( aUid ) );
  if ( task.Exists() )
    {
    // Calls CAknAppUi::OpenFileL(), requires SwEvent capability
    User::LeaveIfError( task.SwitchOpenFile( aFileName ) );
    return ETrue;
    }
  return EFalse;
  }
```

If a file is launched from the S60 Web Browser or File Manager, a different approach is required as there will be no call to `SwitchOpenFile()`. The handler application itself can check for other instances of itself already running in the background. By default, the application framework brings the old instance to foreground, and terminates the new one. It is possible to change this behaviour by overriding `PreDocConstructL()` function from `CAknApplication` class:

```
void CHandlerApplication::PreDocConstructL()
  {
  CEikonEnv* env = CEikonEnv::Static();

  // Check that this app is started as stand-alone
  if (!env->StartedAsServerApp() && !env->EikAppUi())
    {
    RWsSession& ws = env->WsSession();
    const TInt myWgId = env->RootWin().Identifier();
    TInt wgId = 0;
    TUid uid(AppDllUid());

    // Look for another instance of this app
    while (wgId >= 0)
      {
      if (wgId && wgId != myWgId) // Another instance found -> close it
        {
        TApaTask other(ws);
        other.SetWgId(wgId);
        other.EndTask();          // Requires SwEvent capability
        }
      CApaWindowGroupName::FindByAppUid(uid, ws, wgId);
      }
    }

  // call PreDocConstructL from base class
  CEikApplication::PreDocConstructL();
  }
```

Note that above code works only for stand-alone (non-embedded) handler applications, and required **SwEvent** capability.

See also:

TSS000419 - S60 MIME recognizers and opening files for editing
S60 Platform: Document Handler Example and
S60 Platform: Document Handler API Developer?s Guide available for download at Forum Nokia