

| | | | |
|-----------------|--|----------------------|----------------------------|
| ID | TSS000877 | Creation date | April 2, 2008 |
| Platform | S60 3rd Edition, Feature Pack 1, S60 3rd Edition, Feature Pack 2 | Devices | S60 Nseries devices |
| Category | Symbian C++ | Subcategory | Architecture/Design, Music |

Keywords (APIs, classes, methods, functions):

Overview

The Music Player application used in new Nseries devices uses a view system different from the standard S60 Music Player. This may cause problems in 3rd party applications that try to launch the Music Player in a certain view (e.g. *Now Playing*).

Description

Different views of the Music Player (*MPX*) in new Nseries devices cannot be activated using the standard methods of the AVKON view architecture. While there is no public API to control MPX views, view switching is still possible by sending messages to the Music Player application with `TAppTask::SendMessage()` or via command line arguments if the player is currently not running.

Solution

How to find out which Music Player is used in a device?

The UID of the Music Player application is changed in the new version.

```
const TUid KAppUidMPX = { 0x102072C3 };
```

`RApaLsSession::GetAppInfo()` can be used for checking whether an application with this UID exists.

```
TInt ret = apaSession.GetAppInfo(appInfo, KAppUidMPX);
```

TSS000877_-_Launching_the_Nseries_Music_Player_in_a_certain_view

GetAppInfo() returns KErrNone if the device has a new version of the Music Player and KErrNotFound if the device has an old version of the Music Player.

Launching the Music Player in a certain view

```
const TUid KMPXPlaybackViewTypeUid = { 0x101FFCA0 }; // Now Playing View
const TUid KMPXCollectionViewTypeUid = { 0x101FFCA1 }; // Collection View

void LaunchMPXViewL( const TUid& aViewUid )
{
    // Construct MPX parameter string
    TBuf8<16> param;
    param.FillZ(16);
    param[8] = 0x01;
    param[12] = (TUint8) (aViewUid.iUid & 0x000000ff);
    param[13] = (TUint8) ((aViewUid.iUid & 0x0000ff00) >> 8);
    param[14] = (TUint8) ((aViewUid.iUid & 0x00ff0000) >> 16);
    param[15] = (TUint8) ((aViewUid.iUid & 0xff000000) >> 24);

    TApaTaskList taskList( CEikonEnv::Static()->WsSession() );
    TApaTask task = taskList.FindApp( KAppUidMPX );

    if ( task.Exists() )
    {
        task.SendMessage( KAppUidMPX, param );
        return;
    }

    RApaLsSession apaSession;
    TApaAppInfo appInfo;
    User::LeaveIfError( apaSession.Connect() );
    CleanupClosePushL( apaSession );

    User::LeaveIfError( apaSession.GetAppInfo( appInfo, KAppUidMPX ) );

    CApaCommandLine *cmdLine = CApaCommandLine::NewLC();
    cmdLine->SetExecutableNameL( appInfo.iFullName );
    cmdLine->SetCommandL( EApaCommandRun );

    // Collection view is the default view at startup
    if( aViewUid != KMPXCollectionViewTypeUid )
    {
        cmdLine->SetTailEndL( param );
    }

    User::LeaveIfError( apaSession.StartApp( *cmdLine ) );

    CleanupStack::PopAndDestroy( 2 ); // cmdLine, apaSession
}
```