Archived:Finding audio and video formats supported by the Symbian device

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 snippet shows how to find out all audio/video play and record formats supported by the phone.

This snippet can be self-signed.

MMP file

The following libraries are required:

```
LIBRARY MMFControllerFramework.lib
```

Header file

```
#include <mmf\common\mmfcontrollerpluginresolver.h>
#include <mmf\common\mmfcontroller.h>
```

Source file

```
CMMFControllerPluginSelectionParameters* pluginParameters = CMMFControllerPluginSelectionParameters::NewLC();
CMMFFormatSelectionParameters* formatParameters = CMMFFormatSelectionParameters::NewLC();

pluginParameters->SetRequiredPlayFormatSupportL(*formatParameters);
pluginParameters->SetRequiredRecordFormatSupportL(*formatParameters);

RArray<TUid> ids;
User::LeaveIfError(ids.Append(KUidMediaTypeAudio));
User::LeaveIfError(ids.Append(KUidMediaTypeVideo));

//EAllowOnlySuppliedMediaIds - plug-ins that support exact media ID
//EAllowOtherMediaIds - plug-ins that support at least the media ID

//pluginParameters->SetMediaIdsL(ids, CMMFPluginSelectionParameters::EAllowOnlySuppliedMediaIds);
pluginParameters->SetMediaIdsL(ids, CMMFPluginSelectionParameters::EAllowOtherMediaIds);

//It is also possible to set the preferred supplier of the plug-in
//and return e.g. only those plug-ins
//pluginParameters->->SetPreferredSupplierL(_L("SupplierName"),
//    CMMFPluginSelectionParameters::EOnlyPreferredSupplierPluginsReturned);

RMMFControllerImplInfoArray controllers;
CleanupResetAndDestroyPushL(controllers);

//Get all audio/video play and record controllers/formats that are supported
pluginParameters->ListImplementationsL(controllers);

/* //Print results e.g. to global CConsoleBase* console
_LIT(KAudioFormats,"Audio/Video play and record formats detected:\n");
_LIT(KTab, "\t");
_LIT(KNewLine, "\n");
console->Printf(KAudioFormats);
for (TInt index=0; index<controllers.Count(); index++)
  {
  const RMMFFormatImplInfoArray& playFormats = controllers[index]->PlayFormats();
  const RMMFFormatImplInfoArray& recordFormats = controllers[index]->RecordFormats();
  console->Printf(controllers[index]->DisplayName());
  console->Printf(KNewLine);
  for (TInt i=0; i<playFormats.Count(); i++)
    {
    console->Printf(KTab);
    console->Printf(playFormats[i]->DisplayName());
    console->Printf(KNewLine);
    }
  console->Printf(KNewLine);
  for (TInt j=0; j<recordFormats.Count(); j++)
    {
    console->Printf(KTab);
    console->Printf(recordFormats[j]->DisplayName());
    console->Printf(KNewLine);
    }
  }
*/

CleanupStack::PopAndDestroy(3);//controllers, formatParameters, pluginParameters
```


(c) Copyright Nokia 2013. All rights reserved.

Postconditions

The controllers array contains all audio/video play and record formats supported by the phone.

(c) Copyright Nokia 2013. All rights reserved.

 (c) Copyright Nokia 2013. All rights reserved.