如何让用户在程序启动时选择不同的语言

标题：如何让用户在程序启动时选择不同的语言

概述

缺省Symbian/S60程序将使用当前手机语言作为其语言选择，但很多开发者希望能给用户自己选择程序的语言，但不要改变手机的语言。

"用户来选择语言" http://www.developer.nokia.com/Community/Discussion/showthread.php?177512-Language-selection-by-user

解决方案

这里有一个解决方案可以让用户在程序启动时选择自己喜欢的语言

步骤1：在程序开始时，我们提供一个全局列表显示所有支持的语言。然后我们记下用户的选择，在示例工程中我们定义了P&S用来存储用户的选择

```
// begin by chen
...
#include <badesca.h>
#include <akngloballistquery.h>
#include <e32property.h>

void MainL()
 {
 _LIT(KPhoneLanugage, "Phone language");
 _LIT(KEnglish, "English");
 _LIT(KChinese, "Chinese");
 CDesCArray* array = new(ELeave) CDesCArrayFlat(5);
 CleanupStack::PushL(array);
 array->AppendL(KPhoneLanugage);
 array->AppendL(KEnglish);
 array->AppendL(KChinese);
 // the application framework has not be created yet so we the notifier service to show the list
 CAknGlobalListQuery* query = CAknGlobalListQuery::NewLC();
 TRequestStatus status = KRequestPending;
 query->ShowListQueryL(array, status);
 User::WaitForRequest(status);
 TInt ret = status.Int();
 if(ret>=0)
  {
               // ret is the index of the user selected item
               // here we map the user selection to a language code
  TLanguage language = ELangEnglish;
  switch(ret)
   {
   case 0:
    {
    language = User::Language();
    break;
    }
   case 1:
    {
    language = ELangEnglish;
    break;
    }
   case 2:
    {
    language = ELangPrcChinese;
    break;
    }
   default:
    {
    language = User::Language();
    break;
    }
   }
  // here we define a property to hold the user selection, but this is definitely not the only way, for example you can also save it
  RProperty::Define(KUidUrzpvtchApplication, 0, RProperty::EInt);
  RProperty::Set(KUidUrzpvtchApplication, 0, (TInt)language);
  }
 CleanupStack::PopAndDestroy(2, array);
 }

GLDEF_C TInt E32Main()
 {
// begin by chen
// at this stage we have to create the cleanup stack by ourself
 // Cleanup stack needed
 CTrapCleanup* cleanup = CTrapCleanup::New();
 if(cleanup)
  {
  TRAPD(err, MainL());
  if(err != KErrNone)
   {
   _LIT(KUserPanic,"Failed to start");
   User::Panic(KUserPanic, err);
   }
  delete cleanup;
  }
// end by chen
 return EikStart::RunApplication( NewApplication );
 }
```

步骤2：在程序框架初始化时我们将缺省的资源文件替换为用户选择的文件。可以通过重载**ResourceFileName**()来替换缺省内容。

Copyright © Nokia 2013. All rights reserved.

```
#include <coeutils.h>
...
TFileName CUrzpvtchApplication::ResourceFileName() const
 {
 TFileName filename = CAknApplication::ResourceFileName();
 TInt language = 0;
 TInt err = RProperty::Get(KUidUrzpvtchApplication, 0, language);
 if(err==KErrNone)
  {
  TFileName fn = filename;
  _LIT(KExtFormat, "r%d");
  TBuf<16> buf;
  buf.Format(KExtFormat, language);
  if(buf.Length()<=2) // for example "r1"
   {
   _LIT(KZero, "0");
   buf.Insert(1, KZero); // now should be "r01"
   }
  TInt pos = fn.LocateReverse('.');
  fn.SetLength(pos+1); // remove the extention
  fn.Append(buf); // add new extention
  // make sure that the filename actually exists, or there will be a User 23 panic
  TBool exists = ConeUtils::FileExists(fn);
  if(exists)
   {
   filename=fn;
   }
  }
 return filename;
 }
```

源代码

完整示例: Urzpvtch(LanguageSelection).zip

注意

1. 标题无法改变，因为它定义在_reg.rsc中

2.系统字符串如CBA的“选项”“选择”“取消”无法改变，因为它们是定义在avkon.rsc中的

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2010-2013. All rights reserved.