How to build a COM object on Symbian OS

Component Object Model (COM)

Component Object Model (COM) is a platform for software componentry introduced by Microsoft in 1993. It is used to enable interprocess communication and dynamic object creation in any programming language that supports the technology. The term COM is often used in the software development world as an umbrella term that encompasses the OLE, OLE Automation, ActiveX, COM+ and DCOM technologies. Although COM was introduced in 1993, Microsoft did not begin emphasizing the name COM until 1997.

Build a simple COM object

COM is a binary standard (also said to be language agnostic) and may be developed in any programming language capable of understanding and implementing its binary defined data types and interfaces.

All COM components must (at the very least) implement the standard IUnknown interface, and thus all COM interfaces are derived from IUnknown. The IUnknown interface consists of three methods: AddRef() and Release(), which implement reference counting and controls the lifetime of interfaces; and QueryInterface(), which by specifying an IID allows a caller to retrieve references to the different interfaces the component implements. The effect of QueryInterface() is similar to dynamic_cast<> in C++ or casts in Java and C#.

Let's code

```
//globally unique identifiers (GUIDs)
//{00000319-0106-236C-7B9D-0090C9908A90}
DEFINE_GUID(IID_ISomeInterface, 00000319, 0x106, 0x236C, 0x7B, 0x9D, 0x0,
   0x90, 0xC9, 0x90, 0x8A, 0x90);

#undef  INTERFACE
#define INTERFACE   ISomeInterface

DECLARE_INTERFACE_(ISomeInterface, IUnknown)
{
    /*
     * IUnknown methods
     */
    STDMETHOD(QueryInterface) (THIS_
    REFIID riid,
    void** ppvObj) PURE;

    STDMETHOD_(ULONG32,AddRef) (THIS) PURE;

    STDMETHOD_(ULONG32,Release) (THIS) PURE;

    /*
     * ISomeInterface methods
     */

    /*************************************************************************
     * Method:
     *  ISomeInterface::GetSomething
     *
     */
    STDMETHOD(GetSomething) (THIS_
```


(C) Copyright Nokia 2014. All rights reserved.

```
    REF(IOtherInterface*) pOther) PURE;
};
```

GUID(globally unique identifiers)

A Globally Unique Identifier or GUID is a special type of identifier used in software applications in order to provide a reference number which is unique in the context for which it is used, for example, in defining the internal reference for a type of access point in a software application, or for creating unique keys in a database.

The GUID is a 16-byte (128-bit) number:

```
#define REF(type)   type&
#define EXTERN_C    extern "C"
typedef struct  _GUID
    {
    ULONG32  Data1;
    UINT16  Data2;
    UINT16  Data3;
    UCHAR  Data4[ 8 ];
    } GUID;
#define DEFINE_GUID(name, l, w1, w2, b1, b2, b3, b4, b5, b6, b7, b8) \
        EXTERN_C const GUID name \
                = { l, w1, w2, { b1, b2,  b3,  b4,  b5,  b6,  b7,  b8 } }
```

Language-neutral

The essence of COM is a language-neutral way of implementing objects such that they can be used in environments different from the one they were created in, even across machine boundaries.

- DECLARE_INTERFACE_

```
#define _INTERFACE struct
#define DECLARE_INTERFACE_(iface, baseiface) _INTERFACE iface : public baseiface
```

- STDMETHOD

```
typedef LONG32 COM_RESULT;
#define STDMETHODCALLTYPE __stdcall
#define STDMETHOD(method) virtual COM_RESULT STDMETHODCALLTYPE method
```

- Others

```
#define STDMETHOD_(type,method) virtual type STDMETHODCALLTYPE method
#define PURE = 0
#define THIS_
typedef GUID IID;
#define REFIID const IID &
```

COM on Symbian

```
class CSomeObject :   public ISomeInterface
{
```


(C) Copyright Nokia 2014. All rights reserved.

```
 public:

 EXTERN_C CSomeObject* CreateInstance();

    /*************************************************************************
     * Method:
     *  ISomeInterface::GetSomething
     *
     */
    STDMETHOD(GetSomething)
        (THIS_REF(IOtherInterface*) pOtherObject);
}

GLDEF_C TInt E32Dll(TDllReason)
{
   return KErrNone;
}

EXTERN_C CSomeObject* CSomeObject::CreateInstance()
{
   return new (ELeave) CSomeObject;
}

STDMETHOD(CSomeObject::GetSomething)(
      THIS_REF(IOtherInterface*) pOtherObject)
{
...
}
```

Here we have builded a simple COM object on Symbian OS.

Best use of COM on Symbian

HelixPlayer

Usefull links

COM IS LOVE

(C) Copyright Nokia 2014. All rights reserved.