简易COM实现

接口设计

接口的设计原则：接口不变原则

- **虚表的顺序不变**

用户接口

以`UnKnow`类作为虚基类

```
class IMyUnknow
{
public:
    virtual HRESULT QueryInterface(const GUID& type, void** ppObject) = 0; //查询接口
    virtual HRESULT Release() = 0;  // 自杀式释放接口
};
```

其他类派生自`UnKnow`，功能上的更新或扩展依次继承其父类即可

```
class ISuperMath :public IMyUnknow
{
public:
    virtual HRESULT  Add(int n1, int n2, int* ret) = 0;
    virtual HRESULT  Sub(int n1, int n2, int* ret) = 0;
    virtual HRESULT  Mul(int n1, int n2, int* ret) = 0;
};


class ISuperMath2 :public ISuperMath
{
public:
    virtual HRESULT  Mul(int n1, int n2, int* ret, void* p = NULL) = 0;
};
```

赋予每个类一个`GUID`，查找对象接口利用`GUID`确定获取到的对象

```
// {DA0E0B52-6815-41af-BD59-28F86DA4ADCA}
static const GUID IID_ISuperMath2 =
{ 0xda0e0b52, 0x6815, 0x41af, { 0xbd, 0x59, 0x28, 0xf8, 0x6d, 0xa4, 0xad, 0xca }
};


// {A75B18F9-9FB7-4a13-B502-91365CAA74DD}
static const GUID IID_IMyUnknow =
{ 0xa75b18f9, 0x9fb7, 0x4a13, { 0xb5, 0x2, 0x91, 0x36, 0x5c, 0xaa, 0x74, 0xdd }
};
```

```
// {40A4A166-6548-4c66-A06D-5FD8BAFFBC39}
static const GUID IID_ISuperMath =
{ 0x40a4a166, 0x6548, 0x4c66, { 0xa0, 0x6d, 0x5f, 0xd8, 0xba, 0xff, 0xbc, 0x39 }
};

// 获取类对象接口
typedef HRESULT (*MYGETCLASSOBJECT)(const GUID& iid, void** ppObject);
extern "C" __declspec(dllexport) HRESULT MyGetClassObject(const GUID & iid,
void** ppObject);
```

实现

继承接口类，实现接口

```
class CSuperMath :public ISuperMath // 继承ISuperMath接口
{
public:
    // 以GUID获取类对象，通过强转从而可以得到不同类的虚表
    virtual HRESULT QueryInterface(const GUID& iid, void** ppObject)
    {
        if (memcmp(&iid, &IID_IMyUnkown, sizeof(GUID)) == 0)
        {
            //有接口
            *ppObject = (IMyUnknow*)this;    // Unknow
            return S_OK;
        }
        else if (memcmp(&iid, &IID_ISuperMath, sizeof(GUID)) == 0)
        {
            //有接口
            *ppObject = (ISuperMath*)this;  // ISuperMath
            return S_OK;
        }

        //没接口
        return E_NOINTERFACE;
    }


    virtual HRESULT Release()
    {
        delete this;
        return S_OK;
    }

    virtual HRESULT  Add(int n1, int n2, int* ret)
    {
        *ret = n1 + n2;
        return S_OK;
    }

    virtual HRESULT  Sub(int n1, int n2, int* ret)
    {
        *ret = n1 - n2;
        return S_OK;
    }

    virtual HRESULT  Mul(int n1, int n2, int* ret)
    {
        *ret = n1 * n2;
        return S_OK;
    }
};


// 导出获取类对象接口函数
extern "C" __declspec(dllexport)
HRESULT MyGetClassObject(const GUID& iid, void** ppObject)
{
    CSuperMath* pObject = new CSuperMath();
    if (pObject == NULL)
        return E_OUTOFMEMORY;

    return pObject->QueryInterface(iid, ppObject);
}
```