

Serialisierung

CObject <-> CObArray <-> CArchive <-> CFile

1. Eigene Klasse serialisierbar machen

EigeneKlasse.h

```
class CEigeneKlasse : public CObject
{
    DECLARE_SERIAL(CEigeneKlasse)
public:
    void Serialize(CArchive &ar);
    CString m_sText;
    int m_nZahl;
    ...
};
```

EigeneKlasse.cpp

```
IMPLEMENT_SERIAL(CEigeneKlasse, CObject, 1)
CEigeneKlasse::CEigeneKlasse ()
{
    ...
}

CEigeneKlasse::~CEigeneKlasse (){}

void CEigeneKlasse::Serialize(CArchive &ar)
{
    // Serialize der Basisklasse
    CObject::Serialize(ar);

    if(ar.IsStoring())
    {
        // Speichern
        ar << m_sText << m_nZahl;
    }
    else
    {
        // Laden
        ar >> m_sText >> m_nZahl;
    }
}
```

2. Klasse in CObArray verwalten

in DlgKlasse (bzw. DockKlasse) Membervariable (z.B. m_oaFeld) vom Typ CObArray erstellen

Objekt zum Array hinzufügen mit Add:

```
m_oaFeld.Add(new CEigeneKlasse);
```

Wertzuweisung an Objekt mit []:

```
CEigeneKlasse *pEigeneKlasse;  
pEigeneKlasse = (CEigeneKlasse*)m_oaFeld[iIndex];  
pEigeneKlasse->m_sText = „Blabla“;
```

Wert auslesen aus Objekt mit []:

```
CEigeneKlasse *pEigeneKlasse;  
pEigeneKlasse = (CEigeneKlasse*)m_oaFeld[iIndex];  
CString sText;  
sText = pEigeneKlasse->m_sText;
```

Objekt löschen mit delete und RemoveAt:

```
delete (CEigeneKlasse*)m_oaFeld[iIndex];  
m_oaFeld.RemoveAt(iIndex);
```

3. Speichern des ObArrays in einer Datei

Datei öffnen zum Schreiben

```
CFile Datei("C:\\\\Personen.dat", CFile::modeCreate |CFile::modeWrite);
```

Archiv zum Schreiben in Datei verbinden

```
CArchive ar(&Datei, CArchive::store);
```

in Archiv schreiben

```
m_oaFeld.Serialize(ar);
```

4. Laden des ObArrays aus einer Datei

Datei öffnen zum lesen

```
CFile Datei("C:\\\\Personen.dat", CFile::modeRead);
```

Archiv zum Lesen aus Datei verbinden

```
CArchive ar(&Datei, CArchive::load);
```

in Archiv schreiben

```
m_oaFeld.Serialize(ar);
```