

Hochschule Harz	FB Automatisierung und Informatik
MFC	Grafische Nutzerschnittstellen mit MFC Thema: SDI-Programmierung mit Forms und Serialize

Versuchsziele

Vertiefung im Verständnis der SDI-Programmierung mit mit Forms und Serialize unter mit Visual Studio 2008.

Überblick:

Diese Übung zeigt die Verwendung der binären Stream-Speicherung und zeigt die Verknüpfung der Daten mit GUI-Elemente in einem SDI-Fenster.

Folgende Aufgaben müssen Sie durchführen:

- Erstellen eines SDI-Projektes mit MFC und einem FormView (step1)
- Einbau der Attribute (Name, Matrnr, Prüfungsindex)
- Initialisierung der Attribute
- Einbau der Methoden
- Einbau der GUI-Elemente (String-Editfeld, Spin-Feld, Liste)
- Einfügen eines Dialogfensters, zum Ändern der Einträge der Liste mit Prüfungen
- Ändern der Speichern und Ladens
- Aufrufen des Dialogfenster

Grundlagen ListForms:

Die Komponente „ListeForm“ erlaubt die Darstellung von GUI-Elemente. Es ist also ein Dialogfenster, welches aber mit Menü und optional eine Dokument-View-Architektur besitzen kann.

Aufgaben:

1) Grundprojekt erstellen:

Erstellen eines SDI-Projektes mit CFormView

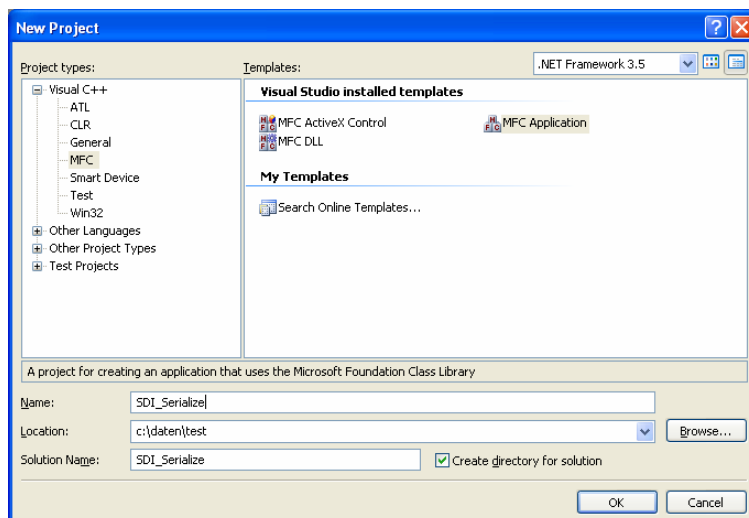


Abbildung 1 MFC-Anwendung erstellen

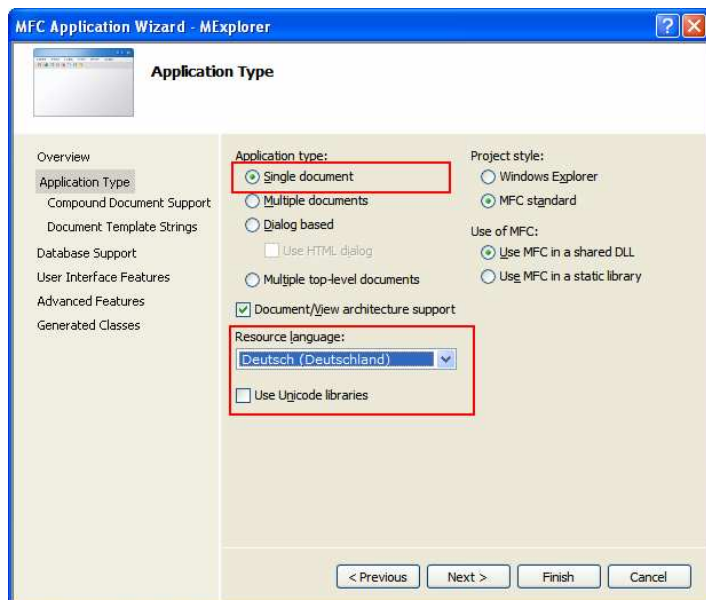


Abbildung 2 SDI-Anwendung

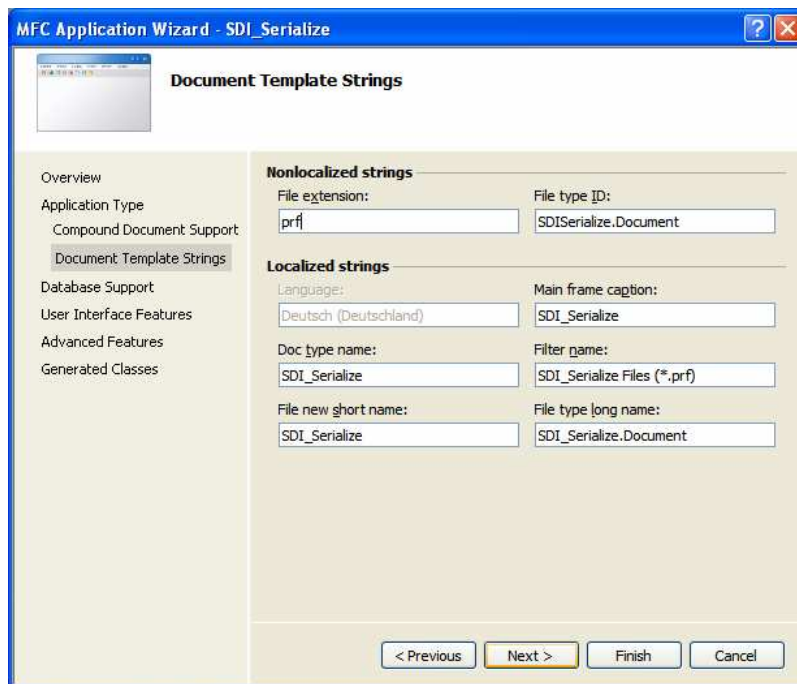


Abbildung 3 Eintragen der Dateierweiterung prf

Wichtig: Hier Änderung von CView in CFormView

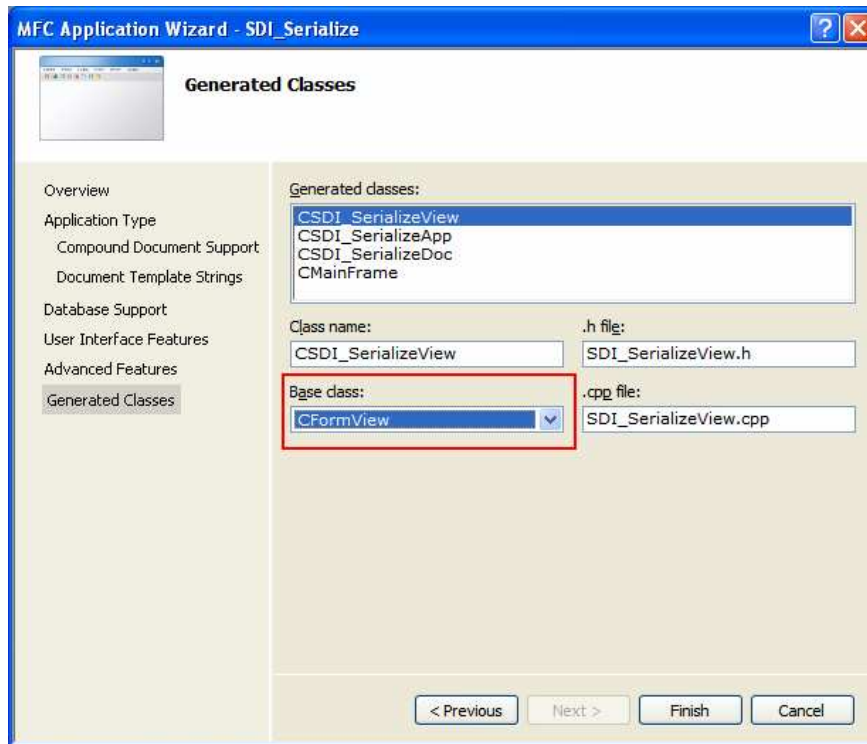


Abbildung 4 Ende des Wizards

Step1

2) Member-Variablen erstellen

In der Datei „SDI_SerializeDoc.h“ werden folgende Attribute eingetragen:

- CString m_name
- unsigned int m_matrnr;
- int m_pruefungslisten_index;

3) Get- und Set-Methoden erstellen

In der Datei „SDI_SerializeDoc.h“ werden folgende Methoden eingetragen:

- CString getName();
- void setName(CString name);
- unsigned int getMatrnr();
- void setMatrnr(unsigned int matrnr);
- int getPruefungslistenIndex();
- void setPruefungslistenIndex(int index);

4) In der Methode OnNewDocument die Member-Variablen initialisieren

- m_name="Meier";
- m_matrnr=14721;
- m_pruefungslisten_index=2;

5) Vervollständigen der Set- und Get-Methoden in Datei SDI_SerializeDoc.cpp

```
CString CSDI_SerializeDoc::getName() {
    return m_name;
}

void CSDI_SerializeDoc::setName(CString name) {
    m_name = name;
}

unsigned int CSDI_SerializeDoc::getMatrnr() {
    return m_matrnr;
}

void CSDI_SerializeDoc::setMatrnr(unsigned int matrnr) {
    m_matrnr=matrnr;
}

int CSDI_SerializeDoc::getPruefungslistenIndex() {
    return m_pruefungslisten_index;
}

void CSDI_SerializeDoc::setPruefungslistenIndex(int index) {
    m_pruefungslisten_index = index;
}
```

6) Speichern und Laden definieren in Datei SDI_SerializeDoc.cpp

Methode: Serialize

Hinweis: ar steht immer links

```
void CSDI_SerializeDoc::Serialize(CArchive& ar)
{
    if (ar.IsStoring())
    {
        // TODO: Hier Code zum Speichern einfügen
        ar << m_name;
        ar << m_matrnr;
        ar << m_pruefungslisten_index;
    }
    else
    {
        // TODO: Hier Code zum Laden einfügen
        ar >> m_name;
        ar >> m_matrnr;
        ar >> m_pruefungslisten_index;
    }
}
```

7) GUI-Elemente hinzufügen

In der Baumdarstellung den Eintrag „recourcen“ auswählen
Danach das Dialogfenster, die Form, auswählen

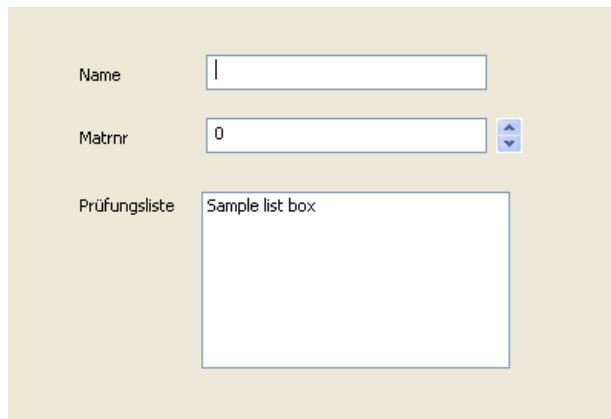
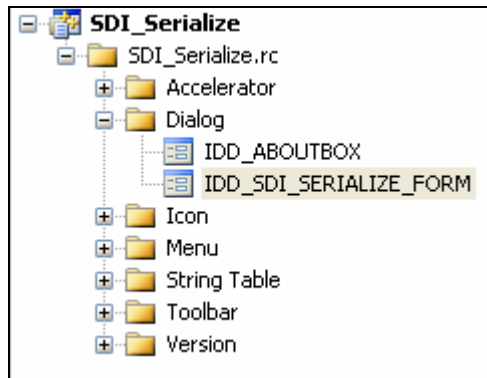


Abbildung 5 GUI-Elemente

Bitte nicht vergessen, die Eigenschaften „Set Buddy Integer“ und „Auto-Buddy“ zu setzen

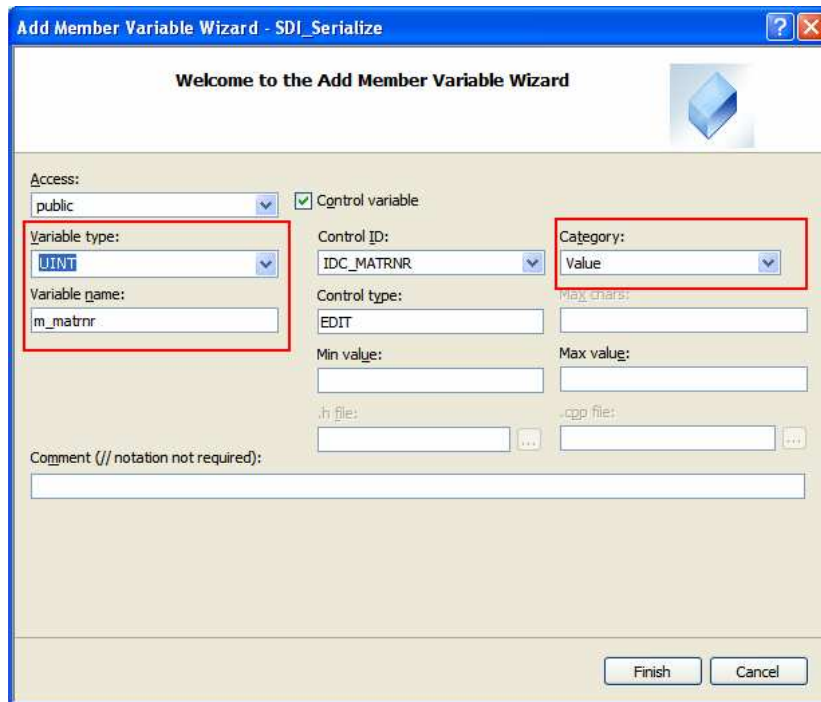
ID:

- IDC_NAME
- IDC_MATRNR
- IDC_PRUEFUNG

8) Member-Variablen erstellen

GUI-ID	Name
IDC_NAME	m_name
IDC_MATRNR	m_matrnr
IDC_PRUEFUNG	m_pruefung_ctrl
IDC_SPIN_MATRNR	m_matrnr_spin_ctrl

Wichtiger Eintrag:



5) Initialisierung der Liste

Datei: SDI_SerializeView.cpp
 Methode: OnInitialUpdate

```
m_pruefung_ctrl.AddString("Grundlagen der Informatik 1");
m_pruefung_ctrl.AddString("Grundlagen der Informatik 2");
m_pruefung_ctrl.AddString("Betriebssysteme");
m_pruefung_ctrl.AddString("Grafische Nutzerschnittstellen");
m_pruefung_ctrl.AddString("Parallele Algorithmen");
```

```
m_matrnr_spin_ctrl.SetRange(1,31999);
```

6) Eintragen der onDraw-Methode

Datei: SDI_SerializeView.h
 virtual void OnDraw(CDC* pDC); // Überschrieben, um diese Ansicht darzustellen

Datei: SDI_SerializeView.cpp

```
void CSDI_SerializeView::OnDraw(CDC* pDC) {
    CSDI_SerializeDoc* pDoc = GetDocument();
    CString sName;
    int matrnr;
    int index;
    sName = pDoc->getName();
    matrnr = pDoc->getMatrnr();
    index = pDoc->getPruefungslistenIndex();

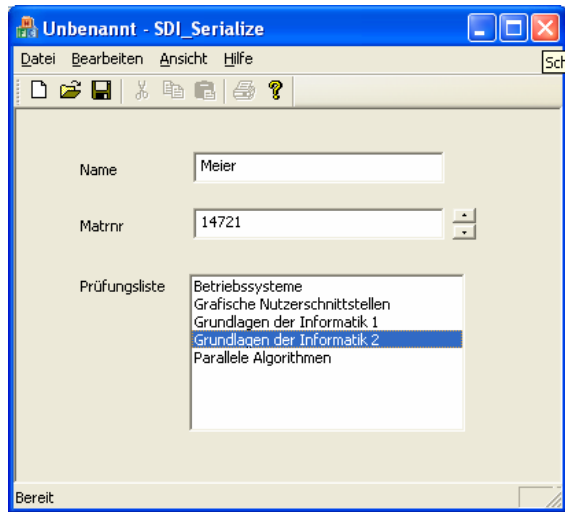
    m_name = sName;
    m_matrnr = matrnr;
```

```

UpdateData(false);
m_pruefung_ctrl.SetCurSel(index);
UpdateData(false);
}

```

Aufruf des aktuellen Programms:



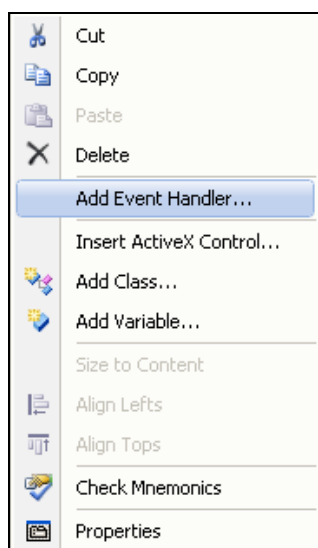
Ende step4

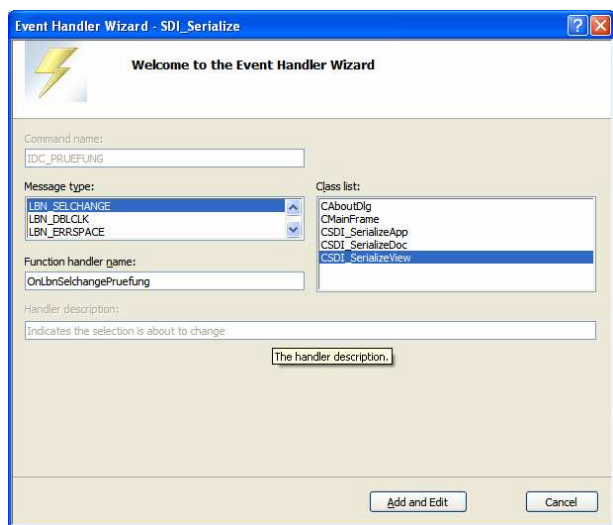
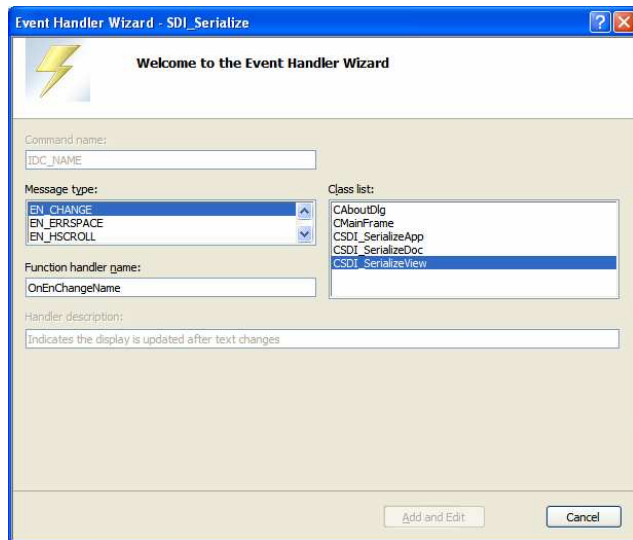
7) Eintragen der Change-Methode

Die folgenden Methoden stellen sicher, dass Änderungen in das Dokument eingetragen werden und dann automatisch gespeichert werden.

Change-Event für den Name

- Wechseln ins Dialogfenster
- Rechter Maustaste über den Namen
- Auswahl „Add Event Handler“





```
void CSDI_SerializeView::OnEnChangeName(){
    CSDI_SerializeDoc* pDoc = GetDocument();
    CString sName;
    UpdateData(true);
    sName = m_name;
    pDoc->setName(sName);
    //Invalidate();
    //pDoc->UpdateAllViews(NULL);
    pDoc->SetModifiedFlag(true);
}
```

```
void CSDI_SerializeView::OnEnChangeMatnrn(){
    CSDI_SerializeDoc* pDoc = GetDocument();
    unsigned int matnrn;
    UpdateData(true);
    matnrn = m_matnrn;
    pDoc->setMatnrn(matnrn);
    //Invalidate(); // okay
    //pDoc->UpdateAllViews(NULL);
    pDoc->SetModifiedFlag(true);
}
```

```
void CSDI_SerializeView::OnLbnSelchangePruefung(){
    CSDI_SerializeDoc* pDoc = GetDocument();
    int index;
```



```

UpdateData(true);
index=m_pruefung_ctrl.GetCurSel();
pDoc->setPruefungslistenIndex(index);
//Invalidate();
//pDoc->UpdateAllViews(NULL);
pDoc->SetModifiedFlag(true);
}

```

Hinweis:

Befehle:

- Invalidate();
- pDoc->UpdateAllViews(NULL);

Erläuterung:

Die beiden unteren Befehle zeigen an, dass der Inhalt des aktuellen Views sich geändert hat.

Invalidate erzwingt ein Neuzeichen.

Mit UpdateAllViews erhalten alle Views die Nachricht, dass sich der Inhalt der „Datei“ geändert hat. Observer-Modell.

Alternativ kann man auch folgenden Code schreiben:

```
GetDocument ()->UpdateAllViews (this, 0x5A, (CObject*) (LPCTSTR) sData);
```

Der erste Parameter, zeigt an, dass das aktuelle View NICHT die Update-Methode erhält.

Wichtig dabei, man muss die OnUpdate-Methode implementieren:

6) Eintragen der onUpdate-Methode

Datei: SDI_SerializeView.h

```
virtual void OnDraw(CDC* pDC); // Überschrieben, um diese Ansicht darzustellen
```

Datei: SDI_SerializeView.cpp

```

void CSDI_SerializeView::OnUpdate(CView* pSender, LPARAM lHint, CObject* pHint) {
    CSDI_SerializeDoc* pDoc = GetDocument();
    CString sName;
    int matrnr;
    int index;
    sName = pDoc->getName();
    matrnr = pDoc->getMatrnr();
    index = pDoc->getPruefungslistenIndex();

    m_name = sName;
    m_matrnr = matrnr;
    UpdateData(false);
    m_pruefung_ctrl.SetCurSel(index);
    UpdateData(false);
}

```