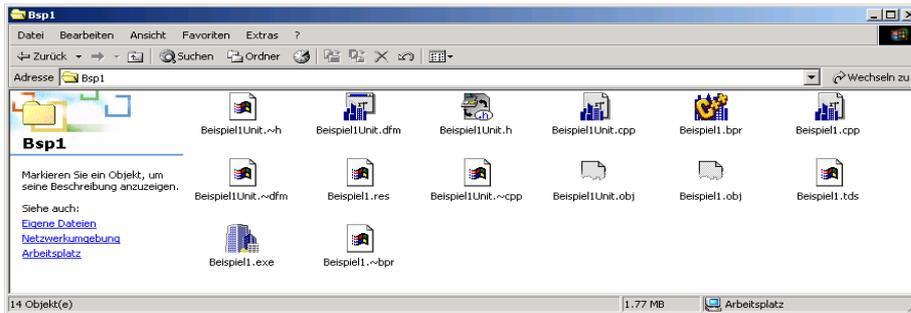


Eine neue Anwendung erstellen

- Nach Start des Borland C++ Builder wird standardmässig eine leere Anwendung erzeugt.
- Eine Borland-Anwendung besteht typisch aus den Komponenten:



- Zweckmässigerweise speichert man das Projekt in einem Verzeichnis der Wahl und vergibt neue Filenamen:

```
project1.bpr      beispiel1.bpr      // Projektfile enthält alle Einstellungen und Referenzen
project1.cpp      beispiel1.cpp      // Startfile
unit1.cpp         Beispiel1Unit.cpp  // Code zu Formular
Form1.dfm        Beispiel1Unit.dfm  // Formular mit Komponenten
```

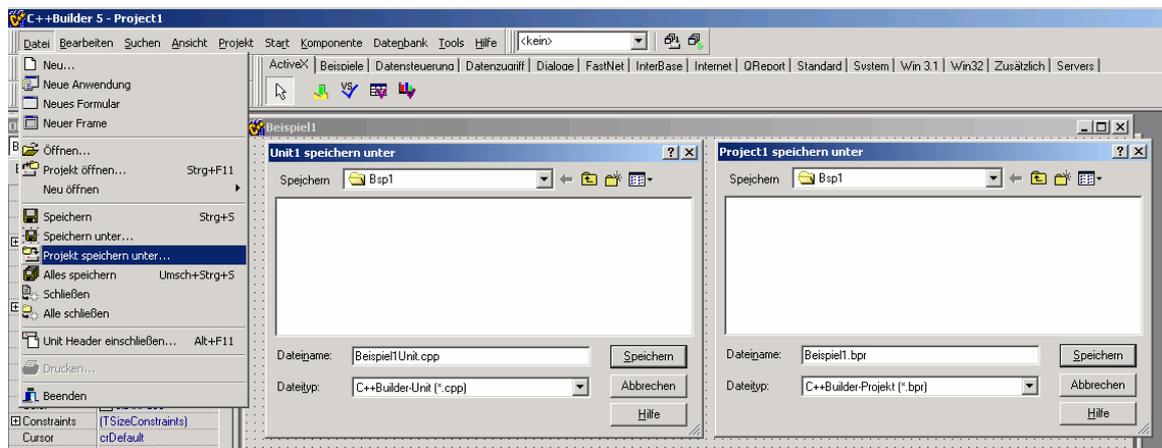
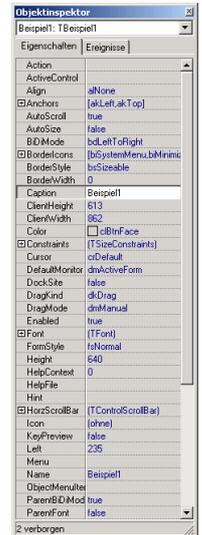
Wichtig:

Dir Unit darf nicht denselben Namen haben wie das Projekt selbst.
Grund: Es wird automatisch ein Hauptprogramm mit dem Namen des Projektes erzeugt. Ein ev. bereits bestehendes File mit dem gleichen Namen wird ohne Rückfrage überschrieben.

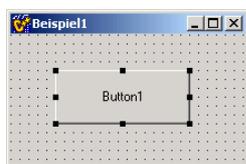
- Die Umbenennungen erfolgen beim Speicherdialog.

Beispiel: Erzeugen eines Projektes

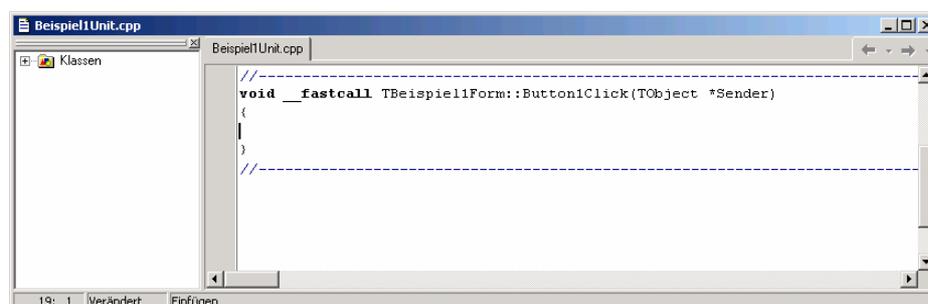
1. Borland C++-Builder starten. Ein leeres Projekt erscheint.
2. Formularname in Objektinspektor umbenennen
3. Speicherort für Projekt und Filenamen umbenennen. Dies erfolgt im Speicherdialog. Eventuelle Referenzen in Files werden automatisch nachgeführt.



4. Formular auf die benötigte Größe zurechtziehen.
5. Schaltknopf auf das Formular setzen.
Die Größe und Position kann durch Ziehen und Verschieben mit der Maus angepasst werden.
Alle anderen Eigenschaften und Verhalten des Knopfes werden nachträglich in den Objekteigenschaften geändert.



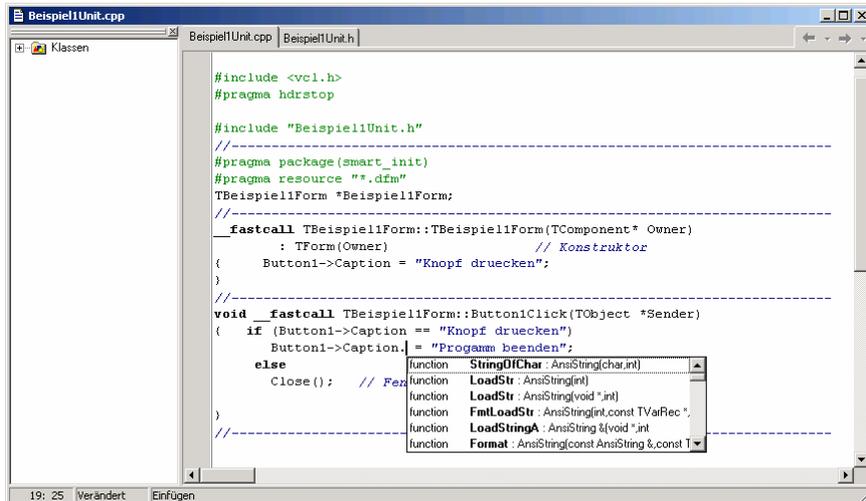
6. Durch Doppelklick auf den Knopf wird automatisch ein (leeres) Codegerüst für die Ereignisbehandlung „Knopf wurde gedrückt“ erzeugt:



7. Wir implementieren folgende Funktionalität:
Beim Programmstart -> Caption des Knopfes auf „Knopf druecken“ setzen.

Nach 1x Drücken -> Caption „Programm beenden“ wechseln.

Nach nochmaligem Druck Programm beenden.



```
#include <vc1.h>
#pragma hdrstop

#include "BeispielUnit.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TBeispielForm *BeispielForm;
//-----
fastcall TBeispielForm::TBeispielForm(TComponent* Owner)
: TForm(Owner) // Konstruktor
{
    Button1->Caption = "Knopf druecken";
}
//-----
void __fastcall TBeispielForm::Button1Click(TObject *Sender)
{
    if (Button1->Caption == "Knopf druecken")
        Button1->Caption = "Programm beenden";
    else
        Close(); // Fenster schließen
}
//-----
function StringOfChar : AnsiString(char,int);
function LoadStr : AnsiString(int);
function LoadStr : AnsiString(void *,int);
function FmtLoadStr : AnsiString(int,const TVarRec );
function LoadStringA : AnsiString (&void *,int);
function Format : AnsiString(const AnsiString &const T
```

8. Projekt kompilieren. Dazu im Menüpunkt Projekt „erzeugen“ wählen. Dies startet die Kompilation und den Linkvorgang. Der Fortschritt und Erfolg wird in einem Fenster angezeigt:

