

**MÄLARDALENS HÖGSKOLA**  
Institutionen för ekonomi och informatik

# **Komponenter med COM (och COM+/VC++ 7.0)**

Med Visual C++ 7.0 COM-komponent

**EI0230**  
**Komponentbaserad applikationsutveckling**  
**oktober 2003**

## Om denna sammanfattning

Denna sammanfattning innehåller motsvarande exempel i kapitel 4 i sammanfattningen *Komponenter med COM (och COM+)* fast med Visual C++ 7.0 i Visual Studio.NET v. 1.0. D.v.s. denna sammanfattning är ett komplement till ovan nämnda sammanfattning. ☺

Programvara från Microsoft som använts är *Visual Studio.NET v.1.0* och *Visual Basic 6.0*.

Jag är givetvis tacksam för alla konstruktiva synpunkter på sammanfattningens utformning och innehåll.

Eskilstuna, oktober 2003

Björn Persson, e-post: [bjorn.persson@mdh.se](mailto:bjorn.persson@mdh.se)  
Institutionen för ekonomi och informatik  
Mälardalens högskola  
<http://www.eki.mdh.se/personal/bpn01/>

## Innehållsförteckning

4	EXEMPEL: EN FÖRSTA KOMPONENT I C++ (VC++ 7.0) .....	3
4.1	KOMPONENTEN .....	3
4.2	TESTPROGRAM I VISUAL BASIC .....	8

## 4 Exempel: En första komponent i C++ (VC++ 7.0)

I detta kapitel kommer vi att göra om den enkla komponenten (i kapitel 4 i sammanfattningen *Komponenter med COM (och COM+)*) i Visual C++ 7.0 (C++.NET).

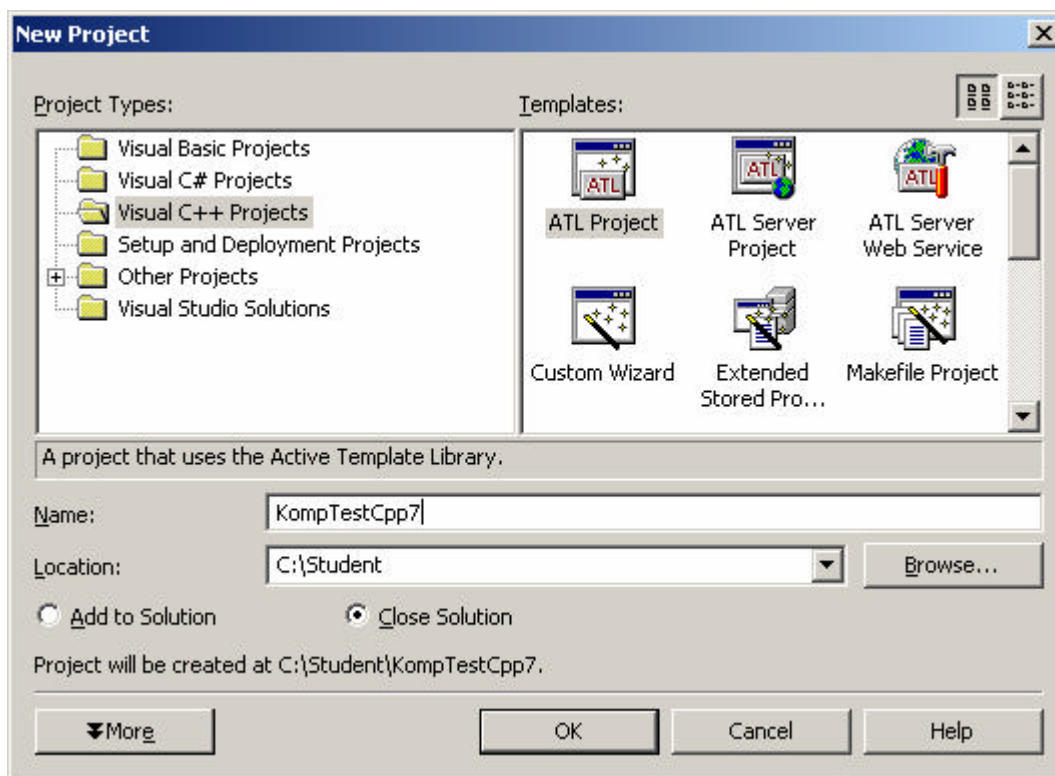
**Observera** att detta exempel är gjort för Microsoft Visual C++ 7.0 som är en del av Visual Studio.NET v. 1.0. Se sammanfattningen *Komponenter med COM (och COM+)* för exempel med Visual C++ 6.0.

**Observera** även att exemplet använder standard C++ (eller så standard det kan bli i Microsofts version av C++ ☺) och inte *Managed C++*, d.v.s. under .NET Framework!

### 4.1 Komponenten

Precis som exemplet i Visual C++ 6.0 så kommer vi göra en slumpalsgenerator som komponent. Komponentens uppgift är att ta fram ett slumpantal och sända det tillbaka. Komponentens kallar vi `SlumpCpp7` och projektet (COM-servern) `KompTestCpp7`.

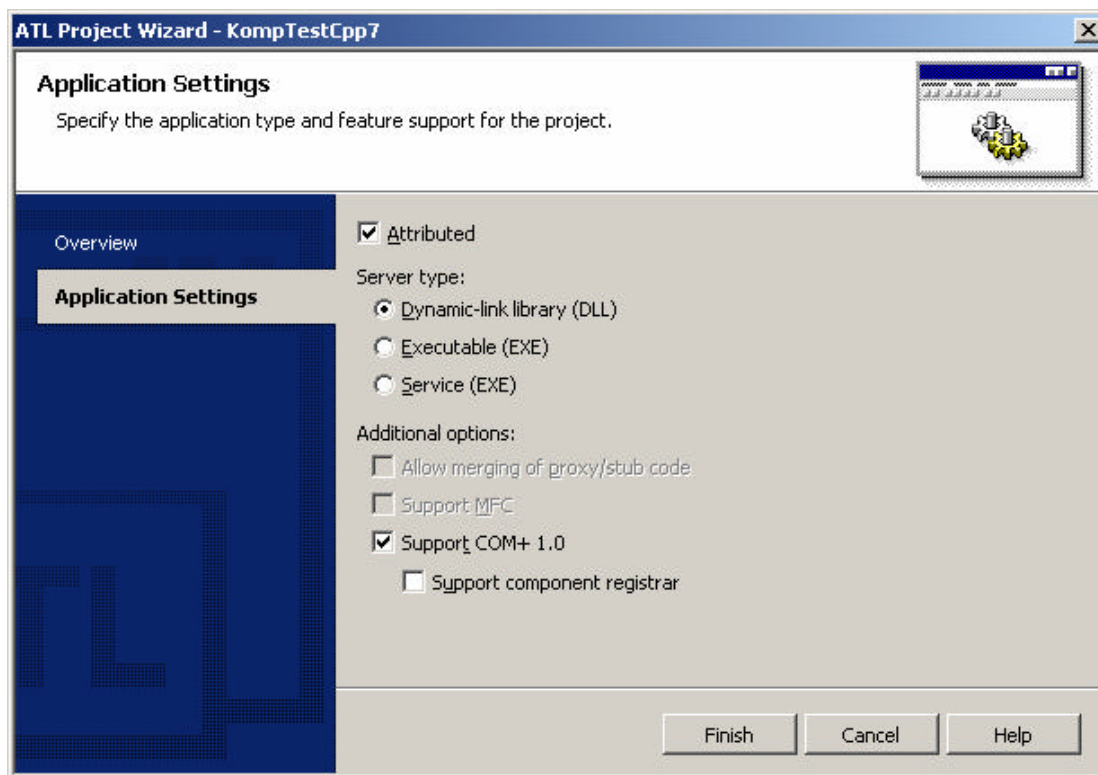
1. Starta Visual Studio.NET.
2. Öppna dialogrutan New Project (d.v.s. skapa ett nytt projekt).
3. Välj grenen **Visual C++ Projects** (Project Types:) och sen **ATL Project** (Templates:). Välj mapp (katalog) som du vill att projektets mapp ska skapas i i textrutan **Location:**.<sup>1 2</sup> Fyll sen i namnet på projektet (COM-servern) i textrutan **Name:**, t.ex. `KompTestCpp7`. Klicka på OK för att starta ATL-guiden.



<sup>1</sup> Genom att välja mapp först så kommer VC++ skapa en mapp med samma namn som projektet under den mapp som vi anger i Location:.

<sup>2</sup> När ni jobbar med Visual C++ 7.0 (d.v.s. Visual Studio.NET) så jobbar vi lämpligast mot lokal hårddisk för prestanda och för att undvika problem med säkerhetsinställningar.

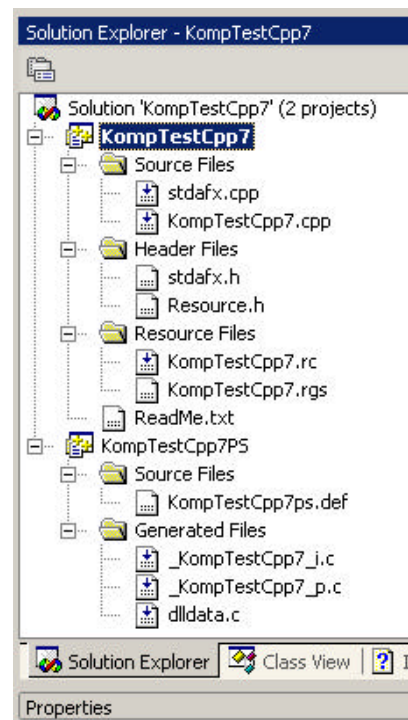
4. Markera fliken Application Settings till vänster i dialogruta. Kontrollera att radioknappen **Dynamic-link library (DLL)** är markerat under **Server Type**. Bocka för kryssrutan **Support COM+ 1.0**. Klick på **Finish** för att stänga dialogrutan **ATL Project Wizard**.



Ni har nu fått ”skalkod” för en COM-server i en DLL-fil. Om ni vill titta på antalet filer och innehållet i dem så klickar ni på fliken **Solution Explorer** underst i fönstret *Solution Explorer*, kallat projektfönstret i fortsättningen (fönstret till höger i IDE:n – se bild till höger). I filerna finns främst kod för att ladda och ladda ur DLL-filen samt andra funktioner som en DLL-fil måste stödja. För att kunna titta på IDL-filen (för senare jämförelse) måste vi först ”bygga” (kompilera) projektet. Filen heter samma som projektet men med ett ”understreck” (“\_”) som prefix och filändelsen .IDL samt vi måste titta på den ifrån Utforskaren.<sup>3</sup>

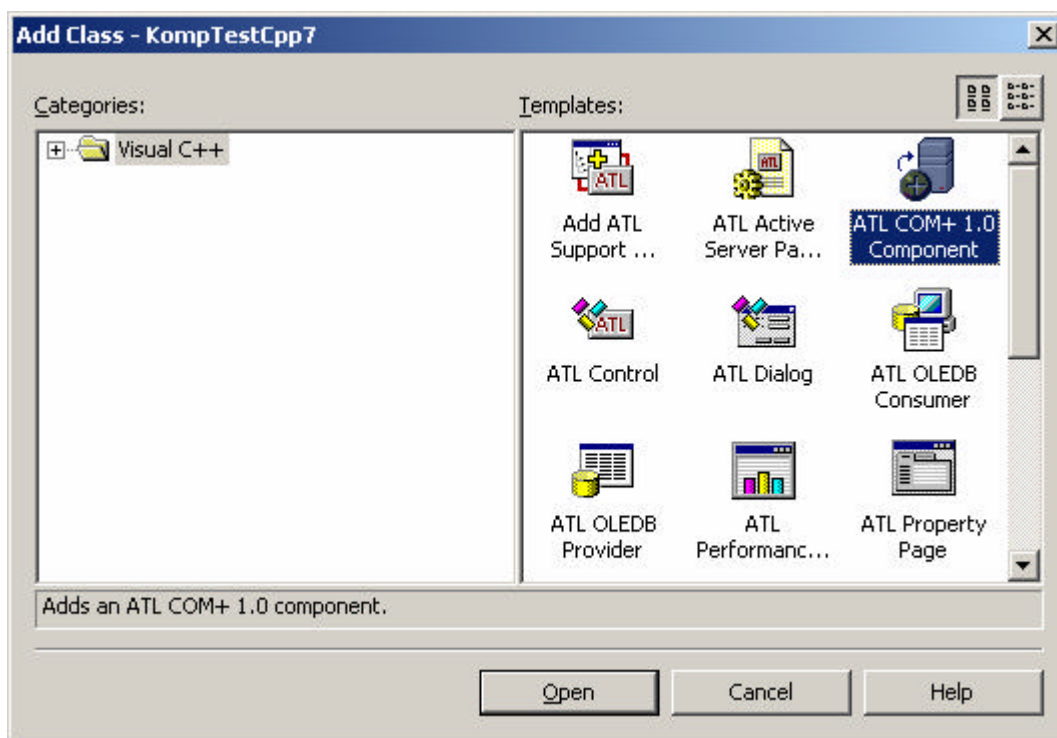
Nästa steg är att skapa klasser som ska bli komponenterna och för detta kommer vi använda fler ATL-guiden.

5. Välj fliken **ClassView** i projektfönstret och högerklicka på projektnamnet med fet text längst upp (*KompTestCpp7* om ni kallade projektet *KompTestCpp7*) samt välj **Add** och sen **Add Class...** (eller bara **Add Class...** från Project-menyn).

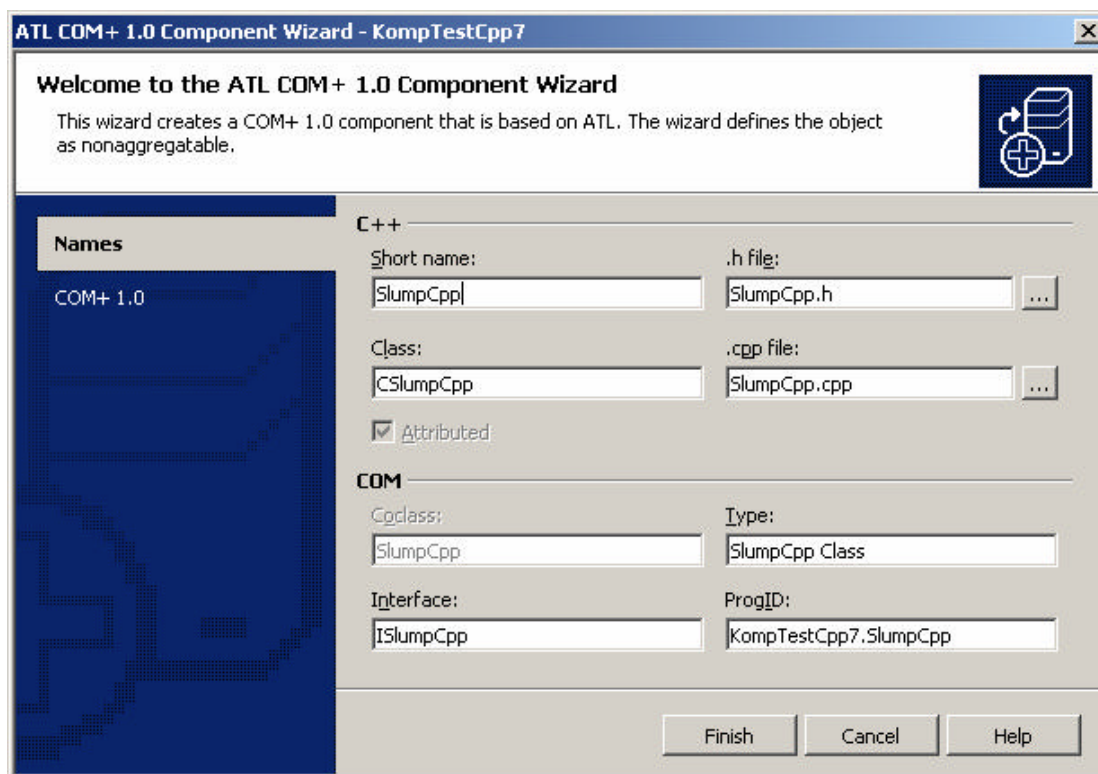


<sup>3</sup> I.a.f. när jag testade detta exempel. ☺

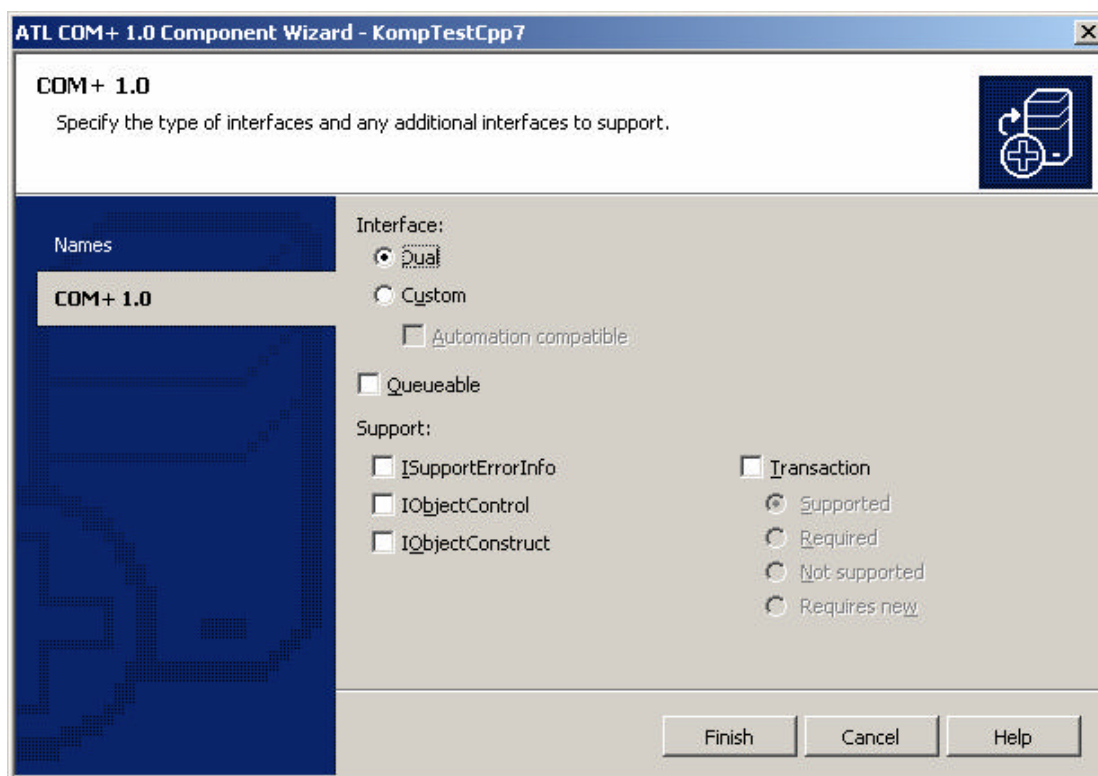
6. Markera alternativet **ATL COM+ 1.0 Component** i listrutan **Templates** (till höger) (samt **Visual C++** i listrutan **Categories** – till vänster – om det finns fler alternativ). Klicka sen på **Open** för att starta guiden.



7. Fyll i namnet (`SlumpCpp`) på komponenten i textrutan **Short Name**. Övriga fält fylls i automatiskt, bl.a. kommer gränssnittet att heta `ISlumpCpp`.



Klicka på fliken **COM+ 1.0** (till vänster i dialogruta). Här kan vi inte längre se att trådmodellen är *Apartment* men att komponenten kommer stödja *dual interfaces*.



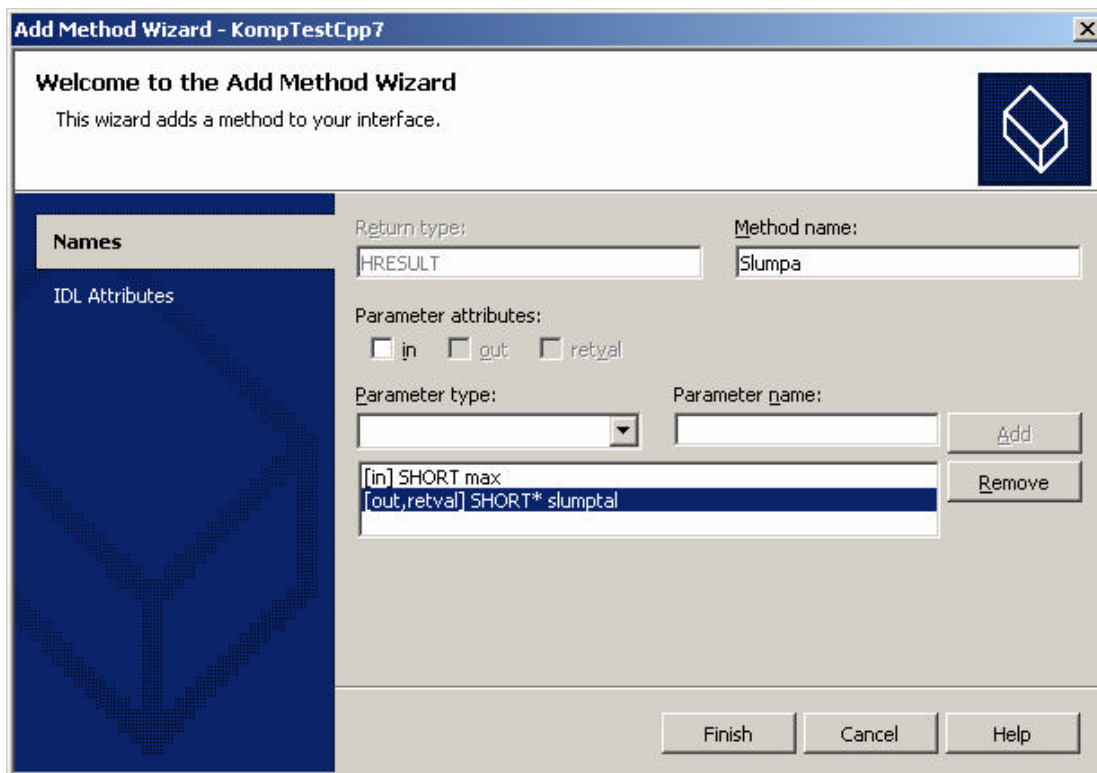
8. Klicka på **Finish** för att stänga dialogrutan **ATL COM+ 1.0 Component Wizard** och för att generera filer samt kod.

Nu har vi alltså skapat komponenten och i detta läge kan vi kompilera komponenten om vi vill. Om vi tittar på fliken **FileView**igen så ser vi att det tillkommit två filer: *header*-filen och kodfilen (.CPP) för klassen `SlumpCpp` (RGS-filen, för att registrerar komponenten i registret, fanns redan). För att kunna tittar på ändringar i IDL-filen så måste vi ”bygga” projektet igen. När vi nu laddar IDL-filen igen så ser vi att gränssnittet för komponenten lagts till.

Nästa steg är att lägga till metoden `Slumpa()` och skriva koden för metoden. Först kommer vi använda dialogrutan **Add Method Wizard** för att namnge metoden och ange parametrar.

9. Högerklicka på gränssnittet, *ISlumpCpp* om du namngav komponenten `SlumpCpp`, samt välj **Add** och sen **Add Method...** (eller markera gränssnittet och välj bara **Add Method...** från Project-menyn).
10. Fyll i namnet `Slumpa` i textrutan **Method Name:**. För att lägga till första parametern så bockar vi för kryssrutan **in**, väljer vi `SHORT` i listrutan **Parameter type:**, skriver `max` i textrutan **Parameter name:** samt klicka på knappen **Add**. Utparametrar kan endast läggas till efter att vi valt en pekare som parametertyp. Så därför väljer vi först `SHORT*` i listrutan **Parameter type:** samt därefter bockar vi för kryssrutan **retval (out)** (markeras automatiskt), fyller i namnet på parametern, `slumptal`, i textrutan **Parameter name:** och sist klicka på knappen **Add**.

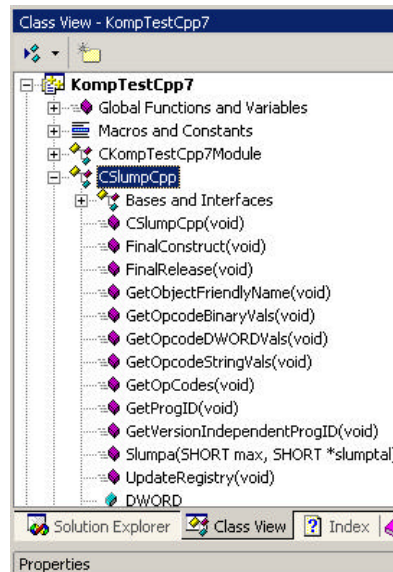




Stäng dialogrutan genom att klicka på **Finish**.

(Om allt har gjorts i en följd bör källkodsfilen för att implementera metoden visas. Annars följ instruktioner i punkt nedan.)

11. Expandera klassen *CSlumpCpp* (under fliken **ClassView**) (se bild till höger). Dubbelklicka på metoden *Slumpa()* (med den rosa ikonen framför) för att visa koden för metoden.



12. Skriv av koden nedan så metoden ser ut enligt följande:

```
STDMETHODIMP CSlumpCpp::Slumpa(int max, int *slumptal)
{
    // TODO: Add your implementation code here
    srand( (unsigned)time( NULL ) ); //skapa ett frö

    *slumptal = rand() % (max + 1); //returnera slumptalet

    return S_OK;
}
```

13. Lägg till `#include <time.h>` högst upp i filen `SlumpCpp.cpp` under raden `#include "SlumpCpp.h"`.

14. Kompilera DLL-filen genom att välja **Build Solution** från Build-menyn (eller tryck `Ctrl-Shift-B` på tangentbordet ☺). Kontrollera att följande visas i resultatfönstret längst

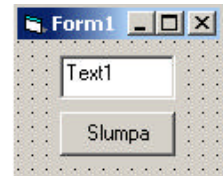
ner i IDE:n (*Build: 1 succeeded* är ”nyckelorden” då komponenten nu blivit registrerad i registret och kan användas direkt):<sup>4</sup>

```
----- Done -----  
Build: 1 succeeded, 0 failed, 1 skipped
```

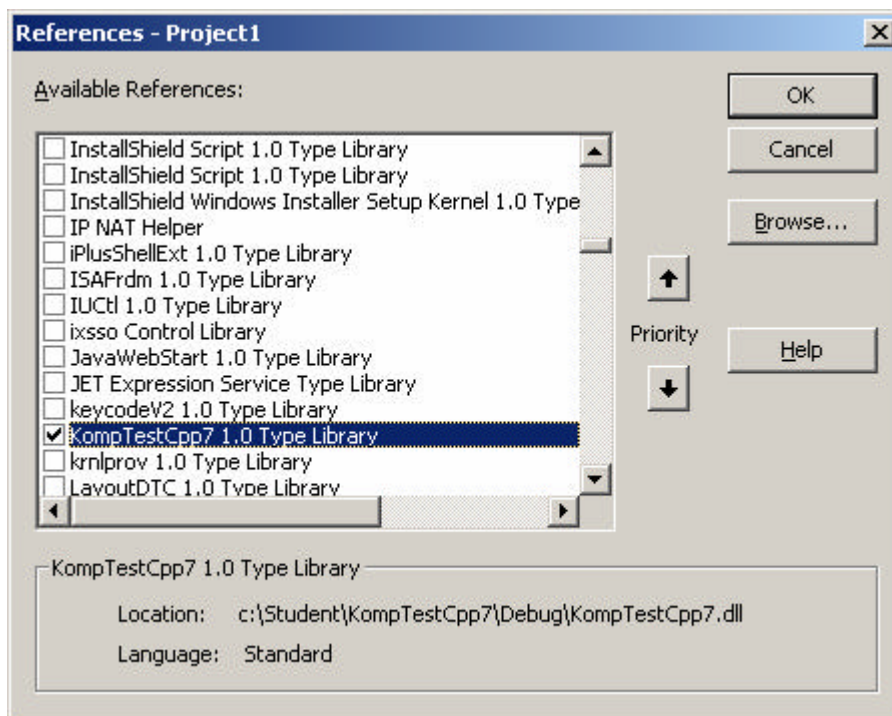
15. Komponenten är klar!

## 4.2 Testprogram i Visual Basic

För testprogrammet kommer vi använda Visual Basic igen (och vi skulle kunna använda det klientprojekt vi skapade för vår VC++ 6.0-komponent).



1. Skapa ett nytt Standard EXE-projekt och döp projektet till `KompTestCpp7Klient`.
2. Placera en textruta och en kommandoknapp på formuläret (se bild till ovan).
3. Sätt referens till den nya DLL-filen (COM-servern) genom att välja **References...** från Project-menyn. Observera att den nya COM-servern heter **KompTestCpp7 1.0 Type Library** (se bild nedan).



4. Dubbelklicka på formulärets bakgrund och på kommandoknappen (för att skapa ”skalkoden” för metoderna `Form_Load()` resp. `Command1_Click()`). Fyll i nedanstående kod (d.v.s. deklarerar variabel och implementera de två metoderna) så att koden blir enligt följande:

<sup>4</sup> Projektet som hoppades över var proxy-/stub-objekten, d.v.s. inget att oroa sig över. ©



```
Option Explicit
Dim objSlump As KompTestCpp7.CSlumpCpp

Private Sub Command1_Click()
    Text1.Text = objSlump.Slumpa(100)
End Sub

Private Sub Form_Load()
    Set objSlump = New KompTestCpp7.CSlumpCpp
End Sub
```

5. Kör programmet och klicka på kommandoknappen.

Som vi ser av koden så ger Visual C++ 7.0 (som standard) COM-servern ett ProgID som motsvaras av namnet på projektet (men utan ordet "Lib"), men ger komponenten ett prefix "C" (CSlumpCpp). Återigen skickar vi endast ett argument till metoden Slumpa(), inte två som implementationen av metoden i C++ har.

(Denna slumptalsgenerator fungerar ännu sämre än den i Visual C++ 6.0... varför vet jag inte i skrivande stund – september 2003 – men jag återkommer till det... ☺)