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. O

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 Institutionen för ekonomi och informatik Mälardalens högskola http://www.eki.mdh.se/personal/bpn01/

Innehållsförteckning

4	EXEN	MPEL: EN FÖRSTA KOMPONENTI 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++ O) 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 slumptalsgenerator som komponent. Komponentens uppgift är att ta fram ett slumptal och sända det tillbaka. Komponenten 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:.^{1 2} Fyll sen i namnet på projektet (COM-servern) i textrutan Name:, t.ex. KompTestCpp7. Klicka på OK för att starta ATL-guiden.

ew Project				
Project Types:		Templates:		000 b-
Visual Basic P Visual C# Pro Visual C+ Pro Setup and De Other Project Visual Studio	rojects ijects rojects eployment Projects ts Solutions he Active Template Library	ATL Project	ATL Server Project Extended Stored Pro	ATL Server Web Service
Name:	KompTestCpp7			
Location:	C:\Student			Browse
C Add to Solution	⊙ ⊆lose Solutio	n		
Project will be created	at C:\Student\KompTestC	pp7.		
1				1410

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

 $^{^{2}}$ 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**.

ATL Project Wizard - KompTe	stCpp7	×
Application Settings Specify the application type	and feature support for the project.	
Overview	Attributed	
Application Settings	Server type: © Dynamic-link library (DLL) © Executable (EXE) © Service (EXE)	
	Additional options:	
	I Support component registrar	
	Finish Cancel Help	>

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 projeckt men med ett "understreck" ("_") som prefix och filändelsen .IDL samt vi måste titta på den ifrån Utforskaren.³

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

 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 Projectmenyn).



³ I.a.f. när jag testade detta exempel. ©

 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.

acegories:	<u>T</u> emplates:		00 0-
⊕ · 🔄 Visual C++			- 🛃 - É
	Add ATL Support	ATL Active Server Pa	ATL COM+ 1.0 Component
	VATL		T
	ATL Control	ATL Dialog	ATL OLEDB Consumer
	_		
	ATL OLEDB Provider	ATL Performanc	ATL Property Page
Adds an ATL COM+ 1.0 component			

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

Welcome to the ATL This wizard creates a CO as nonaggregatable.	t Wizard - KompTestCpp7 . COM+ 1.0 Component Wizar M+ 1.0 component that is based on AT	d IL. The wizard defines the object
Names	C++	
COM+ 1.0	SlumpCpp	SlumpCpp.h
	Class:	.cpp file:
	CSlumpCpp	SlumpCpp.cpp
		SlumpCop Class
	I <u>n</u> terface:	Prog <u>I</u> D:
	I Character	KompTectCop7 SlumpCop

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*.

ATL COM+ 1.0 Compone	ent Wizard - KompTestCpp7		×
COM+ 1.0 Specify the type of inte	erfaces and any additional interfaces to supp	ort.	
Names	Interface:		
COM+ 1.0	C Custom		
	Automation compatible		
	Queueable		
	Support:		
	ISupportErrorInfo	Transaction	
	IObjectControl	🕝 Supported	
	IObjectConstruct	C Required	
		C Not supported	
		C Requires new	
		Finish Cancel	Help

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**.

Class View - KompTestCpp7

庄 🖘 Global Functions and Variables

--=∳ CSlumpCpp(void) --=∳ FinalConstruct(void)

👒 FinalRelease(void)

GetObjectFriendlyName(void)
GetOpcodeBinaryVals(void)

GetOpcodeDWORDVals(void)
 GetOpcodeStringVals(void)
 GetOpcodeStringVals(void)
 GetOpCodes(void)
 GetProgID(void)

UpdateRegistry(void)

🌄 Solution Explorer 🛛 🐼 Class View 🛛 🕐 Index | 🇔

DWORD

Properties

GetVersionIndependentProgID(void)
 Slumpa(SHORT max, SHORT *slumptal)

E Macros and Constants
 CompTestCpp7Module
 CSlumpCpp
 Bases and Interfaces

▶\$ ▼ 10 □ 20 KompTestCpp7

Add Method Wizard - Kom	pTestCpp7	×
Welcome to the Add This wizard adds a methor	Method Wizard d to your interface.	\bigotimes
Names	Return type:	Method name:
	HRESULT	Slumpa
IDL Attributes	Parameter attributes: in gut retyal Parameter type: [in] SHORT max [out,retval] SHORT* slumptal	Parameter name: Add Remove
		Finish Cancel Help

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:



- 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 ⁽²⁾). 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):⁴

```
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.

	orm1 💶 🗖	X
: : : : : :	Text1	
	Slumpa	
::		

- 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).

References - Project1	×
<u>A</u> vailable References:	ОК
InstallShield Script 1.0 Type Library InstallShield Script 1.0 Type Library InstallShield Windows Installer Setup Kernel 1.0 Type IP NAT Helper IPlusShellExt 1.0 Type Library ISAFrdm 1.0 Type Library IUCt1 1.0 Type Library JavaWebStart 1.0 Type Library JET Expression Service Type Library KompTestCop7 1.0 Type Library Knlprov 1.0 Type Library LavoutDTC 1.0 Type Library	Cancel <u>B</u> rowse ↑ ↑ ↑ ↓ ↓ ↓ ↓
KompTestCpp7 1.0 Type Library Location: c:\Student\KompTestCpp7\Debug\KompTe Language: Standard	estCpp7.dll

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

⁴ 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...