COM Instrukcja do ćwiczenia laboratoryjnego z przedmiotu Programowanie składnikowe w modelu COM sem 7

ĆWICZENIE 1: SKŁADNIKI WEWNĄTRZPROCESOWE

Składniki wewnątrzprocesowe (ang. in-process components), rezydujące w plikach DLL, należą do najczęściej stosowanych. Mimo, że nie zapewniają takiego poziomu stabilności, jak składniki zewnątrzprocesowe, ich użycie jest opłacalne ze względu na wysoka wydajność. Ponieważ działają w przestrzeni adresowej klienta, nie ma potrzeby przekazywania danych poprzez granice procesów; nie występują też elementy proxy i stub, a klient ma bezpośredni – poprzez wskaźnik – dostęp do tablicy wirtualnej vtable. W odróżnieniu od typowych składników – kontrolek ActiveX (które zresztą są odmianą składników wewnątrzprocesowych), typowe składniki wewnątrzprocesowe nie posiadają interfejsu użytkownika. Można je więc traktować jako uniwersalne, niezależne od języka biblioteki obiektowe.

W Visual Basicu składniki wewnątrzprocesowe tworzymy wybierając podczas startu programu, w oknie New **B** *visual Basicu* składniki wewiątrzprocesowe tworzynij wyoratące pozietal project ikonę ActiveX DLL. Utworzony projekt (początkowo o nazwie Project1) może zawierać jedną lub więcej klas komponentów. Domyślnie tworzona jest klasa Class1. W oknie nawigacyjnym projektu możemy zmienić nazwę projektu i klasy.

Składniki wewnątrzprocesowe, podobnie, jak wszystkie składniki COM, zawierają zestaw metod. Składniki realizowane na bazie technologii ActiveX mogą dodatkowo posiadać właściwości i zdarzenia. Wszystkie składniki tworzone w środowisku Visual Basic są takimi właśnie składnikami.

Właściwości to zmienne składowe obiektów - tak, jak się je precyzuje w programowaniu obiektowym. Dostęp do właściwości jest w istocie realizowany poprzez parę funkcji dostępowych (Get oraz Let/Set).

Metody to funkcje składowe obiektów.

Zdarzenia nie mają bezpośredniego odpowiednika w typowych językach obiektowych, np. w C++. Wyzwolenie zdarzenia przez składnik (serwer) powoduje wywołanie odpowiedniej metody po stronie klienta. W tym sensie wywołanie zdarzenia powoduje chwilowe odwrócenie relacji klient – serwer zachodzącej pomiędzy modułem wywołującym a składnikiem COM

Najprostszym sposobem wprowadzenia do modułu klasy komponentu nowej metody lub właściwości jest wy-VR wołanie opcji Tools/Add Procedure... W wyświetlonym oknie dialogowym można określić nazwę procedury, jej typ (metoda będąca procedurą lub funkcją, właściwość lub zdarzenie) oraz zakres.

Uwaga 1: właściwości są obsługiwane za pomocą pary funkcji Let i Get. Często wymaga to wprowadzenia dodatkowej zmiennej, przechowującej jej wartość. Należy ją zdefiniować ze specyfikatorem Private. Czasami jednak wartości właściwości są stosowane nie wprost – na przykład przeliczane, modyfikowane lub przenoszone do innych obiektów. Można też skasować jedną z funkcji, tworząc w ten sposób właściwość tylko do odczytu lub tylko do zapisu.

Uwaga 2: serwer COM nie zawiera implementacji zdarzeń (są one oczywiście implementowane przez klienta). Wyżej wspomniana opcja wprowadza do tekstu źródłowego deklarację zdarzenia. Wyzwolenie zdarzenia następuje w wyniku wywołania instrukcji RaiseEvent nazwa_zdarzenia [(parametry)].

Istnieje też bardziej zaawansowany, ale też wygodniejszy sposób rozbudowy klasy. Należy w tym celu wykorzystać składnik Add-in. Jeśli w menu Add-Ins nie jest dostępna opcja Class Builder Utility, wówczas należy wybrać opcję Add-Ins/Add-In Manager, a następnie włączyć odpowiedni składnik Add-In. Obsługa tego narzędzia jest prosta i nie wymaga szczegółowego opisu (przynajmniej dla studentów IV roku!)

PRZYKŁAD: KLASA SKŁADNIKA LICZNIK

Oto prosta, przykładowa klasa składnika wewnątrzprocesowego. Zawiera ona jedną właściwość - Wartość. Jedyna metoda tej klasy, Zmniejsz, zmniejsza Wartość o 1, jeśli jest ona dodatnia. Zastosowano też dwa zdarzenia: zdarzenie Zmieniony jest wywoływane każdorazowo po zmianie Wartośći, zaś zdarzenie Zero jest wywoływane przy próbie zmniejszenia Wartośći, gdy jest ona równa 0.

Private m_wartość As Integer	'lokalna kopia właściwości	Public Property Let Wartość (ByVal vData As Integer)
Public Event Zero()	' zdarzenia	m_wartość = vData
Public Event Zmieniony(war As Integer)		RaiseEvent Zmieniony(m_wartość)
		End Property
Public Sub Zmniejsz()		
If Wartość > 0 Then		Public Property Get Wartość() As Integer
Wartość = Wartość - 1		Wartość = m_wartość
Else		End Property
RaiseEvent Zero		
End If		
End Sub		

W powyższym przykładzie zdarzenia mogą być łatwo wykorzystane przez klienta celem, na przykład, odświeżenia interfejsu użytkownika. Oto przykładowa aplikacja, wyświetlająca bieżącą wartość licznika. Podczas startu licznik jest ustawiany na 10; metodę Zmniejsz wywołuje się poprzez przyciśnięcia przycisku. Zauważ, że zmienne będące obiektami (przedstawicielami klas komponentów) definiuje się za pomocą instrukcji Dim i inicjalizuje instrukcją Set; jeśli ze zmienną mają być związane zdarzenia, wówczas definiuje się je za pomocą instrukcji Dim WithEvents. Tak zdefiniowane zmienne pojawiają się na rozwijalnej liście w lewej, górnej części okna edycyjnego, co ułatwia tworzenie procedur obsługi zdarzeń. Nazwy takich procedur powinny być zgodne ze składnią obiekt_zdarzenie, tak, jak przy obsłudze zda-rzeń pochodzących od zwykłych okien.

	Dim WithEvents s As Licznik	Private Sub s_Zmieniony(war As Integer) Label1 = war
	Private Sub Form_Load() Set s = New Licznik	End Sub
Labell	s.Wartość = 10 End Sub	Private Sub s_Zero() MsgBox ("ZERO!") End Sub
	Private Sub Command1_Click() s.Zmniejsz End Sub	

VB Składniki wewnątrzprocesowe nie mogą być uruchamiane jak zwykłe aplikacje. Aby przetestować składnik wewnątrzprocesowy, najlepiej dodać nowy projekt do już istniejącego, i stworzyć w ten sposób grupę projektów. W tym celu wywołujemy opcję Add Project i w oknie dialogowym wybieramy pozycję Standard EXE. W oknie nawigacji widzimy teraz drugi projekt (możemy zmienić jego nazwę), będący normalnym projektem aplikacji. Aby przy próbie uruchomienia właśnie ten projekt został wywołany, musimy go ustawić jako projekt startowy. W tym celu korzystamy z menu kontekstowego odpowiedniej pozycji w oknie nawigacyjnym projektów, i wybieramy opcję Set as Start Up. Ostatnim krokiem przygotowawczym jest udostępnienie stworzonej przez nas wcześniej klasy dla nowej aplikacji: w tym celu zaznaczamy projekt aplikacji w oknie nawigacyjnym i wywołujemy opcję Project/References i tam zaznaczamy pozycję odpowiadającą nazwie projektu zawierającego składnik.

VB Aby skompilować składnik wewnątrzprocesowy, należy zaznaczyć projekt skłądnika w oknie nawigacyjnym, a nastepnie z menu wybrać pozycję *Make Xxx.dll*, gdzie *Xxx* to nazwa naszego projektu. Odtąd składniki będą dostępne dla całego systemu operacyjnego. Użycie składnika w obrębie modułów tworzonych w Visual Basicu wymaga wywołania opcji *Project/References* i zaznaczenia pozycji odpowiadającej nazwie projektu zawierającego potrzebne składniki.

PLAN ĆWICZENIA:

- uruchom i przeanalizuj podany wyżej przykład klasy *Licznik*,
- zrealizuj wskazany przez Prowadzącego składnik wewnątrzprocesowy,
- zrealizuj aplikację implementującą interfejs użytkownika dla stworzonego składnika.

Stworzony w ramach ćwiczenia interfejs użytkownika powinien być sterowany zdarzeniami generowanymi przez składnik, tak, jak to zademonstrowano w klasie *Licznik*.

Pamiętaj, że w ramach składnika implementujemy niewidoczną, algorytmiczną funkcjonalność aplikacji!

Propozycje składników wewnątrzprocesowych do realizacji w ramach ćwiczenia:

- Automat do napojów (zawiera metody i właściwości pozwalające na zasilenie magazynku w napoje, wrzucanie monet i wybór napoju, oraz zdarzenia informujące interfejs użytkownika między innymi o wydaniu napoju, wydaniu reszty i o wystąpieniu braku określonego napoju).
- Automat do rezerwacji biletów kolejowych (zawiera właściwości i metody pozwalające na określenie informacji o dostępnych pociągach i liczbie wolnych miejsc, a także na zarezerwowanie miejsca na określony pociąg, oraz zdarzenia informujące interfejs użytkownika o wyniku rezerwacji oraz o wyczerpaniu miejsc na poszczególne pociągi).
- Wielofunkcyjny zegarek z budzikiem z możliwością ustawiania ustawienia jednocześnie wielu alarmów, różnego typu (rozpoznawanych np. poprzez różne identyfikatory), cyklicznych i jednorazowych. Moduł powinien być w stanie obsłużyć zarówno aplikację typu "Budzik", jak i klasyczny scheduler.
- System bankowy aplikacja kliencka powinna implementować interfejs użytkownika bankomatu.
- Wypożyczalnia filmów on-line.
- Jednoręki bandyta.

Czas realizacji ćwiczenia: dwa spotkania

ZAWARTOŚĆ SPRAWOZDANIA:

Sprawozdanie powinno zawierać dokładną specyfikację (opis) wszystkich metod, właściwości i zdarzeń stworzonych w ramach składnika oraz aplikacji interfejsu użytkownika.