

ĆWICZENIE 3: KONTROLKI ACTIVEX

Od pozostałych typów obiektów *COM* kontrolki *ActiveX* odróżniają się na ogół faktem posiadania interfejsem użytkownika (choć oficjalna specyfikacja tak tego nie określa). Oprogramowanie korzystające z kontrolki (klient) nazywa się kontenerem. Kod wykonywalny kontrolki *ActiveX* umieszczony jest w specjalnym pliku o rozszerzeniu *OCX*. Technicznie rzecz biorąc plik ten nie różni się od zwykłego pliku *DLL*. Warto też pamiętać, że kontrolki *ActiveX* są w istocie składnikami wewnątrzprocesowymi. Obecność interfejsu użytkownika wymaga w rzeczywistości obsługi wielu różnych standardowych interfejsów – wszystkie szczegóły jednak są na ogół ukryte przed programistą.

VB W *Visual Basicu* kontrolki *ActiveX* tworzymy wybierając podczas startu programu, w oknie *New project* ikonę *ActiveX Control*. Utworzony projekt (początkowo o nazwie *Project1*) może zawierać jedną lub więcej kontrolki. Domyślnie tworzona jest kontrolka *UserControl1*. W oknie nawigacyjnym projektu możemy zmienić nazwę projektu i klasy.

Cały proces projektowania kontrolki nie odbiega od procesu projektowania zwykłego składnika wewnątrzprocesowego; dotyczy to również testowania utworzonej kontrolki. Istotniejszą różnicą jest jedynie fakt, iż włączenie kontrolki do projektu klienta wymaga jej wskazania w oknie wyświetlonym za pomocą opcji *Project/Components* (a nie – jak w zwykłych składnikach – *References*). Spowoduje to dodanie ikony kontrolki do palety elementów wizualnych, które można umieszczać na formatce – odtąd kontrolka jest gotowa do użycia. Dla tak wykorzystanej kontrolki dostępne są wszystkie metody i właściwości zdefiniowane w projekcie kontrolki.

Kontrolki można też testować na stronie WWW – wystarczy nacisnąć F5 i wybrać opcję *Start component!*

Uwaga! Kontrolka *ActiveX* nie jest dostępna dla innych projektów w czasie, gdy jest ona poddawana edycji w oknie projektowym *Visual Basica!*

TRWAŁOŚĆ WŁAŚCIWOŚCI:

Przydatną funkcją kontrolki *ActiveX* jest trwałość ich właściwości (ang. *persistency*). Trwałe właściwości (*persistent properties*) to właściwości, których wartości są przechowywane poza okresami aktywności kontrolki. Jeśli chcemy ustawić właściwości w arkuszu właściwości *Visual Basica* lub konfigurować je ze strony WWW, musimy uczynić je trwałymi. Z drugiej strony trwałość właściwości wydłuża czas ładowania i inicjalizacji kontrolki, tak więc nie należy mnożyć trwałych właściwości bez potrzeby.

VB Aby obsłużyć trwałość właściwości w VB, należy obsłużyć metody kontrolki o nazwach: *ReadProperties* i *WriteProperties*. Są one wywoływane odpowiednio na początku i na końcu okresu wykorzystania kontrolki. Obydwie metody pobierają parametr będący obiektem klasy *PropertyBag*; posiada on zdolność przechowywania informacji w czasie, gdy kontrolka jako taka nie istnieje; w szczególności wykorzystać należy funkcje *ReadProperty* i *WriteProperty* – osobno dla każdej właściwości, której chcemy zapewnić trwałość.

Uwaga! Wszystkie właściwości kontrolki są wyświetlane w okienku *Properties Visual Basica*, jednak właściwości, którym nie zapewniono trwałości, nie są poprawnie obsługiwane!

PLAN ĆWICZENIA:

Ćwiczenie polega na przerobieniu stworzonej na poprzednich zajęciach aplikacji tak, by była zrealizowana w postaci kontrolki *ActiveX*. Należy ją przetestować zarówno w środowisku prostej aplikacji testowej – kontenera, jak i na stronie WWW. Wprowadź też jakieś trwałe właściwości i steruj ich wartościami z poziomu kodu HTML.

Wskazówka: By dowiedzieć się, jak należy wstawić kod HTML instalujący kontrolkę, uruchom ją na stronie WWW z poziomu *Visual Basica* i podglądaj wygenerowaną automatycznie stronę.

Dla zaawansowanych: Jeżeli chcesz się dowiedzieć, w jaki sposób strona *HTML* może wchodzić w interakcję z kontrolką *ActiveX*, sprawdź stronę <http://www-zo.iinf.polsl.gliwice.pl/~jfrancik/lectures/download/mfc\ox.html> i przeanalizuj jej kod w HTML. Zawiera ona fragmenty w języku *VBScript* współdziałające z kontrolką *ActiveX*. Nazwy *GameOver*, *NowMove*, *IllegalMove* odnoszą się do zdarzeń generowanych przez kontrolkę.