

KOMPONENS fejlesztése

Delphi

-ben

A Delphi alatt végzett fejlesztések azért olyan hatékonyak, mert programjainkat komponensekből építhetjük fel, amelyek igen sok feladatot megoldanak külön programozás nélkül. A Delphiben azonban nem csak a vele adott komponenseket használhatjuk, hanem igen sok le is tölthető az Internetről, sőt mi magunk is készíthetünk ilyeneket, amelyeket utána ugyan úgy használhatunk, mint a beépítetteket.

Nagyon érdekes, hogy a Delphi alatt fejlesztők nagy része még soha nem írt komponenset, pedig ezzel igen megkönnyítené és meggyorsítaná saját munkáját. Ebben a sorozatban azt szeretném megmutatni, hogy hogyan lehet komponenset készíteni.

Én a komponensírást sokkal izgalmasabbnak, szórakoztatóbbnak tartom, mint magát a program írását. Sokan azért nem írtak eddig komponenset, mert igazából soha nem jártak utána annak, hogy hogyan is kell ezeket elkészíteni.

Talán kezdjük egy igen egyszerű, de ennek ellenére hasznos komponens megírásával! A Delphivel adott StatusBar komponensre nem tudunk más komponenseket ráhelyezni, pedig időnként igen kellemes lenne egy-egy kép, vagy nyomógombot elhelyezése a felületén. Írjunk egy olyan komponenset, ami ezt a hiányosságot kiküszöböli.

A feladat megoldása igen egyszerű, a StatusBar stílusának rendelkeznie kell a csAcceptsControls tulajdonsággal. Ez a tulajdonság teszi lehetővé, hogy más elemeket ráhelyezzünk az adott komponensre.

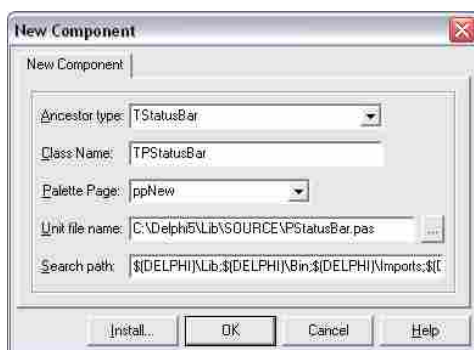
Akkor kezdjünk is hozzá az új komponensünk elkészítésének. Válasszuk ki a Component->New Component menüpontot. Itt hozhatjuk létre a legegyszerűbben az új komponenseinket. Az új TPStatusBar komponensünket a TStatusBar komponensből származtatjuk, így tulajdonságai meg fognak egyezni az őstulajdonságaival, csak a stílusát fogjuk most kiegészíteni egy új elemmel!

Az „Ancestor type:”-nál az őstulajdonságok megadnak,

vagyis azt, amiből származtatni kívánjuk az új komponensünket. Jelen esetben ez a TStatusBar. (A származtatás osztályairól majd később bővebben is szó lesz!)

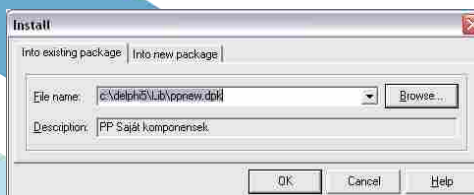
A „Class name:” mezőbe az új komponensünk nevét kell megadni, jelen esetben például írjuk be a TPStatusBar-t! A „Palette page:” mezőben annak a palettának a nevét adjuk meg, vagy válasszuk ki, amelyekre az új komponens helyezni szeretnénk. Itt természetesen új nevet is megadhatunk, vagy a már meglévőkből is választhatunk!

A „Unit file name” mezőben az új unit neve szerepel. Ezt a Delphi automatikusan generálja a komponens nevéből, de természetesen ez is átírható! A „Search path:”-ban a keresési útvonalat kell megadni, ebben szerepelni kell a komponensünk alkönyvtárának!



Amennyiben ezzel készen vagyunk, létre is hozhatjuk új komponensünket. Az „OK” gomb lenyomására elkészül az új unit, az „Install” megnyomására az új komponens hozzáadódik a megadott csomaghoz, és felkerül a kiválasztott palettára. Most nyomjuk meg az „Install” gombot, hogy a komponensünk a palettán megjelenjen.

Ekkor megnyílik egy ablak, ahol megadhatjuk azt a csomagot, amibe be szeretnénk rakni az új komponensünk.



Az első fülön egy már meglévő csomagot választhatunk ki, az „Into new package” fülnél pedig egy új csomagot hozhatunk létre. A „File name” mezőben meg kell adni a csomag nevét, (ez mindig DPK kiterjesztésű), a „Description” sorban pedig szövegesen megadhatjuk a csomag rövid leírását.

Az OK gomb megnyomásával létre is jön az új komponensünk forrása, és már bele is fordíthatjuk a kiválasztott csomagba! Amennyiben ezt tettük, az új komponensünk meg fog jelenni a kiválasztott palettán, és már ebben az állapotában teljesen megegyezik a TStatusBar komponenssel, de már létrejött, és használható!

Talán nagyon röviden nézzük meg a komponensünk forrását:

```
TPStatusBar = class(TStatusBar)
private
    declarations
    protected
    {Protected
declarations }
    public
    {Public
declarations }
    published
    {Published
declarations }
end;

procedure Register;

implementation

procedure Register;
begin
    RegisterComponents('ppNew'
, [TPStatusBar]);
end;
```

Az első sorban jön létre az új osztály a TStatusBar komponensből. Ebben az állapotban az osztályunk teljesen megegyezik az őstulajdossal. (Származtattuk!)

Van egy érdekes „Register” függvény is a forrásban, tulajdonképpen ez a sor gondoskodik arról, hogy a komponensünk megjelenjen a palettán. Ezt a „RegisterComponents,” függvény

végzi, ahol először meg kell adni a paletta nevét, majd fel kell sorolni a palettára kerülő komponenseket. (Többet is megadhatunk itt!)

Azért azt már láthatjuk, hogy egy új komponens, legalábbis az alap osztály létrehozása nem is olyan nehéz dolog.

Akkor most folytassuk a munkát, és változtassuk meg a komponensünk stílusát úgy, hogy elemeket tudjon fogadni. A Create konstruktort kell ehhez átírunk. Először is az objektum „public” részében írjuk felül az őstulajdonság függvényét:

```
public
    constructor Create(AOwner:
TComponent); override;
```

Az „override” az őstulajdonság felülírása miatt kell! Nagyon fontos, származtatáskor soha nem szabad elhagyni! Ezzel biztosítjuk azt, hogy az őstulajdonság azonos nevű függvénye is lefusson!

```
constructor
TPStatusBar.Create(AOwner:
TComponent);
begin
    inherited;
    ControlStyle:=ControlStyle+[cs
AcceptsControls];
end;
```

A Create függvényben az „inherited” soral gondoskodunk arról, hogy az őstulajdonság konstruktora fusson le először, ezzel létrejön az osztály, majd állítjuk stílusát.

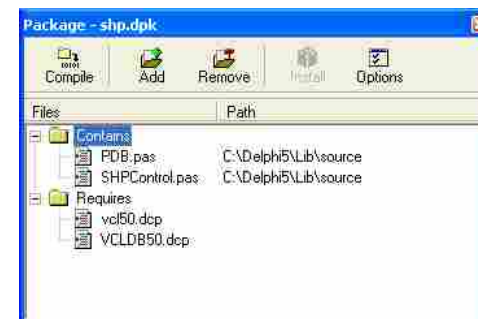
Ezzel már készen is vagyunk az első saját készítésű komponensünkkel! Ha megnézzük a palettát, ahová az új komponensünk került, láthatjuk, hogy a képe megegyezik a StatusBar képevel, (amiből származtattuk) de ez könnyen megváltoztatható az Image Editor programmal, és egyedi képet készíthetünk komponenseinknek.

Most már láttuk, hogy hogyan tudunk gyorsan új komponens létrehozni, de nézzük meg azt is, hogy hogyan tudunk egy már kész komponenset, vagy komponens csomagot beilleszteni a Delphibe.

Komponensekhez két formában juthatunk. Egyrészt kaphatunk egy „Delphi Package”-et, vagyis egy komponens csomagot, ami tartalmazza azokat az állományokat, amire a

csomaghoz szükség van.

A csomagot a kiterjesztéséről ismerhetjük meg, ez a „.DPK”. Amennyiben egy letöltött állományban „.DPK” kiterjesztésű fájl is találunk, igen könnyű dolgunk van, ezt kell csak beilleszteni a Delphibe. Másoljuk be az állományokat egy alkönyvtárba, majd az Fájlok->Open menüpontot választva válasszuk ki a „.DPK” kiterjesztésű állományt, és nyissuk meg. Ekkor a megnyíló „Package” ablakban láthatjuk a csomagban található komponenseket, amikből természetesen több is lehet!



Az első dolgunk a csomag lefordítása a „Compile” gomb megnyomásával. A hibátlan fordítás után, az „Install”-al tudjuk beilleszteni a csomagban lévő komponenseket a Delphi alá. Amennyiben a fordítás alatt olyan hibaüzenetet kapunk, hogy nem talál a fordító valamilyen meglévő állományt, akkor a Tools->Environment ablak Library fülén adjuk hozzá a „Library Path”-hoz a komponensünk alkönyvtárát is! (Ez elég gyakori probléma szokott lenni!)

Amennyiben itt minden rendben volt, akkor az új komponensek már meg is jelennek a palettán, és használhatjuk is őket!

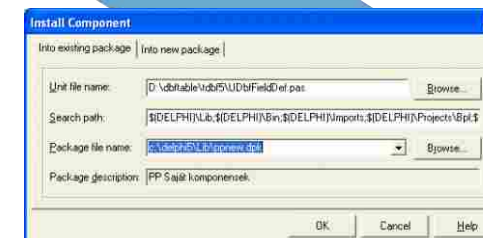
A „Package” ablakban lehetőség van komponensek hozzáadására, és törlésére is, a megfelelő gomb megnyomásával.

A másik esetben nincs „.DPK” állomány, ilyenkor nekünk kell létrehozni, vagy beletenni a komponens egy csomagba. Azért ez sem egy bonyolult dolog, nézzük erre is egy példát.

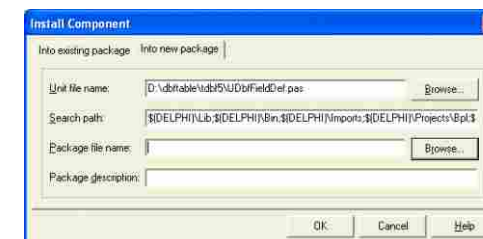
Készítsünk egy olyan csomagot, amiben ezeket a letöltött komponenseket fogjuk gyűjteni. A komponens állományait először másoljuk be egy alkönyvtárba.

Ezek után hozzunk létre egy új csomagot, amibe bele fogjuk tenni az új

komponensünket. Ezt a „Component->Install component” menüpontban tehetjük meg:



A megnyíló ablakban egyrészt már meglévő csomagba („Into existing package”), vagy egy új csomagba tehetünk bele egy új komponens („Into new package”). Először is adjuk meg a komponens forrását, amit a csomagba kívánunk tenni (Unit file name). A „Search path”-hoz adjuk hozzá a komponensünk alkönyvtárát, amennyiben az még nem szerepel benne. (Ezekben az alkönyvtárakban keresi az állományokat a fordító!) Már létező csomag esetén a lenyíló ablakból válasszuk ki a csomagot, amibe a komponens kívánjuk helyezni. Új csomag létrehozása esetén válasszuk az „Into new package” fület:



Itt a „Package file name” mezőben adjuk meg az új csomag pontos nevét „.DPK” kiterjesztéssel, a „Package description” sorban pedig a csomag szöveges megnevezését, ami utal a csomag tartalmára. Ide például írhatjuk az „Internetről letöltött komponensek” szöveget! Az „OK” gombra kattintva a komponens bekerül a már meglévő, vagy az új csomagba. Ezután megnyílik a csomag, benne a komponensekkel. Itt már az előbb bemutatott eljárással le kell fordítani a csomagot, és installálni kell a Delphi alá. Ha minden rendben volt, akkor ne felejtjük elmenteni a kész csomagot a benne lévő komponensekkel!