

# **Programarea Aplicațiilor Windows – curs 9**

**Conf. dr. Cristian CIUREA**  
**Departamentul de Informatică și Cibernetică Economică**  
**Academia de Studii Economice București**  
**[cristian.ciurea@ie.ase.ro](mailto:cristian.ciurea@ie.ase.ro)**

# Agenda

1. Lucrul cu Clipboard-ul
2. Operația de Drag and Drop
3. Data Binding

## **Lucrul cu Clipboard-ul**

Punerea și scoaterea unui obiect în/din Clipboard presupune ca obiectul să știe să se transforme în mai multe formate pentru a se adapta la capacitățile de înțelegere ale aplicației care îl va prelua din Clipboard.

## Lucrul cu Clipboard-ul

Obiectele se pun în Clipboard printr-un apel al metodei:

```
Clipboard.SetDataObject( );
```

Transferul de date în Windows Forms este permis pentru clasele care implementează interfața **IDataObject**.

Aplicația care folosește obiectele puse în Clipboard cere mai întâi o referință la subobiectul **IDataObject**, invocând metoda: **Clipboard.GetDataObject( );**

## Lucrul cu Clipboard-ul

Se definește:

```
IDataObject o =  
Clipboard.GetDataObject();
```

Având la dispoziție referința la interfața **IDataObject**, aplicația se interesează dacă ceea ce există în Clipboard este de un anumit tip de obiect sau se poate converti într-un anumit tip. Astfel, se invocă metoda **GetDataPresent()**, având ca parametru tipul căutat:

```
bool vb =  
o.GetDataPresent(typeof(Bitmap));
```

## Lucrul cu Clipboard-ul

Daca se raspunde afirmativ în urma acestui apel, adică "vb" are valoare "true", atunci se poate lansa o cerere de extragere a obiectului din Clipboard, folosind metoda `GetData()`:

```
if (o.GetDataPresent(typeof(Bitmap)))  
{  
    Bitmap img =  
        (Bitmap)o.GetData(typeof(Bitmap));  
    g.DrawImage(img, panel1.ClientRectangle);  
}
```

# Lucrul cu Clipboard-ul

Etape lucru cu Clipboard:

- pentru a pune obiecte în Clipboard:  
`Clipboard.SetDataObject(object);`
- pentru a realiza transferul de date între formulare, clasele trebuie să implementeze interfața `IDataObject`;
- pentru a lua obiecte din Clipboard se utilizează `Clipboard.GetDataObject();`
- se extrage conținutul prin metoda `IDataObject.GetData();`
- înainte de a extrage date din Clipboard le verificăm dacă sunt de tipul celor cu care aplicația știe să lucreze  
`Object.GetDataPresent(format_verificat);` 7

## Operatia de Drag and Drop

Inițierea operației de Drag and Drop se face printr-un apel al metodei **DoDragDrop()** și se sincronizează cu evenimentul **MouseDown**, care oferă informații detaliate despre poziția mouse-ului. În urma apelului **DoDragDrop()** se stabilesc datele ce fac obiectul drag-ării, precum și drepturile acordate acestora: copiere sau mutare.



## Operatia de Drag and Drop

Următorul eveniment care intervine în operația de Drag and Drop este **DragEnter**, declanșat la intrarea în regiunea unui control care permite dropping (are proprietatea **AllowDrop** pe "true"). Cel de-al treilea eveniment este **DragDrop** și se generează în momentul relaxării mouse-ului în timpul derulării operației de dragare.

## Operatia de Drag and Drop

Astfel, prin apelul `DoDragDrop()` se stabilește sursa de date ce face obiectul drag-ării, precum și efectele permise pentru destinație (Copy sau Move). La intrarea mouse-ului pe suprafața controlului destinație (evenimentul `DragEnter`) se testează compatibilitatea formatului datelor dragate cu formatul destinației posibile.

# Data Binding

Legarea datelor de interfata utilizator (Data Binding) poate fi:

- **simplă** (simple data binding);
- **complexă** (complex data binding).

Legarea datelor se consideră simplă, dacă de proprietatea unui control se leagă un singur element dintr-o colecție, și complexă, dacă controlul este capabil să afișeze toată colecția de date.

# Data Binding

Controalele au o colecție **DataBindings**, de tip **ControlBindingsCollection**. Această colecție conține obiecte de tip **Binding**, reprezentând legături ale diverselor proprietăți ale controlului cu valori ale unor variabile.

# Data Binding

**Simple Binding** conectează un singur element dintr-un vector de o singură proprietate a unui control.

**Complex Binding** conectează un control cu întreaga colecție de date. De exemplu, un **GridView** afișează o întreagă tabelă dintr-o bază de date, cu toate câmpurile ei.

De legarea complexă beneficiază doar controalele orientate pe colecții, precum **ComboBox**, **ListBox**, **GridView**.

# Data Binding

Se face diferența dintre:

- **legare unidirecțională**, când modificările din sursa de date sunt operate și în controale, dar modificările din controale nu au efect asupra datelor din sursă;
- **legare bidirecțională**, când modificările din sursa de date sunt vizibile și în controale, iar modificările operate în controale se observă și asupra datelor din sursa de date.

## Bibliografie

- [1] I. Smeureanu, M. Dârdală, A. Reveiu – *Visual C# .NET*, Editura CISON, București, 2004.
- [2] C. Petzold – *Programming Microsoft Windows with C#*, Microsoft Press, 2002.
- [3] L. O'Brien, B. Eckel – *Thinking in C#*, Prentice Hall.
- [4] J. Richter – *Applied Microsoft .NET Framework Programming*, Microsoft Press, 2002.
- [5] <http://acs.ase.ro/paw>