

ASP.NET

A. Lorenzi, E. Cavalli

**INFORMATICA PER ISTITUTI TECNICI
TECNOLOGICI**

Atlas

Copyright © Istituto Italiano Edizioni Atlas

Web server

- Software per la gestione di un computer *host*, o *server di rete*, basato sul protocollo HTTP
- Web server standard per i sistemi Microsoft Windows: **IIS** (*Internet Information Services*)
- directory **Inetpub** sottocartella **wwwroot**
- Avvio da *Pannello di controllo* (Sistema e sicurezza, Strumenti di amministrazione)
- Per controllare che il server Web sia attivo:

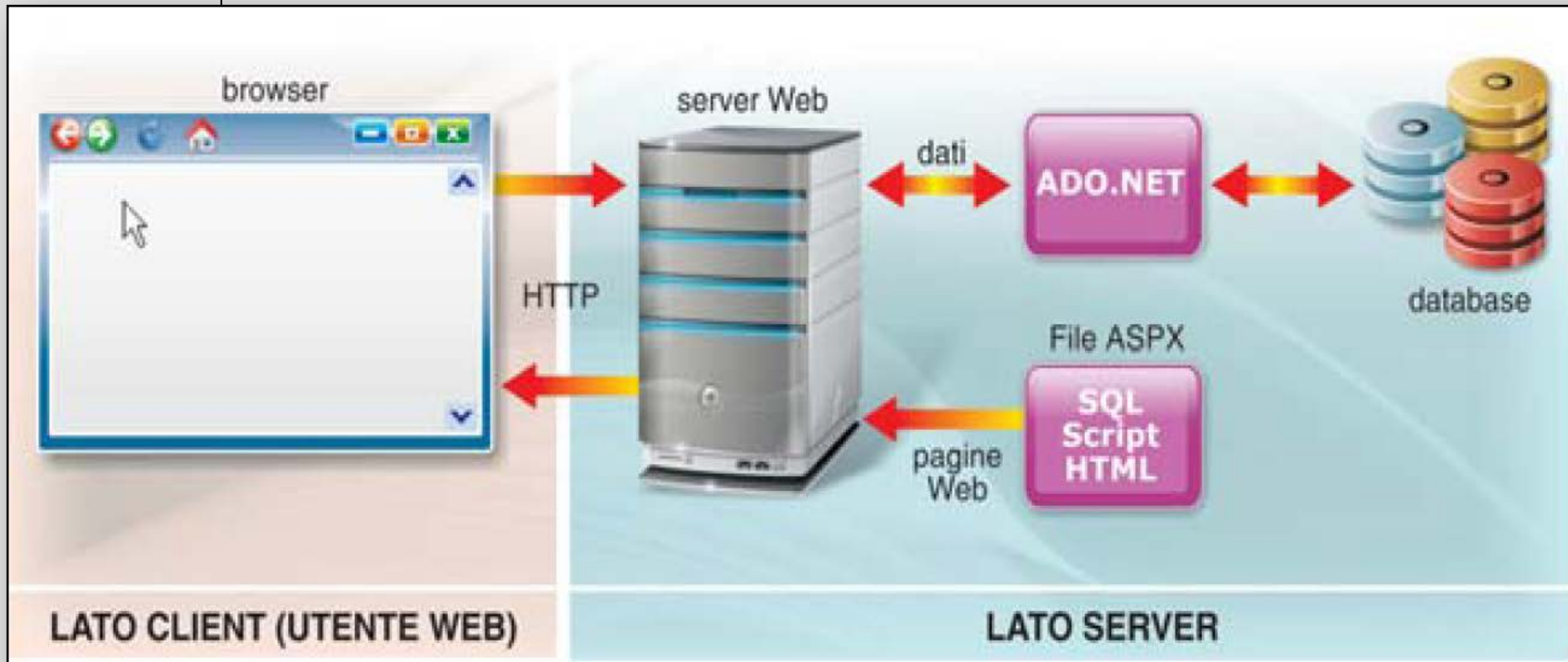
<http://localhost>

Tecnologia ASP.NET

- **ASP.NET** (*Active Server Pages*): tecnologia Microsoft che consente di scrivere codice eseguibile (**script**)
- **ADO.NET** (*Active Data Objects*) per la connessione ai database in rete
- **Windows Forms** per la creazione di interfacce grafiche per l'utente in ambiente Windows.

Pagine ASP.NET

- Una pagina ASP.NET è eseguita sul server in risposta a una richiesta proveniente dal browser dell'utente



Pagine ASP.NET

- Posizionare la pagina nella directory del Web server e lanciarne l'esecuzione dalla barra *Indirizzo* del browser scrivendo:

`http://nomeserver/pagina.aspx`

Linguaggio Visual Basic

```
<%@ Page Language="VB" %>
```

```
'Riga1 di commento
```

```
Dim NomeVariabile As Short
```

```
Const PiGreco = 3.14
```

```
Dim A(5) As Short
```

Struttura di selezione

```
If condizione Then  
    istruzioni1  
Else  
    istruzioni2  
End If
```

Selezione multipla

```
Select Case Selettore  
  Case valore1, valore2, valore3  
    istruzioni1  
  Case valore4, valore5  
    istruzioni2  
  Case Else  
    istruzioni3  
End Select
```


Strutture di ripetizione

```
For i = ValoreIniziale to ValoreFinale  
    istruzioni  
Next
```

```
Do While condizione  
    istruzioni  
Loop
```

Programmazione ad oggetti

- Il linguaggio Visual Basic utilizza i concetti di base della **programmazione a oggetti**
- A ogni oggetto sono associati **metodi** (procedure o funzioni) e **proprietà** (attributi o caratteristiche dell'oggetto).

Oggetti

- chiamata di un metodo:

```
Oggetto.Metodo(parametri)
```

- utilizzo delle proprietà:

```
Oggetto.Proprietà
```

Oggetti lato server

- Controlli **HTML server**: sono i normali tag HTML, ma interpretabili dal server;
- Controlli **Web server**: sono i tag speciali ASP.NET interpretabili dal server.

Oggetti lato server

- Per rendere questi controlli utilizzabili all'interno del codice: l'attributo **runat="server"**.

```
<asp:tipo id="nome" runat="server"></asp:tipo>
```

- etichetta (**Label**):

```
<asp:Label id="label1" runat="server" text="prova"></asp:Label>
```

Forma abbreviata

```
<asp:Label id="label1" runat="server" text="prova" />
```

Image e Button

- **Image** (immagine) e **Button** (pulsante di comando).

```
<asp:Image id="Image1" runat="server" />
```

```
<asp:Button id="button1" Text="Invia"  
OnClick="Visualizza" runat="server" />
```

DropDownList

- Controllo di tipo *Web server per* la casella combinata

```
<asp:DropDownList id="drop1" runat="server">  
<asp:ListItem Value="...">testo1</asp:ListItem>  
<asp:ListItem Value="...">testo2</asp:ListItem>  
...  
...  
</asp:DropDownList>
```

Pagina ASP.NET

- le pagine ASP.NET separano il codice di scripting dai tag HTML

```
<script runat="server">  
Sub  
    . . . . .  
End Sub  
</script>  
  
<html>  
<body>  
    <form runat="server">  
        . . . . .  
        . . . . .  
    </form>  
</body>  
</html>
```


Validazione dei dati

Controlli **Validation server**:

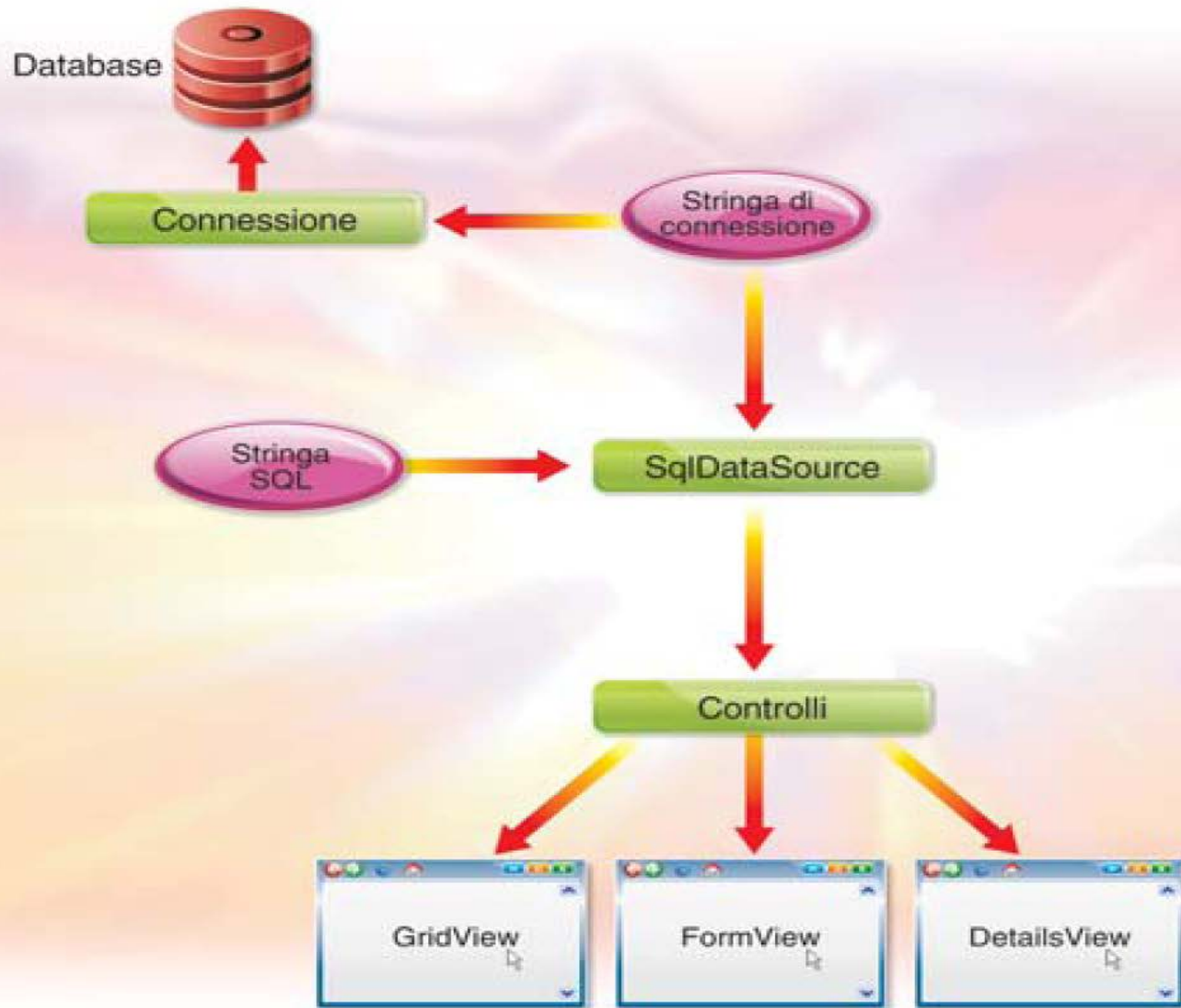
- CompareValidator
- RangeValidator
- RequiredFieldValidator

Accesso ai database

- **ADO.NET** (*ActiveX Data Objects*) è la tecnologia di accesso ai dati
- La configurazione viene scritta nel file **Web.config** (sezione *connectionStrings*)

```
<connectionStrings>  
<add name="conn2"  
connectionString="Provider=Microsoft.ACE.OLEDB.12.0;  
Data Source=Northwind.accdb"  
providerName="System.Data.OleDb" />  
</connectionStrings>
```

Applicazioni Web e database



Visualizzazione dei dati di una tabella

```
<form id="form1" runat="server">  
<asp:SqlDataSource  
id="SqlDataSource1"  
runat="server"  
SelectCommand="SELECT campi FROM nometabella"  
ConnectionString="<%= $connectionStrings:conn2 %>"  
ProviderName=  
"<%= $connectionStrings:conn2.providerName %>" >  
</asp:SqlDataSource>  
<asp:GridView id="GridView1"  
DataSourceID="SqlDataSource1"  
runat="server">  
</asp:GridView>  
</form>
```

Controllo **SqlDataSource**

I dati del controllo **SqlDataSource** sono ottenuti dall'esecuzione di un comando SQL contenuto nella stringa dell'attributo **SelectCommand**.

Le informazioni sulla stringa della connessione (attributo **ConnectionString**) e sul Provider (attributo **ProviderName**) sono contenute nell'elemento *connectionStrings* del file di configurazione *Web.config*.

Parametri e operazioni di manipolazione

- Parametro: nome preceduto dal simbolo @
@param1.
- Attributi di *SqlDataSource* per la manipolazione al posto di *SelectCommand*:
 - **InsertCommand**
 - **DeleteCommand**
 - **UpdateCommand**

```
DeleteCommand="Delete From nometabella  
                Where campo = @param1"
```

Parametri

- collezione **DeleteParameters**

```
<asp:SqlDataSource id="SqlDataSource1" runat="server"
DeleteCommand="Delete From tabella Where campo = @param1 "
OnDeleted="subroutine"
ConnectionString="<%%$ connectionStrings:conn2%"
ProviderName="<%%$ connectionStrings:conn2.providerName %">
  <DeleteParameters>
    <asp:ControlParameter ControlID="nome" Name="param1 "
    PropertyName="Text" Type="Int32" />
  </DeleteParameters>
</asp:SqlDataSource>
```

- In modo analogo **InsertParameters** e **UpdateParameters**

Accesso a un documento XML

- controllo Web server di ASP.NET denominato **Xml**
- un file di trasformazione XSLT da applicare al documento XML

```
<asp:Xml id="Xml1"  
DocumentSource="siti.xml"  
TransformSource="siti.xsl"  
runat="server" />
```


Uso di Dataset

- modalità di lettura del documento XML sull'oggetto **DataSet**.

```
Dim ds As New DataSet  
ds.ReadXml( MapPath( "siti.xml" ) )
```

- Controllo *griglia* di tipo **GridView** associato al *DataSet*

```
griglia.DataSource = ds  
griglia.DataBind( )
```

Controllo XmlDataSource

```
<asp:XmlDataSource id="sitiweb"  
DataFile="siti.xml"  
TransformFile="sitiXSLT.xsl"  
runat="server" />
```

Ricerca dei dati con Xpath

- l'espressione *XPath* definita come una proprietà dell'oggetto **XmlDataSource** (*sitiweb*)
- scelta dall'utente tramite una casella di testo (*categoria*)

```
sitiweb.XPath =  
"/lista/sito[categoria='" +  
categoria.Text + "']"
```