Agenda

1. What is Routing
2. Mapping Route to Physical file
3. Reading Route Parameters
4. Route Expression
5. Generate URL using Markup

What is Routing

Routing is a technique where asp.net webforms can be accessed with a friendly and descriptive URL's. Hence, the ugly URL's with querystring can now be converted to a meaningful and descriptive which will be easy to remember and will improve the search engine rank.

Rather than showing a URL like: http://mywebsite.com/userprofiles.aspx?userid=1, developers would prefer to show the URL as: http://mywebsite.com/john.

To Map a route to the URL following must be done

1. Define the mapping in Global.asax, which maps a route pattern to the aspx page.

```
protected void Application_Start(object sender, EventArgs e)
{
   System.Web.Routing.RouteTable.Routes.MapPageRoute("RouteName",
                                                     "FixedString/{RequestParameterName}","~/PhysicalPage.aspx");
}
```

2. Writing code in the target page to grab the route parameters and use them in rendering its content

   In PhysicalPage.aspx

```
string parameter = Page.RouteData.Values["RequestParameterName"].ToString();
```

Mapping Route to Physical file:

In Global.aspx

```
RouteTable.Routes.MapPageRoute("", "SalesReportSummary/{year}", "~/sales.aspx");

RouteTable.Routes.MapPageRoute("SalesRoute", "SalesReport/{country}/{year}", "~/sales.aspx",true,
                new RouteValueDictionary {   //Default values
                  { "country", "USA" },
                  { "year", DateTime.Now.Year.ToString() } },
                new RouteValueDictionary {   //Contraints
                  { "country", "[a-z]{3}" },
                  { "year", @"\d{4}" } });
```

Reading Route Parameters

SalesReport.aspx

```
Sales Report of
<asp:Literal ID="ltrCountry" runat="server"></asp:Literal> for the Year
<asp:Literal ID="ltrYear" runat="server"></asp:Literal>
```

SalesReport.aspx.cs

```
if (Page.RouteData.Values["country"] == null)
    ltrCountry.Text = "All Countries";
 else
```

```
    ltrCountry.Text = Page.RouteData.Values["country"].ToString();

 ltrYear.Text = Page.RouteData.Values["year"].ToString();
```

Route Expression

Sales Report of

```
 <asp:Literal ID="Literal1" Text="<%$ RouteValue:country %>" runat="server"></asp:Literal>,
 for the Year
<asp:Literal ID="Literal2" Text="<%$RouteValue:year%>" runat="server"></asp:Literal>
```

Default.aspx:

```
<a href="~/salesreportsummary/2010">Sales Report - All countrys, 2010</a>
<a href="~/salesreport/WA/2012">Sales Report - IND, 2012</a>
<a href="~/salesreport">Expense Report - Default country and Year (USA, current year)</a>
```

Generating URLs from Route

```
<asp:HyperLink runat="server" NavigateUrl="<%$RouteUrl:year=2011%>">
      Sales Report - All countries, 2012
</asp:HyperLink>

<asp:HyperLink  runat="server" NavigateUrl="<%$RouteUrl:country=IND,year=2009,routename=salesroute%>">
    Sales Report - IND, 2009
</asp:HyperLink>
```

To get path using code:

```
<asp:HyperLink ID="HyperLink1" runat="server">
   Expense Report - CA, 2008
</asp:HyperLink>
RouteValueDictionary parameters =  new RouteValueDictionary
{
   {"country", "IND" },
   { "year", "2008" } ,
   { "category", "recreation" }
};
VirtualPathData vpd = RouteTable.Routes.GetVirtualPath(null, "ExpensesRoute", parameters);
HyperLink1.NavigateUrl = vpd.VirtualPath;
```