

Review of Telerik toolsets for ASP.NET Web Forms and Core

WHITEPAPER



Contents

[ASP.NET Web Forms / 4](#)

[ASP.NET Core / 4](#)

[Telerik UI for ASP.NET AJAX / 6](#)

[Telerik UI for ASP.NET Core / 6](#)

[Relevance / 7](#)

[First Steps / 8](#)

[Submit Form Sample / 9](#)

[Data-bound Grid Sample / 12](#)

[Conclusion / 17](#)

Introduction

What is the purpose of this document?

This guide aims to assist developers in familiarizing quickly with the overall idea of ASP.NET Web Forms and ASP.NET Core and what Telerik has to offer for both technologies. It also provides couple of fundamental steps for customers who've decided to create a new application based on ASP.NET Core, or modernize their existing Web Forms App by switching to a Core App.

ASP.NET Web Forms

Strong Presence—Mature Experience

In existence since 2002 and still actively used, Web Forms technology has successfully passed the harsh tests of time and the rigorous demands of web developers.

Some of the key features and recognizable traits of Web Forms are the ViewState and Event-driven development. In MVC and Core these are replaced with the concept of Model-View-Controller pattern. You can check Microsoft's own comparison between Web Forms and MVC type of development in this article: [Deciding When to Create a Web Forms Application](#)

One key point worth mentioning is that Web Forms is not supported by ASP.NET Core, which is the new focus of Microsoft as a complete rewrite over .NET framework: [Choose between ASP.NET 4.x and ASP.NET Core](#)

ASP.NET Core

The Microsoft Flagship for Web Development

ASP.NET Core is a kid no longer. It has grown up to take its place as the dominant Microsoft technology for Web Development. But this is not the final stage - the future is called .NET 5 and will be the upcoming version of .NET family: [Introducing .NET 5](#)

To better understand how this happened and what .NET 5 is, let's look at some history of this sequence:

Web Forms (.NET)

-> Model-View-Controller (.NET)

-> Model-View-Controller (.NET Core) or Razor Pages (.NET Core)

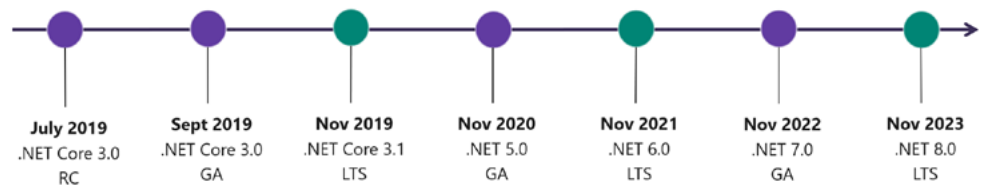
For Web Forms and some useful comparison links you can check this section:

ASP.NET Web Forms

The old .NET Framework is a powerhouse. Its version 4.8 is out and announced to be its last one. This framework is like a proud father, providing so much for the development community and now passing the baton to the upcoming generation. However, there are still many years until retirement.

The .NET Core is the grown-up kid. Officially, the baton was relayed. Microsoft advises .NET Core to be used in new apps and projects. Since its introduction, more and more functionality and API were added so that the capability gap with old .NET Framework is minimized.

.NET Schedule



- .NET Core 3.0 release in September
- .NET Core 3.1 = Long Term Support (LTS)
- .NET 5.0 release in November 2020
- Major releases every year, LTS for even numbered releases
- Predictable schedule, minor releases if needed

New versions of .NET will step on the foundations of Core.

Although ASP.NET Core does not support Web Forms, it supports another technology which is present for years and built a strong community as the primary Microsoft Web Development Technology – the famous Model-View-Controller formation: [Get started with ASP.NET Core MVC](#)

Microsoft suggests yet another technology for building Web Apps called Razor Pages: [Introduction to Razor Pages in ASP.NET Core](#)

Telerik's offer for ASP.NET Core supports both these technologies:

- [Telerik UI for ASP.NET MVC Demos](#)
- [Telerik UI for ASP.NET Core Demos](#)

Explore our ASP.NET Core UI components by downloading a [30-day FREE trial](#)

Telerik UI for ASP.NET AJAX

Over 100 components. Everything you are looking for under one roof. Top stability and utility.

This is the product that elevated Telerik to be a first-class renowned brand around the globe, trusted and chosen by Software Developers galore:

[ASP.NET AJAX Controls](#)

You will find all the functionality you need to be already provided built-in by this suite. And even if you have a specific scenario, there is a big chance that you will discover a possible implementation via custom code somewhere in the plethora of forums, resource base and library samples.

Many users choose this product for building enterprise-level projects and improving existing ASP.NET Web Forms applications. With Controls variety at their fingertips, the developers are enabled to provide a professional full functioning and accessibility compliant application faster and without much hassle.

Telerik UI for ASP.NET Core

Modern, Convenient, Powerful. Compatible with the newest Microsoft releases. Priority product trusted by thousands of customers.

This product's popularity and user base has grown substantially with the inception of ASP.NET Core as the leading Microsoft web development platform:

[ASP.NET Core Components](#)

It uses the well-known [Kendo UI for jQuery framework](#) as its base and wraps HTML helpers and tags on top of its components. This brings richness and constant enhancement to the toolset as Kendo gains more and more ground.

In one of the recent releases, Telerik UI for ASP.NET Core also introduced support for Scaffolding Templates, which provide a convenient way to quickly build up a working solution: [Scaffolding](#)

Here are some of the other features of the suite:

- **80+ ready-to-use components**
- **Three Major Releases per year**
- **In-line with the latest Microsoft has to offer**
- **Compatible with the preview versions of .NET Core 5.0**
- **Mobile support and responsiveness**
 - Mobile has increasing 50%+ rate of the global browser share
- **Powerful CRUD implementation**
- **Build-in customizable themes**
 - Including Bootstrap v4 and Material
- **Automatic database operations**
 - Paging, sorting, filtering, and grouping
- **Visual Studio IntelliSense**
- **Theme Builder**
- **Virtualization**
- **Exporting to Excel, Word, PDF**
- **Built-in Accessibility**

Play around with all of our products by downloading a [30-day FREE DevCraft trial](#)

Relevance

Are these two Toolsets compatible? Can there be automatic migration?

The answer here is simple – no. The .NET Core is not just an upgrade on top of the old .NET framework. As mentioned in the [ASP.NET Core](#) section, it is a complete overhaul. Multiply that by the fact that the technologies have different architectures and you get a no2.

It is possible that you can re-use your Data Layers and some C# or VB classes if the dependencies they are using are also present in .NET Core 3.0+. But a complete rewrite will be needed in the UI part. There can be some tricky solutions over the Net to force an automatic migration, but these are not tested, nor supported by Telerik.

The next sections provide some pointers and getting-started articles to get you going and build a new runnable Core app resembling your existing Web Forms project.

First Steps

Suggestions for building a new Core App

When starting your new ASP.NET Core Web App, we suggest that you check our Introduction page: [Welcome to Telerik UI for ASP.NET Core](#)

It contains [Learning Resources](#) and [Next Steps](#) sections which will guide you through the process to have a running app in no time.

Now, if you have an already existing project and want to achieve the same functionality in the new app, you will need to do the migration manually.

If you have an MVC app using the old .NET framework and want to switch to MVC .NET Core, you can make avail of the procedure explained in the following MSDN article: [Migrate from ASP.NET MVC to ASP.NET Core MVC](#)

If you have a Web Forms app, additional coding will be required. It will require some effort, but it is still possible to achieve. Here is a relation table between Web Forms and Core MVC apps, which will help you clarify some of the major concepts:

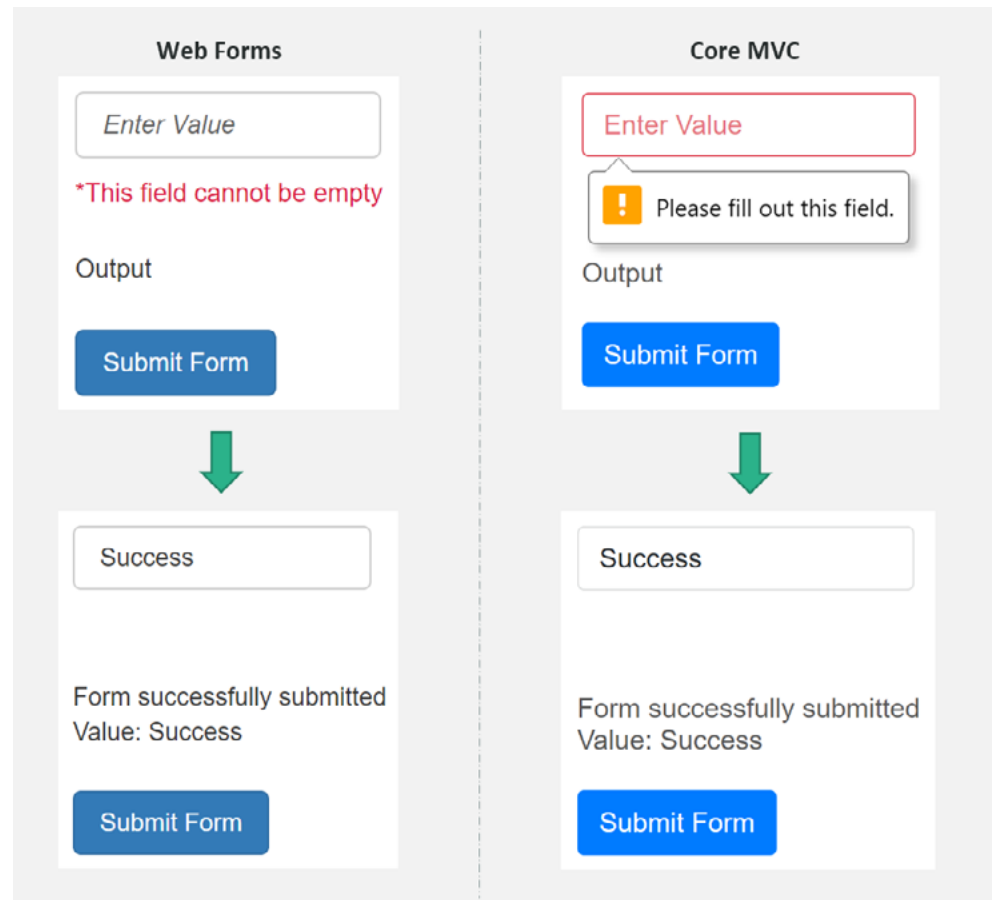
Page	→	View
Event	→	Action
Global.asax	→	Startup.cs
Code-behind	→	Controller
Master Page	→	Layout File
System.Web	→	Microsoft.AspNetCore
Entity Framework	→	Entity Framework Core
Handlers/Modules	→	Middleware
Web User Control	→	Partial View
Static content (images, files)	→	wwwroot Folder

For achieving anything more specific, you can check our live demos, documentation, KBs, How-To articles; in other words – any kind of resource to achieve the same functionality of your Web Forms app. And of course, you can use the Forums or Ticketing system to contact our Support Experts to assist you in the process.

Submit Form Sample

Comparison Example: Basic form with input, button and validation

This example demonstrates basic implementation of a submit form to compare the syntax in both technologies.



Web Forms Page – SubmitFormSample.aspx

```
<form id="form1" runat="server">
  <telerik:RadScriptManager ID="RadScriptManager1" runat="server">
    <Scripts>
      <asp:ScriptReference Assembly="Telerik.Web.UI" Name="Telerik.Web.
UI.Common.Core.js" />
      <asp:ScriptReference Assembly="Telerik.Web.UI" Name="Telerik.Web.
UI.Common.jQuery.js" />
      <asp:ScriptReference Assembly="Telerik.Web.UI" Name="Telerik.Web.
UI.Common.jQueryInclude.js" />
    </Scripts>
  </telerik:RadScriptManager>
  <telerik:RadTextBox ID="RadTextBox1" runat="server"
    EmptyMessage="Enter Value">
  </telerik:RadTextBox>
  <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
    ErrorMessage="*This field cannot be empty" ForeColor="Crimson"
    ControlToValidate="RadTextBox1" Style="display: block; margin: 10px 0
    20px 0;">
  </asp:RequiredFieldValidator>
  <telerik:RadLabel ID="RadLabel1" runat="server" Text="Output"
    Style="display: block; margin-bottom: 20px;">
  </telerik:RadLabel>
  <telerik:RadButton ID="RadButton1" runat="server" Text="Submit Form"
    Primary="true" OnClick="RadButton1_Click">
  </telerik:RadButton>
</form>
```

Core View – Index.cshtml

```
<form id="form1" method="post" action="/Home/Index">
  @(Html.Kendo().TextBox()
    .Name("UserValue")
    .HtmlAttributes(new
    {
      placeholder = "Enter Value",
      required = "required",
      @style = "display: block; margin-bottom: 60px;"
    }
  ))
  <span style="font-size: 16px; display: block; margin-bottom: 20px; width:
  210px;">
    @ViewBag.LabelText
  </span>
  @(Html.Kendo().Button()
    .Name("Button1")
    .Content("Submit Form")
    .HtmlAttributes(new
    {
      @class = "k-primary"
    }
  ))
  <br />
</form>
```

Web Forms Code-Behind – SubmitFormSample.aspx.cs

```
using System;

public partial class SubmitFormSample : System.Web.UI.Page
{
    protected void RadButton1_Click(object sender, EventArgs e)
    {
        RadLabel1.Text = "Form successfully submitted" +
            "<br/>Value: " + RadTextBox1.Text;
    }
}
```

Core Controller – HomeController.cs

```
using Microsoft.AspNetCore.Mvc;

namespace KendoCoreTrialPack1.Controllers
{
    public class HomeController : Controller
    {
        public IActionResult Index()
        {
            ViewBag.LabelText = "Output";
            return View();
        }
        [HttpPost]
        public IActionResult Index(string UserValue)
        {
            ViewBag.LabelText = "Form successfully submitted" +
                "Value: "+ UserValue;
            return View();
        }
    }
}
```

Data-bound Grid Sample

Comparison Example: Grid configuration and binding

Another common scenario for real-time applications is to contain a grid component. This example shows the basic configuration of a Web Forms grid and its Core MVC counterpart.

Telerik Grid for Web Forms AJAX

OrderID	OrderDate	Freight	ShipName	ShipCountry
1	11/19/2019	1.11	Name 1	Country 1
2	11/19/2019	2.22	Name 2	Country 2
3	11/19/2019	3.33	Name 3	Country 3
<input checked="" type="checkbox"/>	4	4.44	Name 4	Country 4
<input type="checkbox"/>	5	5.55	Name 5	Country 5
<input checked="" type="checkbox"/>	6	6.66	Name 6	Country 6
<input checked="" type="checkbox"/>	7	7.77	Name 7	Country 7
<input type="checkbox"/>	8	8.88	Name 8	Country 8
<input type="checkbox"/>	9	9.99	Name 9	Country 9
<input type="checkbox"/>	10	11.1	Name 10	Country 10
Count : 81		Sum : \$3,686.31		

Telerik Grid for Core

Order ID	Order Date	Freight	Ship Name	Ship City
0	9/15/2016	0	ShipName 0	ShipCity 0
1	9/16/2016	10	ShipName 1	ShipCity 1
2	9/17/2016	20	ShipName 2	ShipCity 2
3	9/18/2016	30	ShipName 3	ShipCity 3
<input checked="" type="checkbox"/>	4	40	ShipName 4	ShipCity 4
<input type="checkbox"/>	5	50	ShipName 5	ShipCity 5
<input checked="" type="checkbox"/>	6	60	ShipName 6	ShipCity 6
<input checked="" type="checkbox"/>	7	70	ShipName 7	ShipCity 7
<input type="checkbox"/>	8	80	ShipName 8	ShipCity 8
<input type="checkbox"/>	9	90	ShipName 9	ShipCity 9
Count : 80		Sum : \$31,600.00		

Web Forms Page – RadGridSample.aspx

```
<telerik:RadGrid ID="RadGrid1" runat="server" AllowPaging="True"
CellSpacing="0"
    GridLines="None" Width="1000px" OnNeedDataSource="RadGrid1_
NeedDataSource"
    AllowFilteringByColumn="true" AllowMultiRowSelection="true"
    ShowFooter="true">
    <ClientSettings>
        <Selecting AllowRowSelect="true" />
    </ClientSettings>
    <ItemStyle HorizontalAlign="Center" />
    <AlternatingItemStyle HorizontalAlign="Center" />
    <HeaderStyle HorizontalAlign="Center" />
    <FooterStyle HorizontalAlign="Center" Font-Bold="true"/>
    <MasterTableView AutoGenerateColumns="False" DataKeyNames="OrderID"
        EditMode="Batch" CommandItemDisplay="Top">
        <CommandItemSettings ShowExportToExcelButton="true"
            ShowExportToPdfButton="true" />
        <Columns>
            <telerik:GridClientSelectColumn UniqueName="SelectColumnName">
                <HeaderStyle Width="55px" />
            </telerik:GridClientSelectColumn>
            <telerik:GridBoundColumn DataField="OrderID" DataType="System.
                Int32"
                FilterControlAltText="Filter OrderID column"
                HeaderText="OrderID"
                ReadOnly="True" SortExpression="OrderID"
                UniqueName="OrderID"
                FilterControlWidth="40px" Aggregate="Count">
                <HeaderStyle Width="100px" />
            </telerik:GridBoundColumn>
            <telerik:GridDateTimeColumn DataField="OrderDate"
                DataType="System.DateTime"
                FilterControlAltText="Filter OrderDate column"
                HeaderText="OrderDate"
                SortExpression="OrderDate" UniqueName="OrderDate"
                DataFormatString="{0:d}">
            </telerik:GridDateTimeColumn>
            <telerik:GridNumericColumn DataField="Freight" DataType="System.
                Decimal"
                FilterControlAltText="Filter Freight column"
                HeaderText="Freight"
                SortExpression="Freight" UniqueName="Freight"
                FilterControlWidth="140px"
                Aggregate="Sum" FooterAggregateFormatString="Sum : {0:C}">
            </telerik:GridNumericColumn>
            <telerik:GridBoundColumn DataField="ShipName"
                FilterControlAltText="Filter ShipName column"
                HeaderText="ShipName"
                SortExpression="ShipName" UniqueName="ShipName">
            </telerik:GridBoundColumn>
            <telerik:GridBoundColumn DataField="ShipCountry"
                FilterControlAltText="Filter ShipCountry column"
                HeaderText="ShipCountry"
                SortExpression="ShipCountry" UniqueName="ShipCountry">
            </telerik:GridBoundColumn>
        </Columns>
    </MasterTableView>
</telerik:RadGrid>
```

Core View – Index.cshtml

```
<style>
    .k-grid-excel, .k-grid-pdf {
        float: right;
    }

    .k-grid table tr {
        text-align: center;
    }
</style>

@(Html.Kendo().Grid<KendoCoreTrialPack1.Models.OrderViewModel>()
    .Name("grid")
    .Columns(columns =>
    {
        columns.Select().Width(50);
        columns.Bound(p => p.OrderID).ClientFooterTemplate("<b>Count :
        #=count#</b>");
        columns.Bound(p => p.OrderDate).Format("{0:d}");
        columns.Bound(p => p.Freight).ClientFooterTemplate("<b>Sum : #=
        kendo.format('{0:C}', sum) #</b>");
        columns.Bound(p => p.ShipName);
        columns.Bound(p => p.ShipCity);
    })
    .Pageable()
    .PersistSelection()
    .Editable(editable => editable.Mode(GridEditMode.InCell))
    .Filterable(ftb => ftb.Mode(GridFilterMode.Row))
    .HtmlAttributes(new { style = "width: 1200px;" })
    .ToolBar(toolbar =>
    {
        toolbar.Create();
        toolbar.Save();
        toolbar.Excel();
        toolbar.Pdf();
    })
    .DataSource(dataSource => dataSource
        .Ajax()
        .Batch(true)
        .Aggregates(aggregates =>
        {
            aggregates.Add(o => o.OrderID).Count();
            aggregates.Add(o => o.Freight).Sum();
        })
        .ServerOperation(false)
        .Model(model => model.Id(o => o.OrderID))
        .Read(read => read.Action("Orders_Read", "Grid"))
    )
)
```

Web Forms – RadGridSample.aspx.cs

```
using System;
using System.Data;
using Telerik.Web.UI;

public partial class RadGridSample : System.Web.UI.Page
{
    protected void RadGrid1_NeedDataSource(object sender,
GridNeedDataSourceEventArgs e)
    {
        RadGrid1.DataSource = GetGridSource();
    }
    private DataTable GetGridSource()
    {
        DataTable dataTable = new DataTable();

        DataColumn column = new DataColumn();
        column.DataType = Type.GetType("System.Int32");
        column.ColumnName = "OrderID";
        dataTable.Columns.Add(column);

        column = new DataColumn();
        column.DataType = Type.GetType("System.DateTime");
        column.ColumnName = "OrderDate";
        dataTable.Columns.Add(column);

        column = new DataColumn();
        column.DataType = Type.GetType("System.Decimal");
        column.ColumnName = "Freight";
        dataTable.Columns.Add(column);

        column = new DataColumn();
        column.DataType = Type.GetType("System.String");
        column.ColumnName = "ShipName";
        dataTable.Columns.Add(column);

        column = new DataColumn();
        column.DataType = Type.GetType("System.String");
        column.ColumnName = "ShipCountry";
        dataTable.Columns.Add(column);

        DataColumn[] PrimaryKeyColumns = new DataColumn[1];
        PrimaryKeyColumns[0] = dataTable.Columns["OrderID"];
        dataTable.PrimaryKey = PrimaryKeyColumns;

        for (int i = 0; i <= 80; i++)
        {
            DataRow row = dataTable.NewRow();
            row["OrderID"] = i + 1;
            row["OrderDate"] = DateTime.Now;
            row["Freight"] = (i + 1) + (i + 1) * 0.1 + (i + 1) * 0.01;
            row["ShipName"] = "Name " + (i + 1);
            row["ShipCountry"] = "Country " + (i + 1);

            dataTable.Rows.Add(row);
        }

        return dataTable;
    }
}
```

Core Model – OrderViewModel.cs

```
using System;
using System.ComponentModel.DataAnnotations;

namespace KendoCoreTrialPack1.Models
{
    public class OrderViewModel
    {
        public int OrderID { get; set; }

        public decimal? Freight { get; set; }

        [Required]
        public DateTime? OrderDate { get; set; }

        public string ShipCity { get; set; }

        public string ShipName { get; set; }
    }
}
```

Core Controller – GridController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Kendo.Mvc.UI;
using KendoCoreTrialPack1.Models;
using Kendo.Mvc.Extensions;

namespace KendoCoreTrialPack1.Controllers
{
    public class GridController : Controller
    {
        public ActionResult Orders_Read([DataSourceRequest]DataSourceRequest
request)
        {
            var result = Enumerable.Range(0, 80).Select(i => new
OrderViewModel
            {
                OrderID = i,
                Freight = i * 10,
                OrderDate = new DateTime(2016, 9, 15).AddDays(i % 7),
                ShipName = "ShipName " + i,
                ShipCity = "ShipCity " + i
            });

            var dsResult = result.ToDataSourceResult(request);
            return Json(dsResult);
        }
    }
}
```


Conclusion

Both options – the right choice

Some developers have strong ground in their ASP.NET Web Forms experience. They enjoy this technology and take comfort in their knowledge, as well as in the long path Web Forms has traveled through the years. They just want to continue using it – and who can blame them? Is there something Web Forms is lacking in the functionality department? This technology is still relevant to current Web Development demands and requirements.

On the other hand, Microsoft has shifted their focus to ASP.NET Core. They are actively developing it adding new enhancements and cutting-edge features. They've also made clear that this will be the future path. This technology has already proven to be a steady ship to sail on. And if you are the person (or team) who prefers to catch the latest shiny wave under the sun, definitely check ASP.NET Core out.

Telerik has a strong presence and commitment for both these technologies and our dedicated and experienced Support Team is ready to assist with any migration questions that may arise.



About the author

Eyup Yusein is a Technical Support Officer working on the Progress Telerik web developer tools. His main field of strength is handling various implementations with RadGrid. He likes to discover simple and unconventional explanations for complicated matters and to learn about the newest technological achievements. He also dearly enjoys spending time with his family.



Get in touch with our product specialists team to learn more about your options to migrate!

About Progress

Progress (NASDAQ: PRGS) offers the leading platform for developing and deploying strategic business applications. We enable customers and partners to deliver modern, high-impact digital experiences with a fraction of the effort, time and cost. Progress offers powerful tools for easily building adaptive user experiences across any type of device or touchpoint, award-winning machine learning that enables cognitive capabilities to be a part of any application, the flexibility of a serverless cloud to deploy modern apps, business rules, web content management, plus leading data connectivity technology. Over 1,700 independent software vendors, 100,000 enterprise customers, and two million developers rely on Progress to power their applications. Learn about Progress at www.progress.com or +1-800-477-6473.

