

ComponentOne SSRSDataSource for WinForms

ComponentOne SSRSDataSource for WinForms能够生成分页的SSRS报表并能把报表加载到C1PrintPreviewControl中以便查看和导出到各种格式，如Adobe PDF, Microsoft Word和Excel。

C1SrsDataSource当前版本仅支持SSRS报表分页视图。

ComponentOne Studio for WinForms帮助

开始

想要获得ComponentOne Studio for WinForms的安装信息，证书信息，技术支持，命名空间以及如何使用控件生成一个程序，请参阅Getting Started with Studio for WinForms。

新特性

想要获得ComponentOne Studio for WinForms的最新特性列表，请参阅What's New in Studio for WinForms。

主要特性

ComponentOne SSRS DocumentSource for WinForms 能够借助于 C1PrintPreviewControl 轻松创建多种应用程序。

功能齐全的预览控件

- **C1PrintPreviewControl 控件**

C1PrintPreviewControl 控件提供了随时可用的功能齐全的 UI，包括缩略图，大纲视图，文本搜索，预定义工具栏。

- **连接并查看报表**

连接并查看 SQL Server Reporting Services (SSRS) 报表。

- **缩放和导航**

使用鼠标导航和缩放页面。

- **导出并保存报表**

导出并保存本地报表为 Adobe PDF (.pdf), Microsoft Word (.doc), Microsoft Excel (.xls), Web archive (.mhtml), CSV (.csv), Tiff Image (.tiff), Bmp image (.bmp), Enhanced Metafile (.emf), 和 Gif image (.gif) 格式。

- **打印报表**

支持自定义页面设置和分页打印预览打印报表。

- **支持缩略图**

以页面缩略图查看报表。

SSRSDataSource快速入门

C1SrsDataSource允许你定义要查看的SSRS报表的位置和名称。然后它将启用C1PrintPreviewControl控件来显示分页好的SSRS报表。

它不是一个可视化的控件，所以它虽然能被加到一个窗体里，但是它只显示在组件托盘里。

注意： 在开始这个此章节前，你必须已经访问过SSRS报表的服务器端，并且在服务器上已经有了至少一个报表。

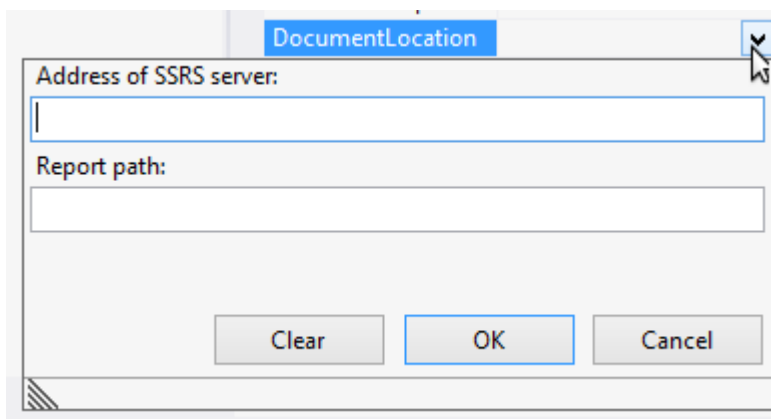
想要创建一个新的WinForms应用程序，使用C1PrintPreviewControl控件来预览或者导出一个SSRS报表，必须使用.NET framework4.0或者以上版本来完成下面的步骤：

使用设计器

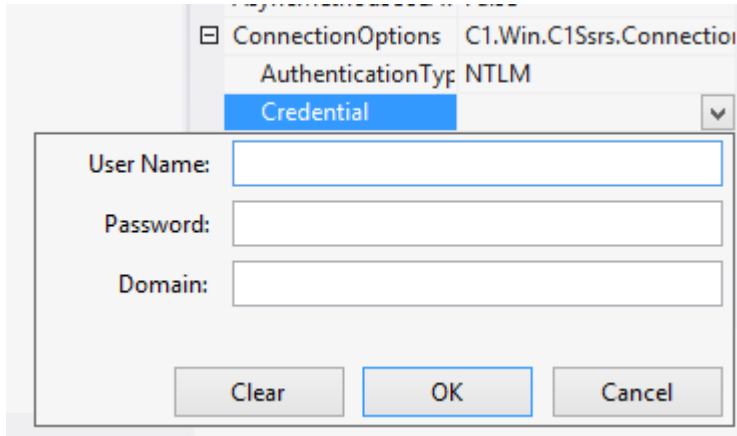
1. 从工具箱中将C1SSRSDataSource组件添加到窗体中，该组件就出现在组件托盘中。
如果在工具箱中你无法找到这个组件，右键点击并选择Choose items.


注意： ChooseToolboxItems对话框就会出现。该组件就在C1.Win.C1Document.dll中，然后将它添加到工具箱里。

2. 点击C1SSRSDataSource组件，然后点击智能标签来打开C1SSRSDataSource任务菜单。
3. 选中C1PrintPreviewControl旁边的复选框，这样就能将C1SSRSDataSource组件赋给了C1PrintPreviewControl控件的Document属性。
4. 右键点击C1SSRSDataSource并且选择Properties来打开属性窗口。
5. 点击DocumentLocation属性旁边的下拉箭头。
6. 使用下面的对话框定义SSRS服务器端的地址和报表的完整地址，然后点击OK。



7. 扩展ConnectionOptions属性组。
8. 点击Credential属性旁边的下拉箭头。
9. 使用下面的对话框定义用户名，密码和域名，然后点击OK按钮。



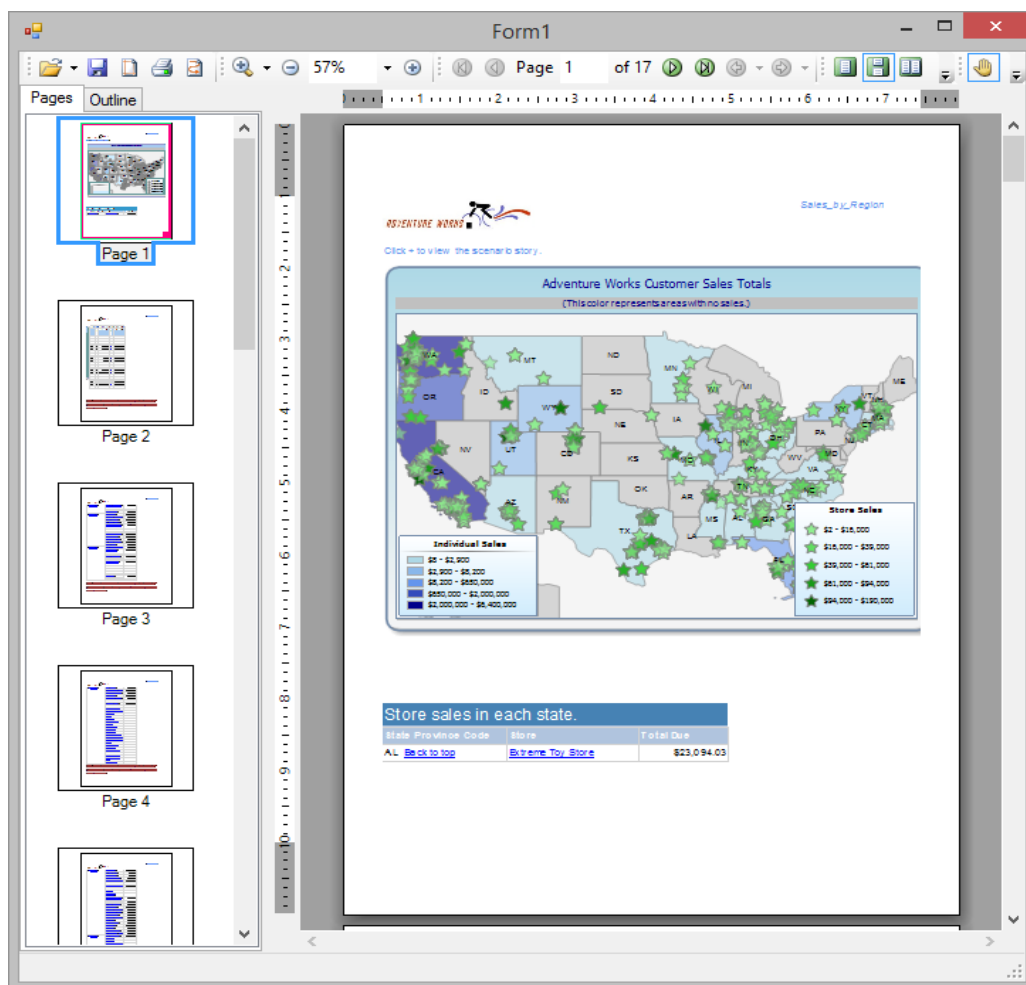
 **注意:** 确认下面的DLL添加到了程序的引用中。

- C1.Win.C1Document.dll
- C1.Win.dll
- C1.Win.C1Srs.dll
- C1.Win.C1Report.dll
- C1.C1Report.dll

你也能够在代码中设定文件位置和网络证书。更多信息请参照在代码中设定文件名和在代码中设置证书

你完成了什么

基于你所看到的在预览控件中定义好的SSRS报表，你也许只需要设定几个参数。



你现在可以将它导出成一系列外部程序支持的格式。

在代码中设定网络证书（Credentials）

你能够通过设置Credentials属性来定义访问SSRS报表所需的用户名和密码。这个属性在C1SSRSDataSource中的SecurityError事件里。

当访问SSRS服务器端因为缺少网络证书而被拒绝时，C1SSRSDataSource中的SecurityError事件就会被触发，这样就允许你设定需要的网络证书并重新访问服务器。

在设计器中

1. 打开C1SSRSDataSource中的事件窗口。
2. 双击SecurityError事件。这样在你的代码中就会产生一个此事件的空的处理块。

在代码中

1. 在代码中，像下面所示修改c1SSRSDataSource1_SecurityError事件处理块

C#

```
private void c1SSRSDataSource1_SecurityError(object sender,
C1.Win.C1Document.SecurityEventArgs e)
{
    var ds = (C1.Win.C1Document.C1SSRSDataSource) sender;
    ds.Credential = new System.Net.NetworkCredential("myUserId", "myPassword");
    e.Retry = true;
}
```

VB

```
Private Sub C1SSRSDataSource1_SecurityError(sender As Object, e As
C1.Win.C1Document.SecurityEventArgs)
    Handles C1SSRSDataSource1.SecurityError
    Dim ds = DirectCast(sender, C1.Win.C1Document.C1SSRSDataSource)
    ds.Credential = New System.Net.NetworkCredential("myUserId",
"myPassword")
    e.Retry = True
End Sub
```

2. 在上面的代码中，用有效的网络证书为SSRS报表服务器重新设定了"myUserId"和"myPassword"。当你的窗体加载时，C1SSRSDataSource组件就会尝试访问定义好的报表。

 **注意：**使用不正确的网络证书会抛出一个安全错误。所以你必须定义一个处理块，用该处理块定义正确的网络证书并且重复访问服务器端的操作。

3. 运行应用程序。

在代码中文件位置（DocumentLocation）

你可以通过代码在Form_Load事件中设定SSRS文件的位置。
参见SSRSDataSource快速入门学习如何使用设计器设定文件的位置。

代码

C#

```
c1SSRSDataSource1.DocumentLocation = new  
C1.Win.C1Document.SSRSSReportLocation("http://ssrs.abc.com/ReportServer", "/Adventure  
Works/Sales");
```

VB

```
c1SSRSDataSource1.DocumentLocation = New  
C1.Win.C1Document.SSRSSReportLocation("http://ssrs.abc.com/ReportServer", "/Adventure  
Works/Sales")
```

显示文件的状态和页数

该章节描述如何使用C1SrsDataSource组件的PageCount 和 State属性来显示文件源的状态以及已经产生的页数。如快速入门所示的创建一个应用程序然后完成下面几个步骤。

在设计器中

1. 在窗体中放置两个label。
2. 设定第一个label的Name属性为tbReport。
3. 设定第二个label的Name属性为tbState。
4. 设定第一个label的Text属性为Report: None。这个标签将显示报表的位置。
5. 设定第二个label的Text属性为State: Unknown。这个标签将显示报表的状态和产生的页的页数。

在代码中

在Form_Load的事件中添加下面的代码，这样就可以将报表的位置，状态和页数显示在标签上。

C#

```
SSRSReportLocation rl = c1SSRSDataSource1.DocumentLocation as SSRSReportLocation;
if (rl != null)
    tbReport.Text = C1.Win.C1Srs.ReportSession.Combine(rl.ReportServer,
rl.ReportPath);
else
    tbReport.Text = "None";
switch (c1SSRSDataSource1.State)
{
    case C1DocumentSourceState.Generated:
    case C1DocumentSourceState.GeneratedWithErrors:
    case C1DocumentSourceState.Generating:
        tbState.Text = string.Format("{0}, Pages: {1}", c1SSRSDataSource1.State,
c1SSRSDataSource1.PageCount);
        break;
    default:
        tbState.Text = c1SSRSDataSource1.State.ToString();
        break;
}
```

VB

```
Dim rl As SSRSReportLocation = TryCast(c1SSRSDataSource1.DocumentLocation,
SSRSReportLocation)
If rl IsNot Nothing Then
    tbReport.Text = C1.Win.C1Srs.ReportSession.Combine(rl.ReportServer,
rl.ReportPath)
Else
    tbReport.Text = "None"
End If
Select Case c1SSRSDataSource1.State
    Case C1DocumentSourceState.Generated,
C1DocumentSourceState.GeneratedWithErrors, C1DocumentSourceState.Generating
```



```
        tbState.Text = String.Format("{0}, Pages: {1}",  
c1SSRSDataSource1.State, c1SSRSDataSource1.PageCount)  
        Exit Select  
    Case Else  
        tbState.Text = c1SSRSDataSource1.State.ToString()  
        Exit Select  
End Select
```

你完成了什么

当运行你的项目时，注意你放置在窗体上的标签显示了报表的位置，状态和已经生成的页面的页数。

