

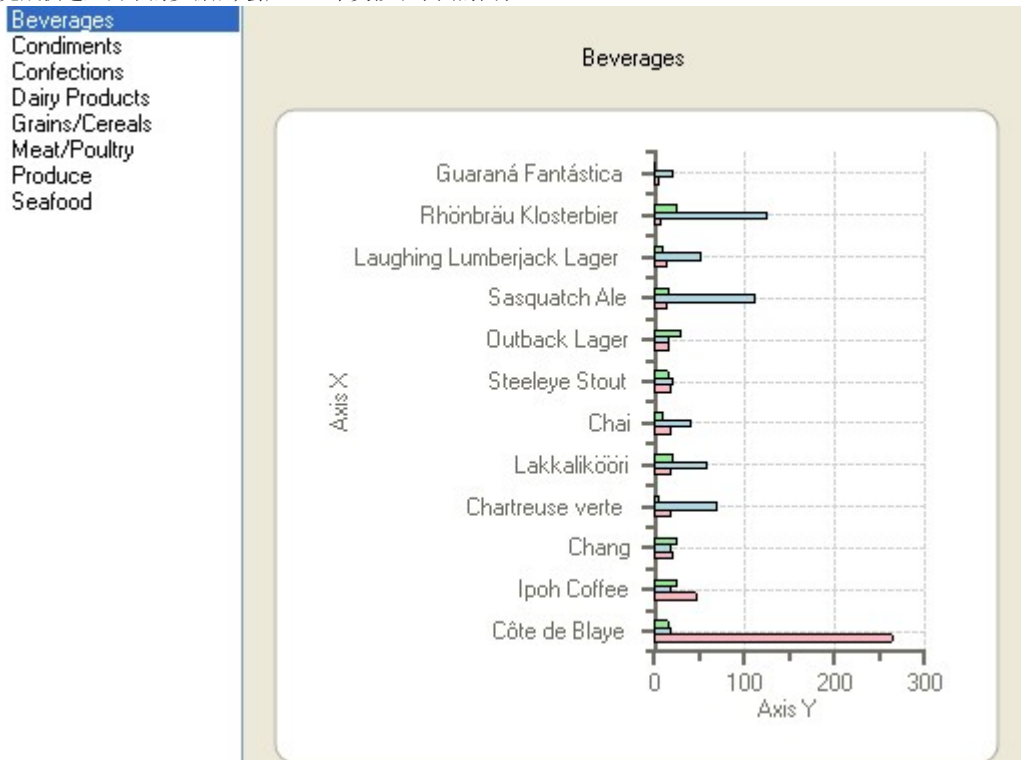
Chart for WinForms 快速入门

本部分引导您一步步地建立一个报告,它描述了产品的单价,单位库存,和按照类别组织的再订货标准.这个报告使用一个简单的条形图来展示信息,该图中 Y 轴表述了产品的名称,然后 X 轴代表了产品的单价,单位库存,和再订货标准.该图表使用三个序列来绘制产品的单价,单位库存,和再订货标准.一个图例用来说明每个序列的颜色

图表使用一个简单的 Access 数据库 Nwind.mdb 中的数据.快速入门假设 C:\Nwind.mdb 文件存储在“<PersonalDocumentsFolder>\ComponentOne\Samples\Common”路径下,其中的

<PersonalDocumentsFolder>路径时用户的 Documents 文件夹,会随着用户和平台的差异而不同.

完成快速入门中的步骤后,会产生一个类似于下图的图表.



四个步骤中的第一步:为图表创建数据

该步骤创建一个数据源,后续动作可以使用 Chart Properties 设计器来将该数据源绑定到图表中.创建一个 .NET 工程,并完成以下的步骤:

添加一个新的数据源

1. 在工程的工具栏中,从Data菜单中选择Add New Data Source. Data Source Configuration Wizard对话框会出现.
2. 选择Database,然后点击Next按钮.
3. 点击New Connection.
4. 在Choose Data Source对话框中,选择Microsoft Access Database File,然后点击 Continue.
5. 在Add Connection对话框中点击Browse.
6. 在Select Microsoft Access Database File对话框中,选择Nwind.mdb文件(默认放在 C:\Program Files\ComponentOne\Studio for Winforms \Common路径下),点击 Open,然后点击OK.
7. 点击Next来继续操作.一个对话框将会出现,它提示您是否添加数据到您的工程中并且修改链接.因为没有将数据拷贝到工程的必要,所以点击NO.
8. 确认Yes, save the connection as复选框是选中的,然后点击Next来继续.

a. 链接字符串被当做NwindConnectionString保存

1. 展开Tables节点并选中Categories和Products对象.
2. 点击Finish按钮

NwindDataSet.xsd会添加到您的工程中

添加一个OleDbDataAdapter

1. 在工具栏中,双击OleDbDataAdapter组件注意:在 Visual Studio 2005 中,右键点击工具栏.然后选择 Choose Items.在对话框中的.NET Framework Components 选项卡页中,选择 OleDbDataAdapter.

组件托盘上将会出现OleDbDataAdapter,并且Data Adapter Configuration Wizard将会出现.

1. 在Data Adapter Configuration Wizard中,在下拉列表中选择您将为数据适配使用的连接串(在本例中, C:\Program Files\ComponentOne\Studio for Winforms Common\Nwind.mdb \Nwind.mdb).然后点击Next按钮.
2. Use SQL Statements选项默认是选中的,然后点击Next按钮.
3. 在Data Adapter Configuration Wizard的文本输入框中复制和粘贴如下的SQL语句:

```
SELECT CategoryID, ProductName, UnitPrice, UnitsInStock, ReorderLevel FROM Products ORDER BY UnitPrice DESC
```

1. 点击Next然后点击Yes来在您的查询中增加主键列。
2. 点击Finish. 请注意OleDbConnection1组件会自动地出现在组件托盘中。

生成一个DataSet 生成一个关于 OleDbDataAdapter1 对象的 DataSet, 需要完成以下步骤:

1. 选中OleDbDataAdapter1, 然后点击它的智能标签, 然后点击Generate DataSet . 然后 Generate DataSet对话框将会出现。
2. 请确认Existing单选按钮是选中的. Products表是选中的. 然后Add this dataset to the designer选项也是选中的, 然后点击OK.

nwindDataSet1会被添加到组件托盘中。

添加第二个OleDbDataAdapter

1. 在工具栏中, 双击OleDbDataAdapter组件, 来添加另外一个OleDbDataAdapter组件到组件托盘中。
2. 选中显示了目录的数据连接, 并从下拉列表中选择C:\Program

Files\ComponentOne\Studio for Winforms \Common\Nwind.mdb .

1. 点击Next来继续。
2. 请确认Use SQL statements是选中的, 然后点击Next.
3. 在Data Adapter Configuration Wizard的文本输入框中, 输入以下的SQL语句:

```
SELECT CategoryName, CategoryID FROM Categories
```

1. 点击Next, 然后点击Finish.

为OleDbDataAdapter2生成一个DataSet 生成一个关于 OleDbDataAdapter2 对象的 DataSet, 需要完成以下步骤:

1. 选中OleDbDataAdapter2, 然后点击它的智能标签, 然后点击Generate DataSet.
2. 在Generate DataSet对话框中点击New, 然后命名它为categoriesDataSet.
3. 请确认Categories表是选中的. 然后Add this dataset to the designer选项也是选中的, 然后点击OK.

categoriesDataSet1会被添加到组件托盘中。

填充DataSets 在Form1_Load事件中, 添加以下的代码来给DataSet中填充数据:

- Visual Basic

```
oleDbDataAdapter1.Fill(nwindDataSet1) OleDbDataAdapter2.Fill(categoriesDataSet1)
```

- C#

```
oleDbDataAdapter1.Fill(nwindDataSet1); OleDbDataAdapter2.Fill(categoriesDataSet1);
```

添加 DataView 组件返回到设计视图, 然后完成以下步骤:

1. 在工具栏中, 双击DataView组件来将其添加到组件托盘中。

注意:在 Visual Studio 2005 中, 右键点击工具栏. 然后选择 Choose Items. 在对话框中的.NET Framework Components 选项卡页中, 选择 DataView.

1. 在DataView属性窗口中以以下方式设置属性:
 - AllowDelete = False
 - AllowEdit = False
 - AllowNew = False
 - Table = nwindDataSet1.Products

恭喜您!您已经成功地创建了一个数据源. 下一个步骤将会向您展示如何添加一个图表并将其绑定到既存的数据源中, 并且还有如何使用 Chart Properties 设计器来轻松地定制您的图表。

四个步骤中的第二步:绑定数据源到 C1Chart 中

为了将您在前述步骤中创建的数据源绑定到图表中, 请完成以下步骤:

1. 在工具栏中双击 C1Chart 组件来将其添加到窗体中. 将 C1Chart 控件的大小设置为

400X400.

1. 在图表上右键点击, 然后在菜单中选择 Chart Properties. 然后 Chart Properties 设计器将会出现。
2. 点击Gallery, 然后选择Simple Types选项卡中的SelectBar. 在右边的面板中, 选择第一行中的第一个条形图图片. 然后点击 Apply 按钮。
3. 点击 Data 然后选择 Series 3. 点击 Remove 来在条形图表中删除 Series 3.
4. 将 Series 0 标记为单价, 将 Series 1 标记为单位库存, 将 Series 2 标记为再订货标

准. 然后点击 Apply 按钮.

1. 在树视图中, 选择 Data>Binding. 然后从 Data source 的下拉列表中选择

DataView1.

1. 在Data Series下拉列表中点击Data来选择单价. 在数据绑定中, 将X轴的数据字段设置为 ProductName. 然后将 Y 轴的数字字段设置为

UnitPrice.

2. 选择单位库存序列. 然后将 X 轴的数据字段设置为 ProductName. 然后将 Y 轴的数字字段设置为 UnitsInStock.
3. 选择再订货标准序列. 然后将 X 轴的数据字段设置为 ProductName. 然后将 Y 轴的数字字段设置为 ReorderLevel.
4. 这会将 DataSet 绑定到图表上. 在这之后如果您检查

ChartGroups>Group0>ChartData>SeriesList, 您将会得到适合 Y 数据字段的数据字段名称.

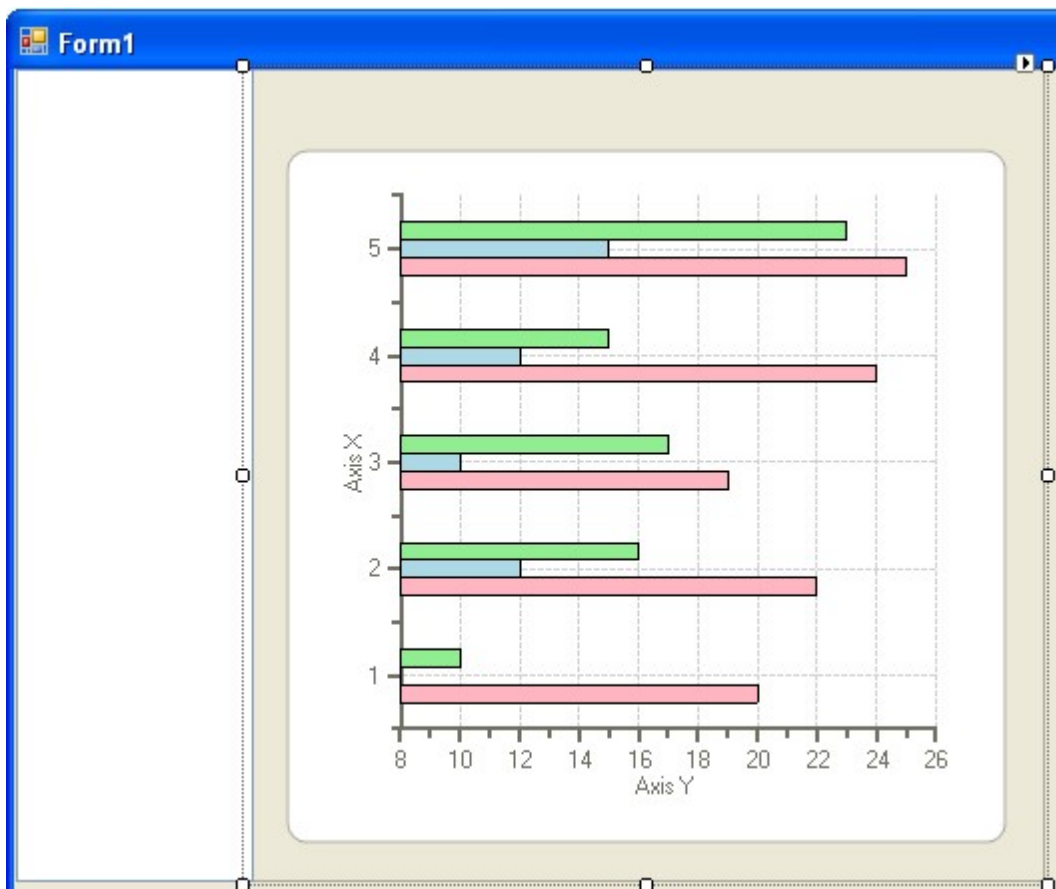
1. 在树视图中, 选择 Appearance>Header.
2. 在 Appearance [Header]选项卡上, 点击 Visible 复选框来启用图表表头.
3. 点击 OK 来关闭 Chart Properties.
4. 运行应用程序

您已经成功地连接图表到数据源并且配置好了图表的设置. 为了让我们的数据在图表上更加的可读. 在下一个步骤中我们将给数据添加一个列表框来过滤产品种类.

四个步骤中的第三步: 将列表框绑定到 DataSet 中

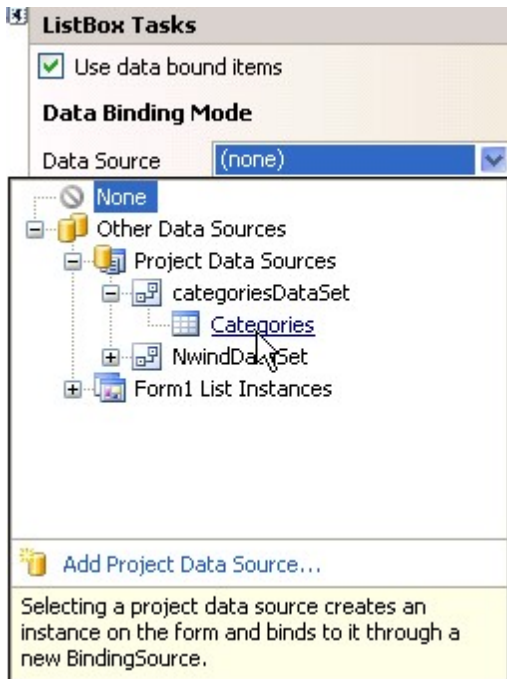
为了将数据进行分类, 完成以下步骤:

1. 在工具栏中, 双击ListBox控件来将其添加到窗体上去. 将其停靠靠在C1Chart控件的左边, 以达到下图的效果:



1. 选择ListBox控件并点击其智能标签来打开菜单. 选择Use databound items, 然后在

Data Source下拉列表中按照Other Data Sources>Project Data Sources>CategoriesDataSet的顺序选择Categories.



1. 设置DisplayMember为CategoryName.
2. 在ListBox上双击来产生listbox1_SelectedIndexChanged事件的处理.
3. 在Listbox1_SelectedIndexChanged事件处理函数中添加以下的代码, 来在用户选择一个分类项目是在下拉列表中过滤CategoryID.

- Visual Basic

```
Private Sub listBox1_SelectedIndexChanged(sender As Object, e As System.EventArgs) Handles
listBox1.SelectedIndexChanged
If listBox1.SelectedIndex >= 0 Then
Dim categoryID As String =
Me.categoriesDataSet1.Categories(listBox1.SelectedIndex).CategoryID.ToString ()
Me.dataView1.RowFilter = "CategoryID = " + categoryID
Me.clChart1.Header.Text = listBox1.Text
End If
End Sub
```

- C#

```
private void listBox1_SelectedIndexChanged(object sender, System.EventArgs e) {
if (listBox1.SelectedIndex >= 0)
{
string categoryID =

this.categoriesDataSet1.Categories[listBox1.SelectedIndex].CategoryID.ToStri ng(); this.dataView1.RowFilter = "CategoryID
= " + categoryID; this.clChart1.Header.Text = listBox1.Text;

}
}
```

1. a. 在Form1_Load事件中添加如下的代码, 以在重新填充后进行新的计算. 这样做的话, 将会展示第一个分类的产品元素而不是所有未分类的元素.

- Visual Basic

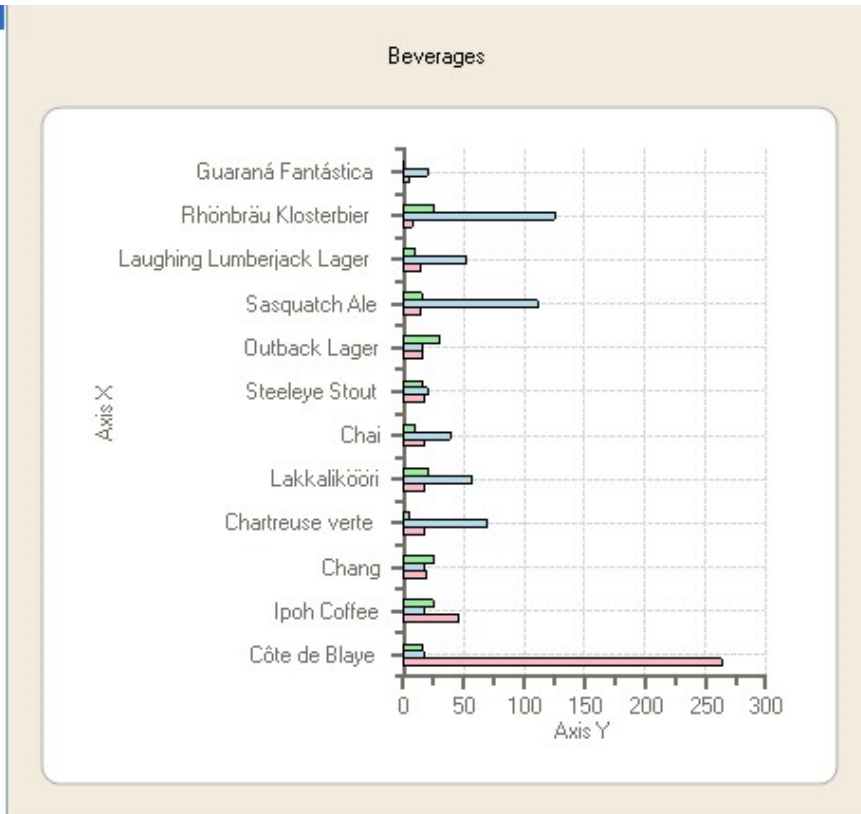
```
'force the new calculation after the refill
listBox1_SelectedIndexChanged(Me.listBox1, New EventArgs())
```

- C#

```
//force the new calcaulation after the refill
listBox1_SelectedIndexChanged(this.listBox1, new EventArgs());
```

1. a. 运行程序并在下拉列表中选择分类来观察进行过数据过滤的图表恭喜您!您已经将数据绑定到了图表上. 在下一个步骤中, 您将修改图表的外观.

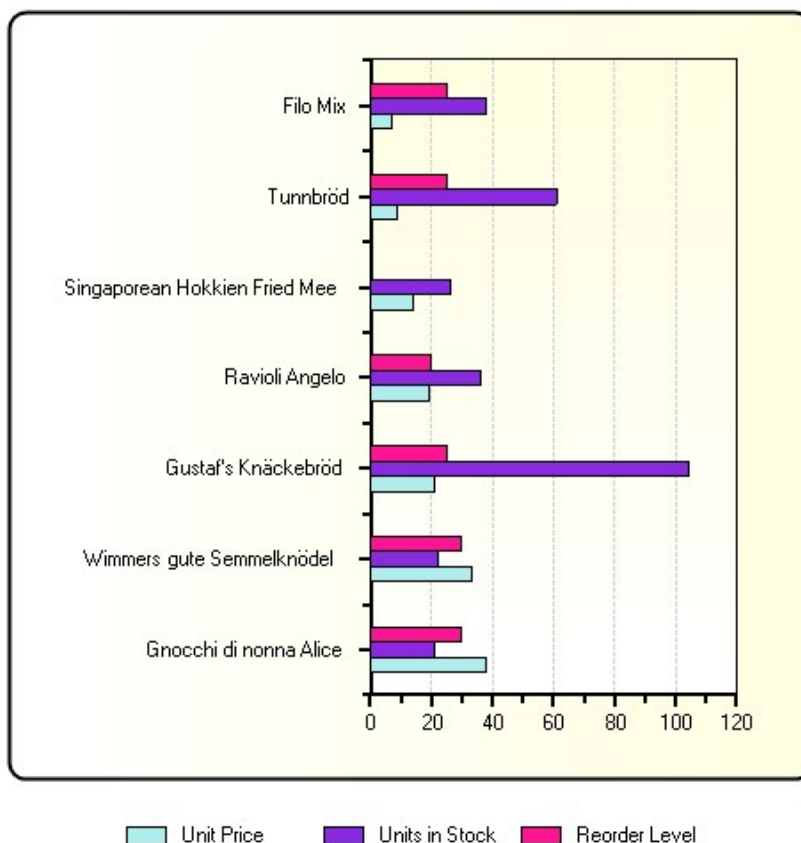
Beverages
Condiments
Confections
Dairy Products
Grains/Cereals
Meat/Poultry
Produce
Seafood



四个步骤中的第四步:自定义图表

在这个部分,您将学习如何使用内置的格式化效果来创建更有视觉效果和看起来更加专业的图表。
当您完成指南的步骤后,您的图表将看起来如下:

Grains/Cereals



C1Chart 将图表的背景色格式化为一个控件的颜色. 然后将绘制区域涂成白色. 但是您可以选择一个其它的颜色, 或者不使用颜色, 或者使用两个颜色来创造出很有创意的渐变效果.

改变数据序列的颜色

1. 在C1Chart控件上右键点击, 并选择Chart Properties.
2. 在Chart Properties设计器中, 选择Data.
3. 在ChartData选项卡中, 在数据序列分组框中选择各个序列, 然后在填充下拉列表复选框上点击, 来选择一个特定的填充色.
4. 选择Simple填充类型, 然后将单价的颜色设置为PaleTurquoise, 库存单位的颜色设置为BlueViolet, 再进货标准的颜色设置为DeepPink.
5. 点击OK.

请注意图表序列上的亮色和暗色的对比会更容易地看出图表序列上的差异.

为了在图表中给数据留有更多的空间, 我们可以调整图表图例的位置修正图例的位置

1. 在C1Chart图表上右键点击, 并选择Chart Properties.
2. 在Chart Properties设计器中, 选择Appearance>Legend.
3. 在Position下拉复选框中, 选择South.
4. 点击OK.

这将会释放水平方向上的空间. 所以图表将会看起来更加的吸引人和更加的可读.

为了让图表看起来不是那么的拥挤. 我们可以删除水平网格线. 因为我们仅需要垂直网格线来帮助我们阅读数据.

删除水平网格线

1. 在C1Chart图表上右键点击, 并选择Chart Properties.
2. 在Major grid group中, 点击Visible复选框来取消选中. AxisX网格线会在图表中消失.
3. 点击OK. 为了在图表中更加地突出文字的效果, 您可以修改图表的字体样式和颜色.

改变图表表头的字体样式

1. 在C1Chart图表上右键点击, 并选择Chart Properties.
2. 在Chart Properties设计器中, 选择Appearance>Header.
3. 在ForeColor下拉列表中点击, 然后选择Black.
4. 在Font按钮上点击来打开Font对话框. 改变字体为Arial, 样式为Bold, 然后大小为10.
5. 点击OK.

下一步, 我们将改变图表的边框颜色从控件色到黑色来让它跟图表文字的颜色形成鲜明的对比.

改变图表边框颜色

1. 在C1Chart图表上右键点击, 并选择Chart Properties.
2. 在Chart Properties设计器中, 选择Appearance>ChartArea
3. 在Appearance [ChartArea]选项卡中, 点击Border下拉列表复选框, 然后选择Solid 样式, Black颜色, 和2的厚度.
4. 点击OK. 为了给图表添加一个样式, 我们可以给整个图表添加一些渐变填充效果.

给图表添加渐变填充效果

1. 在C1Chart图表上右键点击, 并选择Chart Properties.
2. 在Chart Properties设计器中, 选择Appearance
3. 在Appearance选项卡中, 点击Fill下拉列表复选框, 然后选择Gradient选项. 分组框中会出现几种渐变选项. 选择第一种渐变并选择Light Yellow作为Color1的颜色, White 作为Color2的颜色. 然后点击OK. 运行程序来观察图表的新外观.

恭喜您!您已经完成了 C1Chart 的快速入门. 在这次的快速入门中您学习到了: 静态地绑定数据表到图表中使用 Chart Properties 设计器来自定义 C1Chart 的属性.