

VisualC 2005中开发自定义绘图控件 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/253/2021_2022_VisualC__2_c97_253835.htm 在您决定开发 Windows 提供的常规免费自定义控件范围之外的控件之后，您必需确定自己的控件将有多少独到之处 在功能和外观两方面。例如，我们假定您正在创建一个类似于计速表的控件。由于公共控件库 (ComCtrl32.dll) 中没有类似的控件，您完全需要自己进行以下操作：编写所有控件功能需要的代码，进行绘制，默认终端用户的交互，以及控件与其父窗口之间需要的任意消息处理。另一方面，还包括一些您只想调整公共控件功能的情况。例如，我们假定您想创建一个屏蔽编辑控件，它只允许接受指定的字符。如果使用 MFC，通常涉及从 MFC 提供的类派生一个类，该类封装了一个公共控件（在屏蔽编辑控件中，通常为 CEdit），重写必需的虚函数（或处理指定的消息），然后加入自定义的代码。本文讨论的重点介于两者之间 公共控件赋予您想要的大部分功能，但控件的外观并不是您想要的。例如，列表视图控件提供在许多视图风格中显示数据列表的方式 小图标、大图标、列表和详细列表（报告）。然而，如果您想要一个网格控件，那结果怎样呢？尽管公共控件库里没有特别包含网格，但是列表视图控件与它较为接近，它以行和列显示数据，并有一个相关的标头控件。因此，许多人以一个标准的列表视图控件为起点创建自己的网格控件，然后重写该控件及其子项的呈现方式或绘制方式。主宰绘图操作 即使“只”进行绘制，您仍然有至少四种选项可用，它们都具有鲜明的优缺点：处理 WM_PAINT 所有者绘制 自定义绘制 处理

WM_CTLCOLOR 处理 WM_PAINT 最极端的选择是执行一个 WM_PAINT 处理程序，并且自己完成所有的绘制。这意味着，您的代码将需要进行一些与呈现控件相关的琐事：创建适当的设备上下文（一个或多个），决定控件的大小和位置，绘制控件等。在绘制过程中，很少需要这种级别的控件。所有者绘制 控制控件绘制的另一种方法是利用所有者绘制。事实上，您也许听开发人员提到过所有者绘制控件，因为它是用于开发自定义控件最普通的技术。该技术普遍使用的主要原因在于，Windows 可为您提供很多帮助。在呈现控件的那一刻，Windows 就已经创建并填写了设备上下文，决定了控件的大小和位置，并且向您传递信息以使您了解此刻绘制的需求。对于列表控件（例如，列表框和列表视图），Windows 将为列表中的每一项调用绘制代码，这意味着您只需绘制这些项，而无需考虑控件的其他方面。注意，所有者绘制可用于大多数控件。然而，它不能用于编辑控件；并且考虑到列表控件，它只能用于报表视图样式。自定义绘制 对于绘制自己的控件而言，这可能是最少为人所知的技术。事实上，许多技术能力较高的开发人员也混淆了术语所有者绘制 (owner-draw) 和自定义绘制 (custom-draw)。关于自定义控件，首先需要了解，它仅针对于指定的公共控件：标头、列表视图、rebar、工具栏、工具提示、跟踪条和树视图。此外，尽管所有者绘制只允许绘制报告视图风格的列表视图控件，而自定义绘制则使您能够处理列表视图控件所有视图风格的绘制。使用自定义绘制的另一个明显优势是，您可以对希望绘制的内容进行严格挑选。实现方式是，在控件绘制的每个阶段由 Windows 向代码发送一个消息。这样，您可以决定在

每个阶段是自己进行所有的绘制工作，增加默认的绘制，还是允许 Windows 为该阶段执行所有的绘制。（鉴于自定义绘制是本文的一个主题，因此您很快会看到它的工作方式。）

处理 WM_CTLCOLOR 这可能是帮助决定如何呈现控件最简单的方式。正如消息名所指，当要绘制一个控件，并且它能让您的代码决定要使用的画笔时，发送 WM_CTLCOLOR 消息。通常情况下，如果您只想更改控件的颜色，并且不提供除控件本身之外的更多功能，则使用该技术。此外，对于由 Internet Explorer 引入的公共控件（列表视图、树视图、rebar 等），不发送该消息，并且它只与标准控件（编辑、列表框等）协同使用。

实现自定义绘制的三步曲 既然您已经了解了绘制控件可用的各种选项（包括使用自定义绘制的好处），那么，让我们来看看实现一个自定义绘制控件需要的三个主要步骤。执行一个 NM_CUSTOMDRAW 消息处理程序。指定处理所需的绘制阶段。筛选特定的绘制阶段（在这些阶段中，您需要加入自己的特定于控件的绘制代码）。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com