鱼眼视图的源代码应用

——Visual Studio编辑器扩展

软件工程 齐笑尘 指导教师 邬江

【摘要】 软件开发是一项高度复杂的劳动，为了提高编写程序、开发软件的效率，改善开发人员的体验，来自各个领域的许多理论和技术被应用到编程工具和软件开发环境当中，其中应用较多的理论来自人机交互领域。本设计基于信息可视化方向的鱼眼视图理论，通过设计实现主流的软件开发环境Visual Studio编辑器的扩展应用，探究鱼眼视图扩展在该开发环境中的可行性与实际效果。根据需求，设计选择了开发环境本身提供的辅助效果较少的C/C++语言。在设计过程中受到较多技术上的限制，发现Visual Studio提供的扩展性无法完全满足参考文献中鱼眼视图的设计。利用开发环境提供的支持，最终的扩展实现了鱼眼视图主要的框架和基本功能。得到的结论是鱼眼视图的源代码应用与编程语言本身特性有关，在Visual Studio平台下针对C/C++语言的鱼眼视图的完美实现有一定难度。如果开发平台可以提供更加强大的用户界面接口，同时配合相关语言的前端解析支持，鱼眼视图会在Visual Studio平台上提供给用户一个体验更好的浏览编辑代码环境。

【关键词】 鱼眼视图 VisualStudio 扩展

【Abstract】Software development is of high complexity. In order to improve the productivity of programming and software development, theories and techniques from many fields has been applied to the programming tools and integrated development environments(IDE), most of which are from the field of human-computer interaction. This project is based on the Fisheye Visualization theory, and through designing and implementing an extension for Visual Studio editor to explore the possibility and actual effect for applying fisheye to this IDE. The research and development process got quite a few constraints from the techniques available, and the expected fisheye effect is not fully supported by the IDE extensive functionality. The final application includes mainframe of the fisheye design and basic functions. To conclude, the fisheye view effect for source code is closely related to the programming language. If Visual Studio could provide more powerful user interface extensibility and corresponding programming language forefront parser, the fisheye view would be more helpful for browsing and editing code.

【Keywords】FisheyeView VisualStudio Extension

目录

[1 绪论 4](#_Toc324101650)

[1.1 源代码应用鱼眼视图的背景 4](#_Toc324101651)

[1.2 现有的一些鱼眼视图研究 4](#_Toc324101652)

[1.3 本课题的研究范围及应解决的主要问题 4](#_Toc324101653)

[2 鱼眼视图理论 5](#_Toc324101654)

[2.1 鱼眼视图的核心思想 5](#_Toc324101655)

[2.2 兴趣指数（Degree of Interest） 5](#_Toc324101656)

[2.3 鱼眼视图应用需要考虑的问题 5](#_Toc324101657)

[3 VS2010编辑器的扩展性与鱼眼视图的设计 5](#_Toc324101658)

[3.1 5](#_Toc324101659)

[4 编程语言相关性 5](#_Toc324101660)

[4.1 鱼眼视图的核心思想 5](#_Toc324101661)

[5 扩展应用的实现情况 6](#_Toc324101662)

[5.1 鱼眼视图的核心思想 6](#_Toc324101663)

[6 结论 6](#_Toc324101664)

1. 绪论
   1. 源代码应用鱼眼视图的背景

软件开发（编写程序）是一项复杂的智力密集型劳动，是至今为数不多的机器不能够有效地代替人完成的工作之一。随着硬件的升级，软件系统规模越来越大，复杂程度也越来越高，编写程序与维护程序的成本也随之加大。

尤其是大型软件的开发，往往需要耗费大量人力和时间。即使软件一般都会有优良的设计和实现，开发人员也具备丰富的经验，但是，由于代码的数量很大、结构极其复杂，浏览和理解代码对开发人员来说是一个挑战，如果没有相应的展示增强工具，开发效率会受到严重挑战。我在实习工作中参与了一个大型商业软件的开发，使用Visual Studio经常会面对单文件几千行源代码，代码的浏览效果不尽如人意。对于较长的代码，开发人员无论是浏览还是添加新的代码都需要灵活地跳转代码的位置，同时也希望能够浏览到尽量多和重要的代码信息来节约时间。

和源代码展示相关的理论研究领域是信息可视化(Information visualization)，该领域主要关注大规模信息的可视化展示，比如软件中的文件和源代码、图书馆和文献系统数据库、网络与因特网。其中，鱼眼视图(Fisheye views)是一种比较成熟的理论，具有突出重点、节省展示空间的特性。最早提出信息展示和鱼眼视图结合理论的是George W. Furnas，他的设定的鱼眼视图中，通过比较代码行的重要程度来决定突出显示重要代码或者隐藏非重要代码。

* 1. 现有的一些鱼眼视图研究

鱼眼视图其实是受自然界生物的启发，也符合人类视觉效果的一般习惯，即离的越近的物体和主观上越关注的物体越容易察觉到。人类模拟鱼眼制造出了鱼眼镜头产生的就是这样夸张的效果，在有限的展示范围内放大易于捕捉的和希望捕捉到的画面，同时缩小甚至略去难捕捉的和没有兴趣捕捉的画面。

在信息可视化领域，我们接触的较多的都是鱼眼视图在一般人机交互界面的应用。比如Mac OS X的Dock，一些手持设备的图片浏览器，一些软件的复杂菜单等。上述这些例子多为提供一种视觉效果，是作为一种吸引用户的手段出现的。而本文的主题，展示源代码的鱼眼视图应用，自从Furnas于1986年首次提出以来，却未在主流集成开发环境中出现较成熟且广泛使用的解决方案。

* 1. 本课题的研究范围及应解决的主要问题

从程序开发人员的实际的角度来看，更高效地浏览、编辑源代码的需求应该比一般用户浏览工具栏图标和菜单项要迫切。本课题基于如下的情境：使用Visual Studio 2010（以下简称VS2010）查看、编辑编写规范的C/C++源代码文件。这也是我在实习工作中的实际任务需求。

一方面，本课题需要学习鱼眼视图的基本理论，研究VS2010所提供的扩展开发环境，结合其特点来设计实现VS2010编辑器下的鱼眼视图扩展；另一方面，在实现过程中探究在该开发环境下鱼眼视图扩展应用的可行性与实用程度，提出该设计可能存在的一些问题，来解释鱼眼视图未能如预期应用到源代码展示中的原因。

1. 鱼眼视图理论
   1. 鱼眼视图的核心思想

计算机科学以及互联网技术的不断发展使得人们每天面对的信息量呈爆炸式增长，但与此同时用于展示、获取这些信息的交互设备并没有产生相应规模的进化。这主要还是受限于人本身获取信息的特点和能力。鱼眼视图理论就是在这种背景下产生的，提出该理论的是一位认知心理学家，George W. Furnas，他通过跨学科的研究阐释了鱼眼视图的原理和应用构想。

鱼眼视图在视觉上很像鱼眼镜头——距离视觉焦点近的部分被夸张放大，而距离焦点远的部分被缩小、减少细节甚至略去，但是仍然完整地展示一个范围。鱼眼视图提出的动机是为了在获取信息时能兼顾局部细节和整体结构。这种需求本质上符合人类认知事物的特点，其结构也与信息在人脑中的组织相似。举例来说，在一个公司内部，新员工会很快熟悉自己所在部门的同事，但是对于其他部门，开始只是对其领导有印象；地方性的报纸报道的信息，和本地相关的内容涵盖各个方面，比如本地蔬菜价格上涨，但是世界范围内只会选择诸如朝鲜发射火箭这样重要的内容报道。因此，在人类和外界的交互中，自然地就体现出鱼眼视图的特点。如果可以将信息以鱼眼视图的形式展示出来，用户会更容易理解整体并且高效地获取自己所需的部分。

* 1. 兴趣指数（Degree of Interest）

鱼眼视图通过引用一个称为“兴趣指数”（Degree of Interest）的量化标准作为计算标准。按照Furnas提出的公式：DOIfisheye(x|.=y) = API(x) – D(x,y)，该值由两部分计算得出，其中优先重要性(A Priori Importance)是x点在视图中的绝对重要程度；D(x,y)是x点和视觉焦点y的距离。根据公式，兴趣指数和优先数量级呈正相关，与距离呈负相关。

我们将计算机中数据信息组织形式统一抽象为树形结构来具体应用上述计算方法。其实不难发现计算机系统中树形结构还是比较常见的：最容易想到的就是操作系统中的文件系统，除此之外还有目录和菜单等，对于结构化的编程语言也是以树形结构组织的。（源代码经过编译器前端解析，会生成抽象语法树，继而经过后端进行优化并生成机器码。）

在树形结构中，以两个节点的路径表示两点间距离，以某点距离根节点的路径表示其优先重要性，得到公式DOIfisheye(tree)(x|.=y) = -(dtreee(x,y) + dtree(x,root))。通过该公式计算出树形结构中每个节点的兴趣指数之后，我们可以设定一个阈值来限制鱼眼视图最终展示的部分。合适的阈值

* 1. 鱼眼视图应用需要考虑的问题

从实际的角度来看，更高效地浏览、编辑源代码的需求应该比一般用户浏览工具栏图标和菜单项要迫切。本课题基于如下的情境：使用Visual Studio 2010（以下简称VS2010）查看编辑C/C++源代码。这也是我在实习中的工作需求。

1. VS2010编辑器的扩展性与鱼眼视图的设计
2. 编程语言相关性
   1. 鱼眼视图的核心思想

类比到

1. 扩展应用的实现情况
   1. 鱼眼视图的核心思想

类比到

1. 结论