

****

|  |
| --- |
| **畅想器** |
| **Imagination Container** |

|  |
| --- |
| **畅想器** |
| **Imagination Container** |

摘要

用C#在Visual Studio 中的WPF (Microsoft Presentation Foundation)进行开发；构建一个多人即时交流的平台，该平台主要支持多人文字交流，实时合作绘图，可弥补远程会议仅仅通过文字交流造成的不顺畅，并具有相当的娱乐性。本软件目标需求是多人联网平面设计，也支持扩展型需求，基础功能是联网通讯，自定义的数据和功能通讯。采用B/S架构，但是区别于B/S结构的一点是支持局域网，每个客户端都可以是一个服务器。

**关键词**：桌面应用 共同操作 远程连接 实时更新 可扩展

Abstract

By using the programming language C# in WPF, Microsoft Presentation Foundation of Visual Studio, we create a platform for people to communicate, which supports the basic text communication and the real-time collaboration of drawing. Such a platform will make up for the shortage of plain text, and it is fun to draw together to a large extent. The basic function includes communicating through the Internet and self-defining data. According with the B/S framework, the project does more because every client can be a server. Extensions supported.

**Keywords**: desktop application, work together, remote connection, Real-time update, Extensible.

目录

[第一章 绪论 1](#_Toc383210531)

[1.1 项目背景 1](#_Toc383210532)

[1.2 项目意义 1](#_Toc383210533)

[第二章 项目设计 1](#_Toc383210534)

[2.1 系统总体设计 1](#_Toc383210535)

[2.2 功能模块设计 2](#_Toc383210536)

[2.3 界面设计 2](#_Toc383210537)

[第三章 项目功能及原理 3](#_Toc383210538)

[3.1 项目功能简介 3](#_Toc383210539)

[3.2 实现原理： 3](#_Toc383210540)

[第四章 项目创新点 4](#_Toc383210541)

[4.1 创新点一 4](#_Toc383210542)

[4.2 创新点二 5](#_Toc383210543)

[4.3 创新点三 5](#_Toc383210544)

[第五章 市场应用前景 5](#_Toc383210545)

[结论 5](#_Toc383210546)

## 第一章 绪论

### 1.1 项目背景

在线交流平台确实不少，不过不论是QQ，YY，阿里旺旺等等，其主要的交流方式就是文本对话，以及在线语音。这类软件的共性就是并没有实现除了文本，语音之外的其他辅助交流方式。

本平台就是想在原有基础上加上一个共有的画布，在同一个群组里的用户将共同完成对同一张图像的绘制。这样它可以集实用性和娱乐性于一身，在用户之间交流的同时，可以画图辅助说明，而且其他人也可以随时添上一笔，使得交流更加无缝契合。

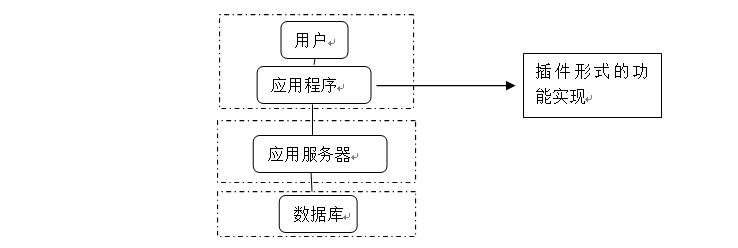
### 1.2 项目意义

本项目采用EMF框架，将软件的功能以插件的形式实现，如此平台的可扩展性大大提升。关于软件的运作方式，我们计划开发离线和多人在线两种模式。在线模块中，将向服务器发请求进行登录，然后邀请用户成为好友或者进入交流界面开始共同绘图等其他操作。为了缓解服务器压力和兼容局域网联机的情况，在进入绘图界面后，我们考虑将服务器转交给邀请人的计算机，并借此临时服务器完成数据更新与广播的任务。

当然，EMF框架保证了软件很好的扩展性，而且我们将计划提供API，支持用户自定义插件。于是便可以实现“什么事情能一起做，那就可以在线在此平台一起做。” 多人文档处理器，多人同时编辑文档，比如多人同时填表，并收纳整理的功能；多人工作流水线；自定义式服务，针对特定行业，多人工作，就开发一个流水线作业；多人电影院，视频播放，多人在线看视频，支持弹幕和实时语音或文字通信；多人游戏平台，便利的多人游戏开发模块，提供底层数据通信，TCP和UDP连接，也可以将IP传入后就放弃底层，完全由自己的程序来控制；数据共享接口便于p2p传输数据等等。只要能想到，就能实现相应功能的插件。

## 第二章 项目设计

### 2.1 系统总体设计



**图 1系统总体结构图**

### 2.2 功能模块设计

应用层将使用WPF来进行界面开发，应用服务器层将使用C#来完成，数据库则采用关系型数据库MySQL进行数据的增删改查。图1并不完全描述了产品特性，因为我们计划采用转接用户端服务器的方式来减轻应用服务器的压力。对此在用户端我们还将实现一个服务器的构建，并且构建多播模型。

**应用程序层**

客户端分两层结构，第一层是软件上层，通过进程间通信和其他软件进行操作的插件。支持要求互连的网络游戏和游戏大厅。客户端的第二层是插件层，各个插件间有可能有相互依赖关系，两个插件也可以相互依赖，只要接口保持不变。

**应用服务器**

服务器将通过操作数据库来响应客户端的一系列请求，将用C#实现服务器的功能，主要有消息更新，服务器多播。

**数据库**

采用关系型数据库MySQL，在数据库中存放用户信息，用户扩展信息，群组，好友关系，系统消息等等。

用户表：id、用户名、密码、昵称、头像、年龄、生日、电话、邮箱、格言、简介、家乡等等

好友关系表： id1 id2 备注 亲密度 社会关系

群组表：id、群组名、群主名、群组人数、群组类型、群组的空间等等

群与用户关系表：表id、群组id、群中用户身份（是群主，管理员，还是一般用户）、用户id、群名片。

系统消息：id、id1、id2、消息类型、时间、内容。

### 2.3 界面设计

界面设计基本上都是客户端的设计了。

登陆界面：登录入口。

注册界面：用户注册并实时返回信息是否有效。

好友列表以及功能列表：选择好友加入群组然后选择功能共同开始操作。

某一功能的界面：每个功能的实现是靠加载插件完成的，界面设计将包含在该插件中，而插件的加入是动态的。例如图2部分界面中的多人绘图(未完成)，将支持在画板上绘图并可以切换选项卡来点击定制画刷的样式和聊天室群聊。

目前的部分界面：



**图 2部分界面**

## 第三章 项目功能及原理

### 3.1 项目功能简介

本项目的功能是动态添加的，采用EMF框架，每一个插件都可以对应一项功能，主框架负责通信以及好友群组系统，而插件则通过调用系统API来实现各种功能。主框架只是一个平台，内置功能只有好友系统，群组以及插件功能加载。

插件将调用主框架提供的API并且需要满足特定的接口规范以供主框架组织运行。

### 3.2 实现原理：

环境：

开发环境 Visual Studio WPF

开发语言 C#

主框架 EMF(Eclipse Modeling Framework Project)

实现：

界面部分：采用WPF的类文件+XAML文档结合，将界面展现出来

服务器部分：应用服务器采用C#并构建两个类Program和MySQL\_Manager

Program类就是服务器，利用线程池模型来响应客户端的请求，并通过MySQL\_Manager来实现数据获取与更新

MySQL\_Manager类则为C#和MySQL数据库之间的连接，以及用C#语言操作数据库。

插件部分：插件的完成体是一个动态链接库(扩展为dll)，于是不存在编程语言的限制，只要可以导出动态链接库，只要实现了规范接口，那么就是可用的扩展插件

接口规范：

public interface Plugin

{

List<User> user\_list;

string Name;

SortedSet<int> Dependencies;

event PluginRunEvent.PluginRunEventHandler runhandle;

byte[] Versions;

void RunPlugin();

void Init();

}

插件的类样例：

[Export(typeof(Plugin))]

class PluginMain : Plugin

{

List<User> \_user\_list;

public string Name

{

get { return "畅想器"; }

}

public byte[] Versions;

public void RunPlugin();

public event PluginRunEvent.PluginRunEventHandler runhandle;

public void Init()

{

Button btn = new Button();

btn.Content= "畅想器";

…….

API.AddUserWindowChild("通信类",btn);

}

Void btn\_PreviewMouseLeftButtonDown(object sender, System.Windows.Input.MouseButtonEventArgs e)

{

Canvas4All mainWindow = new Canvas4All(); //插件窗口类

mainWindow.Show();

var func = runhandle;

if (func != null)

{

func(this, new PluginRunEvent.PluginRunEventArgs(this));

}

}

这段代码即为插件必须实现的一个规范

然后在主框架中

## 第四章 项目创新点

### 4.1 创新点一

高扩展性

每个插件对应一个功能，功能的丰富性是由插件的数量决定的

### 4.2 创新点二

大家一起来

提供一个交流的平台，文字交谈，语音交谈，绘图辅助，一起游戏(插件要求)等等，强调互动，体验新奇

### 4.3 创新点三

服务器转交

在新建房间之后，我们将会把服务器的一部分任务交给房主来做，由房主的计算机担任消息的更新以及多播的实现,于是这也为多人局域网互联提供了实现的可行性

## 第五章 市场应用前景

由其可扩展性的特点来讲，这个平台将可以根据市场的需求来通过插件的更新来适应。同样它还有相当的灵活性。其应用前景在未进行调研的情况下很难描述，因为从需求方面来说，特定插件对应的功能将面向不同的人群。市场的应用前景是取决于插件而不是这个框架。

## 结论

这是一个平台，在这个平台上可以做能想到的几乎任何事(除了煮咖啡)，但是这个平台需要丰富的插件支持，插件才是这个平台的能力。而插件可以开发人员自己提供，也可以是第三方插件。比如多人联网做音乐，联网音乐创作，多个人一起来写音乐段，还可以一起整合起来，每个人在自己的工作空间外还有共享空间；多人联网学习平台，同多人会议室，另加个人白板和考试功能；多人会议室，拥有高质量的通话，和视频聊天功能，便于书写画图的白板和文件共享； 多人代码运行平台，多人连连看、多人台球、多人记事本、多人俄罗斯方块。跨领域的平台，需要保证的是平台的健壮性以及插件的丰富性