CAD secondary development technology based on .NET API

Li Zhang\textsuperscript{1,*}, Peng Zhang\textsuperscript{2}

\textsuperscript{1} Electric Power Research Institute, Yunnan Power Grid Co., Ltd, Kunming, China
\textsuperscript{2} School of Electrical Engineering, Beijing Jiaotong University, Beijing, China

*Corresponding author e-mail: 253097785@qq.com

\textbf{Abstract.} AutoCAD secondary development technology plays an increasingly important role in engineering practice. It can help drafters free from heavy and repetitive tasks and is suitable for different engineering fields. This article details the implementation of secondary development based on the .NET API. This method makes it easier to develop the various plug-ins we need, and to draw various graphics accurately. This paper introduces the development of CAD plug-ins, the drawing of graphics, and the connection between CAD and database. Through the actual development and application, it can be found that the C#-based .NET API technology can easily realize the secondary development of CAD, and the method is easy to grasp and has strong practicability, which can greatly improve work efficiency.

1. Introduction

As a successful computer-aided software, AutoCAD has a wide range of applications in engineering drawing, which can help designers to design drawings. CAD has become the most commonly used tool for modern engineering design, and has been widely used in machinery, construction, electronics and other fields. However, due to the versatility of CAD software, it may cause insufficient or missing functions in some professional fields. Therefore, it is necessary to carry out secondary development of CAD and design professional modules that conform to specific fields. This can extend the application range of CAD, and can also improve the designer's work efficiency and reduce the errors in design, which has strong practical value.

2. AutoCAD .NET API Overview

At present, there are many tools for secondary development of AutoCAD, such as Visual Lisp, VBA and ObjectARX. They each have different advantages and disadvantages. Visual Lisp is developed in Lisp language, which is simple but not intuitive. \textsuperscript{1} VBA is developed in Visual Basic, which is simple and convenient to develop but has insufficient functions. ObjectARX is based on the Visual C++ development language, which is powerful but difficult to master.

The AutoCAD .NET API enables users to manipulate AutoCAD and graphics files programmatically using open assemblies and open libraries. With open objects, they can be accessed by many different programming languages and environments. Implementing the .NET API in AutoCAD has the following advantages:

1) There are more programming environments to access AutoCAD graphics. Previous development tools were limited to using the appropriate language. For example, VBA can only use the Visual Basic language.
2) The .NET framework can be designed for 32-bit and 64-bit operating systems, while VBA can only be designed for 32-bit operating systems.

3) Compared to traditional programming languages, such as C++, .NET has lower learning requirements to access advanced interfaces.

4) Compared with VC++, C# adopts garbage collection mechanism, judges the timing of memory recovery and implements recycling, and automatically releases resources through Dispose function, thus solving the memory leak problem. At the same time, VC# .NET's secondary compilation technology also makes the compiled program successful, after the compiled program can be independent of the hardware, so that the programmer is only focused on the program itself, thus becoming the first choice for programmer development tools. Therefore, under the .NET framework, the C# language is used for secondary development of AutoCAD. It is completely object-oriented and easy to use. It is the most suitable secondary development tool for AutoCAD.

3. Technical route

3.1 AutoCAD ActiveX

AutoCAD displays the programmable objects described by the AutoCAD object model through Automation. These programmable objects can be created, edited, and manipulated by other applications. The application that can access the AutoCAD object model is the Automation control program, and any type of application can access the Automation object displayed in AutoCAD.

C# is a new programming language developed by Microsoft Corporation in the 21st century. It is deeply loved by programmers because of its full object-oriented and powerful library support. By using C# for secondary development of AutoCAD, not only can the powerful functions of the .NET platform be fully utilized for good data encapsulation, object-oriented programming can be performed in a better way, and data exchange with other applications can be better.

AutoCAD provides a rich interface for secondary development. Through these interfaces, users can access AutoCAD built-in objects such as model space and drawing space. The C# application runs in the same process space as AutoCAD. The application obtains an instance of the AutoCAD application to achieve communication between the AutoCAD shared address space and the AutoCAD system. The message mechanism of Windows is used to send a message to the AutoCAD object, so that the AutoCAD core function can be directly called. Using AutoCAD core database structure, graphics system, geometric modeling core and code, it can establish new commands with the same operation mode as AutoCAD's own commands, which has high running efficiency and stability.

3.2 Platform development based on .NET MVC

MVC is one of three ASP.NET programming modes.

MVC is a model for creating web applications using MVC (Model-View-Controller) design:

The Model represents the core of the application (such as a list of database records).

View displays data (database records).

The Controller processes the input (writes to the database record).
The Model is the part of the application that handles the logic of the application data. Usually the model object is responsible for accessing the data in the database.

The View is the part of the application that processes the display of data. Usually views are created from model data.

The Controller is the part of the application that handles user interaction. Usually the controller is responsible for reading data from the view, controlling user input, and sending data to the model.

MVC tiering helps manage complex applications because you can focus on one aspect at a time. For example, you can focus on view design without relying on business logic. It also makes testing the application easier.

MVC tiering also simplifies grouping development. Different developers can develop views, controller logic, and business logic at the same time. [3]

### 3.3 Power Wiring Diagram Component Modeling

The power wiring diagram (CAD diagram) contains a wealth of power professional information, involving a variety of power components, interfaces, wiring information, etc., because there is no uniform drawing standards and modeling standards, for non-electricity industry and even the power industry staff are reading The diagrams bring difficulties. In this paper, the original wiring diagram of the power wiring diagram is mainly used in the XML language. The model details the information, interface information, connection relationship and other information of the components in the power wiring diagram (CAD diagram).
4. Plug-in development in AutoCAD

In the secondary development of AutoCAD, there are generally two implementations. One is to start AutoCAD through an external application and map it in CAD; the other is to load the plugin in CAD, use the plugin to pop up the panel we need, input the parameters on the panel, and draw in CAD Graphics.[4] The implementation methods and principles of the two methods are basically the same, but the implementation is different. Here we take the more general and representative plug-in development as an example to illustrate.

In the secondary development of AutoCAD, in order to use the .NET API to host encapsulated classes, we need to introduce these DLL files first. Start Visual Studio 2008, create a new class library project, then mouse over the project name, right-click on "Reference", and select "Add Reference". In the "Add Reference" dialog box, select "Browse", then find the above AcDbMgd.dll, AcMgd.dll, AcCui.dll file, click OK to add these components to the project. Right now, we have connected C# to the .NET API, and then we can concentrate on CAD development.

The development process is mainly for the recognition and drawing of the power wiring diagram model described by AutoCAD using XML.

![Figure 4. Graphic model recognition and rendering](image-url)
5. Panel design and drawing in AutoCAD

Once you have added the required plug-in menus and their submenus, you can add commands to each sub-menu to implement different functions. Now if you click the submenu and pop up the panel form, then we first need to add a WindowsForm form in the project, and the command of the form corresponds to the submenu command, you can click the submenu to pop up the corresponding Form. The design of the panel is relatively easy.\[5\]We can design the panel we want according to various controls such as text boxes, labels, buttons, etc. provided in the Visual Studio 2008 integrated environment.

The panel is relatively simple, the upper left is the DataGridView control, which is mainly interactively bound with the data in the database, and displays related information in the database through some commands such as query. The lower left is the TextBox control, which is mainly used to input data to issue commands. The PictureBox control on the upper right is mainly used to display the preview image, and the button control is on the lower right. What functions these button controls implement depends on the internal code. Double-click on the control and we can enter the programming interface of the control and write the corresponding program according to the function to be implemented.

The secondary development of AutoCAD is mainly used for drawing. Below we will introduce how to implement drawing in CAD through the panel. From the above, we know that lines, circles, notes, layers, etc. are all AutoCAD objects, so we can use these classes to draw graphics. The following is an example of a flange. After setting the relevant data on the panel, click the drawing button, the corresponding graphic will be displayed in the CAD drawing interface.

6. AutoCAD and database link

In engineering applications, designers may often need to save the designed graphics. If there are too many graphics, it is not convenient to manage, and it is difficult to find them. It is also not conducive to resource exchange within the company. Therefore, it is necessary to link AutoCAD to the database during secondary development. At the same time, the database is also the link between CAD and the enterprise's ERP management system, which facilitates the overall management of the enterprise and can improve management efficiency.

The database is mainly divided into two types: local database and remote database. The local database is installed on the computer used, which is generally small, and is used for information storage on a personal computer; the remote database is generally on the company's server, which is generally large.\[6\]The remote database and the local database are connected through a local area network or the Internet, and the transfer and download between the local and the server are realized. In this article we use MySQL as the local database and SQL Server as the remote database. Because of the similar operation methods between them, we mainly use the MySQL database as an example.

In order to achieve the connection between AutoCAD and MySQL, a component of MySQL, mySql.data.dll, must be introduced into the project.

7. Conclusion

When you design the required plug-ins, you can compile the program, and finally generate a dll file, you can load the Netload command in CAD, but obviously this method is too cumbersome, we hope that each time you start the CAD Automatically loaded. At this point, you can add the command (COMMAND "NetLoad" "D:/ProgramFiles/AutoCAD 2007/Support/devicemodel.dll") at the end of autocad2007.lsp in the CAD installation file support.

This paper mainly introduces an AutoCAD secondary development method combining C# and .NET API technology, and briefly expounds its principle and the process of implementing various
From the introduction in the article, we can see that this method can easily realize the secondary development of AutoCAD, and it is relatively easy to master. It only needs a certain C# foundation and some basic common sense of CAD to carry out secondary development of CAD, and has a very good practical value.

References
[1] Yuxi Ge, Feng Huang. Secondary development of AutoCAD based on VBA and Visual Lisp [J/OL]. Electromechanical Engineering Technology, 2019(10): 86-88+207.
[2] Qian Xu, Feipeng Liu, Jiguo Li. Application of secondary development of cad based on Lisp language in U-shaped aqueduct section design[J].Computer Knowledge and Technology,2019,15(23):275-276+293.
[3] Xiaoqi Zhang, Tianguo Zhang, Zhen Hu. Script customization and application in the secondary development of Auto CAD [J]. Computer Knowledge and Technology, 2019, 15 (09): 92-94.
[4] Xinzhuang Tang, Hongbo Zhu, Dan He. A method for drawing cross-sections of AutoCAD based on .NET[J]. Surveying and Mapping Technology and Equipment, 2018, 20(02): 78-79+74.
[5] Dandan Cui. Secondary development method and application of CAD software [J]. Electronic Technology and Software Engineering, 2018 (10): 40.
[6] Lu Sun. Secondary development and application of CAD software [J]. Light Industry Technology, 2018, 34 (01): 78-79.