

建筑信息模型BIM丛书
AutoCAD Civil 3D系列

AutoCAD Civil 3D .NET二次开发

王磊 编著



同济大学出版社
TONGJI UNIVERSITY PRESS

建筑信息模型 BIM 丛书
AutoCAD Civil 3D 系列

AutoCAD Civil 3D .NET

二次开发

王 磊 编著



同济大学出版社
TONGJI UNIVERSITY PRESS

内 容 提 要

随着 BIM 技术应用的逐步普及,AutoCAD Civil 3D 软件应用于越来越多的行业,诸如交通运输、土地开发、水利项目、市政工程、公路工程、测绘、地质等,正是因为该软件面向的行业众多,用户遍布全球各个国家和地区,所以软件本身不可能完全满足每个行业的需求,也不可能符合每个国家及地区的标准要求,若要利用其完成本行业的应用,顺利实现模型到施工图的转化并满足国家、行业或企业标准,对该软件进行二次开发是必不可少的。

本书从一个程序设计爱好者的角度出发,针对如何学习 AutoCAD Civil 3D .NET 二次开发,通过一系列的实例,向读者展示了需要学习的基本知识点,为工程设计人员、程序设计人员学习 AutoCAD 及 Civil 3D 二次开发指明道路。本书由浅入深,向读者逐步展示了 AutoCAD 及 Civil 3D .NET 二次开发所需的基本计算机语言知识与各项基本操作、进阶应用与相关知识。本书面向的读者对象为熟练掌握 Civil 3D 软件应用并希望学习二次开发的工程设计人员、专职软件设计人员以及从事 BIM 应用研究的在职硕士研究生。

图书在版编目(CIP)数据

AutoCAD Civil 3D .NET 二次开发/王磊编著.

—上海:同济大学出版社,2017.11

ISBN 978-7-5608-7497-5

I. ①A… II. ①王… III. ①AutoCAD 软件
IV. ①TP391.72

中国版本图书馆 CIP 数据核字(2017)第 288372 号

AutoCAD Civil 3D .NET 二次开发

王磊 编著

责任编辑 马继兰 责任校对 徐逢乔 封面设计 安柯 陈益平

出版发行 同济大学出版社 www.tongjipress.com.cn
(地址:上海市四平路 1239 号 邮编:200092 电话:021-65985622)

经 销 全国各地新华书店
排 版 南京月叶图文制作有限公司
印 刷 常熟市华顺印刷有限公司
开 本 787 mm×1092 mm 1/16
印 张 18
字 数 449 000
版 次 2018 年 1 月第 1 版 2018 年 1 月第 1 次印刷
书 号 ISBN 978-7-5608-7497-5
定 价 78.00 元

本书若有印装质量问题,请向本社发行部调换 版权所有 侵权必究

前 言

本书从 AutoCAD Civil 3D .NET 二次开发基础知识入手,详细介绍了 AutoCAD Civil 3D 及 AutoCAD 二次开发所需了解、掌握的理论知识。以 .NET 为主,通过详细的代码,向读者展示了 AutoCAD Civil 3D 的基本技术与操作过程;同时也简单介绍了 COM API 及 ObjectARX 技术在 AutoCAD Civil 3D 二次开发中的应用。

本书面向的读者是熟练掌握 AutoCAD Civil 3D 软件应用并开始接触 AutoCAD Civil 3D .NET 二次开发的初学者,特别是没有计算机语言基础从零开始学习 Civil 3D 二次开发的初学者,也可以作为 AutoCAD 二次开发初学者的学习用书。

作为一名 AutoCAD 爱好者,在大学的最后一个学期,大部分时间是在计算机房度过的——学习 AutoCAD 的使用,当时的版本是 DOS 系统下的 AutoCAD R12 英文版。因工作性质的关系,工作后,使用 AutoCAD 的机会并不多,但偶然的的机会,接触到一些 Lisp 语言的知识,从此知道了 AutoCAD 二次开发的概念,随后的两年内我编写了一些并不完善的小程序,主要用于提高工作效率。因为对二次开发的认识不够准确,当时放弃了二次开发的学习,直到开始使用 AutoCAD Civil 3D 后,发现二次开发是充分发挥软件功能必不可少的手段,我又重新开始学习 AutoCAD 及 Civil 3D 的二次开发知识。

在通读本书之前,读者有必要对本书的定位有一个准确的认识:要知道这是一本关于什么的书,要知道这不是一本什么书。

首先回答第一个问题:

这是一本 AutoCAD 爱好者、AutoCAD Civil 3D 用户所编写的书,我没有深厚的计算机知识背景,也不是计算机软件相关的从业人员,仅仅是一名从事了 15 年房建工程施工的工程师而已。

这是一本由二次开发自学者所写的书,书中的内容仅是我自学过程中所学到的各种知识的汇总,并没有把所有 AutoCAD 及 Civil 3D 二次开发知识全部罗列出来。

这是一本我根据自己自学的心路历程总结所写出的书,我想把自己在自学过程中遇到的问题及如何解决这些问题的经验与有着同样自学需求的朋友分享。

接下来回答第二个问题:

这不是一本介绍计算机语言的书籍,二次开发离不开计算机语言,如果读者要学习相关语言,您不得不参考其他书籍。

这不是一本介绍软件应用的书籍,二次开发的一个基本条件是要求熟练掌握软件的应用,即使熟练掌握了多种计算机语言,如果不熟悉软件的应用,要写出高效率的代码也并不

是一件易事。

这不是一本能解决所有 AutoCAD 与 Civil 3D 二次开发相关问题的书,更确切地讲,书中不少内容只是提出问题,并给出了解决相关问题的思路,但并未给出具体的解决方法,目的是让读者掌握解决问题的方法,而不是学会解决某一个具体问题。

我写这本书的目的是想让读者在读完这本书之后,能够掌握 AutoCAD 及 Civil 3D 二次开发的学习方法,知道要学什么,知道要查找什么,知道如何去查找。

书中的一些示例并不完善,尤其是“类的继承”的示例,我只是想向读者介绍“类的继承”的概念,至于示例是否合适、是否完善,需要打一个很大的问号。

在阅读本书过程中,建议读者边阅读边键入代码进行测试,不要只读不做,看与练是完全不同的,别人轻松完成的事情,到自己手上可能就难以实现,所以一定要亲自动手练习。在键入代码的过程中,还可以发现更多问题并解决问题。

在学习的初期,不要带着很强的目的性去学习,此时了解的知识、方法有限,容易误入歧途,甚至是死胡同。随着掌握的知识的增多,视野会越来越宽,也就能找到更多解决问题的方法,对自己手中的工具有了全面了解后,再用它来完成相应的工作会变得较为轻松。

书中第 4 章之后的示例代码,读者可发送邮件索取,我的 E-mail 地址:465340553@qq.com。

鉴于水平有限,书中难免有错误之处,欢迎读者予以指正。

王 磊

2017 年 9 月于西安

目 录

前言

第 1 部分 基础知识	1
第 1 章 Civil 3D 二次开发简介	2
1.1 了解 AutoCAD Civil 3D 及其 APIs	2
1.2 Civil 3D 二次开发的意义	3
1.3 Civil 3D 二次开发所需的条件	3
1.4 从哪儿开始	4
第 2 章 开发环境的配置	7
2.1 编译环境	7
2.2 .NET 项目设置	8
2.2.1 手动创建解决方案及项目	8
2.2.2 利用向导创建解决方案和项目	10
2.3 混合项目设置	11
2.4 定义命令	12
2.5 编译	13
2.6 加载及运行	13
2.6.1 手动加载程序及运行命令	13
2.6.2 通过工具空间加载并运行	14
2.6.3 通过注册表按需加载	15
2.7 帮助文档	17
2.7.1 帮助文档	18
2.7.2 如何查看帮助文档	18
第 3 章 程序设计基础	20
3.1 数据类型和操作符	20
3.1.1 常量与变量	20
3.1.2 简单数据类型	21
3.1.3 算数运算符	22
3.1.4 关系与逻辑运算符	23

3.1.5 值类型与引用类型	24
3.2 方法	25
3.2.1 方法签名	25
3.2.2 方法访问	26
3.2.3 方法参数	26
3.2.4 返回值	27
3.2.5 通过引用传递参数	27
3.2.6 方法的递归调用	29
3.3 选择语句	30
3.3.1 if 语句	30
3.3.2 switch 语句	32
3.4 循环语句	32
3.4.1 for 循环	32
3.4.2 foreach 循环	33
3.4.3 while 和 do while 循环	34
3.5 字符串的操作	35
3.5.1 字符串的比较	35
3.5.2 字符串的拆分	35
3.5.3 获取指定字符串的位置	36
3.5.4 字符串的提取	37
3.5.5 字符替换	37
3.5.6 大小写转换	38
3.5.7 判断是否存在指定字符	38
3.5.8 插入	38
3.5.9 删除	38
3.5.10 清空空格及指定字符	38
3.6 集合	39
3.6.1 集合中元素数量	40
3.6.2 遍历集合	40
3.7 类型转换	42
3.7.1 隐式转换	42
3.7.2 显式转换	42
3.7.3 字符串与数字	43
3.8 命名空间	45
3.8.1 命名空间的用途	46
3.8.2 导入命名空间	47
3.8.3 AutoCAD 及 Civil 3D 命名空间简介	49

第 2 部分 基本操作	51
第 4 章 访问数据库中的对象	52
4.1 了解 AutoCAD 对象层次结构	53
4.1.1 应用程序	53
4.1.2 文档	54
4.1.3 数据库	56
4.1.4 符号表	56
4.1.5 块表	57
4.1.6 块表记录	58
4.1.7 字典	63
4.2 了解 Civil 3D 对象层次结构	66
4.2.1 应用程序	67
4.2.2 文档	67
4.2.3 集合	67
4.3 由 ObjectId 获取 Object	68
4.4 事务(Transaction)	70
4.5 捕捉异常	71
4.6 人机交互	72
4.6.1 命令行输出	73
4.6.2 输入整数	73
4.6.3 输入实数及关键字	74
4.6.4 拾取点	75
4.6.5 拾取角度	76
4.6.6 拾取单个实体	77
4.6.7 拾取选择集	79
第 5 章 创建对象	82
5.1 创建 AutoCAD 对象	82
5.1.1 创建直线	83
5.1.2 创建图层	86
5.1.3 创建布局及视口	88
5.2 创建 Civil 3D 对象	95
5.2.1 创建几何空间点	96
5.2.2 创建曲面	100
5.2.3 创建采样线	104
5.2.4 小结	109
5.3 创建 Civil 3D 样式	110

5.3.1	创建点样式	110
5.3.2	创建曲面样式	113
5.3.3	创建标签样式	115
5.3.4	小结	117
第 6 章	编辑对象	120
6.1	编辑 AutoCAD 对象	120
6.1.1	修改对象的属性	121
6.1.2	复制、删除、分解对象	124
6.1.3	平移、旋转、缩放对象	126
6.1.4	多段线修改	135
6.2	编辑 Civil 3D 对象	141
6.2.1	修改曲面顶点	142
6.2.2	修改纵断面	144
6.2.3	拆分道路区域	147
6.3	编辑 Civil 3D 设定	150
6.3.1	访问各种设定	150
6.3.2	编辑要素设定	151
6.3.3	编辑命令设定	152
6.3.4	使用属性字段	153
第 3 部分	进阶应用	155
第 7 章	对象信息的提取	156
7.1	创建标签	156
7.2	数据插入表格	161
7.2.1	获取表格样式	162
7.2.2	创建表头	164
7.2.3	填充数据	167
7.2.4	插入表格	170
7.2.5	获取桩号文本	172
7.3	输出数据到外部文件	173
第 8 章	用户界面的应用	177
8.1	自定义对话框	177
8.1.1	界面设计	178
8.1.2	代码实现	184
8.1.3	调用对话框	187
8.2	功能区	190

8.2.1 了解功能区	190
8.2.2 将功能区面板添加至已有选项卡	192
8.2.3 将功能区面板添加至上下文选项卡	197
8.3 面板	200
8.3.1 了解 Palette	200
8.3.2 创建简单的 WPF 用户控件	201
8.3.3 创建面板	202
8.4 上下文菜单	204
第 9 章 程序部署	206
9.1 自动加载简介	206
9.2 BUNDLE 软件包	207
9.2.1 文件夹结构	207
9.2.2 XML 文件基础知识	210
9.3 MSI 安装程序	211
9.3.1 组织安装程序	212
9.3.2 指定应用程序数据	213
9.3.3 配置目标系统	214
9.3.4 定制安装程序界面	215
9.3.5 定义安装需求及行为	215
9.3.6 准备发布	217
第 4 部分 相关主题	221
第 10 章 COM 互操作的应用	222
10.1 了解 COM API	222
10.2 实现 .NET 与 COM 互操作	223
10.2.1 根对象及 COM 中的基本概念	224
10.2.2 访问 Civil 3D 对象	227
10.3 COM 对象与 .NET 对象的转换	232
10.3.1 .NET 对象转换为 COM 对象	232
10.3.2 COM 对象转换为 .NET 对象	233
第 11 章 C++ API 的应用	235
11.1 了解 CustomDraw	235
11.2 自定义纵断面竖轴	237
11.3 绘制挡墙分隔缝	244

第 12 章 LINQ 的应用	249
12.1 了解 LINQ	249
12.1.1 匿名类型	249
12.1.2 扩展方法	251
12.1.3 Lambda 表达式	252
12.2 LINQ 功能初体验	254
12.2.1 排序	254
12.2.2 筛选	255
12.2.3 数据投影	256
12.2.4 分组	256
12.3 针对对象查询	257
第 13 章 创建部件	259
13.1 部件程序的基本结构	259
13.1.1 模板类 SATemplate	259
13.1.2 CorridorState 对象	261
13.1.3 支持文件	261
13.2 创建自定义部件	262
13.3 创建 .atc 文件	269
13.4 通过 .pkt 文件加载部件	272
附录 A 视频部分说明及下载地址	273
索引	274
参考文献	276

第1部分

基础知识

- 第1章 Civil 3D 二次开发简介
- 第2章 开发环境的配置
- 第3章 程序设计基础

——世上无难事，只要肯登攀

本章重点

- ◇ 为什么需要二次开发
- ◇ 学习二次开发的条件是什么

1.1 了解 AutoCAD Civil 3D 及其 APIs

随着建筑信息模型 BIM 应用技术的普及，越来越多的工程技术人员逐步掌握多种 BIM 软件的应用。AutoCAD Civil 3D 作为众多建筑信息模型(BIM)软件中的一员，主要面向土木工程设计及图形文档编制。AutoCAD Civil 3D 在大家熟悉的 AutoCAD 环境中，帮助从事交通运输、土地开发和水利项目等土木工程专业人员更轻松、更高效地探索设计方案，分析项目性能，并提供相互一致、更高质量的图形文档。

AutoCAD Civil 3D 就是根据专业需要进行了专门定制的 AutoCAD，是业界认可的土木工程软件包，可以加快设计理念的实现。它的三维动态工程模型有助于快速完成道路工程、场地、雨水/污水排放系统以及场地规划设计。所有曲面、横断面、纵断面、标注等均以动态方式链接，可更快、更轻松地评估多种设计方案，做出更明智的决策并生成最新的图纸。

AutoCAD Civil 3D 提供了 3 种 API，分别是 .NET API，COM API 和 CustomXXX API(使用 C++ 语言)。

.NET API——允许以任何 .NET 语言(C#、VB .NET 或 C++/CLI)编写 AutoCAD Civil 3D 扩展程序。一般来说，AutoCAD Civil 3D .NET API 的执行速度明显快于 COM API。

COM API——允许从托管(.NET)或非托管代码(C++、Lisp)访问 COM API 从而创建客户端程序，当然也可以在 VBA 程序中应用(VBA 编辑器已不随 Civil 3D 安装包发布，需要单独下载安装)。

CustomXXX API——CustomDraw， CustomEvent 和 CustomUI，以 AutoCAD ObjectARX API 形式出现，允许开发者定制对象的显示方式、创建自定义事件和界面(其本质是使用 C++ 访问 COM API，至于能否使用 C++/CLI 访问 .NET API 创建程序，作者未亲自验证)。

1.2 Civil 3D 二次开发的意义

AutoCAD Civil 3D 在全球多个国家、多个行业被采用,每个国家有每个国家的标准、不同的行业又有不同的标准, Autodesk 作为软件供应商,不可能满足每个国家、每个行业的标准和要求,因此 Autodesk 开放 API,给用户自行定制提供可能。每位用户可以根据自身的需求,对软件进行扩展。

在实际工作中,使用软件现有的基本功能,经常遇到一些简单重复的操作,例如绘制一段踏步的纵断面,需要重复创建或添加变坡点,计算机恰恰擅长处理这些简单重复的操作,因此需要通过简单的二次开发将这些简单重复的操作交由计算机完成,从而提高设计人员的工作效率。

有一些问题,人工操作难以实现,例如这样一个问题:已知弧长 330,弦长 300(图 1-1),绘制这条弧。看起来很简单,您可能已经拿出了纸和笔来列方程求解。这是超越方程,解不出来,怎么办?交给计算机,用二分法查找近似解!

在 BIM 技术日新月异的今天,如果不掌握一些二次开发技术,看似强大的软件工具,在完成实际的工作时,总会差那么一点点而不能尽善尽美完成任务;只有掌握了二次开发技术,才能使手中的软件如虎添翼,发挥更大的作用。

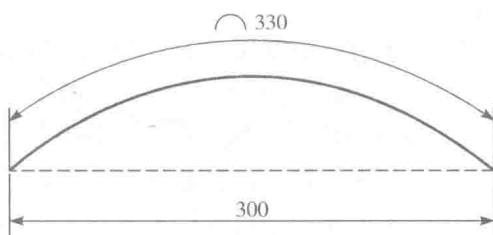


图 1-1 已知弧长弦长画图

1.3 Civil 3D 二次开发所需的条件

学习 Civil 3D 二次开发需要哪些条件呢?

首先要熟悉 Civil 3D 的应用。只有熟悉 Civil 3D 的应用,了解 Civil 3D 对象之间的关系,用代码操纵 Civil 3D 对象才成为可能。另外,一旦熟悉 Civil 3D 应用后,还可避免做无用功,不会出现自己用代码实现了某个功能,最后发现 Civil 3D 自身已有这功能,并且该功能更强大更完善的情况。

其次,至少要掌握一门计算机语言。这是必要条件,如果不掌握计算机语言,编写代码也就无从谈起。至于掌握哪种语言,可以根据读者之前所学的计算机语言进行选择。如果您是零基础学起,建议您学习 C# 语言,个人感觉,在 .NET 语言(C#、VB.NET 或 C++/CLD)中,C# 语言是易学易用的,这也是为什么本书中示例代码以 C# 语言实现的原因。

再次,熟悉 AutoCAD 二次开发也是一个不可缺少的条件。Civil 3D 作为 AutoCAD 的一个超集,对其二次开发,不可避免要涉及 AutoCAD 二次开发的知识。这也是本书中有近一半的内容为 AutoCAD 二次开发相关内容的原因。

除此之外,学习者还应有坚定的信念和毅力。笔者在学习 AutoCAD 二次开发之初,没有清楚地认识到二次开发的意义,认为二次开发没有前途(“钱图”)、没有意义,因此放弃,中断了十年的时间,直到后来工作中需要使用 Civil 3D 完成项目,才发现二次开发的必要性,于是从零开始学习 C# 语言,学习 AutoCAD 及 Civil 3D .NET 二次开发知识,直到现

在与您分享自己学习二次开发的经验并写下这本书。

1.4 从哪儿开始

对于多数的工程设计人员,不完全具备本书 1.3 节所述的相关条件。对于 Civil 3D 的应用可能比较熟悉,但对于计算机语言,只限于上学时所学的一点基础知识(还可能还给老师了),步入工作岗位后与计算机语言基本无缘;对于 C#、VB.NET 可能根本就没有接触过。

只要有一颗坚定的心,坚持学习,就一定能进入 Civil 3D 二次开发的“大门”,写出自己的程序,从而节省时间,调高工作效率。

如果您是一位熟悉 Civil 3D 软件应用的工程设计人员,那么您就要从学习计算机语言开始。如果您是一位精通计算机语言的专业程序设计人员,那么您就要先了解 Civil 3D 的应用。本书面向的读者是第一类人群,如果您是第二类人员,这本书可能不适合您。

学习计算机语言,从哪里学起呢?这是个问题,并且是个不好回答的问题。既然这样,我们就从阅读代码开始,虽然简单直接,但我相信会很快见到成效。

在利用 Civil 3D 创建对象过程中,经常遇到类似如下的需求:要把普通的 AutoCAD 对象,诸如点、文本、块等,转化成 Civil 3D 的几何空间点(CogoPoint)。对于点,Civil 3D 有内部命令,可以直接实现转换,但对于文本、块等对象,没有内部命令,虽然可以利用数据提取等方法提取数据,创建外部文件,然后再创建几何空间点,但总会有些情况需要通过二次开发才能实现。假设需要将 AutoCAD 的块对象批量转换成几何空间点,需要写多少行代码才能实现呢?

阅读以下代码:

```
01 public void BlockReferenceToCogoPoint()  
02 {  
03     //获取 AutoCAD 的 Document 对象  
04     Document doc = Application.DocumentManager.MdiActiveDocument;  
05     //获取 Civil 3D 的 Document 对象  
06     CivilDocument civilDoc = CivilApplication.ActiveDocument;  
07     //获取几何空间点集  
08     CogoPointCollection cogoPts = civilDoc.CogoPoints;  
09     //开始事务  
10     using (Transaction tr = doc.TransactionManager.StartTransaction())  
11     {  
12         //获取块表  
13         BlockTable blockTable = tr.GetObject(doc.Database, BlockTableId, OpenMode.ForRead)  
14             as BlockTable;  
15         //获取模型空间的 ObjectId  
16         ObjectId modelSpaceId = blockTable[BlockTableRecord.ModelSpace];  
17         //获取块表记录(模型空间)  
18         BlockTableRecord modelSpace = modelSpaceId.GetObject(OpenMode.ForRead)  
19             as BlockTableRecord;
```

```

20 //循环处理模型空间的每一个对象
21 foreach (ObjectId id in modelSpace)
22 {
23     //判断对象是否为 BlockReference
24     if (!id.ObjectClass.IsDerivedFrom(RXObject.GetClass(typeof(BlockReference))))
25         continue;
26     //获取 BlockReference 对象
27     BlockReference br = id.GetObject(OpenMode.ForRead) as BlockReference;
28     //向几何空间点集中添加点
29     cogoPts.Add(br.Position, br.Layer, true);
30 }
31 //提交事务
32 tr.Commit();
33 }
34 }

```

这是一段用 C# 语言完成的代码,代码实现的功能是根据模型空间中的块创建几何空间点,几何空间点的位置采用块的插入点位置,几何空间点的描述采用块的图层名称。

这段代码不到 20 行(去除注释),不知道您想到了多少问题?在这里我想让初学者了解的问题可不止 20 个。下面就来看一下我想到了哪些问题:

1. 关于程序设计的基本问题

- (1) 编写这些代码需要什么环境?
- (2) 如何编译?
- (3) 编译后如何加载?
- (4) 加载后如何运行?
- (5) 什么是方法?
- (6) 如何定义方法?
- (7) 如何实现方法?
- (8) 什么是变量?
- (9) 如何声明变量?
- (10) 变量命名规则有没有约定?
- (11) 怎么为变量赋值?
- (12) 数据类型都有哪些?
- (13) 不同数据类型能否实现转换?
- (14) 如何实现循环操作?

2. AutoCAD 二次开发问题

- (1) 程序、文档、数据库之间的关系是什么?
- (2) 如何定义一个在 AutoCAD 中运行的命令?
- (3) AutoCAD 数据库结构是怎样的?
- (4) 什么是块表(BlockTable)?
- (5) 什么是块表记录(BlockTableRecord)?

- (6) 什么是块参照(BlockReference)?
- (7) 从 AutoCAD 数据库中,如何获取指定对象?
- (8) 从 ObjectId 如何获取相应的 Object?
- (9) 事务(Transaction)是什么?
- (10) 如果事务未提交(Commit)会有什么后果?

3. Civil 3D 二次开发问题

- (1) Civil 3D 数据库结构是怎样的?
- (2) 从 Civil 3D 数据库中,如何获取指定对象?
- (3) 如何创建 Civil 3D 对象?
- (4) 如何从集合中获取某一对象?

问题已经提出,带着这些问题阅读本书的前两部分吧。在阅读过程中,若发现本书中有讲解不清楚的地方,您就需要学会网络搜索或查阅其他书籍,本书中个别地方提供了关键字,方便上网搜索。