AutoCAD 应用程序开发系列

AutoCAD ObjectARX 程序开发技术

国防工业出版社
http://www.ndip.cn

李长勋 主编

AutoCAD ObjectARX

[illegible]

AutoCAD 应用程序开发系列

AutoCAD ObjectARX 程序开发技术

李长勋　主编

国防工业出版社
·北京·

图书在版编目(CIP)数据

AutoCAD ObjectARX 程序开发技术/李长勋主编.
北京:国防工业出版社,2005.1
(AutoCAD 应用程序开发系列)
ISBN 7-118-03565-3

Ⅰ.A... Ⅱ.李... Ⅲ.计算机辅助设计-应用软件,
AutoCAD ObjectARX-程序设计 Ⅳ.TP391.72

中国版本图书馆 CIP 数据核字(2004)第 084780 号

国防工业出版社出版发行
(北京市海淀区紫竹院南路 23 号)
(邮政编码 100044)
腾飞胶印厂印刷
新华书店经售
*
开本 787×1092 1/16 印张 24½ 565 千字
2005 年 1 月第 1 版 2005 年 1 月北京第 1 次印刷
印数:1—3000 册 定价:33.00 元

(本书如有印装错误,我社负责调换)

国防书店:68428422 发行邮购:68414474
发行传真:68411535 发行业务:68472764

前　言

AutoCAD 的第一代开发工具 Auto LISP 是 1986 年随 AutoCADv2.18 提供的二次开发工具。它是一种人工智能语言，是嵌入 AutoCAD 内部的 COMMON LISP 的一个子集。第二代开发工具 ADS（AutoCAD Development System）是 AutoCAD R11 开始支持的一种基于 C 语言的灵活的开发环境。ADS 可直接利用用户熟悉的 C 编译器，将应用程序编译成可执行文件后在 AutoCAD 环境下运行，从而既利用了 AutoCAD 环境的强大功能，又利用了 C 语言的结构化编程、运行效率高的优势。Visual AutoLISP、ObjectARX 以及 VBA 是第三代开发工具。

ObjectARX 开发环境提供了一个面向对象的 C++应用程序接口，开发人员可以利用接口使用、修改和扩展 AutoCAD。用 VC++语言编写的程序经过编译、链接，最后生成 ObjectARX 应用程序。ObjectARX 应用程序是一个分享 AutoCAD 的地址空间、并可对 AutoCAD 直接调用的动态链接库。采用可扩展性思想设计出的 ObjectARX 库，包含了用于定义新类的宏，提供了对已存在库中的类进行实时添加新功能的能力。此外，为了使开发者能根据自己的需要和能力选择开发工具，ObjectARX 库提供了与 ADS、AutoLISP 应用程序相链接的接口。ObjectARX 库中有各种各样的系列工具，开发人员可以通过 AutoCAD 的开放结构，直接对 AutoCAD 的数据结构、图形系统和内部命令进行操作。

ObjectARX 编程环境提供了一个面向对象的 C++应用程序开发界面，使开发者能够使用、定做和扩展 AutoCAD。另外，这些库被设计的可以与 Visual LISP 及其他应用程序开发界面协同工作，以便开发者根据自己的需要和经验来选择最适合的开发工具。

在本书的编写过程中，突出应用性强的特点。本书不是单纯讲述诸命令，而是结合实例，介绍常用命令的具体应用和绘制方法，使用了大量的插图、框图，图文并茂使读者更容易阅读，更具操作性，更贴近于计算机屏幕上的操作界面，因此读者也更容易接受所讲述的内容。

本书内容新颖、全面，语言通俗易懂、易于读者掌握，从入门篇、基础篇和高级篇三部分对 ObjectARX 入门知识、ObjectARX 基础知识和 ObjectARX 高级应用等知识进行详细概述。本书编者多年从事 AutoCAD 应用与开发工作，具有丰富的使用经验，书中提供的提示与应用技巧可帮助读者在学习中少走弯路；功能讲述中给出的实例更可帮助读者进一步加深对 AutoCAD 2002 所学知识的理解。

由于编者水平有限，加之时间仓促，缺点和错误在所难免，恳请读者批评指正。

目　录

入门篇

第1章 ObjectARX简介

AutoCAD是美国Autodesk公司研制的软件，是目前微型计算机上应用最广泛的通用计算机辅助绘图和设计软件，经过20多年的不断发展和完善，现已推出第16版——AutoCAD 2002。1996年8月，Autodesk公司不失时机地推出了AutoCAD Runtime eXtension实时扩展(简称ObjectARX)，这一全新的AutoCAD开发环境，利用了VC++的强大功能，使CAD产品的开发变得更容易，功能更强大。

ObjectARX开发环境提供了一个面向对象的C++应用程序接口，开发人员可以利用接口使用、修改和扩展AutoCAD。用VC++语言编写的程序经过编译、链接，最后生成ObjectARX应用程序。ObjectARX应用程序是一个分享AutoCAD的地址空间、并可对AutoCAD直接调用的动态链接库。采用可扩展性思想设计出的ObjectARX库，库中包含了用于定义新类的宏，提供了对已存在库中的类进行实时添加新功能的能力。此外，为了使开发者能根据自己的需要和能力选择开发工具，ObjectARX库提供了与ADS、AutoLISP应用程序相链接的接口。ObjectARX库中有各种各样的系列工具，开发人员可以通过AutoCAD的开放结构，直接对AutoCAD的数据结构、图形系统和内部命令进行操作。

ObjectARX环境可从Internet互联网络中Autodesk公司的网址上下载，大约20MB，主要内容包括ObjectARX库、头文件及帮助文件。开发人员必须在AutoCAD R13或R14以上版本才能使用ObjectARX开发环境。

本章还将同时对其他的AutoCAD二次开发工具做概述，以使读者更好地学习ObjectARX开发工具。

1.1 AutoCAD二次开发工具综述

AutoCAD的强大生命力在于它的通用性、多种工业标准和开放的体系结构。其通用性使得它在机械、电子、航空、船舶、建筑、服装等领域得到了极为广泛的应用。但是，不同的行业标准使得各领域在使用AutoCAD的过程中均需根据自身特点进行定制或开发。Autodesk公司为满足广大用户的需求，自AutoCAD v2.18版至AutoCAD 2002的短短十几年间，就相继推出了三代二次开发工具。可以说，AutoCAD的通用性为其二次开发提供了必要条件，而AutoCAD开放的体系结构则使其二次开发成为可能。

AutoCAD的第一代开发工具当属于AutoLISP语言，第二代开发工具就是基于C语言的开发工具ADS。在随后的几年里又有几种语言被用来对AutoCAD进行二次开发。它们分别是Visual AutoLISP、ObjectARX和VBA。下面来看一看它们各自的特

点。

1.1.1 AutoLISP 语言概述

第一代开发工具 AutoLISP 是 1986 年随 AutoCAD v2.18 提供的二次开发工具。它是一种人工智能语言，是嵌入 AutoCAD 内部的 COMMON LISP 的一个子集。在 AutoCAD 的二次开发工具中，它是惟一的一种解释型语言。使用 AutoLISP 可直接调用几乎所有的 AutoCAD 命令。

AutoLISP 语言最典型的应用之一是实现参数化绘图程序设计，包括尺寸驱动程序和鼠标拖动程序等。另一个典型应用就是驱动 AutoCAD 提供 PDB 模块构成 DCL(Dialog Control Language)文件，创建自己的对话框。

AutoLISP 具有以下优点：

(1) 语言规则十分简单，易学易用；

(2) 直接针对 AutoCAD，易于交互；

(3) 解释执行，立竿见影。

AutoLISP 也具有以下缺点：

(1) 功能单一，综合处理能力差；

(2) 解释执行，程序运行速度慢；

(3) 缺乏很好的保护机制，源程序保密性差；

(4) LISP 用表来描述一切，并不能很好地反映现实世界和过程，跟人的思维方式也不一致；

(5)不能直接访问硬件设备、进行二进制文件的读写。

AutoLISP 的这些特点，使其仅适合于有能力的终端用户完成一些自己的开发任务。

1.1.2 ADS 语言概述

第二代开发工具 ADS(AutoCAD Development System)是 AutoCAD R11 开始支持的一种基于 C 语言的灵活的开发环境。ADS 可直接利用用户熟悉的 C 编译器，将应用程序编译成可执行文件后在 AutoCAD 环境下运行，从而既利用了 AutoCAD 环境的强大功能，又利用了 C 语言的结构化编程、运行效率高的优势。与 AutoLISP 相比，ADS 优越之处在于：

(1) 具备错综复杂的大规模处理能力；

(2) 编译成机器代码后执行速度快；

(3) 编译时可以检查出程序设计语言的逻辑错误；

(4) 程序源代码的可读性好于 AutoLISP。

ADS 不便之处在于：

(1) C 语言比 LISP 语言难于掌握和熟练应用；

(2) ADS 程序的隐藏错误往往导致 AutoCAD，乃至操作系统的崩溃；

(3) 需要编译才能运行，不易见到代码的效果；

(4) 同样功能，ADS 程序源代码比 AutoLISP 代码长很多。

1.1.3 Visual AutoLISP 、ObjectARX 及 VBA

第三代开发工具包括 Visual AutoLISP、ObjectARX 以及 VBA，特性如下。

1. Visual AutoLISP

Visual AutoLISP 是 AutoLISP 的换代产品。它与 AutoLISP 完全兼容，并提供它所有的功能，是新一代的 AutoCAD LISP 语言。Visual AutoLISP 对语言进行了扩展，可以通过 Microsoft ActiveX Automation 接口与对象交互。同时，通过实现反应器函数，还扩展了 AutoLISP 响应事件的能力。作为开发工具，Visual AutoLISP 提供了一个完整的集成开发环境(IDE)，包括编译器、调试器和其他工具，可以提高二次开发的效率。另外，Visual AutoLISP 还提供了用于发布独立的应用程序的工具。

2. ObjectARX(AutoCAD Runtime eXtension)

本章开始已经对 ObjectARX 做了简要介绍。ObjectARX 是 AutoCAD R13 之后推出的一个以 C++语言为基础的面向对象的开发环境和应用程序接口。ObjectARX 程序本质上为 Windows 动态链接库(DLL)程序，它与 AutoCAD 共享地址空间，可直接调用 AutoCAD 的核心函数，还可直接访问 AutoCAD 数据库的核心数据结构和代码，以便能够在运行期间扩展 AutoCAD 固有的类及其功能，创建能够全面享受 AutoCAD 固有命令特权的新命令。ObjectARX 程序与 AutoCAD、Windows 之间均采用 Windows 消息传递机制直接通信。

3. VBA(Visual Basic for Application)

VBA 的语言基础是微软公司开发的 Visual Basic。Visual Basic 是微软公司以最终用户为目标生产的编程工具，在美国，各种版本的 Visual Basic 软件零售价格在（100～500）美元之间。每一个用户都可以在自己的 PC 机上安装 Visual Basic 并且在 Visual Basic 环境下编程。一旦编程完毕，就可以作为独立的 Windows 应用程序执行自己的 Visual Basic 程序。Visual Basic 程序可以是自给自足的，也可以与其他 Windows 应用程序通信。而大家所关心的是 Visual Basic 程序通过 ActiveX Automation 与 AutoCAD 通信。

在 AutoCAD 中，Visual Basic 程序可以通过 AutoCAD 的 ActiveX Automation API 操纵 AutoCAD。ActiveX Automation 是一套微软标准，以前称为 OLE Automation 技术。该标准允许通过外显的对象由一个 Windows 应用程序控制另一个 Windows 应用程序，这也是面向对象编程技术的精髓所在。AutoCAD 从 R14 开始增加了作为 ActiveX Automation 服务器应用程序的功能，使得许多面向对象编译语言和应用程序可以通过 ActiveX 与 AutoCAD 进行通信，并操纵 AutoCAD 的许多功能。

开放的体系结构一直是 AutoCAD 软件极为重要的特性。在 AutoCAD 过去的版本中，提供了一系列软件客户化工具。而现在，通过 ActiveX Automation 系统，更提供了 R14 与其他应用程序集成的客户化工具。

AutoCAD 中的 ActiveX Automation 接口与微软公司出品的 Excel 和 Access 中的接口十分相似，因此十分容易学习和使用。对 R14 的用户来说，因为 Visual Basic 的那些标准特性(例如数学和字符操作)对于所有应用程序都是相同的，因此他们只需要学习与 AutoCAD 相关的部分功能。

在 R14 中实现的 ActiveX Automation 不但是即将交付使用的软件包的一部分，可以立即与其他编程工具，如 Viausal Basic 一起使用，而且将成为未来在 AutoCAD 中实现

的 VBA 的基础。

1.1.4 ObjectARX 与其他开发工具的比较

1. ObjectARX 与 AutoLISP、ADS、ADSRX 的比较

AutoLISP、ADS、ObjectARX 都是 AutoCAD 提供的内嵌式编程语言。AutoLISP 和 ADS 都是通过内部进程通信(IPC)来和 AutoCAD 通信的，它们与 AutoCAD 是相互分离的过程，而 ObjectARX 以 DLL 形式与 AutoCAD 共享地址空间。因此，与前两者相比，其速度更快、运行更稳定、更简单。由于是在 Windows 及 VC++编程环境里运行，所以，对开发者的编程能力要求较高。另外 ObjectARX 应用程序以 C++为基本开发语言，具有面向对象编程方式的数据可封装性、可继承性及多态性的特点，用其开发的 CAD 软件具有模块性好、独立性强、连接简单、使用方便、内部功能高效实现以及代码可重用性强等特点，并且支持 MFC 基本类库，能简捷高效地实现许多复杂功能。

AutoLISP 是一种解释性的语言，它提供了一个简单的扩充 AutoCAD 命令的机制。ADS 是用 C 语言开发编译执行的。然而，对于 AutoCAD 来说，ADS 程序和 AutoCAD 程序没有什么区别。一个 ADS 程序实际上是由一组外部函数组成，它们由 AutoLISP 解释器来加载调用，ADS 程序本身并不能直接和 AutoCAD 进行通信。

ObjectARX 程序在很多方面都和 ADS 程序不同。把它们的区别总结如下：

(1) ObjectARX 程序是一个动态链接库(DLL)，它直接和 AutoCAD 进行通信。ADS 程序是一个可执行文件，它需要通过 AutoLISP 来和 AutoCAD 进行通信。

(2) AutoCAD 是不可重入的，因此 ADS 程序也是不可重入的。而在 ObjectARX 中，每一个命令都有独立的入口。

(3) ObjectARX 程序速度快，但更“脆弱”，ObjectARX 程序和 AutoCAD 共享进程空间，ObjectARX 程序本身是 AutoCAD 的一部分，ObjectARX 程序的崩溃通常会导致 AutoCAD 系统的崩溃。而 AutoLISP 和 ADS 都是通过函数来间接访问 AutoCAD。ADS 程序速度慢，但更“绝缘”，ADS 程序崩溃并不一定导致 AutoCAD 系统崩溃。

(4) ADS 程序类似宏(macro)，ADS 中的函数(如 ads_command)以及和 AutoLISP 的通信使得 ADS 程序的工作类似于自动作用的宏。相比之下，ObjectARX 程序更基本，主程序(AutoCAD)可以调用每一个 ObjectARX 程序注册的命令。

(5) ObjectARX 程序具有 ADS 程序和 AutoLISP 程序所不具备的访问和控制 AutoCAD 的能力。

(6) ObjectARX 提供了面向对象编程的技术。ObjectARX 充分支持 C++，充分支持面向对象编程的技术，而 ADS 仍然只能使用传统的 C 语言编程。

这里顺便提一下 ADSRX。AutoCAD R14 的开发环境 ADSRX 是 ObjectARX 的一个子集，它等效于 ADS。使用 ADSRX，能够用 C 语言编写基于 AutoCAD 的程序，也能很方便地将 ADS 程序移植为 ObjectARX 程序。尽管 ObjectARX 接口是四个 API 中最强有力的，它也具有产生严重编程错误的最大潜在性，如破坏 AutoCAD 数据结构等。其他编程环境要求较少的编程经验，但提供的功能和范围也较小。

通过比较可以说，AutoLISP 着眼于应用程序的交互性，ADS C/C++着眼于应用程序的综合性，而 ObjectARX 则着眼于应用程序的智能性。

下面分别在四个方面(运行速度、稳定性、运行效率和开发难度)将四种开发工具做对比，并采用图像的形式展示出来供参考，如图 1-1 所示。

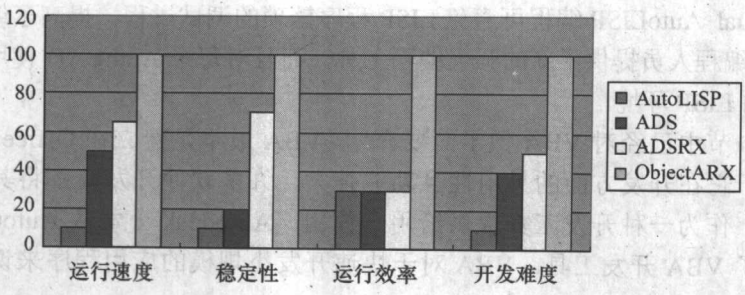

图 1-1　四种开发工具的比较

2. ObjectARX 与 Visual AutoLISP、VBA 的比较

从 AutoCAD R14 开始提供了采用对象编程技术的 Visual AutoLISP 语言，大大增强了 AutoLISP 的编程能力，是 AutoLISP 编程技术自 1985 年嵌入 AutoCAD 的百万用户传统的应用开发手段向新一代对象编程技术的飞跃。

Visual AutoLISP 是一个使用 LISP 语言开发和定制 AutoCAD 的可视化开发环境。它扩展和增强了现有的 AutoLISP 语言，提供了程序的编写和调试环境，可将 LISP 程序编译成 ObjectARX，大大提高了 CAD 编程效率和性能，是一个崭新的一体化可视 CAD 编程环境。Visual AutoLISP 提供标准 Windows 安装界面，安装方便。安装完后，进入 AutoCAD，在命令行上输入 VLIDE 就可以进入 Visual AutoLISP 编程环境。

对使用 AutoLISP 进行二次开发的人员，Visual AutoLISP 既是 LISP 编辑器又是编译器，它提供了一套简单的可视环境去开发和维护用户原有的 AutoLISP 源程序。Visual AutoLISP 新特点如下：

(1) 使用 Visual AutoLISP 可使用户的 AutoCAD 应用程序运行更快，它的编译器将 LISP 源程序编译成 ObjectARX 应用程序，由于 LISP 程序通过 AutoCAD 的 ObjectARX 接口运行，所以比 AutoLISP 加载运行快 3 倍～10 倍，并且省去了每次打开新图再调用的麻烦。

(2) 由于 Visual AutoLISP 采用 ObjectARX 平台，可以将 AutoCAD 和其他的应用程序如 Windows、Office、ActiveX 包含到 AutoLISP 源程序中，改善了 ActiveX 与 AutoCAD 对象模型之间接口特性，提高了应用程序的灵活性。

(3) 由于 Visual AutoLISP 编译成的二进制代码无法直接读取，所以它生成的应用程序更安全。

Visual AutoLISP 可视化编程提供了更多的实用功能：

(1) 提供控制台，在控制台的命令行作 AutoLISP 命令，可以直接看到结果。

(2) 彩色字符源代码检查，可以在编辑窗中同时显示 AutoLISP 和 DCL 源程序的命令，注释、提示等以各种颜色区分表示出来，易于检查。

(3) LISP 程序自动缩进和标准格式化。

(4) 括号匹配检查。

(5) 多窗口同时编辑 LISP 和 DCL 文件，并提供 DCL 对话框预览功能。

(6) 可直接将用户的 LISP＋DCL 文件编译成一个 ObjectARX 程序，并可连接 ObjectARX、VC、Visual Basic、ActiveX 的 AutoCAD 程序。

使用 Visual AutoLISP 编程可避免 LISP 程序繁琐的调试过程，提高了编程效率，它不仅对 CAD 编程人员提供了新的强大编程工具，而且对每个 AutoCAD 使用者都提供了更简单易用的 LISP 环境。

在上面一节中已经对 VBA 做了简要介绍。VBA 最早是建立在 Office 97 中的标准宏语言，由于它在开发方面的易用性且功能强大，许多软件开发商都将其嵌入自己的应用程序中，作为一种开发工具提供给用户使用。Autodesk 公司从 AutoCAD R14 版开始，内置了 VBA 开发工具。VBA 对于快速开发小规模的应用程序来说，是一种最好的选择。

3. 开发工具总结

以上对 AutoCAD 的三代开发工具分别做了介绍以及简单对比。目前，第一代的 AutoLISP 已能被第三代的 Visual AutoLISP 完全替代，第二代的 ADS 在 AutoCAD 2000 中已不再支持，所以，第三代开发工具将成为今后 AutoCAD 二次开发的必然选择。而在第三代工具中具体选择哪一种，笔者认为，主要还应根据用户应用程序的需要和开发人员的编程经验来选择最适合自己的一种开发工具。

1.2 ObjectARX 程序设计环境

1.2.1 ObjectARX 的运行环境

一个 ObjectARX 应用程序是一个分享 AutoCAD 的地址空间并为 AutoCAD 直接调用的动态链接库(DLL)。用户可以向 ObjectARX 编程环境添加新类，并可以导出它们为其他程序所用。用户创建的 ObjectARX 实体实际上与 AutoCAD 内置的实体是没有区别的。用户可以在运行期间向已存在的 AutoCAD 类添加函数以扩展 ObjectARX 协议。

1.2.2 ObjectARX 的功能简介

ObjectARX 编程环境提供了一个面向对象的 C＋＋应用程序开发界面，使开发者能够使用、定做和扩展 AutoCAD。ObjectARX 库包括各种各样的工具，可以使应用程序开发者方便地利用 AutoCAD 的开放结构，这些工具可以方便应用程序对 AutoCAD 数据库结构、图形系统和本地命令进行直接访问。另外，这些库被设计得可以与 Visual AutoLISP 及其他应用程序开发界面协同工作，以便开发者根据自己的需要和经验来选择最适合的开发工具。作为一个开发者，用户可以使用 ObjectARX 完成以下任务：

(1) 访问 AutoCAD 数据库；

(2) 与 AutoCAD 编辑器交互作用；

(3) 使用 MFC 创建用户界面；

(4) 支持多文档界面(MDI)；