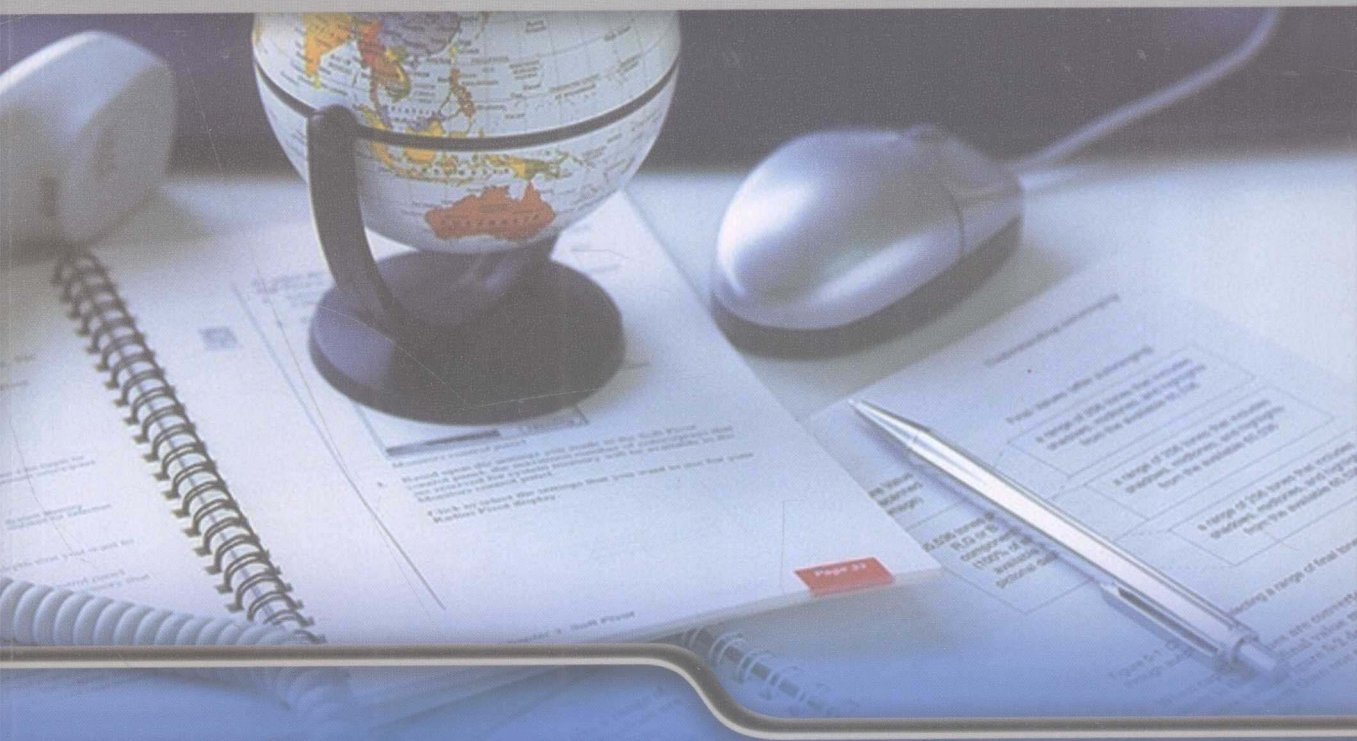


微软（中国）有限公司商务解决方案（MBS）事业部推荐技术图书



微软 ERP Dynamics AX 2009 企业开发指南

彭庆华 编著



清华大学出版社

微软 ERP Dynamics AX 2009

企业开发指南

彭庆华 编著

清华大学出版社
北 京

内 容 简 介

本书从微软 ERP Dynamics AX 2009 (简称 AX 2009) 的技术构架、开发平台开始 AX 2009 的企业开发之旅; 沿途深入剖析了数据字典、X++语言、数据库操作及查询; 小憩时侃了一个古老的 OOP 故事, 登上主峰后踏着华丽的景色生动演绎了用户界面相关的窗体、报表设计和开发, 在崎岖的悬崖边讲解了安全及配置, 最后探索通向外界的第三方接口; 或妙趣横生、或惊心动魄的旅途定让你或低头沉思、或豁然开朗, 掩卷之后定然满载而归、流连忘返。

本书包含 276 幅图, 77 张表格, 225 个完整的范例程序, 其内容丰富、系统、翔实, 可供从事 ERP 系统设计开发的科研人员和工程技术人员学习参考, 同时本书亦可作为大专院校微软 ERP 实验室以及企业 (特别是微软合作伙伴、咨询公司和实施 AX 系统的企业) 的技术培训教材。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

微软 ERP Dynamics AX 2009 企业开发指南 / 彭庆华编著. —北京: 清华大学出版社, 2011.4
ISBN 978-7-302-24951-1

I. ①微… II. ①彭… III. ①企业管理—计算机管理系统, ERP IV. ①F270.7

中国版本图书馆 CIP 数据核字 (2011) 第 030784 号

责任编辑: 王峰松 李 晔

责任校对: 徐俊伟

责任印制: 王秀菊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62795954, jsjic@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 清华大学印刷厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 34.5 字 数: 860 千字

版 次: 2011 年 4 月第 1 版 印 次: 2011 年 4 月第 1 次印刷

印 数: 1~4000

定 价: 69.00 元

序 言 一

作为高速成长型企业的信息化管理系统，微软 ERP Dynamics AX 具有高度灵活、高性价比、易学易用及高度集成等特性。正如德国哲学家莱布尼茨所说：世界上不存在两片完全相同的叶子。我们也认为：每个企业也都拥有自己独一无二的、有异于竞争对手的行业优势，即核心竞争力。那些以抹杀客户个性化特点为代价，而试图把客户引导到标准行业解决方案的系统，显然束缚了企业未来的发展空间。企业的快速发展，导致需求的日新月异，昨天的标准解决方案已难满足今天企业业务流程的需求。

AX 最大的特点在于它是一个开放的平台，我们在这个平台之上高度抽象了所有行业的共同特性，提供了大概 80%通用的标准功能。让精通行业的各合作伙伴提供垂直的行业解决方案，再由 ERP 系统实施商和客户共同完成客户化定制。这是一种全新的“平台+配置+服务（定制开发）”的模式。这种模式使得企业在拥有本行业最前沿、最先进的管理模式的同时，又保留并发扬了企业自身的核心竞争优势。我们所努力打造的是一种和合作伙伴及客户一起成长的“多赢”的“生态系统”。我们异常珍惜这来之不易的“生态圈”，并不遗余力地推动其往健康、良性、可持续发展的方向。

毋庸置疑，在这个“生态圈”中，无论是微软团队内部的开发和合作伙伴的垂直行业解决方案的大规模团队开发，还是客户内部的小规模个性化定制开发，AX 的应用技术都在这一系列过程中扮演着至关重要的作用。

本书作者彭庆华先生，多年来担任 AX 专家顾问，在技术管理及咨询工作的最前沿屡创佳绩，也积累了丰富的经验。从实践上升到理论，作者系统地把 AX 每个技术模块的细节一一呈现出来，深入浅出，举一反三。这不仅对 AX 技术顾问有极大的参考价值，更降低了普通员工和学生学习的技术壁垒。

随着 AX 市场的与日俱增，AX 应用技术的市场空白逐渐成为约束其发展的态势，可以肯定，本书的出版将对 AX 的推广应用，特别是 AX 应用开发技术，起到积极的推动作用。AX 技术的推广和完善，不仅仅凭借本公司的技术研发专家，事实上还依托更多的技术应用部门的专家，在此我要感谢彭庆华先生以及许多未曾谋面的专家。

孙志伟 (Francis. Sun)

微软（中国）有限公司 MBS 大中华区总经理

2011 年 1 月

序 言 二

微软 Dynamics AX 是一种多语言、多货币、针对大中型公司所设计的企业资源规划（ERP）解决方案。其核心优势在于制造和电子商务，而且它包含仓存和服务产业等功能，并可用于批发行业。强大、全面的功能（它们全都在一个解决方案中）可以帮助企业抓住稍纵即逝的商业机会，并赢得竞争优势。

微软 Dynamics AX 的外观和工作方式与微软 Outlook 类似，因此，员工学习和使用起来更为容易。使用者可单独把常用的功能添加到“我的最爱”，这使员工能够从各个不同的地方轻松地启动和使用它们。微软 Dynamics AX 凭借与其他微软的软体相似且简单易用的特点，帮助企业拥有更顺畅的自动化业务流程，并获得优化财务、客户关系、商业服务、人力资源管理和供应链流程管理等全面的功能。此外，它还可以使您的员工和业务流程紧密相连，不论他们在世界的哪个角落，都可以密切地联系在一起协同工作。因此可以帮助您提高公司的生产效率和工作效率，从而助您取得更好的业绩。

作为微软全球最大的合作伙伴，Tectura 战略联盟合作伙伴项目，确保了垂直化的行业解决方案与水平化的平台解决方案的完美结合，确保了最佳实施、合作项目以及全球实施方法论的高度一致性。因此，大中型企业客户将一如既往地享受灵活的、可管理的、具有低成本高效益的解决方案，而其都基于微软市场领先的产品。同时 Tectura 与微软 Dynamics 团队的紧密合作也确保了项目的高质量与成功。

Tectura 一直以来是微软 Dynamics 领先的合作伙伴，向中型公司、大型企业及其分支机构提供基于微软技术的商业解决方案。Tectura 向分销、制造、医药和服务等行业的 5000 多家客户提供软件、顾问和 IT 实施服务，向 60 多个国家的客户提供商业流程再造及优化，使其具备更高的效率、更坚固的竞争优势。Tectura 横跨美洲、欧洲、中东和亚太地区的 21 个国家，拥有 70 多家分公司，超过 1750 名专家，使顾客受益于丰富的经验和可靠的保障。Tectura 还是微软金牌认证合作伙伴，由于杰出的客户服务获得了微软公司的多次认证和表彰，如：微软商业解决方案核心合作伙伴、全球年度合作伙伴、年度发展最快合作伙伴、消费者满意度优秀奖、消费者忠诚奖等等。

基于微软技术的解决方案，帮助我们更有效地服务客户，这也促使我们与合作伙伴能取得持续性的成功。我们为 Tectura 所取得的成就感到骄傲，同时我们在微软合作伙伴项目中将一如既往的合作。

本书基于作者彭庆华多年在技术管理实践中对 AX 技术的深刻理解，系统地、深入地介绍了 AX 各部分的开发技术，并提供很多具有实用价值的范例和技巧，相信无论是刚入门的开发人员，还是资深的顾问都将会从本书中受益。

总之，我们期待本书能够实实在在地帮助客户及读者取得进步！

李君卫（Kenric . Lee）

上海瀚资软件咨询有限公司（Tectura）中国区总经理

2011 年 1 月

前 言

微软 Dynamics AX（以下简称 AX）是微软开发的面向大中型企业的 ERP（企业资源规划）解决方案，以其强大灵活的二次开发能力和与其他微软系产品线无缝衔接能力而闻名于世。AX 拥有微软系产品的显著特点：稳定、易用、安全、开放架构、易于集成，有极高的性价比，且已被 Garter 列入中型市场 ERP 解决方案领导者。

笔者自 2003 年接触 AX 以来一直奋战在系统分析、开发管理及实施的最前线，深知从 AX 入门者到资深技术顾问的心声，因为这就是我的成长历程。本书汇集了我多年实践经验的精髓，涉及诸多具有使用价值的范例和解决方案，内容翔实、丰富，相信不会辜负朋友们的厚望。

阅读本书最好的方法就是纵览全书，建立“索引”，然后在工作中现场查阅具体细节。那些努力记住每个函数及偏僻语法的开发人员和死记茴香豆的“茴”字有四种写法的孔乙己并无本质区别，因为大脑不是一个需要被填满的容器，而是思想的源泉，我们要用更多的时间来思考系统架构、模块设计和业务流程等更有价值的事情。本书就为此目的而写，让您摆脱底层代码对您的束缚，腾出更多的时间来高屋建瓴地思考真正重要的核心问题。

是故，技术者，项目之大事，成败之关键，不可不察也！

◆ 主要内容

本书遵循一般计算机技术类图书的编写模式，先用第 1 章简单回顾了 AX 的历史，介绍了 AX 的技术架构及各种客户端连接方式。

第 2 章介绍 AX 的集成开发环境（IDE）MorphX 中各种工具的功能及使用方法、技巧，熟练地使用这些工具是高效地开发 AX 程序的基础。

第 3 章介绍数据字典下的各种元素及其创建、使用方法。深入介绍系统的缓存机制、系统字段（RECID、recVersion 等）产生机制、多公司账户在系统中的实现机制、索引优化以及各元素的内部关联关系等。

第 4 章介绍所有 X++支持的数据类型及复合数据类型的使用及其常用函数，各种运算符、流程控制、错误异常处理及宏指令的使用等。

第 5 章介绍如何使用 X++语言来操作数据库。首先介绍 CRUD（即创建、读取、更新及删除）操作，接着深入介绍如何在 AX 中实现数据库的各种事务、锁定、交易隔离级别及并发模式。最后介绍从 AX 直接调用数据库的几种方式。

第 6 章在介绍面向对象各术语的同时结合 X++的实现方式，由浅入深地介绍 X++的面向对象特性，包括类的封装、继承、多态、序列化等等。并深入介绍 RunBase 类构架、批处理类的内部机制及实现方法，最后介绍系统重点类及其使用方法。

第 7 章介绍窗体、控件的设计，及窗体和控件对应的类和方法、窗体间的关系等，同时介绍了对话框、下拉框、向导和库存维度的设计及使用等。

第 8 章介绍 AX 中使用到的 MorphX 报表和 SSRS 报表的开发设计方法，同时介绍与报表相关的 RunBaseReportStd 类和 PrintJobSettings 类。

第 9 章介绍菜单、菜单项、内容控制面板和导航栏的创建及使用，开发人员可以通过这些功能发布新开发的功能给最终用户调用。

第 10 章介绍如何在 AX 中添加和引用资源，包括图片资源和其他文件资源。

第 11 章深入介绍 AX 中的反射体系，可以通过反射在运行时获取、创建和调用 AX 中的各种元素，包括类、报表等等。

第 12 章介绍 AX 中如何选择文件、文件夹，处理 CSV 文件、XML 文件、文本文件、Excel 文件、Word 模板文件等，同时介绍如何调用 Outlook 发送邮件。

第 13 章介绍 AX 中的安全及配置体系。解决方案发布者可以通过许可码来控制系统的可用功能，系统实施商再通过配置码去除或激活某些特殊功能，最后系统管理员使用安全码为最终用户分配可用的功能权限及记录级数据权限。

第 14 章介绍 AX 和第三方软件集成的方法。第三方软件不但可以通过 .Net Business Connector 调用 AX 的内核业务逻辑，对 AX 数据库进行 CRUD 操作，还可以使用 Web 服务发布和暴露 AX 的内部业务逻辑供第三方软件调用，同时也可以调用第三方软件供应商提供的 Web 服务。

◆ 本书约定

- 前面带星号 (*) 的章节属于高级课程，初学者可以先略过，即使您暂时不太了解这些功能也照样可以写程序，只不过效率会有些区别或绕点弯子而已。
- 本书中，若无特别说明，数据库指微软 SQL Server 2005 及 SQL Server 2008 数据库，AX 指微软 Dynamics AX，AX 2009 指微软 Dynamics AX 2009。
- 本书所使用的全部范例代码在 Windows Server 2008 + SQL Server 2008 + AX 2009+SP1 环境测试通过，若要使用于其他平台，烦请读者自己测试。
- AOT 中的节点除专有名词外，一概称之为元素，因为对象在 OOP 中另有其意。
- 请不要在正式生产环境中使用本书的范例，一切后果与作者无关。

◆ 本书适合那些人阅读

- Dynamics AX 技术顾问。我一开始决定写此书时，就以达到“即使您是拥有 5 年以上经验的资深 AX 技术顾问，也会从中受益”为目的，所以，每一章节都是由浅入深，并结合实际例子展现各种实用的高级功能。
- Dynamics AX 系统甲方企业开发人员和维护人员。本书不但系统讲解了 AX 的整个开发体系，还提供了一些平时会用到的小技巧。它不但能帮助您完成企业内部的客户化需求，更能让您在应对现场环境各种燃眉之急的问题时淡定而从容。
- Dynamics AX 应用顾问。实施现场总有难以预料的问题发生，而身边没有技术顾问，会不会尴尬之极呢？您只要花两三天读读本书，便可以学会一些实用而常用的技术，边学边用，解决您的燃眉之急。
- ERP 同行。不管您是否承认，AX 已在 ERP 市场上成为了耀眼的新星，那些先进的设计理念和技术构架就在您眼前的书里，难道您不想一探究竟？
- 对 AX 或 ERP 技术感兴趣的读者。AX 顾问市场供不应求，本书将引领您进入这充满魅力舞台。

◆ 源代码

为保护地球、减少碳排放，本书不提供范例代码的光盘，请读者到以下地址下载：

[http://axBook.QiuHao.com/index.asp\\$boardid=61](http://axBook.QiuHao.com/index.asp$boardid=61)

◆ 勘误表

本书在出版后发现的错误将实时更新到勘误表中，也请读者到上述地址获取。

◆ 学习资源

➤ 英文学习资源

Dynamics AX 2009 SDK: <http://msdn.microsoft.com/en-us/library/aa546389.aspx>

Dynamics AX MSDN: <http://msdn.microsoft.com/en-us/dynamics/ax/default.aspx>

Dynamics AX 社区: <https://community.dynamics.com/product/ax/default.aspx>

Dynamics AX 博客: <https://community.dynamics.com/product/ax/b/default.aspx>

◆ 中文学习资源

秋毫 ERP 咨询论坛: <http://www.qiuhao.com/>

MBSCN 论坛: <http://www.mbscn.net/>

◆ 特别鸣谢

感谢我所有家人的支持和理解！

感谢微软（中国）有限公司 MBS 大中华区总经理孙志伟先生和王雷雁女士的大力支持！

感谢瀚资软件咨询有限公司中国区总经理李君卫先生的大力支持！

感谢瀚资软件咨询有限公司北京分公司总经理徐勇和高级顾问经理张强的支持及鼓励！

感谢所有对本书提出意见的朋友及 Tectura 同事！

所有这些帮助都对促进本书最终诞生及出版起到了不可或缺的作用。

由于作者水平有限，书中难免存在错误和缺点，殷切希望读者批评指正。

作者 彭庆华

2010 年 11 月于北京

目 录

第 1 章 初识 Dynamics AX..... 1

- 1.1 Dynamics AX 历史回顾..... 1
- 1.2 ERP 与 Dynamics AX..... 2
- 1.3 AX 系统技术架构..... 3
 - 1.3.1 表现层..... 4
 - 1.3.2 业务逻辑层..... 5
 - 1.3.3 数据层..... 5
- 1.4 小结..... 6

第 2 章 开发环境 MorphX..... 7

- 2.1 介绍..... 7
- 2.2 AOT..... 7
 - 2.2.1 AOT 概览..... 7
 - 2.2.2 加载项..... 11
 - 2.2.3 清除代码缓存..... 11
- 2.3 层..... 12
- 2.4 项目..... 14
 - 2.4.1 项目介绍..... 14
 - 2.4.2 新建和修改项目..... 15
 - 2.4.3 项目类型..... 17
- 2.5 Jobs..... 17
- 2.6 代码移植..... 19
- 2.7 属性表..... 21
- 2.8 X++代码编辑器..... 22
- 2.9 编译器..... 23
- 2.10 调试器..... 25
- 2.11 标签编辑器..... 28
 - 2.11.1 标签简介..... 28
 - 2.11.2 创建标签文件..... 29
 - 2.11.3 创建和编辑标签..... 29
 - 2.11.4 引用标签..... 30
- 2.12 查找工具..... 30
- 2.13 比较工具..... 32
- 2.14 交叉引用..... 33
- 2.15 版本控制..... 34
- 2.16 用户环境设置..... 38

- 2.17 代码事件探查器..... 39

- 2.18 系统文件结构..... 41

- 2.19 小结..... 42

第 3 章 数据字典..... 43

- 3.1 介绍..... 43
- 3.2 数据字典同步..... 44
- 3.3 表..... 45
 - 3.3.1 表浏览器..... 45
 - 3.3.2 创建表..... 46
 - 3.3.3 表属性..... 47
 - 3.3.4 表缓存..... 48
 - 3.3.5 表变量及赋值..... 51
 - 3.3.6 字段..... 52
 - 3.3.7 公司账户..... 58
 - 3.3.8 索引..... 66
 - 3.3.9 表、字段和索引的
名字长度..... 68
 - 3.3.10 表方法..... 69
 - 3.3.11 关联..... 75
 - 3.3.12 删除触发事件..... 78
 - 3.3.13 临时表..... 80
 - 3.3.14 常用系统表..... 87
- 3.4 表映射..... 88
- 3.5 视图..... 90
- 3.6 小结..... 92

第 4 章 X++语言基础..... 93

- 4.1 X++简介..... 93
- 4.2 变量和表达式..... 93
 - 4.2.1 关键字..... 93
 - 4.2.2 标识符命名..... 94
 - 4.2.3 变量声明..... 95
 - 4.2.4 注释..... 97
 - 4.2.5 屏幕打印..... 102
- 4.3 数据类型..... 102
 - 4.3.1 介绍..... 102

4.3.2	基本数据类型	103	5.3.1	数据插入	207
4.3.3	复合数据类型	121	5.3.2	数据更新	212
4.3.4	扩展数据类型	128	5.3.3	数据删除	216
4.3.5	扩展数据类型数组	131	5.3.4	doInsert()、doUpdate() 和 doDelete()	218
4.4	运算符	132	5.3.5	触发日志和预警	219
4.4.1	赋值运算符	133	5.4	*数据查询及操作相关 的系统类	220
4.4.2	算术运算符	133	5.4.1	*RecordInsertList 类及 RecordSortedList 类	220
4.4.3	关系运算符	134	5.4.2	*RecordView Cache 类	223
4.4.4	位运算符	136	5.5	事务	224
4.4.5	运算符优先级	137	5.6	锁定	230
4.5	流程控制	138	5.7	交易隔离级别	231
4.5.1	分支	138	5.8	并发模式	234
4.5.2	循环	144	5.8.1	保守并发模式	234
4.6	错误和异常处理	150	5.8.2	积极并发模式	236
4.6.1	异常种类	150	5.8.3	同一连接中多次 更新记录	240
4.6.2	捕获异常	150	5.8.4	SkipTTSCheck	242
4.7	宏指令	154	5.8.5	相对更新	243
4.7.1	声明指令	155	5.8.6	设定并发模式	244
4.7.2	条件编译指令	155	5.9	直接运行 SQL 脚本及存 储过程	246
4.7.3	宏指令参数	157	5.9.1	连接到当前 AX 数 据库	246
4.7.4	局部 Macro 和全 局 Macro	158	5.9.2	用 ODBC 连接到外 部数据库	250
4.7.5	用 Macro 定义常量	159	5.9.3	ADO 操作外部 数据库	252
4.7.6	在 X++ 查询中 使用 Macro	159	5.10	小结	256
4.7.7	宏注释	160	第 6 章	类与面向对象	257
4.8	小结	161	6.1	什么是类	257
第 5 章	数据查询及操作	162	6.2	定义类	258
5.1	数据查询语言	162	6.3	继承	258
5.1.1	Select 语句	162	6.4	终类	259
5.1.2	聚合函数	177	6.5	抽象类	260
5.1.3	分组查询	179	6.6	接口	260
5.1.4	多表连接查询	181	6.6.1	定义接口	261
5.1.5	*有限字段查询	185	6.6.2	在类中实现接口	262
5.1.6	表达式中的 select 语句	190	6.7	类成员	262
5.1.7	右对齐字段	191	6.7.1	访问限定符	262
5.2	查询	193			
5.2.1	静态查询	194			
5.2.2	代码创建动态查询	202			
5.2.3	复合查询	205			
5.3	数据操作语言	207			

6.7.2	修饰符	264	7.4.6	使用数据源编辑 记录	337
6.7.3	字段	265	7.5	界面设计	339
6.7.4	方法	265	7.5.1	标准控件及组件	339
6.7.5	this 的意义	275	7.5.2	Grid 控件	342
6.7.6	多态	275	7.5.3	Window 控件及存 取图片	348
6.7.7	传值调用与传址 调用	279	7.5.4	预加载图片	353
6.7.8	序列化	282	7.5.5	*树形控件	355
6.7.9	AOS 服务器	285	7.5.6	Display 方法和 Edit 方法	362
6.8	*RunBase 类构架	289	7.5.7	*运行时窗体	364
6.9	*批处理	293	7.5.8	用代码调用窗体	366
6.9.1	简介	293	7.5.9	*窗体运行时控件及 方法	368
6.9.2	创建批处理类	294	7.5.10	*对话框运行时控 件及方法	371
6.9.3	配置批处理服务器	299	7.5.11	*使用窗体对话框	373
6.9.4	启动批处理作业	301	7.5.12	窗体分割	377
6.10	编号规则	304	7.5.13	模式窗体	379
6.11	应用层内核类	307	7.5.14	修改窗体通用属性	380
6.12	重点系统类	310	7.5.15	进度条	381
6.12.1	Object 类型	310	7.6	下拉框	383
6.12.2	Args 类	311	7.6.1	自动下拉框	383
6.12.3	ApplicationVersion 类	312	7.6.2	动态下拉框	383
6.12.4	xppCompiler 类	313	7.6.3	在下拉框中使用 临时表	385
6.13	集合类	314	7.6.4	使用窗体下拉框	387
6.13.1	Set 类	314	7.7	向导	391
6.13.2	List 类	316	7.7.1	简介	391
6.13.3	Map 类	316	7.7.2	创建标准向导	392
6.13.4	Struct 类	317	7.8	库存维度	398
6.13.5	Array 类	318	7.8.1	新建库存维度	398
6.13.6	Stack 类	319	7.8.2	维度显示	405
6.13.7	StackBase 类	319	7.8.3	查找库存维度	410
6.14	小结	320	7.9	财务维度	411
第 7 章	窗体	321	7.10	小结	412
7.1	简介	321	第 8 章	报表	413
7.2	创建窗体	323	8.1	简介	413
7.3	窗体方法	325	8.2	MorphX 报表	413
7.4	数据源	327	8.2.1	报表设计基本准则	413
7.4.1	数据源关联	328	8.2.2	报表向导	414
7.4.2	数据源方法	330			
7.4.3	窗体间关联	331			
7.4.4	访问控制	334			
7.4.5	过滤数据源	335			

8.2.3	创建新报表	421	12.2	选择文件	482
8.2.4	报表的数据源	423	12.3	CSV 文件	484
8.2.5	报表的设计	424	12.3.1	导出 CSV 文件	484
8.2.6	报表方法	429	12.3.2	读取 CSV 文件	486
8.2.7	使用 RunBaseReport 类	435	12.4	文本文件	487
8.2.8	使用 PrintJobSettings 类	438	12.5	XML 文件	488
8.2.9	在报表中使用临时表	439	12.5.1	导出 XML 文件	489
8.3	SSRS	441	12.5.2	读取 XML 文件	490
8.3.1	SSRS 简介	441	12.6	微软 Office Excel 文件	491
8.3.2	创建 AX SSRS 报表	441	12.7	微软 Office Word 文件	494
8.4	小结	447	12.8	调用 Outlook 发邮件	496
8.4	小结	447	12.9	小结	499
第 9 章	菜单	448	第 13 章	安全与配置	500
9.1	菜单项	448	13.1	介绍	500
9.2	菜单	450	13.2	许可码	500
9.3	内容面板	451	13.3	配置码	502
9.4	小结	453	13.4	安全码	505
第 10 章	资源	454	13.5	记录级权限	508
10.1	简介	454	13.6	小结	510
10.2	添加资源	454	第 14 章	第三方应用程序集成	511
10.3	引用资源	455	14.1	简介	511
10.4	小结	456	14.2	CLR	512
第 11 章	反射	457	14.3	.NET Business Connector	514
11.1	概述	457	14.3.1	简介	514
11.2	反射方法	458	14.3.2	调用 AX 方法	514
11.2.1	内置方法	458	14.3.3	插入 AX 数据库	518
11.2.2	数据类型方法	461	14.3.4	查询 AX 数据库	519
11.2.3	类类型方法	461	14.3.5	更新 AX 数据库	520
11.3	反射接口	464	14.3.6	删除 AX 数据库	521
11.3.1	系统表	464	14.4	Web 服务	521
11.3.2	数据字典接口	467	14.4.1	简介	521
11.3.3	AOT 节点接口	474	14.4.2	创建 AX Web 服务	522
11.4	小结	479	14.4.3	发布 AX Web 服务	526
第 12 章	文件处理	480	14.4.4	AX 中调用外部 Web 服务	527
12.1	选择文件夹	480	14.5	小结	529
12.2	选择文件	482	索引	范例索引	531
12.3	CSV 文件	484	参考文献	参考文献	537
12.3.1	导出 CSV 文件	484			
12.3.2	读取 CSV 文件	486			
12.4	文本文件	487			
12.5	XML 文件	488			
12.5.1	导出 XML 文件	489			
12.5.2	读取 XML 文件	490			
12.6	微软 Office Excel 文件	491			
12.7	微软 Office Word 文件	494			
12.8	调用 Outlook 发邮件	496			
12.9	小结	499			

第 1 章 初识 Dynamics AX

1.1 Dynamics AX 历史回顾¹

Dynamics AX 从 1.0 到 3.0 版本都叫做 Axapta，直到微软发布 4.0 之后，才正式改名为 Microsoft Dynamics AX²，简称 AX。AX 是微软 Dynamics 商务解决方案（Microsoft Business Solution，MBS）中功能最强大也最受瞩目的产品，其他 Dynamics 系列的商务解决方案包括 Microsoft Dynamics NAV、Microsoft Dynamics GP、Microsoft Dynamics SL 和 Microsoft Dynamics CRM。

Axapta 相关技术的研发可追溯到 1983 年，尽管前期的产品和技术为 Axapta 的最终诞生奠定了坚实的基础，但它们在名称上都和 Axapta 没有太大关系。丹麦的 Damgaard A/S 公司于 1998 年 3 月正式发布 Axapta 1.0 版，主要面向丹麦和美国市场，以财务、贸易、库存管理、物流和生产等模块的强大功能而备受关注。Axapta 从第一版开始就支持微软的 SQL Server 数据库和 Oracle 数据库。该公司于同年 11 月发布 1.5 版本，主要面向欧洲市场。

1999 年 7 月，Axapta 2.0 发布，在这一版中比较吸引人的特性有项目会计、仓库管理、OLAP 支持、ActiveX 支持、COM 接口支持和早期版本的 AOS 服务器等。

2000 年 1 月，Axapta 2.1 发布，增加了 CSS（Customer Self-Service）模块，并于 SP3 中正式引入 AOS，这使得 Axapta 成为真正意义的三层结构的 ERP 系统。同年 Damgaard 公司和丹麦的另一家公司 Navision 合并为新的公司 Navision Damgaard。Axapta 从 2.5 到 3.0 版本期间，被改名为 Navision Damgaard Axapta。

2000 年 11 月，Axapta 2.5 发布。在这一版中，Axapta 增加和增强了 Web 开发环境、项目管理模块、银行和 OLAP。并在 2001 年 10 月发布的升级版中新增了 CRM 和产品生成器（product builder）。

2002 年夏天，微软兼并了 Navision Damgaard 公司，Navision Damgaard Axapta 也改名为 Microsoft Business Solutions Axapta。

2002 年 10 月，Axapta 3.0 发布。新增了企业门户、多公司交易等功能，增强了用户权限设置和系统配置等功能。

2006 年 3 月，AX 4.0 发布。Axapta 被正式命名为 Microsoft Dynamics AX，对界面进行了重大的提升，同时集成了众多微软的成熟技术。比如 AOS 已经变成了一项系统服务，首次引入 .NET Business Connector，CLR 和 XML 支持，Unicode 支持等。

2008 年 6 月，AX 2009 发布。进一步提升了用户界面，提供了协调世界时 UTC（Coordinated Universal Time）支持，并在仓库维度中引入站点（site），支持 workflow、角色中心、法规遵从中心和不同国家特定的功能等，并提供了 36 个国家和地区的 40 多种语言。

¹ 本节的部分内容参考维基百科：http://en.wikipedia.org/wiki/Microsoft_Dynamics_AX。

² 若无特别说明，本书以下内容中提及的 AX 指微软 Dynamics AX，AX 2009 指微软 Dynamics AX 2009。

微软计划于 2011 年第一季度发布 AX 6 (AX 2011), 2014 年发布 AX 7, 2017 年发布 AX 8。

1.2 ERP 与 Dynamics AX

ERP 是 Enterprise Resource Planning (企业资源计划)简称,其概念最初由美国 Gartner Group 于 1990 年提出,它本身是一种管理的方法和管理模型,而不是软件,但是由于这个管理模式需要的信息量极大、运算极为复杂,即使是中等规模的公司,其信息运算量也几乎是超越了人类手工计算的极限,所以一般使用信息系统来实现 ERP 管理模型,这就是 ERP 系统。世界有名的 ERP 管理软件有 SAP (德国)、Oracle (美国)和微软 (美国)的 Dynamics AX 等,国内有名的 ERP 软件供应商有金蝶、用友等,这些软件都从不同的角度实现了 ERP 管理模型中定义的几乎所有功能,甚至超越了模型所定义的范畴。

Dynamics AX 集多国家(时区)、多语言、多公司和行业解决方案于一个平台之中的 ERP 解决方案,它不但能满足中、大规模的组织及集团公司的需求,还能在一套系统中集成多套第三方合作伙伴开发的行业解决方案,并可以根据企业自身的特点进行企业个性化定制开发。

从纵向的角度看,微软提供了从底层的操作系统、数据库、.NET 平台到应用层的 Office 套件和 Dynamics 系列 ERP 套件等一个完整的产品链,AX 是集成于这套产品体系中的一个应用软件,一旦 AX 和这整套微软产品集成在一起后就会产生 1+1>2 的协同效应,它一方面给企业降低了总体的成本,包括购买成本、实施和培训成本、维护成本和运营成本等,另一方面给企业铸就了稳定的、可拓展的、易用的企业管理平台,避免了传统 ERP 由于集成了多个其他供应商的产品而导致的稳定性及兼容性问题。用户可以在 AX 中和微软自己提供的产品链上的多个企业级的产品或组件进行完美的集成,比如和 Office 套件(常用的如 Excel、Word、Outlook、Project Server 等)、SharePoint、BizTalk、AD、SSRS 商业智能等集成。

从横向角度看,AX 本身已经超越了一个产品的概念,它更是一个平台。任何合作伙伴和客户都可以在这个平台之上开发适合于特定行业的解决方案,同时 AX 把这些不同合作伙伴和客户开发的解决方案按不同层组织在一起,使得企业能非常从容地管理这些解决方案的源代码。事实证明,要用一个软件管理所有的行业是不可能的,那么是否可以使用同一个行业解决方案管理同行的企业呢?答案也是值得商榷的。因为企业的竞争力来源于企业之间的差异化(包括战略、管理、市场、产品等差异化),而这些独特的差异特点构成了该企业独一无二的核心竞争力,那么一个行业解决方案能管理的企业还会有差异化吗?还能体现其独一无二的核心竞争力吗?这些能力是否会被这个行业解决方案磨平呢?这是传统仅以解决方案作为卖点的产品无法解决的问题。而 AX 却以平台(微软提供)+解决方案(合作伙伴)+个性化定制(合作伙伴或客户内部开发)的方式来解决这些棘手的难题,这种模式无形中把各种组织最得意的智慧集成到了一起:微软由于精技术,所以提供一个标准的 ERP 平台;合作伙伴长于行业经验,所以提供解决方案,而客户对自己的管理模式和独特的管理方法最为熟知,所以自己提供个性化的功能,这些个性化开发也可以和合作伙伴共同完成。这样最终实施到客户环境的 AX 就变成了集各公司和组织的智慧于

一体的完整的企业解决方案。

本书所探讨的就是如何为 AX 开发企业级的解决方案、个性化定制以及如何和外部应用集成。

1.3 AX 系统技术架构

AX 2009 是典型的三层结构的系统,从 AX 4.0 开始正式取消两层结构,总体架构上可分为表现层、业务逻辑层和数据库层,如图 1-1 所示。

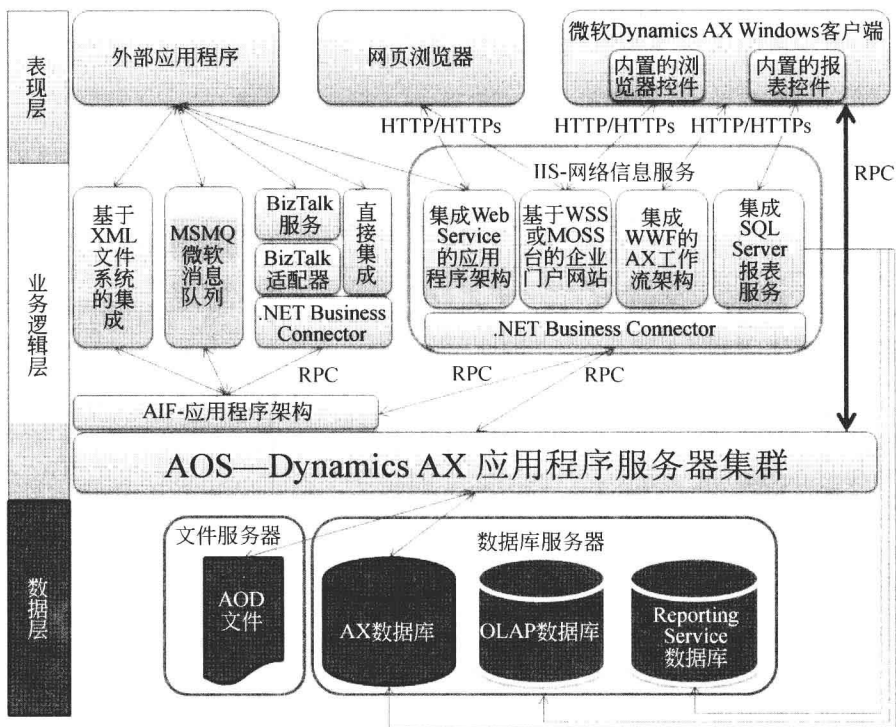


图 1-1

其中表现层主要处理界面显示、用户交互及事件捕捉等,几乎所有业务逻辑相关的处理都集中在应用程序服务器端执行。AX 除支持传统的 C/S (客户端/服务器) 结构的胖客户端调用外,还支持 B/S (浏览器/服务器) 模式及第三方应用程序调用。第三方应用程序可能是 Windows 应用程序、Web 服务或其他终端设备,比如各种手持终端等。业务逻辑层是各种处理业务逻辑相关的服务的总称,这些服务可以运行于一台或多台服务器上,但无论客户端如何调用,AOS 服务器始终是不可或缺的应用程序服务器,而其他所有服务均为可选服务,这得视企业的需求及规模而定。数据层主要用于保存源代码文件、临时文件和各种数据(包括业务数据、系统数据等),源代码文件可以保存于单独的文件服务器中,也可以保存于 AOS 服务器中。AX 所用到的各种数据都保存于数据库服务器中,AX 目前支持 SQL Server 数据库和 Oracle 数据库。

1.3.1 表现层

表现层是直接和用户交互的一层，按使用方式可以分为三类，如图 1-2 所示。

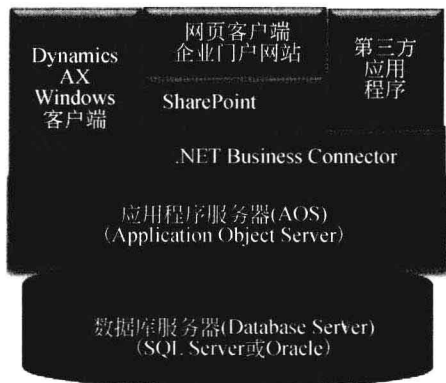


图 1-2

1. 胖客户端

胖客户端 (Rich Client) 也叫 Dynamics AX Windows 客户端，也可翻译为富客户端，它是流行的 C/S 模式的 Windows 应用程序，这是最主要的 AX 客户端，大部分客户的绝大部分操作都通过使用这种客户端来完成。这种客户端目前只能在 AX 集成的 MorphX 开发环境中使用 X++ 语言开发。胖客户端和应用程序服务器的通信使用 RPC 协议 (Microsoft Remote Procedure Call)。

2. 企业门户 (Enterprise Portal)

网页客户端 (Web Client) 是由浏览器解析执行的应用程序，目前网页客户端能实现绝大部分胖客户端的功能，但其易用性、功能性远没有 C/S 客户端强大。开发网页客户端需用到 MorphX 开发环境、Visual Studio 2008 开发环境和 SharePoint。当 IIS 服务器解析执行网页请求时，后台系统同时要调用 ASP.NET 运行时服务、AX 运行时服务和 SharePoint 运行时服务来共同执行完成。

3. 第三方应用程序

AX 允许第三方应用程序调用 AOS 层的业务逻辑，比如 Outlook (同步 CRM 相关数据)、Excel (数据导入、导出等) 和企业或合作伙伴开发的外部应用程序，我们可以通过外部应用程序来完成 C/S 可以完成的功能，比如创建工单、销售单、采购单，甚至过账这些单据等。这些应用程序主要是由 Visual Studio 或者其他符合 WCF (Windows Communication Foundation) 标准的开发工具开发而成。第三方应用程序通过调用 AIF (Application Integration Framework) 或者业务连接器 (Business Connector) 链接到 AOS，并最终由 AOS 的运行时服务解析执行。外部应用程序还可以使用 COM 接口和 AOS 通讯，不过 AX 2009 是支持 COM 技术的最后一个版本，这意味着 AX 的下一个版本将不再支持 COM 调用。

如今公司之间的商业协作日益频繁，比如和供应商、客户间的信息交换等。通过 AX 集成的 Web Service 技术，不但可以容易调用外部的 Web Service，也可以通过 Web Service 暴露 AX 的内部功能，比如读取 AX 内部的库存信息、从 B2C/B2B 平台创建销售单等。本

书的最后将使用实例来讲解如何在 AX 内部调用外部的 Web Service, 以及如何发布 AX 创建的 Web Service 等。

1.3.2 业务逻辑层

业务逻辑层是扩展性非常强的一层。首先, 为了提高多用户访问的性能, 应用程序服务器可实现集群; 另外, 当集成企业门户时, 将添加 IIS 网络信息服务器, 当和外部应用程序集成时, 可增加外部应用程序的中间层服务器。可实现负载均衡的 AOS 服务器集群功能保证了大型企业的高扩展性需求。

AX 的大部分业务逻辑基本上都运行于 AOS 服务器, 比如类、查询和表的静态方法等, 一般都执行于 AOS 服务器 (开发人员可以通过 RunOn 属性或 Server/Client 关键词指定类和方法的执行层), 这就使得相同的业务逻辑可被各种类型的客户端重复调用。由于 AOS 服务器直接和数据库服务器交互执行业务逻辑, 然后返回执行的结果给客户端, 这也使得 AX 客户端降低了对网络带宽的需求, 提高了整体的效率。

AOS 通过 RPC 协议 (Remote Procedure Call) 和客户端通信, 它是一种基于 TCP/IP 的通信协议, 这就使得 AX 客户端可执行于局域网和广域网中, 甚至可以不需要特殊配置就可以运行于企业通过公网建立的 VPN 网络中, 但鉴于目前国内几兆的带宽, 直接运行于 VPN 的客户端效果不是很理想。

AX 的设计理念是让开发人员将更多的精力集中在商业逻辑上, 而不是编码上, 更不是界面设计上, 所以诸多繁杂的功能, 比如界面布局、多语言等都由 AX 的 IntelliMorph 自动完成。我们把这种以模块化构建应用程序的方式叫做“模块驱动”的设计模式。

从 AX 4.0 开始, 只有三层结构的客户端, 也就是任何 AX 的客户端都必须通过 AOS 来访问数据库, 不能直接访问数据库。当然为满足某些特殊需求, X++ 也允许直接调用 SQL 语句操作数据库, 但这是一种以破坏系统严谨性和安全性为代价换取操作方便或少许性能的方法, 直接操作数据库意味着绕开了 AX 的安全机制, 同时由于必须赋予客户端操作数据库的相应的权限, 这将置数据库于危险之中。同时我们还要考虑数据库类型, 不同的数据库对标准 SQL 的兼容性各不相同。总之, 直接操作数据库是一种不被推荐使用的方法。

1.3.3 数据层

数据层主要包括程序代码文件、缓存文件、临时文件和数据。前三者以文件的形式保存于磁盘中, 其中代码文件可以单独保存在文件服务器中, 而业务数据和系统数据都保存于数据库中。AX 目前支持 SQL Server 数据库和 Oracle 数据库, AX 的数据库层统一了各种数据库的访问接口, 所以开发人员不必关心 AX 所使用的数据库类型, 所有对数据库的操作, 包括创建表、字段、索引、查询、修改和删除等, 都使用相同的方式。也不像传统的开发语言那样在调用数据库前要先要创建数据库连接, AX 会为我们的数据库操作语句创建连接, 并根据当前使用的数据库类型翻译成合适的数据库语句提交给底层数据库。从一个开发者的角度看, AX 的内核已经封装了数据库之间的差异, 并对我们提供统一的与数据库无关的访问接口。

为优化性能, AX 的数据库引擎还提供了连接池以及在所有客户端连接之间共享数据缓存的能力, 这意味着一旦客户端的请求满足命中缓存的条件, 就可以直接从 AOS 服务器