

# 1С: ПРЕДПРИЯТИЕ 7.7/8.0 СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ 2-е издание



ПРИМЕНЕНИЕ  
ТЕХНОЛОГИЙ OLE,  
COM, ActiveX

ИСПОЛЬЗОВАНИЕ  
СКРИПТОВ  
WSH И WMI

ПРИМЕНЕНИЕ ADO  
И SQL-DMO ПРИ РАБОТЕ  
С БАЗАМИ ДАННЫХ

ИНТЕГРАЦИЯ  
С MICROSOFT  
OFFICE

ИСПОЛЬЗОВАНИЕ ACTIVE  
DIRECTORY SERVICE  
INTERFACES

**PRO**  
ПРОФЕССИОНАЛЬНОЕ  
ПРОГРАММИРОВАНИЕ

+ CD

**Андрей Михайлов**

**1С:  
ПРЕДПРИЯТИЕ  
7.7/8.0  
СИСТЕМНОЕ  
ПРОГРАММИРОВАНИЕ**  
*2-е издание*

Санкт-Петербург

«БХВ-Петербург»

2007

УДК 681.3.06  
ББК 32.973.26-018.2  
М69

**Михайлов А. В.**

М69 1С:Предприятие 7.7/8.0: системное программирование. 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2007. — 384 с.: ил. + CD-ROM  
ISBN 5-94157-960-8

Содержится информация, предназначенная для практического применения скрытых возможностей операционной системы и разнообразных COM-объектов при разработке собственных информационных систем на платформе "1С:Предприятие" версий 7.7 и 8.0. Рассмотрены особенности использования объектов ActiveX, администрирования Windows средствами WMI и WSH, взаимодействия с базами данных посредством интерфейсов ADO и SQL-DMO, применения технологии OLE Automation и многое другое при работе в среде "1С:Предприятие".

Во втором издании книги добавлены новые описания возможностей использования технологии ADO и интеграции с офисными приложениями Microsoft Word, Excel, Visio и Project.

Материал сопровождается большим количеством наглядных и подробно прокомментированных примеров, полные тексты которых приведены на компакт-диске.

*Для программистов*

УДК 681.3.06  
ББК 32.973.26-018.2

#### **Группа подготовки издания:**

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Анна Кузьмина</i>
Компьютерная верстка	<i>Натальи Караваевой</i>
Корректор	<i>Виктория Пиотровская</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 23.11.06.

Формат 70×100<sup>1/16</sup>. Печать офсетная. Усл. печ. л. 30,96.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов  
в ГУП "Типография "Наука"  
199034, Санкт-Петербург, 9 линия, 12

ISBN 5-94157-960-8

© Михайлов А. В., 2007  
© Оформление, издательство "БХВ-Петербург", 2007

# Оглавление

<b>Введение</b> .....	<b>1</b>
Зачем нужна эта книга .....	1
Структура книги .....	2
Программные требования .....	4
Описание сопроводительного компакт-диска .....	4
Источники информации .....	4
<b>Глава 1. Применение технологий COM и ActiveX</b> .....	<b>5</b>
Технологии COM и ActiveX .....	5
Работа с объектом Internet Explorer .....	7
Определение текущего разрешения экрана .....	7
Анализ HTML-страниц .....	8
Работа с POST-запросами в HTML-формах .....	10
Работа с объектом Microsoft Winsock .....	12
Определение IP-адреса локального компьютера .....	13
Определение IP-адреса удаленного компьютера .....	13
Создание штрихкодов с использованием ActiveX <i>ActiveBarcode</i> .....	14
Использование ActiveX Windows Media Player .....	17
Использование помощников Microsoft Agent .....	19
Понятие MS Agent .....	19
Практическое применение MS Agent .....	20
Интерактивное взаимодействие с MS Agent .....	29
Создание HTML-редактора .....	38
Регулярные выражения .....	46
<b>Глава 2. Администрирование Windows средствами WMI</b> .....	<b>51</b>
Понятие WMI .....	51
Доступ к объектам WMI .....	53
Сбор данных об аппаратном составе .....	60

Работа с программным обеспечением .....	62
Получение списка установленного программного обеспечения .....	62
Удаление программного обеспечения .....	63
Работа со службами и процессами .....	64
Получение списка сервисов .....	65
Установка приоритета процесса .....	66
Завершение процесса .....	66
Запуск и приостановка выполнения сервиса .....	67
Работа с операционной системой .....	67
Определение информации об ОС .....	69
Выключение компьютера .....	69
Определение разрешения экрана .....	69
Список элементов автозагрузки .....	70
Активизация копии Windows XP/Server 2003 .....	70
Просмотр учетных записей и доменов .....	71
Определение домена компьютера .....	71
Определение роли компьютера .....	72
Определение имени текущего пользователя .....	73
Определение локальных групп компьютера .....	73
Определение списка пользователей и групп .....	73
Работа с системной датой и временем .....	74
Определение даты и времени .....	74
Определение временной зоны .....	75
Работа с журналами событий .....	76
Получение информации о журнале событий .....	77
Копирование и очистка журнала .....	77
Просмотр записей журнала .....	78
Работа с файлами и папками .....	78
Переименование файлов .....	79
Поиск всех файлов по расширению .....	79
Открытие общего доступа к папке .....	80
Работа с принтерами .....	80
Добавление нового сетевого принтера .....	81
Установка принтера по умолчанию .....	82
Работа с заданиями планировщика .....	82
Просмотр заданий в планировщике .....	83
Создание заданий .....	83
Работа с системным реестром .....	84
Чтение значений ключей реестра .....	85
Создание нового раздела .....	86
Создание нового параметра .....	86

Работа с сетью .....	86
Отключение сетевого соединения .....	87
Определение MAC- и IP-адресов.....	87
Использование команды <i>ping</i> .....	88
<b>Глава 3. Использование скриптов WSH.....</b>	<b>89</b>
Что такое WSH.....	89
Объектная модель WSH.....	89
Запуск произвольного VB-скрипта (VBS) .....	91
Выполнение операций с файловой системой .....	92
Получение сведений о дисках.....	93
Получение сведений о папках и файлах .....	94
Проверка существования определенного диска, каталога или файла .....	96
Копирование файлов и папок .....	96
Перемещение файлов и папок .....	97
Удаление файлов и папок.....	97
Создание папок .....	98
Работа с текстовыми файлами .....	98
Получение списка всех доступных дисков, папок и файлов .....	100
Чтение свойств MP3-файлов.....	101
Специальные папки .....	104
Работа с ярлыками Windows.....	106
Создание ярлыков для файлов.....	107
Копирование, перемещение и удаление ярлыков.....	108
Создание ярлыков для URL-адресов.....	108
Работа с системным реестром Windows.....	109
Понятие реестра .....	109
Запись, чтение и удаление ветвей реестра.....	110
Работа с сетевыми ресурсами.....	111
Получение информации о сетевой идентификации .....	112
Работа с сетевыми дисками.....	113
Работа с сетевыми принтерами.....	114
Управление программами.....	115
Управление калькулятором.....	118
Управление Microsoft Word .....	118
Запуск встроенного калькулятора .....	119
Закрытие окна сообщений в системе "1С:Предприятие".....	119
<b>Глава 4. Использование OLE Automation .....</b>	<b>121</b>
Понятие OLE Automation.....	121
Использование "1С:Предприятие" в качестве сервера OLE Automation .....	123
Использование сервера OLE Automation в версии 7.7 .....	123

Подключение к базе данных .....	123
Доступ к объектам базы данных .....	124
Вычисление выражений.....	127
Выполнение операторов .....	127
Использование сервера OLE Automation в версии 8.0 .....	128
Доступ к объектам базы данных .....	131
Использование запросов .....	134
Вызов пользовательских интерфейсов .....	134
Доступ к метаданным.....	136
Использование COM-соединения в версии 8.0 .....	136
Работа с Microsoft Word.....	139
Создание и открытие документов MS Word .....	140
Сохранение, печать и закрытие документов .....	141
Вставка текста и объектов в документ и форматирование текста .....	142
Перемещение курсора по тексту .....	144
Создание таблиц.....	146
Обращение к свойствам документа.....	147
Выгрузка метаданных в MS Word .....	150
Динамическое создание и выполнение макросов MS Word .....	154
Проверка орфографии текстов в MS Word.....	156
Работа с Microsoft Excel.....	159
Запуск MS Excel, создание и открытие рабочих книг.....	159
Сохранение, печать и закрытие рабочих книг MS Excel .....	162
Обращение к листам и ячейкам.....	162
Создание диаграмм .....	165
Выгрузка метаданных в MS Excel .....	166
Создание сводных таблиц MS Excel .....	170
Работа с Microsoft PowerPoint .....	173
Запуск MS PowerPoint и открытие презентаций .....	173
Сохранение, печать и закрытие презентаций .....	174
Демонстрация слайдов .....	176
Работа с Microsoft Visio .....	178
Создание и открытие документов MS Visio .....	179
Сохранение, печать и закрытие документов MS Visio.....	181
Оформление рисунков.....	182
Рисование графических примитивов .....	185
Связывание фигур.....	187
Формирование структуры конфигурации в MS Visio .....	188
Работа с Microsoft Project .....	195
Создание и открытие документов MS Project .....	195
Сохранение, печать и закрытие документов MS Project.....	195

Получение свойств открытых проектов .....	198
Получение списка задач и ресурсов.....	199
Создание и редактирование задач .....	201
Работа с Adobe Photoshop .....	202
<b>Глава 5. Использование ADO .....</b>	<b>205</b>
Понятие ADO .....	205
Соединение с источником данных.....	207
Выполнение SQL-запросов и хранимых процедур .....	212
Обращение к результатам запроса.....	218
Запись и чтение изображений из базы данных.....	223
Работа со структурой базы данных (ADOX) .....	226
Чтение структуры базы данных.....	227
Создание баз данных и их объектов.....	230
Удаление объектов базы данных.....	233
Доступ к данным Microsoft Excel через OLE DB .....	234
Доступ к данным Microsoft Project через OLE DB.....	239
<b>Глава 6. Использование SQL-DMO .....</b>	<b>251</b>
Понятие SQL-DMO.....	251
Получение списка доступных серверов .....	252
Перечисление спецификаций баз данных .....	253
Получение списка таблиц и спецификаций полей .....	255
Получение списка представлений .....	257
Получение списка хранимых процедур .....	257
Изменение структуры баз данных .....	258
Создание и удаление баз данных.....	258
Создание и удаление таблиц и их полей.....	260
Создание SQL-скриптов.....	270
Выполнение SQL-запросов.....	273
Резервирование и восстановление базы данных .....	276
Настройка ограничений доступа к данным.....	278
<b>Глава 7. Команды управления Windows, использующие rundll32 .....</b>	<b>287</b>
Понятие rundll32 .....	287
Запуск элементов Панели управления.....	288
Запуск мастеров .....	295
Работа с Интернетом .....	297
Установка и удаление принтера.....	299
Прочие команды .....	303



<b>Глава 8. Интернет-технологии.....</b>	<b>307</b>
Работа с электронной почтой .....	307
Использование команды <i>mailto</i> .....	307
Использование интерфейса MAPI .....	308
Использование компонента V7Plus.dll .....	312
Использование компонента Rom-Mail.dll.....	315
Использование объекта <i>Почта</i> .....	322
Использование объекта <i>ИнтернетПочта</i> .....	325
Работа с протоколом FTP.....	330
Использование программы ftp.exe .....	330
Использование объекта <i>FTPCoединение</i> .....	337
<b>Глава 9. Использование Active Directory Service Interfaces .....</b>	<b>341</b>
Понятие ADSI .....	341
Формирование дерева объектов WinNT Provider.....	343
Вывод списка всех доменов локальной сети .....	345
Вывод списка пользователей группы .....	346
Создание и удаление пользователей.....	346
Создание и удаление групп пользователей.....	347
Манипулирование учетными записями.....	348
Вывод информации о компьютере и пользователе .....	350
<b>Приложение. Описание компакт-диска.....</b>	<b>357</b>
<b>Предметный указатель.....</b>	<b>363</b>

# Введение

## Зачем нужна эта книга

Предметом книги является широкий спектр вопросов по профессиональному программированию в системе "1С:Предприятие" версий 7.7 и 8.0, а также использование скрытых и неочевидных возможностей системы.

Книга представляет собой практическое руководство, предназначенное для программистов, разработчиков и администраторов системы "1С:Предприятие". Рассматриваются следующие темы: работа с технологиями ActiveX, COM, WMI и WSH, OLE Automation, интеграция с базами данных через интерфейсы ADO и SQL-DMO, работа с Интернетом, электронной почтой и FTP, интеграция с внешними приложениями и многое другое. Отдельное внимание уделено интеграции с Microsoft Word, Excel, PowerPoint, Visio и Project.

Главная цель книги — дать читателю основы понимания того, что возможности системы "1С:Предприятие" не ограничиваются решениями задач для автоматизации документооборота; что в системе можно использовать абсолютно разные технологии программирования, которые активно применяются программистами в других средах и языках программирования.

Книга не является традиционным учебником по программированию в "1С:Предприятии", а сосредотачивает внимание читателя на неявных, скрытых, но не менее важных возможностях платформы, операционной системы и разнообразных СОМ-объектов.

Книга содержит большое количество примеров. Примеры использования различных приемов и технологий приводятся для обеих версий системы (7.7 и 8.0).

## Структура книги

**Глава 1. Применение технологий COM и ActiveX.** Обсуждаются основы технологий COM и ActiveX. Здесь вы найдете краткий обзор возможностей этих технологий и получите всю необходимую информацию. Материал этой главы будет особенно полезен начинающим разработчикам, которые пока еще не работали с технологиями COM и ActiveX.

Теоретическая часть главы подкреплена множеством примеров использования данных технологий. В частности рассмотрены приемы работы с Internet Explorer, Microsoft Winsock, ActiveBarcode, Windows Media Player ActiveX, Microsoft Agent и многие другие.

**Глава 2. Администрирование Windows средствами WMI.** Вы узнаете, каким образом возможно автоматизировать службу системного администратора средствами Windows Management Instrumentation (WMI). Вы научитесь выполнять различные операции с учетными записями пользователей, доменами, операционной системой, файлами, реестром, оборудованием, входящим в локальную вычислительную сеть. Вам полезно будет также узнать о дополнительных функциях, включенных в интерфейс WMI.

**Глава 3. Использование скриптов WSH.** Данная глава описывает возможности стандартного компонента операционной системы Windows Script Host (WSH). В главе подробно рассмотрена объектная модель WSH, с помощью которой возможно работать с файловой системой, системным реестром, специальными папками и ярлыками Windows, ресурсами локальной сети, а также запускать процессы и контролировать ход их выполнения.

Даже если вы уже знакомы с интерфейсом WSH, прочитайте данную главу, чтобы выяснить все его возможности и закрепить знания на подробно разобранных примерах.

**Глава 4. Использование OLE Automation.** Глава посвящена практическому использованию технологии OLE Automation. В главе приводится обзор самых популярных OLE-серверов, таких как Microsoft Word, Excel, PowerPoint, Visio, Project и Adobe Photoshop. Помимо этого рассматриваются возможности использования системы "1С:Предприятие" версий 7.7 и 8.0 в качестве сервера OLE Automation, а также приемы работы с COM-соединением в версии 8.0.

Помимо практической части в главе раскрываются понятия OLE-объекта, контейнера приложения, сервера приложения и другие основополагающие термины и принципы данной технологии.

**Глава 5. Использование ADO.** Эта глава книги полностью посвящена вопросам интеграции разрабатываемых конфигураций с произвольными базами данных при помощи технологии Microsoft ActiveX Data Objects (ADO).

В главе рассматривается объектная модель интерфейса ADO, используемая для соединения с источником данных, для чтения, добавления, удаления и модификации данных.

Помимо ADO в главе содержится описание технологии ADO Extensions for Data Definition and Security (ADOX), которая представляет собой набор объектов, позволяющих манипулировать метаданными в базах данных и управлять объектами, отвечающими за безопасность.

**Глава 6. Использование SQL-DMO.** Данная глава описывает еще одну технологию доступа к базам данных Microsoft SQL Server посредством технологии SQL Distributed Management Objects (SQL-DMO).

В главе приводятся примеры манипуляции базами данных Microsoft SQL Server, таблицами, представлениями, хранимыми процедурами и SQL-запросами. Также в главе приведены примеры работы с сервисными возможностями Microsoft SQL Server, такими как резервирование и восстановление баз данных и настройка ограничений доступа к данным.

**Глава 7. Команды управления Windows, использующие rundll32.** В этой главе узнаете возможности использования утилиты командной строки Rundll32, которая позволяет запускать некоторые команды, заложенные в DLL-файлах.

В данной главе будут приведены примеры запуска элементов Панели управления, работы с Интернетом, принтерами и многое другое.

**Глава 8. Интернет-технологии.** Данная глава посвящена использованию интернет-технологий в конфигурациях, разрабатываемых в системе "1С:Предприятие" версий 7.7 и 8.0.

В главе приводятся практические примеры работы с электронной почтой, включая как объекты доступа к почтовым ящикам посредством интерфейса Messaging Application Program Interface (MAPI), так и объекты, предназначенные для прямой работы с почтовыми серверами.

Помимо этого в главе рассматриваются различные приемы работы с протоколом FTP.

**Глава 9. Использование Active Directory Service Interfaces.** В последней главе книги рассматриваются вопросы работы с технологией Active Directory Service Interfaces (ADSI).

В главе приведены примеры манипуляции службой WinNT Provider, которая позволяет работать с доменами, рабочими станциями, пользователями и группами локальной сети.

## Программные требования

Для запуска приведенных в книге практических примеров необходима установленная система "1С:Предприятие" версии 7.7 или 8.0. Для разработки рекомендуется использовать операционную систему Microsoft Windows 2000 или XP.

Для выполнения отдельных примеров требуется наличие установленных приложений или некоторых COM-объектов, которые располагаются на сопроводительном компакт-диске.

## Описание сопроводительного компакт-диска

В прилагаемом к книге компакт-диске содержатся все исходные тексты примеров, приведенных в книге, а также использованные в книге COM-объекты. Помимо примеров компакт-диск содержит off-line-версию интернет-проекта ERP Group ([www.erpg.ru](http://www.erpg.ru)), основателем которого является автор книги.

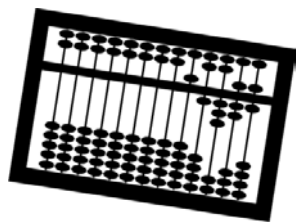
## Источники информации

При написании данной книги использовалось множество интернет-ресурсов, основными из которых являются:

- ❑ <http://www.osp.ru> — издательство "Открытые системы";
- ❑ <http://msdn.microsoft.com> — Microsoft Developer Network;
- ❑ <http://rusproject.narod.ru> — "Русский проект", автор А. Климов;
- ❑ <http://admtech.mrsu.ru> — Мордовский государственный университет им. Н. П. Огарева;
- ❑ <http://avb1c.narod.ru> — домашняя страница Алексея Бажитова;
- ❑ <http://www.erpg.ru> — проект Андрея Михайлова "ERP Group".

Автор книги выражает искреннюю благодарность авторам данных ресурсов.

## Глава 1



# Применение технологий COM и ActiveX

## Технологии COM и ActiveX

Аббревиатура COM расшифровывается достаточно просто. По существу это краткая запись понятия — Component Object Model (компонентная объектная модель). Иногда говорят — "модель COM".

Построение компонентной объектной модели осуществляется при помощи соответствующей информационной технологии. Суть ее заключается в том, что программы строятся из компонентов, которые в свою очередь состоят из объектов. Эти компоненты и объекты могут представлять собой непосредственно исполняемый двоичный код или файлы (DLL, EXE), которые никак не надо "связывать" с проектом, для которого они построены. Их достаточно зарегистрировать в операционной системе, и они будут доступны любой программе, исполняющейся на данной машине. Таким образом, использование их в программе производится без применения операций сборки модуля. Более того, модель COM позволяет вызывать объекты какого-либо компонента из своей программы без указания того, где они расположены. Здесь достаточно знать только имя объекта.

Объекты COM можно создавать на любом языке, способном поддерживать данный интерфейс. После этого обращаться к методам этих объектов можно будет на любом языке, в том числе и на встроенном языке системы "IS:Предприятие", позволяющем осуществлять вызовы данного двоичного интерфейса. Ни объект, ни его клиент не знают (да и зачем им это), на каком языке написаны другие.

Технология COM (в дальнейшем просто — COM) разработана Microsoft, а поэтому первоначально была доступна для программ, работающих под управлением Microsoft Windows 9x и Microsoft Windows NT. Теперь Microsoft предоставляет поддержку COM и для Macintosh. Хотя Microsoft

не поддерживает COM на других операционных системах, этот пробел заполнен третьими фирмами. Несколько компаний, больших и малых, предоставляют реализации COM и основанных на ней технологий для широкого диапазона операционных систем.

Технология *ActiveX* построена на основе компонентов COM. ActiveX — это технология, разработанная фирмой Microsoft с целью стандартизации программных компонентов. Это системная технология, представляющая совокупность средств, с помощью которых объекты, разработанные различными разработчиками на разных языках программирования и работающие в разных средах, могут взаимодействовать друг с другом без какой-либо модификации их исполняемых модулей (двоичных кодов).

Все ActiveX-компоненты можно разделить на два типа:

- ❑ визуальные (например, Microsoft Agent, Progress Control, Slider Control, календарь, таймер и пр.);
- ❑ не воспринимаемые визуально, но обеспечивающие какую-либо функцию интеграции разных по происхождению компонентов.

Преимущества использования технологии ActiveX заключаются в следующем:

- ❑ *быстрое написание программного кода*, которое выражается в том, что разработка приложений становится очень похожей на программирование, в котором используются отдельные "программные кирпичики" (элементы ActiveX);
- ❑ *открытость и мобильность*, являющиеся следствием спецификации данной технологии, недавно переданной в Open Group, как основы открытого стандарта (кроме того, Microsoft совместно с компаниями Metrowerks и Bristol заканчивает реализацию технологий ActiveX для платформ Macintosh и UNIX);
- ❑ *возможность написания приложений с использованием любых средств разработки*, выражающаяся в том, что программные элементы ActiveX могут использоваться в Visual Basic, Visual C++, Borland Delphi, Borland C++, "1С:Предприятие" и других средствах разработки приложений;
- ❑ *большое количество уже существующих программных элементов ActiveX*, причем некоторые из разработанных компонентов являются бесплатными и могут применяться в приложениях независимых разработчиков;
- ❑ *стандартность*, выражающаяся в том, что технология ActiveX основана на широко используемых стандартах Интернета (TCP/IP, HTML, Java), с одной стороны, и стандартах, введенных в свое время Microsoft и необходимых для сохранения совместимости с COM и OLE, с другой.

Как ни странно, при рассмотрении проблемы расширения встроенного языка системы "1С:Предприятие" речь заходит только о внешних компонентах, в основу которых положена идеология COM, но которые должны быть написаны по единому стандарту "1С". Следствием этого является невозможность использования компонентов, которые написаны в средах других языков программирования. В связи с этим, в данной главе будет рассмотрено большое число возможностей использования стандартных COM- и ActiveX-объектов в системе "1С:Предприятие".

В системе "1С:Предприятие" (версия 7.7) новый экземпляр COM-объекта создается с помощью конструкции Идентификатор = СоздатьОбъект("ИмяОбъекта"). Для версии 8.8 оператор СоздатьОбъект заменяется на Новый СОМОбъект. Методы и свойства COM-объектов в дальнейшем становятся доступными через созданный объект.

## Работа с объектом Internet Explorer

COM-объект, предоставляемый браузером Microsoft Internet Explorer, позволяет выполнять множество функций. Например, получение данных с различных Web-сайтов и использование полученной информации в своих целях, автоматическая регистрация сайта во всех поисковых системах сразу, решение задач посылки GET- и POST-запросов и др. Поскольку данный браузер присутствует в любой современной операционной системе семейства Windows, то использование данного компонента является одним из самых универсальных способов получения сторонних HTML-данных.

Рассмотрим ряд приемов использования данного COM-объекта.

### Определение текущего разрешения экрана

Для того чтобы определить текущее экранное разрешение, находясь в браузере, необходимо запустить объект InternetExplorer и определить, как это показано в листинге 1.1, текущее разрешение.

**Листинг 1.1. Определение текущего разрешения экрана для версии 7.7**

```
objIE = СоздатьОбъект("InternetExplorer.Application");  
objIE.Visible = 0; // Отключаем видимость  
objIE.Navigate("about:blank"); // Загружаем пустую страницу  
Пока objIE.readyState < 4 Цикл // Ждем, пока она загрузится  
КонецЦикла;
```



```
oScreen = objIE.document.parentWindow.screen;  
Сообщить ("Текущее разрешение экрана: " + oScreen.width + "x" +  
    | oScreen.height);
```

Приведенный код работает в системе "1С:Предприятие 7.7". Для версии 8.0 необходимо в первой строке заменить СоздатьОбъект на Новый СОМОбъект.

### Внимание

Если в цикле ожидания готовности объекта `objIE` условие `objIE.readyState < 4` заменить на `objIE.busy` (распространенная ошибка), то в Windows 9x приложение будет завершаться с ошибкой примерно один раз из четырех.

Несмотря на всю привлекательность использования Internet Explorer, вышеописанный подход имеет и значительные недостатки, основной из которых — это существенное понижение общей производительности приложения. Для более эффективного определения текущего разрешения экрана рекомендуется использовать объекты WSH или WMI, о которых будет подробно рассказано в последующих главах.

## Анализ HTML-страниц

Среди множества методов и свойств Microsoft Internet Explorer есть весьма ценное свойство `Document`, которое позволяет получить доступ к содержимому объектной модели HTML-документа (т. е. возвращает ссылку на тот же объект, что и строка клиентского JScript: `window.document`). Это позволяет в свою очередь легко и просто разбирать HTML-документы безо всяких регулярных выражений.

Рассмотрим примеры использования Internet Explorer для анализа всех ссылок на HTML-странице, получая информацию с сайта автора [www.erpg.ru](http://www.erpg.ru) (листинги 1.2 и 1.3). Данную информацию можно использовать, например, для автоматического определения наличия необходимой ссылки на свой сайт при обмене кнопками и баннерами (рекламными блоками) с дружественными сайтами.

### Листинг 1.2. Получение всех ссылок на сайте для версии 7.7

```
objIE = СоздатьОбъект ("InternetExplorer.Application");  
objIE.Visible = 0;  
objIE.Silent = 0;  
objIE.Navigate (Адрес);  
Пока objIE.readyState<4 Цикл // Ждем, пока она загрузится
```

```

КонецЦикла;
objDoc = objIE.Document;
AllTags = objDoc.all.tags("A"); // Получаем всю коллекцию ссылок
Для Ном = 0 По AllTags.length-1 Цикл
    Сообщить (AllTags.item(Ном).href);
КонецЦикла;
objIE.Quit(); // Закрываем приложение
objIE = "";

```

### Листинг 1.3. Получение всех ссылок на сайте для версии 8.0

```

objIE = Новый СОМОбъект("InternetExplorer.Application");
objIE.Visible = Ложь; // Отключаем видимость
objIE.Silent = Ложь;
objIE.Navigate(Адрес);
Сообщить ("Идет загрузка страницы. Ждите...");
Пока objIE.readyState<4 Цикл // Ждем, пока загрузится страница
КонецЦикла;
objDoc = objIE.Document;
Для каждого а Из objDoc.all.tags("A") Цикл // Перебираем все ссылки
    Сообщить (a.href);
КонецЦикла;
objIE.Quit(); // Закрываем приложение
objIE = Неопределено;

```

Существует несколько важных отличий в реализации данной задачи на разных версиях системы "1С:Предприятие":

- ❑ создание СОМ-объекта в версии 7.7 происходит с помощью конструкции СоздатьОбъект, а в версии 8.0 — Новый СОМОбъект;
- ❑ конструкция objDoc.all.tags("A") возвращает коллекцию СОМ-объектов с ссылками HTML-страницы, заключенными в тег <A>, но версия 7.7 не позволяет напрямую работать с подобными коллекциями. Поэтому для этой версии сначала определяется количество элементов в коллекции с помощью структуры AllTags.length, а только потом каждый элемент получаем методом item(<НомерЭлемента>). В версии 8.0 обход элементов любой коллекции происходит с помощью конструкции Для каждого...Из...Цикл...КонецЦикла, причем каждый полученный элемент цикла уже будет соответствовать очередному элементу коллекции СОМ-объектов.

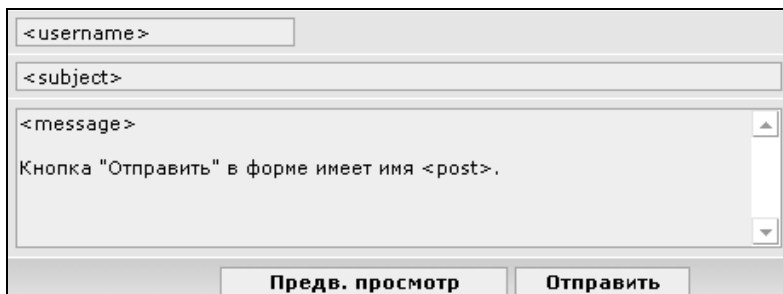
## Работа с POST-запросами в HTML-формах

Данный тип HTTP-запросов чаще всего используется для транслирования сторонней информации в тех случаях, когда для доступа к ней требуется либо идентификация прав доступа (предварительный ввод имени пользователя и пароля), либо автоматическое заполнение определенной HTML-формы и отправки этой информации куда-либо, например, на форум.

Обычно, для передачи данных POST-запроса, во всех популярных языках программирования применяется метод `Navigate` объекта `InternetExplorer` со специальным параметром `postData`. Однако, несмотря на это, его практическое использование в среде "IC:Предприятие" сопряжено с некоторыми трудностями. Дело в том, что данный аргумент должен иметь тип — указатель на структуру `SafeArray`, который не поддерживается системой "IC:Предприятие" версии 7.7.

Поэтому самым простым способом отправить POST-запрос посредством `InternetExplorer.Application` можно считать "эмуляцию" нажатия кнопки **Submit** (управляющая кнопка, используемая для передачи данных формы на сервер) в какой-либо реальной HTML-форме. Этот способ особенно интересен с учетом того, что многие скрипты при обработке POST-запросов, как раз для защиты от подобной автоматизации, учитывают `HTTP_REFERER`, т. е. с какой страницы пришел запрос, и (или) выставляют специальные сессионные переменные или cookie на той странице, где находится форма, посылающая POST-запрос.

Листинги 1.4 и 1.5 демонстрируют примеры подобного программного эмулирования "интерактивного пользователя". При этом программа, написанная для версии 7.7, приведенная в листинге 1.4, демонстрирует заполнение HTML-формы (рис. 1.1) форума, расположенного по адресу <http://forum.erpg.ru>.



The image shows a screenshot of a web form for a forum. It contains three input fields: a text box for the username labeled "<username>", a text box for the subject labeled "<subject>", and a larger text area for the message labeled "<message>". The message field contains the text "Кнопка "Отправить" в форме имеет имя <post>.". Below the message field are two buttons: "Предв. просмотр" (Preview) and "Отправить" (Submit).

Рис. 1.1. Элементы HTML-формы форума

**Листинг 1.4. Заполнение HTML-формы сообщения на форуме для версии 7.7**

```
objIE = СоздатьОбъект("InternetExplorer.Application");
objIE.Visible = 0;
objIE.Silent = 0;
objIE.Navigate("http://erpg.ru/forum/posting.php?mode=reply&t=34");
Пока objIE.readyState<4 Цикл // Ждем, пока она загрузится
КонецЦикла;
objDoc = objIE.Document;
objForm = objDoc.forms(0); // Получаем первую форму в документе
Попытка
    // Заполняем поле формы "Имя пользователя"
    objForm.elements("username").value = СокрЛП(Имя);
Исключение
КонецПопытки;
objForm.elements("subject").value = СокрЛП(Тема);
objForm.elements("message").value = СокрЛП(Описание);
objIE.Visible = 1;
Сообщить("HTML-форма форума заполнена. Для отправки сообщения необходимо
| нажать кнопку Отправить в Internet Explorer.");
```

К сожалению, версия 7.7 не позволяет выполнить эмуляцию нажатия определенной кнопки HTML-формы из-за того, что система не опознает метод `click()` элемента формы, поэтому для окончательной отправки сообщения на форум пользователю необходимо вручную нажать кнопку **Отправить** в окне Internet Explorer. Но если на форме имеется только одна кнопка, используемая по умолчанию, то для программной эмуляции ее нажатия можно использовать метод `submit: objForm.submit()`.

Версия 8.0 системы "1С:Предприятие" более корректно работает с COM-объектами, поэтому процедура отправки сообщения на форум полностью автоматизирована. Рассмотрим процедуру автоматического заполнения той же формы форума и эмуляции нажатия кнопки **Отправить** (листинг 1.5).

**Листинг 1.5. Отправка сообщений на форум (метод Post) для версии 8.0**

```
objIE = Новый СОМОбъект("InternetExplorer.Application");
objIE.Visible = Ложь; // Отключаем видимость
objIE.Silent = Ложь;
objIE.Navigate("http://erpg.ru/forum/posting.php?mode=reply&t=34");
```

```

Сообщить ("Идет загрузка страницы. Ждите...");
Пока objIE.readyState<4 Цикл // Ждем, пока загрузится страница
КонецЦикла;
objDoc = objIE.Document; // Получаем объект HTML-документ
objForm=objDoc.forms(0); // Получаем первую форму в документе
Попытка
    // Заполняем поле формы "Имя пользователя"
    objForm.elements("username").value = Имя;
Исключение
КонецПопытки;
objForm.elements("subject").value = Тема;
objForm.elements("message").value = Описание;
objForm.elements("post").click(); // Нажимаем кнопку "Отправить"
Сообщить ("Ваше сообщение отправлено на форум по адресу: " +
|         "http://erpg.ru/forum/viewtopic.php?t=34");
objIE.Quit(); // Закрываем приложение
objIE = Неопределено;

```

По-моему, приведенный код должен быть понятен без дополнительных объяснений, за исключением использования конструкции Попытка...Исключение... КонецПопытки при заполнении имени пользователя на форме (элемент с именем `username`). Это делается для того, чтобы предотвратить ошибку системы в том случае, если поле `username` не существует. На форуме используется система авторизации пользователей, и если пользователь был авторизован ранее, то поле `username` не появляется на форме.

## Работа с объектом Microsoft Winsock

*Microsoft Winsock* (`MSWinsock`) — это объект ActiveX, который обеспечивает простой доступ к *TCP* или *UDP* сетевым операциям. Чаще всего его используют для того, чтобы подсоединиться к удаленному компьютеру и обмениваться данными в обоих направлениях.

Объект `MSWinsock` имеет следующие свойства:

- ❑ `ByteReceived` — возвращает количество полученных данных, находящихся в данный момент в принимающем буфере;
- ❑ `LocalHostName` — возвращает имя местного компьютера;
- ❑ `LocalIP` — возвращает IP-адрес местного компьютера в формате (xxx.xxx.xxx.xxx);
- ❑ `LocalPort` — возвращает или устанавливает местный порт использования;

### Примечание

Для клиента порт используется для отправки данных. Укажите порт 0, если приложение не требует специфического порта. В этом случае элемент управления выберет случайный порт. После установления соединения этот местный порт используется для TCP-соединений. Для сервера этот порт является принимающим. Если указан порт 0, будет использоваться случайный порт. После вызова функции `Listen` свойство содержит порт, который был выбран.

- `Protocol` — возвращает или устанавливает протокол, используемый `Winsock` (либо TCP, либо UDP);
- `RemoteHost` — возвращает или устанавливает имя или IP-адрес удаленного компьютера, с которым нужно обмениваться данными (можно указать IP-адрес, имя компьютера или FTP-адрес);
- `RemoteHostIP` — возвращает IP-адрес удаленного компьютера;
- `RemotePort` — возвращает или устанавливает порт удаленного компьютера для соединения (порты, устанавливаемые по умолчанию: 80 — HTTP, 26 — FTP);
- `State` — возвращает состояние элемента управления в числовом выражении.

## Определение IP-адреса локального компьютера

Для определения IP-адреса локального компьютера используется свойство `LocalIP` объекта `MSWinsock`. В листинге 1.6 приведен фрагмент программы, показывающий вариант реализации этого действия.

### Листинг 1.6. Определение IP-адреса локального компьютера

```
Winsock = Новый COMОбъект("MSWinsock.Winsock");  
Сообщить(Winsock.LocalIP);  
Winsock = Неопределено;
```

## Определение IP-адреса удаленного компьютера

Для определения IP-адреса удаленного компьютера используется свойство `RemoteHostIP` объекта `MSWinsock`. В листинге 1.7 приведен фрагмент программы, показывающий вариант реализации этого действия.

**Листинг 1.7. Определение IP-адреса удаленного компьютера**

```
Winsock = Новый СОМОбъект("MSWinsock.Winsock");  
Winsock.Connect(ИмяСервера, 139);  
Пока Winsock.State = 4 Цикл // Ждем, пока произойдет соединение  
КонецЦикла;  
Сообщить(Winsock.RemoteHostIP);  
Winsock = Неопределено;
```

Код довольно простой, но в нем есть некоторые особенности:

- ❑ для TCP-соединения с удаленным сервером используется порт 139, который, как правило, доступен всегда;
- ❑ метод `Connect` является асинхронным, т. е. возврат из него осуществляется сразу же, не дожидаясь установления соединения;
- ❑ для ожидания соединения с удаленным компьютером в цикле анализируется состояние соединения `State`.

## Создание штрихкодов с использованием **ActiveX ActiveBarcode**

Для эффективного учета движения товаров в организации необходима технология, позволяющая присваивать каждому товару уникальный код и обеспечивать быстрое его считывание при минимальных ошибках. Именно этим условиям и удовлетворяет технология штрихового кодирования товаров. На данный момент насчитывается около 20 форматов штрихкодов (включая их модификации), различающихся, в основном, областями применения. Некоторые форматы имеют механизм контроля корректности, заключающийся в вычислении одной части кода по другой.

Безусловно, система "1С:Предприятие" позволяет использовать технологию штрихкодирования при автоматизации предприятия с использованием входящего в поставку многих конфигураций ActiveX-элемента — `ActiveBarcode`. Он позволяет формировать образ штрихкода непосредственно в "1С:Предприятии". Этот элемент поддерживает практически все известные форматы штриховых кодов, управление цветом и шрифтом выводимого штрихкода, автоматическое определение формата кода, а также вычисление контрольных цифр.

Компонент `ActiveBarcode` имеет множество свойств, методов и событий. Рассмотрим основные свойства, которые чаще всего используются:

- ❑ `Text` — текст, выводимый под штрихкодом (обычно номер);
- ❑ `Type` — тип штрихкода (полный список кодов можно найти на сайте производителя <http://www.activebarcode.com>);
- ❑ `AutoType` — признак автоматического определения типа исходя из номера штрихкода;
- ❑ `ShowText` — признак включения или отключения вывода текста под штрихкодом;
- ❑ `ForeColor` — цвет текста штрихкода;
- ❑ `BackColor` — цвет фона;
- ❑ `Font` — шрифт текста штрихкода.

Фактически данный компонент представляет собой файл `Barcode.ocx`, который может располагаться где угодно, но перед использованием его необходимо зарегистрировать в системе командой `regsvr32 Barcode.ocx`.

`ActiveBarcode` является визуальным компонентом, т. е. его можно разместить в диалоге или таблице. Однако следует заметить, что использование данного ActiveX-компонента несколько отличается в разных версиях системы "1С:Предприятие". Основное различие состоит в том, что в версии 7.7 невозможно размещать ActiveX-компоненты на формах (разрешено только в табличных документах). В версии 8.0 таких ограничений нет.

Есть еще одна крайне неприятная особенность версии 7.7. Она заключается в том, что в этой версии нельзя штатными средствами вставить данный элемент в табличный документ. Чтобы все-таки это сделать, необходимо вставить `ActiveBarcode` в любом другом приложении (например в MS Word, "1С:Предприятие" 8.0 и др.), а потом через буфер обмена перенести его в табличный документ 7.7. Далее, для динамического вывода нужного номера на штрихкоде надо в свойствах данного объекта прописать необходимые свойства объекта (рис. 1.2).

Для вывода табличного документа используется стандартный код "1С:Предприятия" (листинг 1.8).

#### Листинг 1.8. Вывод табличного документа для версии 7.7

```
Таблица.ИсходнаяТаблица ("Таблица");  
Таблица.ВывестиСекцию ("ШтрихКод");  
Таблица.Показать ("Штрихкод");
```





Рис. 1.2. Настройка параметров динамического формирования ActiveBarCode

В данном примере предполагается, что:

- ❑ элемент ActiveBarCode в табличном документе находится в секции ШтрихКод;
- ❑ переменные Номер, Тип и Показывать, введенные в свойствах элемента, объявляются и заполняются в коде до вывода таблицы (например, из диалога или справочника).

В версии 8.0 системы "1С:Предприятие" все намного проще. В этом случае ActiveX-элементы можно устанавливать как на форму (**Форма | Вставить ActiveX** и затем выбрать ActiveBarCode), так и в табличный документ (**Таблица | Рисунки | Вставить объект**).

Так, если элемент ActiveBarCode находится на форме и имеет имя ШтрихКод, то для его формирования достаточно написать код, приведенный в листинге 1.9.

#### Листинг 1.9. Формирование штрихкода средствами ActiveBarCode для версии 8.0

```

ЭлементыФормы.ШтрихКод.Туре = Тип;
ЭлементыФормы.ШтрихКод.Текст = Номер;
ЭлементыФормы.ШтрихКод.ShowText = Показывать;
ЭлементыФормы.ШтрихКод.AutoType = Автоопределение;

```

В результате выполнения этого кода на форме появится необходимый номер со штрихкодом (рис. 1.3).



Рис. 1.3. Результат формирования штрихкода для версии 8.0

## Использование ActiveX Windows Media Player

ActiveX-компонент Microsoft Windows Media Player позволяет просматривать видео, прослушивать аудиофайлы, а также просматривать рисунки различных форматов (JPG, BMP и др.). Данный компонент бывает очень полезен, когда необходимо просмотреть определенную мультимедийную информацию о товаре или поставщике.

Существует несколько версий Windows Media Player. Для определенности будем рассматривать версию 9.0, которую можно загрузить на сайте производителя по адресу <http://www.microsoft.com/windows/windowsmedia/download/default.asp>.

### Примечание

К сожалению, в "1С:Предприятие" версии 7.7 невозможно использование данного компонента, поэтому все дальнейшее рассмотрение будет актуально только для восьмой версии платформы.

Для работы Windows Media Player в системе "1С:Предприятие" необходимо выполнить всего два действия:

1. Для размещения ActiveX-компонента на форме в меню **Форма** следует выбрать пункт **Вставить ActiveX**. В появившемся списке выбрать **Windows Media Player**.

2. Написать код, который бы передавал файл с мультимедийной информацией в ActiveX-компонент (листинг 1.10).

#### Листинг 1.10. Запуск мультимедиафайла в ActiveX Windows Media Player

Попытка

```
ЭлементыФормы.WMP.settings.autoStart = True; // Автовоспроизведение
ЭлементыФормы.WMP.URL = ИмяФайла;
```

Исключение

```
Предупреждение(ОписаниеОшибки(), 10, "Ошибка");
```

КонецПопытки;

В данном примере применена конструкция `Попытка...Исключение...КонецПопытки` для предотвращения ошибок, связанных с отсутствием в системе данного компонента. Элемент формы `WMP` как раз и является ActiveX-компонентом.

С помощью ActiveX Windows Media Player можно также просматривать свойства файла (`tag`, `Bitrate` и пр.). В листинге 1.11 демонстрируется пример просмотра свойств произвольного файла.

#### Листинг 1.11. Получение свойств файла с помощью ActiveX Windows Media Player

Попытка

```
ЭлементыФормы.WMP.settings.autoStart = False;
ЭлементыФормы.WMP.URL = ИмяФайла;
```

```
// Получаем объект, предназначенный для работы с выбранным файлом
CM = ЭлементыФормы.WMP.currentMedia;
```

```
// Получение значения свойства по имени
```

```
Сообщить("Полное имя файла: " + CM.getItemInfo("SourceURL"));
```

```
// Получение всего списка свойств
```

```
Для Ном = 0 по CM.attributeCount-1 Цикл
```

```
ИмяСвойства = CM.getAttributeName(Ном);
```

```
ЗначениеСвойства = CM.getItemInfo(CM.getAttributeName(Ном));
```

```
Сообщить(ИмяСвойства + " = " + ЗначениеСвойства);
```

```
КонецЦикла;
```

Исключение

```
Предупреждение(ОписаниеОшибки(), 10, "Ошибка");
```

КонецПопытки;

# Использование помощников Microsoft Agent

## Понятие MS Agent

*Microsoft Agent* — это набор программных средств, которые поддерживают воспроизведение интерактивных анимированных персонажей. Разработчики могут использовать эти персонажи для создания различных интерактивных приложений, справочных систем, в том числе и в системе "1С:Предприятие".

Также Microsoft Agent предлагает поддержку распознавания голосовой информации (на основе Microsoft Speech Application Programming Interface, SAPI 4.0), позволяя управлять приложениями на уровне голосовых команд. Персонажи могут воспроизводить синтезированную речь (text-to-speech, TTS), аудиофрагменты или текст.

Персонажи отображаются в собственных окнах, что делает интерфейс более удобным и гибким. В состав Microsoft Agent входит компонент ActiveX, доступный из системы "1С:Предприятие".

Эта технология напрямую используется в MS Office 2000 и включена в состав ядра Windows 2000. Для установки Microsoft Agent в Windows 98 и Windows NT 4.0 требуется загрузить соответствующее программное обеспечение с Web-узла фирмы Microsoft (<http://www.microsoft.com/msagent/>).

В настоящее время на сайте Microsoft доступны четыре стандартных персонажа, которые можно использовать. Это маг — Merlin, джин — Genie, робот — Robby и попугай — Peedy (рис. 1.4).

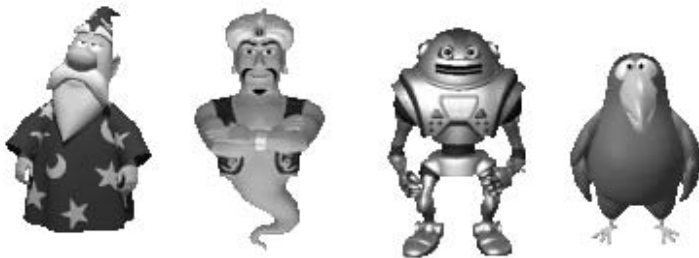


Рис. 1.4. Основные персонажи Microsoft Agent

Кроме стандартных персонажей в Интернете существуют персонажи, разработанные другими разработчиками на любой вкус.

**Примечание**

На том же сайте Microsoft можно загрузить Microsoft Text-To-Speech для того, чтобы "агенты" воспроизводили речь (русский голосовой модуль — Lernout&Hauspie TTS3000), а также редактор "агентов" и официальную документацию Microsoft. Так, при написании материала данной главы были использованы Microsoft Agent Control 2.0 и персонаж Merlin.

## Практическое применение MS Agent

Microsoft Agent очень удобно применять в приложениях, где необходимо выводить какие-либо сообщения, подсказки и указания пользователю, но не хочется ограничиваться штатными методами системы. Microsoft Agent позволяет все это сделать красиво, удобно и необычно. Наверняка работать со столь дружественным интерфейсом, как показано на рис. 1.5, будет приятно любому пользователю.

В системе "1С:Предприятие" доступ к объекту Microsoft Agent создается с помощью конструктора СоздатьОбъект("Agent.Control.2"), для версии 7.7, и Новый СОМОбъект("Agent.Control.2"), для версии 8.0. После создания объекта минимальное, что нужно сделать, — это подключиться к созданному объекту, загрузить необходимый персонаж и вывести его на экран (листинг 1.12).

### Листинг 1.12. Создание объекта MS Agent и вывод персонажа

Попытка

```
// Устанавливаем ссылку на объект
Agent = Новый СОМОбъект("Agent.Control.2");
```

Исключение

```
Сообщить("MS Agent не установлен!");
Сообщить("Загрузить MS Agent можно с сайта Microsoft —
|      http://www.microsoft.com/msagent/");
```

Возврат;

КонецПопытки;

```
Agent.Connected = Истина;           // Подключаемся к объекту
Agent.Characters.Load("Merlin", "Merlin.acs"); // Загружаем персонаж
Agent.Characters("Merlin").Show(); // Выводим на экран
```

Рассмотрим код, приведенный в листинге 1.12, более подробно.

Прежде чем приступить к анимации персонажа, необходимо загрузить его, используя метод Load, и в качестве аргументов передать два параметра: иден-

тификатор персонажа и полное имя файла персонажа или его HTTP-адрес. Microsoft Agent по умолчанию ищет персонажи в каталоге %WinDir%\MSAgent\Chars (где %WinDir% — это переменная окружения, которая указывает путь к каталогу операционной системы Microsoft Windows). Если файл вашего персонажа находится в другом месте, указывайте для него полный путь. Microsoft Agent поддерживает два формата для хранения данных доступных анимаций: одиночный файл (ACS) для локального применения и множественный формат (ACF, ACA), который хранит отдельные виды анимаций и используется для загрузки анимации отдельно через HTTP-сервер.

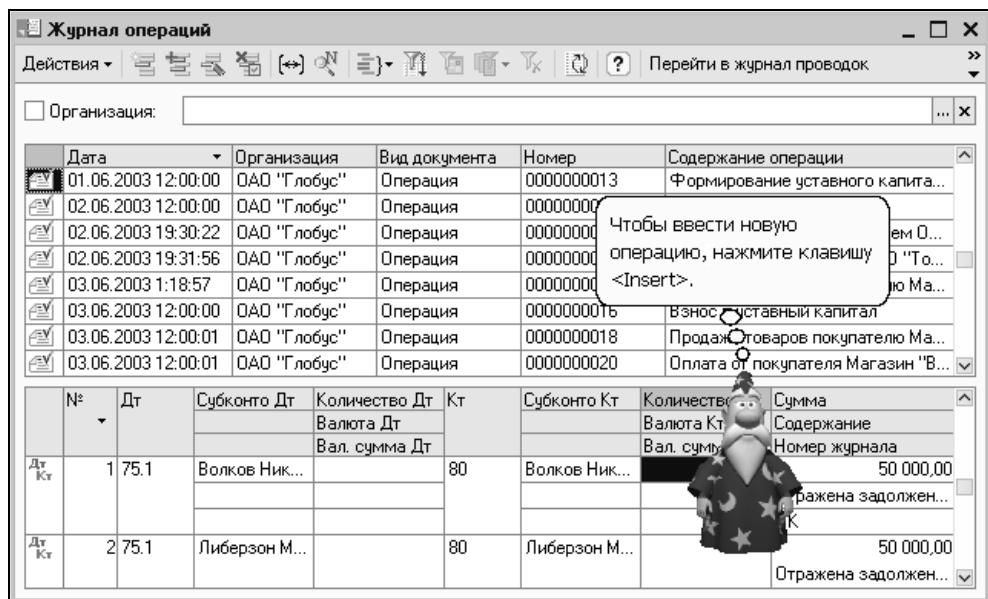


Рис. 1.5. Пример вывода сообщения Microsoft Agent

Приложение может загрузить только единственный экземпляр заданного персонажа. Но при этом можно использовать одновременно разные персонажи.

Метод Show выводит заданный персонаж на экран с одновременным проигрыванием соответствующей анимацией и не имеет параметров. Для скрытия персонажа существует метод Hide, который тоже параметров не имеет.

Вместо загрузки заданного персонажа, жестко определяя его имя, вы можете загрузить персонаж по умолчанию (листинг 1.13). Метод ShowDefaultCharacterProperties объекта Agent выводит панель настройки свойств для выбора персонажа по умолчанию. В качестве аргументов в метод