

Модификация дистрибутивов на основе Windows Installer



Иван Коробко

Возможность модификации дистрибутивов на основе Windows Installer всегда прельщала системных администраторов и специалистов системной поддержки. Рассмотрим один из компонентов Windows Installer Resource Kit – *orca.exe*. Эта утилита специально разработана для решения задач такого рода.

Решение задач автоматической установки программного обеспечения не ограничивается использованием ключей системы инс-

талляции Windows Installer. Получив возможность изменить пакет инсталляции, доступ к базе данных всех параметров дистрибутива, администра-

тор может создавать пакеты для автоматической установки, не используя внешние файлы, содержащие различные настройки.

Уделим внимание программному управлению базой данных, хранящейся в msi-файлах.

Windows Installer

Это сервис установки и конфигурирования программных продуктов, который входит в состав операционной системы. Также он может устанавливаться как пакет обновления или отдельный дистрибутив.

В настоящее время используется Windows Installer 4.5, который можно бесплатно загрузить с сайта Microsoft в виде заплатки (hotfix) KB842288 – <http://download.microsoft.com/download/2/6/1/261fca42-22c0-4f91-9451-0e0f2e08356d/WindowsXP-KB942288-v3-x86.exe>.

Windows Installer, как и любое другое быстрое исправление, поддерживает ряд ключей командной строки, обеспечивающих управление автоматической установкой пакета:

- **/norestart** – не перезагружать компьютер после установки обновления;
- **/quiet** – выполнить установку в «тихом» режиме;
- **/help** – вывод справки на экран, содержащей ключи и их краткое описание.

Файл msisexec.exe

Двойным щелчком мыши по файлу с расширением *.msi осуществляется запуск утилиты:

```
%SystemRoot%\System32\msisexec.exe /i *.msi
```

В таблице 1 приведен список часто используемых параметров командной строки, обеспечивающих автоматическую установку приложения.

Обычно для установки приложения в автоматическом режиме используют команду:

```
%SystemRoot%\System32\msisexec.exe /fa /qn /jm *.msi
```

где *.msi – полный путь к файлу с расширением MSI, например:

```
c:\Windows\system32\msisexec.exe /fa /qn /jm "acdsee10.msi"
```

Windows Installer SDK

Пакет утилит, который можно загрузить бесплатно с сайта компании Microsoft,

Таблица 1. Параметры командной строки msisexec.exe

Параметр	Описание
/i	Установка приложения в обычном режиме
/f [p o e d c a l u m s v]	Переустановка приложения. Если задан только ключ /f, то имеется в виду команда /frcsm: <p>p – восстановление удаленных файлов; o – обновление до более новой версии; e – полная переустановка приложения той же версии; d – восстановление удаленных файлов и перезапись файлов более старой версии; c – восстановление удаленных файлов или перезапись файлов, если CRC (контрольная сумма) установленных файлов не совпадает с CRC файлов в дистрибутиве; a – переустановка всех файлов; u – перезапись всех ключей реестра, касающихся данного приложения в ветвях HCKU, HCU; m – перезапись всех ключей реестра, касающихся данного приложения в ветви HCLM; s – перезапись всех ярлыков</p>
/a	Выполнение административной установки приложения
/x	Деинсталляция продукта
/j [u m g]	Управление публикацией программного продукта: <p>u – публикация для текущего пользователя; m – публикация для всех пользователей; g – идентификатор языка (управление установкой языкового интерфейса)</p>
/q [n b l r f]	Определение уровня пользовательского интерфейса во время установки: <p>n – отсутствие интерфейса; b – базовый интерфейс (по умолчанию); r – сокращенный интерфейс; f – полный пользовательский интерфейс</p>

по адресу – <http://download.microsoft.com/download/7/c/4/7c426dfc-46e2-4ded-bab4-3b33600ad7d1/msi45sdk.msi>, используется для различного рода манипуляций с MSI-файлами.

По умолчанию каталог установки пакета Windows Installer SDK – %ProgramFiles%\Windows Installer 4.5 SDK.

В его комплекте несколько утилит, в том числе дистрибутив программы ORC, расположенный в папке Tools.

Orca.msi

Программа orca.exe, дистрибутив которой находится в папке %ProgramFiles%\Windows Installer 4.5 SDK\Tools, предназначена для модификации *.msi-файлов. С помощью этой утилиты можно не только управлять устанавливаемыми компонентами, но и указать серий-

ный номер продукта, изменить выводимые сообщения и др.

Установка orca.exe осуществляется в папку %:ProgramFiles%\Orca запуском файла orca.msi.

Orca.exe

Запуск программы осуществляется через «Пуск → Программы → orca.exe» (см. рис. 1). После успешного запуска приложения необходимо открыть msi-файл, который необходимо модифицировать. В данном случае в качестве опытного образца использовать дистрибутив AcdSee10 – файл ACDSee 10 Photo Manager.msi, находящийся в каталоге c:\ACDSee.

Замечание: дистрибутив ACDSee 10 представляет собой *.exe-файл – архив, который при запуске распаковывает msi-файл в папке C:\Documents

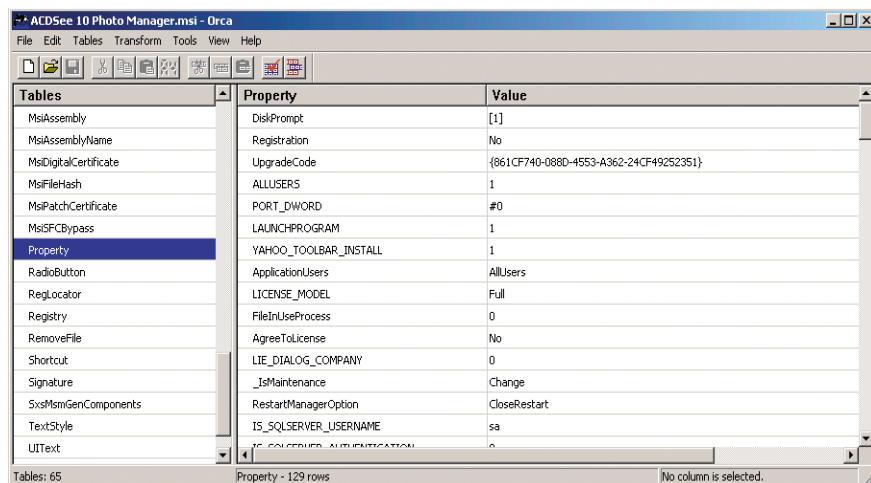


Рисунок 1. Внешний вид утилиты orca.exe

Feature	Feature_Parent	Title	Description	Dis...	Level	Directory	Attribu...
Binary							
CheckBox							
ComboBox							
Component							
Condition							
Control							
ControlCondition							
ControlEvent							
CreateFolder							
CustomAction							
Dialog							
Directory							
DrLocator							
Error							
EventMapping							
Feature							
FeatureComponents							
File							
HelpFile							
HelpFileToNamespace							
Media	ImageEncodingDecodingPlugIns	ID_Media Plug-in	ID_Media Plug-in	28	1		8
Effects	VCDWizard	NewFeature1		0	1	INSTALLDIR	24
Welcome	VCDWizard	NewFeature1		0	1	INSTALLDIR	24
ImageEncodingDecodingPlugIns		Image Plug-ins	Plug-ins for decoding and encodin...	20	1		8
ACDSee		ACDSee 10 Photo Manager	The easy way to organize, enhan...	2	1	INSTALLDIR	24
AlbumGenerator	ACDSee			0	1	INSTALLDIR	8
BuyNowACDSeeTrial	ACDSee			0	1	INSTALLDIR	8
DevDetect		ACDSee Device Detector	USB Camera/Device Detector	10	10	INSTALLDIR	8
VCDWizard	MediaSupport	NewFeature1		0	1	INSTALLDIR	24
ACDSeeInTouch	ACDSee			0	1	INSTALLDIR	24
ACDSeeLicenseManager	ACDSee			0	1	INSTALLDIR	24
ACDSeeMFC71	ACDSee			0	1		24
ACDSeeShowroom	ACDSee	ACDSee 10 Showroom	ACDSee 10 Showroom	5	1	INSTALLDIR	8
ACDSeeStd	ACDSee			0	1	INSTALLDIR	24
CommonVideoComponents	VCDWizard	NewFeature1		0	1	INSTALLDIR	24
ACDV	CommonVideoComponents	NewFeature1		0	1	INSTALLDIR	24
ArchivePlugIns		Archive Plug-ins	Plug-ins for encoding and decodin...	12	1		8
ACE	ArchivePlugIns	ACE Archive ExtractorPlug-in	ACE Decoder	2	1		8
ARJ	ArchivePlugIns	ARJ Archive Extractor Plug-in	ARJ Decoder	4	1		8
Adobe	ImageEncodingDecodingPlugIns	Adobe Photoshop File Export Plug-In Abs...	Adobe Photoshop File Export Plu...	2	1		8

Рисунок 2. Создание новых полей в таблице

and Settings\%username%\Local Settings\Application Data\Downloaded Installations\{0051E50E-CECD-4E04-A71A-B8C49733542A}\ACDSee 10 Photo Manager.msi. Именно этот файл необходимо открыть утилитой orca.exe.

Orca.exe рассматривает msi-файл как SQL-базу данных, где msi-файл – это база данных, левая колонка таблицы (см. рис. 2), в которой хранятся различные настройки и списки. Правая часть – список полей выбранной таблицы с соответствующими значениями.

Структура msi-файлов

В базе данных пакета среди множества таблиц, которые можно объединить в группы, выделим самые важные из них:

- базовые таблицы;
- файловые таблицы;
- таблицы информации о программе.

Базовые таблицы

К базовым таблицам относят таблицы, описывающие параметры работы инсталлятора, устанавливаемые компоненты. Группа состоит из 11 связанных таблиц, краткое описание которых приведено в таблице 2.

Файловые таблицы

Эта группа таблиц содержит информацию обо всех файлах, входящих в комплект поставки программного продукта. Большая часть этих файлов перечислена в таблице File. Хотя таблица Directory не входит в эту группу, она тесно связана с ней, так как отражает структуру каталогов приложения. В таблице 3 приведен список файловых таблиц и их краткое описание.

Таблицы информации о программе

Таблицы этой группы содержат важную информацию о пакете инсталля-

ции, используемую на протяжении всего процесса инсталляции. В таблице 4 приведена информация о назначении таблиц.

Доступ к msi-файлам программным способом

Для обеспечения программного управления базой данных, хранящейся в msi-файле, используется шаблон, приведенный в листинге 1. Необходимо отметить, что шаблон чтения данных отличается от шаблона коррекции данных. О нем будет рассказано отдельно.

Листинг 1. Шаблон доступа к базе в msi-файле

```
Set obj = CreateObject ("WindowsInstaller.Installer")
Set base = obj.OpenDatabase (MSI,Mode)
base.OpenView(SQL-query) .Execute
base.Commit
```

В приведенном шаблоне в первой строке осуществляется подключение к COM-объекту, с помощью которого осуществляется доступ к базе данных.

На следующей строке осуществляется подключение к базе данных с помощью метода OpenDatabase, который имеет два аргумента. Первый из них – имя базы, ко-

Таблица 2. Базовые таблицы MSI Installer

Имя таблицы	Краткое описание
Feature	Содержит список всех функций программного продукта
Condition	Содержит описание условий, определяющих порядок установки каждой функции, описанной в таблице
FeatureComponents	Связывает функции с компонентами
Component	Содержит список всех компонентов приложения
Directory	Содержит список всех каталогов, необходимых для инсталляции
PublishComponent	Содержит список функций и компонентов, публикуемых для использования в других приложениях
Assembly	Задаёт установки для сборок .NET Framework CLR и Win32
AssemblyName	Задаёт схему для именования сборок .NET Framework CLR и Win32
Complus	Содержит информацию, необходимую для установки приложений COM+
IsolatedComponent	Связывает компонент, заданный в столбце Component_Application (обычно .exe), с компонентом, заданным в столбце Component_Shared (обычно .dll)
Upgrade	Содержит информацию для значительных обновлений программного продукта

торым является полный путь к msi-файлу. Второй – режим доступа к базе данных:

- 0 – только чтение;
- 1 – чтение / запись.

На третьей – реализовано выполнение задаваемого запроса. SQL-запрос задается с помощью метода OpenView, а выполнение запроса – с помощью метода Execute.

Напомним, что в SQL-запросе регистр названия полей и имен таблиц имеет значение.

В последней строке осуществляется запись сделанных изменений в базе с помощью команды Commit.

Работа с таблицами в orca.exe

Рассмотрим подробнее структуру таблиц, содержащихся в msi-файле. Orca – мини-редактор SQL-таблиц, хранящихся как составная часть дистрибутива.

С полями таблиц баз можно совершать следующие действия:

- чтение существующих полей;
- создавать новые поля;
- удалять поля;
- изменять поля.

Чтение существующих полей

Для реализации шаблона поиска используется шаблон (см. **листинг 2**), отличный от приведенного в **листинге 1**. Это объясняется тем, что запись, удаление, замена осуществляются у единичного поля, а вывод на экран осуществляется для всех полей.

Трансформация листинга осуществляется в третьей строке **листинга 1**. Она разбивается на две подстроки – это строки 3 и 4 в **листинге 2**. Здесь разносятся методы OpenView и Execute.

Далее осуществляется чтение массива строк. Обратите внимание на то, что в запросе SELECT поля могут быть перечислены только по именам. Значение «*» в данном случае не работает. А во время чтения данных необходимо указывать нумерацию полей (Record.StringData(1)), начиная с единицы (1).

В **листинге 2** приведен пример чтения всех полей (Property, Value) таблицы Property файла ACDSee 10 Photo Manager.msi:

Листинг 2. Чтение полей в указанной таблице

```
Set obj = CreateObject("WindowsInstaller.Installer")
Set ob = obj.OpenDatabase ("C:\ACDSee\ACDSee 10 Photo Manager.msi", 0)
set View=obj.OpenView("SELECT Property, Value FROM `Property`")

View.Execute
Do
    Set Record = View.Fetch
    If Record Is Nothing Then Exit Do
```

Таблица 3. Файловые таблицы MSI Installer

Имя таблицы	Краткое описание
File	Представляет собой список файлов, которые будут распакованы в ходе инсталляции. Эта таблица связана с таблицей Directory, в которой зафиксирована структура каталогов устанавливаемого приложения и таблица Component, с помощью которой обеспечивается связь с компонентами
RemoveFile	Содержит список файлов, которые необходимо удалить при выполнении операции RemoveFiles
Font	Включает в себя список шрифтов, которые необходимо зарегистрировать в операционной системе
SelfReg	Содержит список саморегистрирующихся модулей. Installer не регистрирует EXE-файлы
Media	В таблице описывается набор дисков инсталляции
BindImage	Содержит информацию о привязках исполняемых файлов или DLL
MoveFile	Содержит список файлов, которые необходимо перенести во время инсталляции из исходного каталога в заданный каталог
DuplicateFile	Содержит список дублируемых файлов либо в другой каталог с тем же именем, что и исходный файл, либо в тот же каталог, но с другим именем
Environment	Используется для задания переменных окружения
Icon	Хранит файлы иконок. Каждая иконка этой таблицы во время инсталляции копируется в отдельный файл на диске
MsiFileHash	Эта таблица хранит 128-разрядное хэш-значение для исходных файлов в пакете инсталляции

Таблица 4. Файловые таблицы программы

Имя таблицы	Краткое описание
Property	Хранятся все свойства пакета инсталляции
Binary	Содержатся двоичные данные для иконок, растров и т. п. Также здесь хранятся данные для пользовательских операций
Error	Используется для поиска шаблонов форматирования при обработке ошибок. Installer имеет свой собственный механизм обработки ошибок
Shortcut	Здесь хранится вся информация, необходимая для создания файловых ярлыков
ReserveCost	Эта таблица содержит информацию о необходимом дисковом пространстве для каждого компонента приложения

```
Wscript.Echo Record.StringData(1) + `
vbTab+vbTab+Record.StringData(2)

Loop
Set View = Nothing
```

Создание новых полей

Создание новых полей используется достаточно часто, например, для того чтобы записать в дистрибутив программы ее серийный номер. Для этого необходимо добавить в таблицу Property поле PIDKEY с соответствующим значением, содержащим серийным номер продукта. Чтобы добавить новое поле, необходимо установить курсор на нужную таблицу (см. **рис. 3**) и, нажав в произвольной ее части правую кнопку мыши, выбрать пункт Add Row или нажать комбинацию клавиш <CTRL+R>. В появившемся диалоговом окне будет приведен список доступных полей. Необходимые значения можно присвоить в этом же окне.

Аналогичную операцию можно выполнить программным способом (см. **листинг 3**). Для этого необходимо вызвать объект WindowsInstaller.Installer. На основе шаблона, приведенного в **листинге 1**, сформируем **листинг 3**. В качестве второго параметра метода OpenDatabase необходимо указать 1, поскольку выполняется операция записи данных.

Самым сложным в данном листинге правильно сформировать запрос. Особенность запроса INSERT INFO заключается в том, что в первых по счету скобках, в которых указывается название полей, имя поля должно быть задано с родителем. Все имена параметров в запросе указы-

ваются без кавычек; все присваиваемые значения – в кавычках и, наконец, названия полей и таблиц чувствительны к регистру.

Листинг 3. Создание нового атрибута в таблице

```
Set a = CreateObject("WindowsInstaller.Installer")
Set b = a.OpenDatabase ("C:\acdsee\ACDSee 10 Photo Manager.msi",1)
b.OpenView("INSERT INTO Property
(Property,Property,Property.Value)
VALUES ('PIDKEY','1234567890')").Execute
b.Commit
```

Удаление существующих полей из таблицы

Удаление какого-либо поля осуществляется в соответствии со следующим алгоритмом:

- установить курсор на таблицу, в которой необходимо удалить параметр;
- установить курсор на удаляемый параметр в таблице;
- нажать кнопку на клавиатуре или правую кнопку мыши, а в контекстном меню – DELETE;
- подтвердить намерение удалить параметр.

Эту же задачу можно решить программным способом (см. **листинг 4**), используя шаблон 1. Для обеспечения возможности записи в таблице необходимо установить второй параметр метода OpenDatabase равным единице (1).

Листинг 4. Удаление существующего атрибута в таблице

```
Set a = CreateObject("WindowsInstaller.Installer")
```

```
Set b = a.OpenDatabase ("C:\acdsee\ACDSee 10 Photo Manager.msi",1)
b.OpenView("DELETE FROM Property WHERE
'PIDKEY'='1234567890')").Execute
b.Commit
```

Изменение значений существующих полей

Изменение значения осуществляется путем установки курсора в необходимую ячейку двойным щелчком левой кнопки мыши (см. **рис. 3**).

Программное изменение свойства осуществляется по тем же правилам. По сравнению с предыдущими двумя случаями, изменяется только запрос.

В **листинге 5** осуществляется отключение компонента Device Detector.

Листинг 5. Изменение значений существующих полей таблицы

```
Set a = CreateObject("WindowsInstaller.Installer")
Set b = a.OpenDatabase ("C:\acdsee\ACDSee 10 Photo Manager.msi",1)
b.OpenView("UPDATE Feature SET Level='0' WHERE
Feature='DevDetect')").Execute
b.Commit
```

Заключение

Внимательно изучив предложенный вариант, читатель без труда сможет либо с помощью сценария на VBScript, либо с помощью программы orca.exe значительно облегчить себе задачу массовой установки программного обеспечения, что позволит сэкономить массу времени и сил.

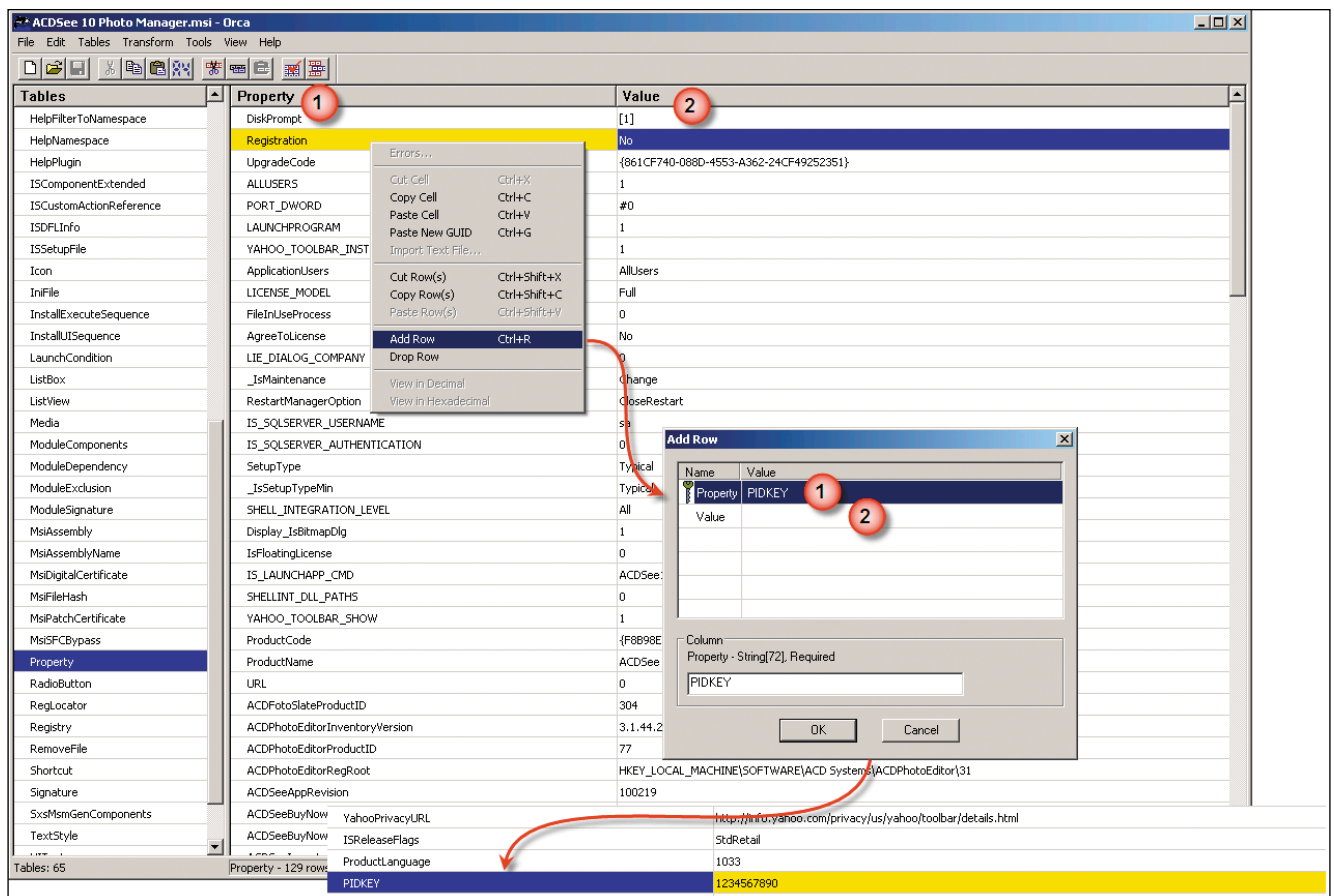


Рисунок 3. Изменение параметра