

# Глава 8<sup>1</sup>. Подсистема ввода-вывода

Подсистема ввода-вывода в Windows состоит из набора компонентов исполнительной системы, которые совместно управляют устройствами и предоставляют этим устройствам интерфейсы для приложений и системы. В этой главе мы прежде всего поговорим о том, какие цели, ставившиеся при проектировании подсистемы ввода-вывода, оказали влияние на ее реализацию. Затем мы перейдем к рассмотрению ее компонентов, в том числе диспетчера ввода-вывода, диспетчера устройств, поддерживающих технологию Plug and Play (PnP), и диспетчера электропитания. Будут проанализированы структура и компоненты подсистемы ввода-вывода и различные типы драйверов устройств. Вы познакомитесь с основными структурами данных, которые описывают устройства, драйверы устройств и запросы на ввод и вывод. Затем мы рассмотрим этапы обработки этих запросов. Завершают главу сведения о способах распознавания устройств, установки драйверов и управления электропитанием.

## Компоненты подсистемы ввода-вывода

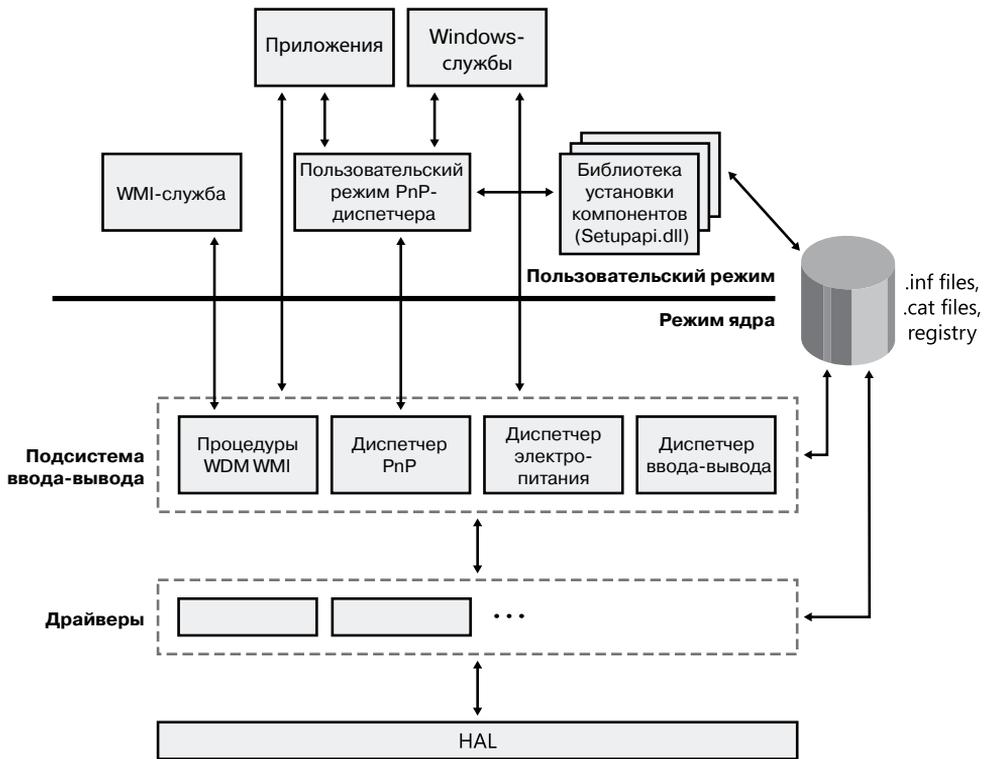
Подсистема ввода-вывода в Windows проектировалась как абстрактный интерфейс приложений для аппаратных (физических) и программных (виртуальных, или логических) устройств, обладающий определенными функциональными возможностями:

- ❑ Стандартные средства безопасности и именованя устройств предназначены для защиты общих ресурсов (политики безопасности Windows описываются в главе 6 части I).
- ❑ Высокопроизводительный асинхронный пакетный ввод-вывод служит для поддержки масштабируемых приложений.
- ❑ Специальные службы позволяют писать драйверы устройств на высокоуровневом языке и упрощают их перенос на машины с другой архитектурой.
- ❑ Многоуровневая модель и расширяемость обеспечивают возможность добавлять драйверы, меняющие поведение других драйверов или устройств без необходимости модификации последних.
- ❑ Динамическая загрузка и выгрузка драйверов устройств позволяют выполнять данные процедуры по запросу, экономя системные ресурсы.
- ❑ Поддержка технологии Plug and Play обеспечивает обнаружение и установку драйверов для нового оборудования и выделение им нужных аппаратных ресурсов, давая приложениям возможность находить и задействовать интерфейсы устройств.

---

<sup>1</sup> Уважаемые читатели! Не удивляйтесь, что эта книга начинается с восьмой главы. Авторы разделили свой труд на две части. Первая часть книги вышла в издательстве «Питер» в 2013 году.

- ❑ Подсистема управления электропитанием позволяет системе или отдельным устройствам переходить в состояния с низким энергопотреблением.
- ❑ Поддерживается установка различных файловых систем, в том числе FAT, CDFS, UDF и NTFS (типы и архитектура файловых систем подробно рассматриваются в главе 12).
- ❑ Благодаря поддержке технологии WMI (Windows Management Instrumentation – инструментарий управления Windows) и средств диагностики управление драйверами и текущий контроль осуществляются при помощи WMI-приложений и WMI-сценариев (подробно технология WMI рассматривается в главе 4 части I).  
Для реализации этой функциональности в подсистеме ввода-вывода Windows предусмотрен ряд компонентов и драйверов устройств (рис. 8.1).



**Рис. 8.1.** Компоненты подсистемы ввода-вывода

- ❑ Сердцем подсистемы ввода-вывода является одноименный диспетчер; он соединяет приложения и системные компоненты с виртуальными, логическими и физическими устройствами, создает поддерживающую драйверы устройств инфраструктуру.

- Драйвер устройства, как правило, обеспечивает конкретные устройства интерфейсом ввода-вывода. Он представляет собой программный модуль, интерпретирующий высокоуровневые команды, такие как `read` или `write`, и выполняющий низкоуровневые команды, связанные с устройством, например запись в регистр управления. Драйверы устройств принимают от диспетчера ввода-вывода команды, предназначенные для управляемых ими устройств, и уведомляют диспетчер о выполнении этих команд. Данный диспетчер часто используется драйверами устройств для пересылки команд ввода-вывода другим драйверам, задействованным в реализации интерфейса того же устройства и участвующим в управлении им.
- PnP-диспетчер работает совместно с диспетчером ввода-вывода и такой разновидностью драйверов устройств, как драйвер шины. Он управляет выделением аппаратных ресурсов, а также распознает устройства и реагирует на их подключение или отключение. Именно PnP-диспетчер и драйверы шин обеспечивают загрузку соответствующего драйвера при обнаружении нового устройства. Если нужный драйвер устройства отсутствует, компоненты исполнительной системы, отвечающие за поддержку технологии PnP, вызывают службы установки устройств PnP-диспетчера в пользовательском режиме.
- Диспетчер электропитания также тесно связан с диспетчером ввода-вывода и PnP-диспетчером. Он управляет переходами в различные состояния энергопотребления как самой системы, так и отдельных драйверов устройств.
- Процедуры поддержки инструментария управления Windows (WMI), называемые WMI-провайдером модели драйверов в Windows (Windows Driver Model, WDM), позволяют драйверам устройств выступать в роли провайдеров, взаимодействуя с WMI-службой в пользовательском режиме через провайдер WDM WMI (подробно технология WMI рассматривается в соответствующем разделе главы 4 части I).
- Реестр представляет собой базу данных с описанием основных подключенных к подсистеме устройств, а также параметров инициализации драйверов и конфигурации (подробно они рассматриваются в главе 4 части I).
- Файлы установки драйверов (с расширением `.inf`) связывают аппаратные устройства с драйверами, управляющими этими устройствами. Содержимое такого файла состоит из напоминающих сценарий инструкций, описывающих собственно устройство, исходное и целевое положение файлов драйвера, вносимые в реестр при установке драйвера изменения и сведения о зависимостях драйвера. Удостоверяющие файлы драйверов цифровые подписи, проверенные лабораторией WHQL (Microsoft Windows Hardware Quality Lab), хранятся в файлах с расширением `.cat`. Цифровые подписи также применяются для предотвращения взлома драйвера или его INF-файла.
- Уровень аппаратных абстракций (Hardware Abstraction Layer, HAL) изолирует драйверы от специфических особенностей конкретных процессоров и контроллеров прерываний, поддерживая прикладные программные интерфейсы, скрывающие межплатформенные различия. В сущности, HAL является драйвером шины для устройств на материнской плате компьютера, которые не контролируются другими драйверами.

## Диспетчер ввода-вывода

Центральным элементом подсистемы ввода-вывода является *диспетчер ввода-вывода* (I/O manager), задающий инфраструктуру для доставки драйверам устройств запросов на ввод и вывод. Данная подсистема имеет *пакетное управление*. Большинство запросов представлены именно *пакетами запросов на ввод-вывод* (I/O Request Packets, IRP), передаваемыми от одного компонента системы к другому. (В разделе «Быстрый ввод-вывод» вы познакомитесь с исключением из этого правила, когда IRP-пакеты не используются.) Подобное проектное решение позволяет отдельному программному потоку приложения одновременно управлять целым набором запросов на ввод и вывод. Такая структура данных, как IRP-пакет, содержит информацию, полностью описывающую запрос на ввод и вывод (подробно эта тема рассматривается в разделе «Пакеты запросов на ввод и вывод» далее в этой главе).

Диспетчер ввода-вывода представляет операции ввода и вывода в памяти в виде IRP-пакетов. При этом он передает указатель на IRP нужному драйверу и после завершения операции удаляет пакет. А драйвер, получивший IRP, выполняет указанную в пакете операцию и возвращает пакет диспетчеру ввода-вывода, либо сигнализируя о завершении операции, либо с целью передачи пакета другому драйверу для дальнейшей обработки.

В дополнение к созданию и уничтожению IRP-пакетов диспетчер ввода-вывода предоставляет различным драйверам общий код, который они используют при обработке ввода-вывода. Подобное объединение задач упрощает драйверы и делает их более компактными. В частности, диспетчер ввода-вывода предоставляет функцию, позволяющую драйверу вызывать другие драйверы. Также он управляет буферами запросов на ввод и вывод, обеспечивает время ожидания для драйверов и регистрирует загруженные в операционную систему устанавливаемые файловые системы. Диспетчер ввода-вывода содержит почти сотню процедур, которые могут вызываться драйверами устройств.

Также диспетчер ввода-вывода предоставляет гибкие службы ввода-вывода, позволяющие подсистемам окружения (например, Windows и POSIX) реализовывать соответствующие функции. В частности, сюда относятся тщательно разработанные службы асинхронного ввода-вывода, предоставляющие разработчикам возможность создавать высокопроизводительные масштабируемые серверные приложения.

Единый модульный интерфейс драйверов позволяет диспетчеру ввода-вывода вызывать их даже при отсутствии сведений об их структуре и внутреннем устройстве. Операционная система обрабатывает запросы на ввод и вывод так, будто они адресованы файлам; запрос к виртуальному файлу преобразуется драйвером в запрос к конкретному устройству. Драйверы могут вызывать друг друга (через диспетчер ввода-вывода), обеспечивая многоуровневую независимую обработку запроса на ввод или вывод.

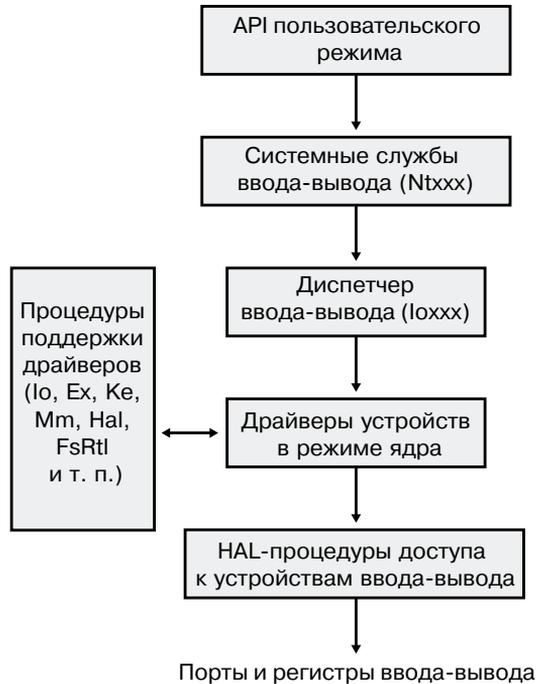
Кроме обычных функций открытия, закрытия, чтения и записи подсистема ввода-вывода в Windows предоставляет ряд усовершенствованных механизмов, например асинхронного, прямого, буферизованного и фрагментированного ввода-вывода. Эти механизмы рассматриваются в разделе «Типы ввода-вывода» далее в этой главе.

## Стандартная обработка ввода-вывода

Большая часть операций ввода и вывода не требует участия всех компонентов подсистемы ввода-вывода. Как правило, запрос на ввод или вывод поступает от приложения,

выполняющего соответствующую операцию (например, чтение данных с устройства); такие операции обрабатываются диспетчером ввода-вывода, одним или несколькими драйверами устройств и HAL.

Как уже упоминалось, в операционной системе Windows операции ввода и вывода программные потоки выполняют с виртуальными файлами. Термин «виртуальный файл» относится к любому источнику или приемнику запроса на ввод-вывод, который рассматривается как файл (это может быть файл, папка, канал или почтовая ячейка). Операционная система рассматривает все запросы на ввод и вывод как операции над виртуальным файлом, так как диспетчер ввода-вывода ни с чем другим работать не умеет. При этом за преобразование файловых команд (`open`, `close`, `read`, `write`) в команды для конкретного устройства отвечает драйвер. Подобная абстракция обеспечивает единый программный интерфейс для всех устройств. Приложения в режиме пользователя (как в Windows, так и в POSIX) вызывают документированные функции, которые, в свою очередь, обращаются к внутренним функциям подсистемы ввода-вывода для чтения из файла, записи в файл и прочих операций. Диспетчер ввода-вывода динамически направляет эти адресованные виртуальным файлам запросы к драйверам соответствующих устройств. Рисунок 8.2 демонстрирует базовую схему обработки запроса на ввод-вывод.



**Рис. 8.2.** Схема обработки типичного запроса на ввод-вывод

Далее мы детально рассмотрим эти компоненты, поговорим о различных типах драйверов, их структуре, загрузке, инициализации, а также способах обработки ими

запросов на ввод-вывод. Затем вы познакомитесь с функциями и ролью PnP-диспетчера и диспетчера электропитания.

## Драйверы устройств

Для интеграции с диспетчером ввода-вывода и прочими компонентами одноименной подсистемы драйвер устройства должен быть написан в соответствии с правилами, установленными для указанного типа драйверов устройств и выполняемых этим драйвером операций. В этом разделе рассматриваются типы поддерживаемых в Windows драйверов устройств и их внутренняя структура.

### Типы драйверов устройств

В операционной системе Windows поддерживается широкий спектр типов драйверов устройств и сред разработки. Последние могут различаться даже для драйверов устройств одного типа. Это зависит от особенностей устройств, для которых предназначен драйвер. Кроме того, драйверы делятся на работающие в пользовательском режиме и в режиме ядра. В Windows поддерживаются несколько типов драйверов пользовательского режима:

- ❑ Драйверы принтеров в Windows переводят аппаратно-независимые запросы на графические операции в понятные принтеру команды, которые затем передаются драйверу порта в режиме ядра. Например, драйвер порта принтера универсальной последовательной шины (USB) называется `Usbprint.sys`.
- ❑ Драйверы, являющиеся компонентами среды UMDF (User-Mode Driver Framework), как понятно из их названия, работают с аппаратным обеспечением на уровне пользователя. С библиотекой поддержки UMDF в режиме ядра они общаются через механизм ALPC. Более подробно мы поговорим об этом в разделе «Среда UMDF».

В этой главе в основном рассматриваются драйверы, работающие в режиме ядра. Их можно разбить на следующие категории:

- ❑ *Драйверы файловой системы* (file system drivers) принимают запросы к файлам на ввод-вывод и на их основе выдают более конкретные запросы к драйверам запоминающих или сетевых устройств.
- ❑ *Драйверы PnP-устройств* (Plug and Play drivers) работают с аппаратным обеспечением и объединены с диспетчером электропитания и PnP-диспетчером. В эту категорию входят драйверы запоминающих устройств, видеоадаптеров, устройств ввода и сетевых адаптеров.
- ❑ *Драйверы устройств, не поддерживающих технологию Plug and Play* (Non-Plug and Play drivers), включают в себя также расширения ядра и делают систему более функциональной. Как правило, они не интегрированы с PnP-диспетчером или с диспетчером электропитания, так как не связаны с физическими аппаратными устройствами. К этой категории относятся драйверы протоколов и сетевого при-

кладного программного интерфейса, а также описанный в главе 4 части I драйвер Process Monitor.

Драйверы режима ядра подразделяются на группы в зависимости от модели и роли в обслуживании запросов к устройствам.

### WDM-драйверы

WDM-драйверы являются драйверами устройств, соответствующими модели WDM (Windows Driver Model). WDM поддерживает управление электропитанием, технологию Plug and Play и инструментарий управления Windows. Большинство драйверов PnP-устройств соответствует модели WDM. Драйверы данной категории делятся на три типа:

- ❑ *Драйверы шины* (bus drivers) управляют логической или физической шиной. Это могут быть шины PCMCIA, PCI, USB и IEEE 1394. Драйвер шины отвечает за распознавание подключенных к шине устройств и оповещение о них PnP-диспетчера, а также за управление электропитанием шины.
- ❑ *Функциональные драйверы* (function drivers) управляют устройствами конкретного типа. Драйверы шины представляют устройства функциональным драйверам через PnP-диспетчер. Функциональным называется драйвер, экспортирующий в операционную систему рабочий интерфейс устройства. В общем случае именно он лучше всего осведомлен о работе устройства.
- ❑ *Фильтрующие драйверы* (filter drivers) могут располагаться как выше, так и ниже функционального и шинного драйверов. Они дополняют или меняют поведение устройства или другого драйвера. Например, служебная программа для перехвата ввода с клавиатуры может быть реализована при помощи фильтрующего драйвера клавиатуры, работающего поверх функционального драйвера клавиатуры.

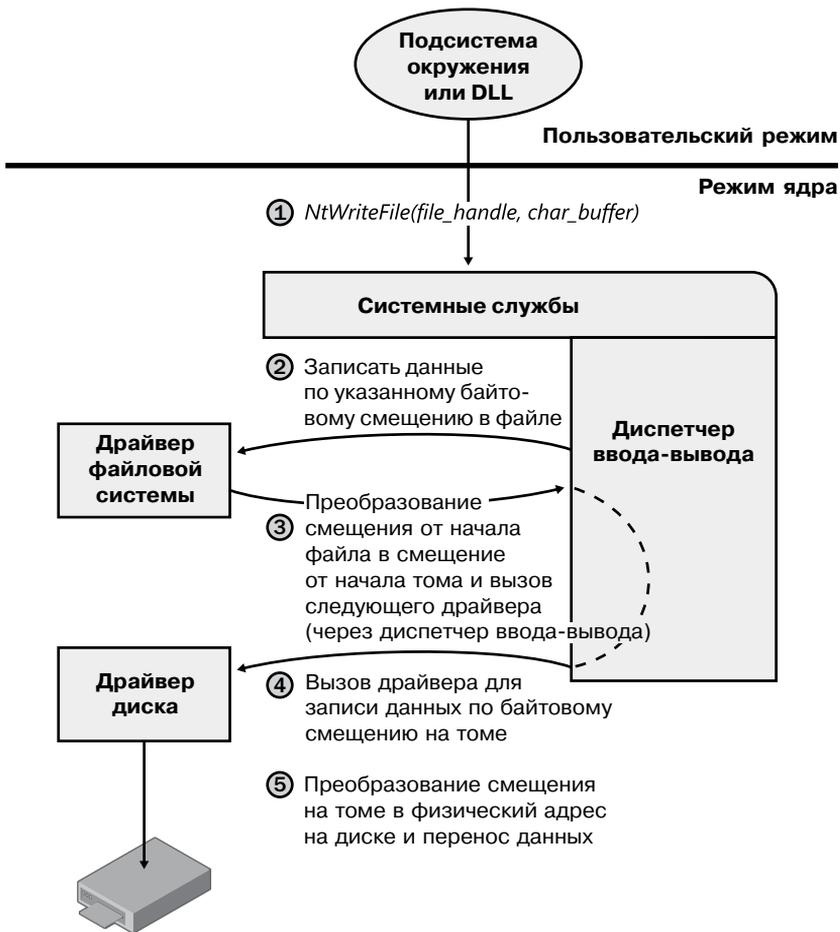
Ни один WDM-драйвер не отвечает полностью за все аспекты управления устройством. Драйвер шины занимается отслеживанием состава устройств на шине (путем подключения или отключения), помогая PnP-диспетчеру регистрировать эти устройства, обращаясь к относящимся к шине конфигурационным регистрам и в некоторых случаях управляя электропитанием подключенных устройств. К аппаратной части устройства обычно обращается только функциональный драйвер.

### Многоуровневые драйверы

Поддержка отдельного устройства часто реализуется целым набором драйверов, каждый из которых предоставляет часть функциональности, необходимой для корректной работы. Кроме WDM-драйверов шины, функциональных и фильтрующих драйверов, поддержка аппаратного обеспечения может обеспечиваться еще и другими компонентами:

- ❑ *Драйверы классов* (class drivers) устройств отвечают за обработку ввода-вывода для устройств конкретного класса, таких как жесткий диск, клавиатура или компакт-диск, со стандартизированными аппаратными интерфейсами, позволяющими одному драйверу обслуживать устройства от различных производителей.

- *Драйверы мини-классов* (miniclass drivers) реализуют обработку ввода-вывода, заданную производителем для определенного класса устройств. К примеру, несмотря на наличие стандартного драйвера класса элементов питания от Microsoft, интерфейсы источников бесперебойного питания (Uninterruptible Power Supplies, UPS) и элементов питания портативных компьютеров у различных производителей различаются настолько, что не обойтись без мини-класса. Принадлежащие к данной категории драйверы по сути представляют собой динамически подключаемые библиотеки (DLL) режима ядра и не умеют напрямую обрабатывать IRP-пакеты — они приводятся в действие драйвером класса и именно оттуда импортируют нужные функции.



**Рис. 8.3.** Многоуровневое представление драйвера файловой системы и драйвера диска