
МАСС-СПЕКТРОМЕТРИЯ ДЛЯ БИОТЕХНОЛОГИИ

УДК 621.384.668.8

© Д. М. Петров

**СПОСОБЫ УНИФИЦИРОВАННОГО
УПРАВЛЕНИЯ СИСТЕМАМИ РЕГИСТРАЦИИ
ВРЕМЯПРОЛЕТНОГО МАСС-СПЕКТРОМЕТРА**

В статье описывается проблема взаимодействия между собой различных уровней управления системой регистрации, получения данных, их обработки и представления результатов. Рассматриваются и анализируются существующие методы ее решения. Предлагаемая технология основана на OLE/COM-интерфейсах и COM-технологии, разработанной Microsoft.

ВВЕДЕНИЕ

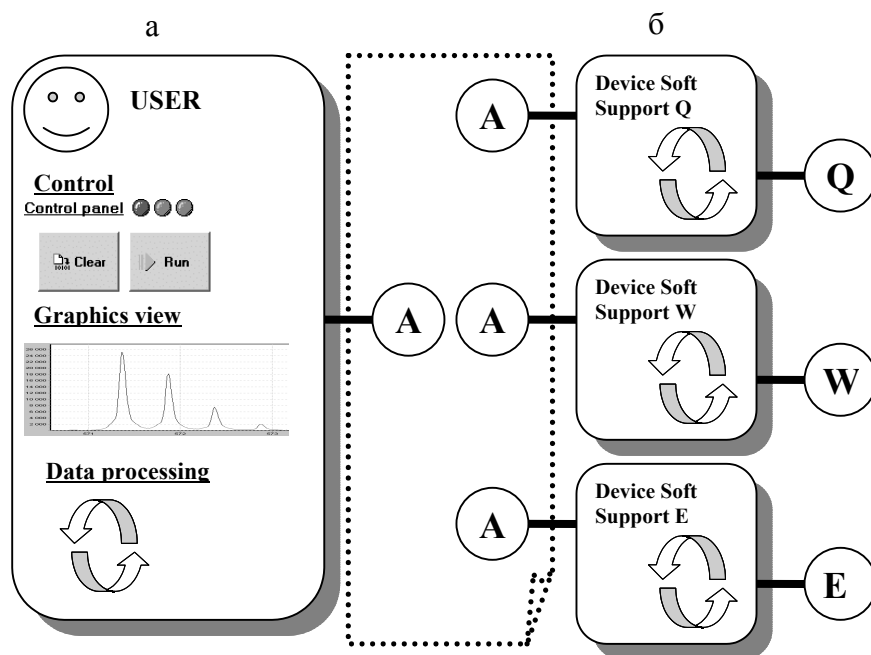
Отсутствие каких-либо общепринятых стандартов разработки программного обеспечения в конкретной области является причиной многих проблем. В этом случае при разработке нового устройства или модификации существующего требуется разрабатывать новое программное обеспечение (ПО). Сложность и качество могут варьироваться в зависимости от требований и методов разработки. Каждый разработчик реализует ту функциональность устройства, которую считает нужной, и теми способами, которыми может. Каждый отдельный случай отражает специфическую логику разработчика, которая для конечного пользователя оборачивается как минимум разнообразием пользовательских интерфейсов и вариантов совместимости с другими аппаратными и программными комплексами. Такая проблема более остро ощущается в тех областях, где профессиональная разница между разработчиком и конечным пользователем становится достаточно велика. В масс-спектрометрии производители — это физики и конструкторы, а пользователи — химики и биологи. В этом случае важны не столько способ управления прибором и специфика получаемых данных, сколько результат различной обработки этих данных. Для производителя необходимо обеспечить возможность корректной настройки прибора и получение данных (спектра), удовлетворяющих требуемым параметрам. Но в конечном счете для пользователя необходимо иметь возможность просто запустить эксперимент и получить конкретный ответ — состав вещества в его пробе. А это — результат многочисленных операций над уже полученными данными.

ОБСУЖДЕНИЕ ЗАДАЧИ УНИФИКАЦИИ ПО

Задача унификации разделяется по функцио-

нальности на две четкие группы. На рисунке показана схема программно-аппаратно-независимого комплекса с общим интерфейсом. Первая группа — это взаимодействие с устройством (нижний уровень, на рисунке — б). Вторая — это обработка данных, получаемых устройством, и взаимодействие с пользователем (верхний уровень, а). Эти группы практически не пересекаются в реализации, а потому могут (и должны) разрабатываться независимо друг от друга. В противном случае производителю требуется разрабатывать весь комплекс в целом, что является избыточным для поддержки только одного устройства (в этом случае страдает, как правило, вторая группа) и коммерчески невыгодным.

В тех областях, где количество производителей, а тем более пользователей велико, задача унификации до некоторой степени решена. Таким решением является ставший стандартом *de facto* динамический обмен данными (DDE). Интерфейс широко применяется в операционных системах Windows. При таком включении компоненты являются информационным источником, взаимодействующим с приборами. Изначально протокол DDE применялся в первых человеко-машинных интерфейсах в качестве механизма разделения данных между прикладными системами и устройствами типа ПЛК (программируемые логические контроллеры). Для преодоления недостатков DDE, прежде всего для повышения надежности и скорости обмена, разработчики предложили свои собственные решения (протоколы), такие как AdvancedDDE или FastDDE — протоколы, связанные с пакетированием информации при обмене с ПЛК и сетевыми контроллерами. Но такие частные решения приводят к ряду проблем: каждый раз пишется свой драйвер для поставляемого на рынок оборудования; в общем случае два пакета не могут иметь доступ к одному драйверу в одно и то же время, поскольку каждый из них поддерживает



Программно-аппаратно-независимый комплекс с общим интерфейсом. "Верхний уровень" (а) включает в себя интерфейс пользователя, элементы управления, отображение информации, методы обработки данных. "Нижний уровень" (б) содержит только программную поддержку управления конкретной аппаратной реализацией устройства. Взаимодействие осуществляется через интерфейс А, который оба уровня должны поддерживать

обмен именно со своим драйвером; изменения в аппаратных реализациях могут привести к необходимости замены некоторых драйверов и т. д.

Но главная цель унификации достигнута — мы можем, например, набирать текст в "любимом" текстовом редакторе, независимо от того какой марки принтер, на котором собираемся его распечатывать, лазерный он или матричный. Кроме того, производителю принтера не требуется писать собственный уникальный редактор, а у разработчиков редактора "не болит голова" в попытках организовать взаимодействие с различными аппаратными средствами.

Технология унификации должна исключать недостатки использования протокола DDE и в то же время являться открытой и независимой как для производителей устройств, так и для разработчиков программного обеспечения. Технология должна являться связующим звеном между аппаратным и программным производителем. Это механизм, который обеспечивает получение данных от устройства и передачу их прикладной программе некоторым стандартным путем. Необходимо, чтобы различные программные системы, созданные

с помощью различных средств, установленных на различных платформах, работающих на разных компьютерах, умели "договариваться". То есть они знали, как запросить друг у друга данные и как послать друг другу "указания". Для основы такой технологии больше всего подходят OLE/COM-интерфейсы и COM-технология, разработанная Microsoft.

Таким образом, программный комплекс делится на два компонента. Один — это "компонент-устройство", создаваемый изготовителем устройства, — осуществляет минимальную программную поддержку устройства на низком уровне. Здесь изготовителю предоставляется полная свобода в выборе методов разработки, которые могут не отличаться от методов разработки драйвера DDE и при этом не требуют публикации внутренней архитектуры. Главное, компонент должен содержать механизм взаимодействия со вторым компонентом. Второй — это "компонент-пользователь", который содержит всю смысловую функциональность: интерфейс пользователя, обработку данных, взаимодействие с другими программами, базами данных, устройствами и т. д. Реализация такого механизма

требует минимальных затрат для обеих групп разработчиков. Усилия при разработке этих компонентов направлены на решение конкретных задач — для каждого компонента своей. Изменения во внутренней структуре одного компонента не требуют вмешательства в структуру другого. Благодаря этому, на программном уровне мы получаем гибкий, независимый, легко организуемый комплекс. А пользователь получает программу, функциональность которой его удовлетворяет и, что немаловажно, работать с которой он привык, независимо от того с каким устройством и какого производителя он работает в данный момент или собирается работать в будущем.

Как и было описано выше, производители систем регистрации поставляют программное обеспечение для поддержки своих устройств. Оно отличается по функциональности и может не удовлетворять тем параметрам, которые необходимо достичь. Тем более, если разрабатывается собственная система регистрации, на свет появляется еще один комплекс-самородок. В то же время мы хотим использовать свои методы обработки сигналов, реализовать ту функциональность и с теми параметрами, которые нас интересуют.

РЕАЛИЗОВАННЫЕ В ПРИБОРЕ РЕШЕНИЯ

Описываемая технология унификации позволила разрабатывать программный комплекс, ориентированный только на скоростную обработку сигналов и мониторинг данных в режиме реального времени, а также взаимодействие с пользователем. Комплекс имеет возможность в любой момент переподключиться к требуемой системе регистрации для управления и получения данных, а также работать и без нее — с ранее сохраненными данными. С другой стороны, были разработаны компоненты поддержки используемых систем регистрации.

Для разработки общего интерфейса взаимодействия было проведено обобщение функциональности различных существующих систем регистрации. Был выявлен минимально необходимый и достаточный набор методов (термин "метод" здесь применен в смысле объектно-ориентированного языка программирования), который удовлетворял любой системе регистрации время-

пролетного масс-спектрометра. Интерфейс *IDevice* содержит методы подключения и отключения от системы регистрации на уровне компонентов. Интерфейс *IDeviceProp* содержит методы оперирования со свойствами системы регистрации, такие как чтение, редактирование, запись и т.д. Интерфейс *IDeviceEngine* содержит методы аппаратного управления устройством: инициализация, запуск и т.д. Интерфейс *IDeviceEvents* является клиентским и предназначен для передачи клиенту различной информации: сообщений об ошибках, получаемых данных и т.д.

Практическое применение описываемой технологии было использовано в разработке программного комплекса для времяпролетного масс-спектрометра. В этом случае конечным устройством, которое требует программной поддержки, является система регистрации.

ЗАКЛЮЧЕНИЕ

На данный момент аналогичную технологию применяют крупные компании-производители аппаратных средств. Практически у всех она является закрытым внутренним стандартом коммерческих организаций. Компания "National Instruments" разработала свой стандарт взаимодействия с устройствами, благодаря которому существует возможность разработать приложение верхнего уровня, используя средства разработки, предоставляемые компанией, которое будет взаимодействовать с любыми устройствами, имеющими драйверы, удовлетворяющие внутреннему стандарту компании. При таком подходе разработчики как устройств, так и программ для обработки данных становятся и коммерчески, и/или функционально независимы от компаний-производителей не только в приобретении их продукта, но и в его использовании.

*Институт аналитического приборостроения РАН,
Санкт-Петербург*

Материал поступил в редакцию 7.04.2004.

METHODS OF UNIFIED DATA SYSTEM MANAGEMENT FOR A TIME-OF-FLIGHT MASS SPECTROMETER

D. Petrov

Institute for Analytical Instrumentation RAS, Saint-Petersburg

The problem of interaction of different management levels for the system of data acquisition, processing and presentation of results is discussed. The existing methods of solving this problem are considered and analyzed. The technology offered is based on OLE/COM interfaces and the COM technology developed by Microsoft.