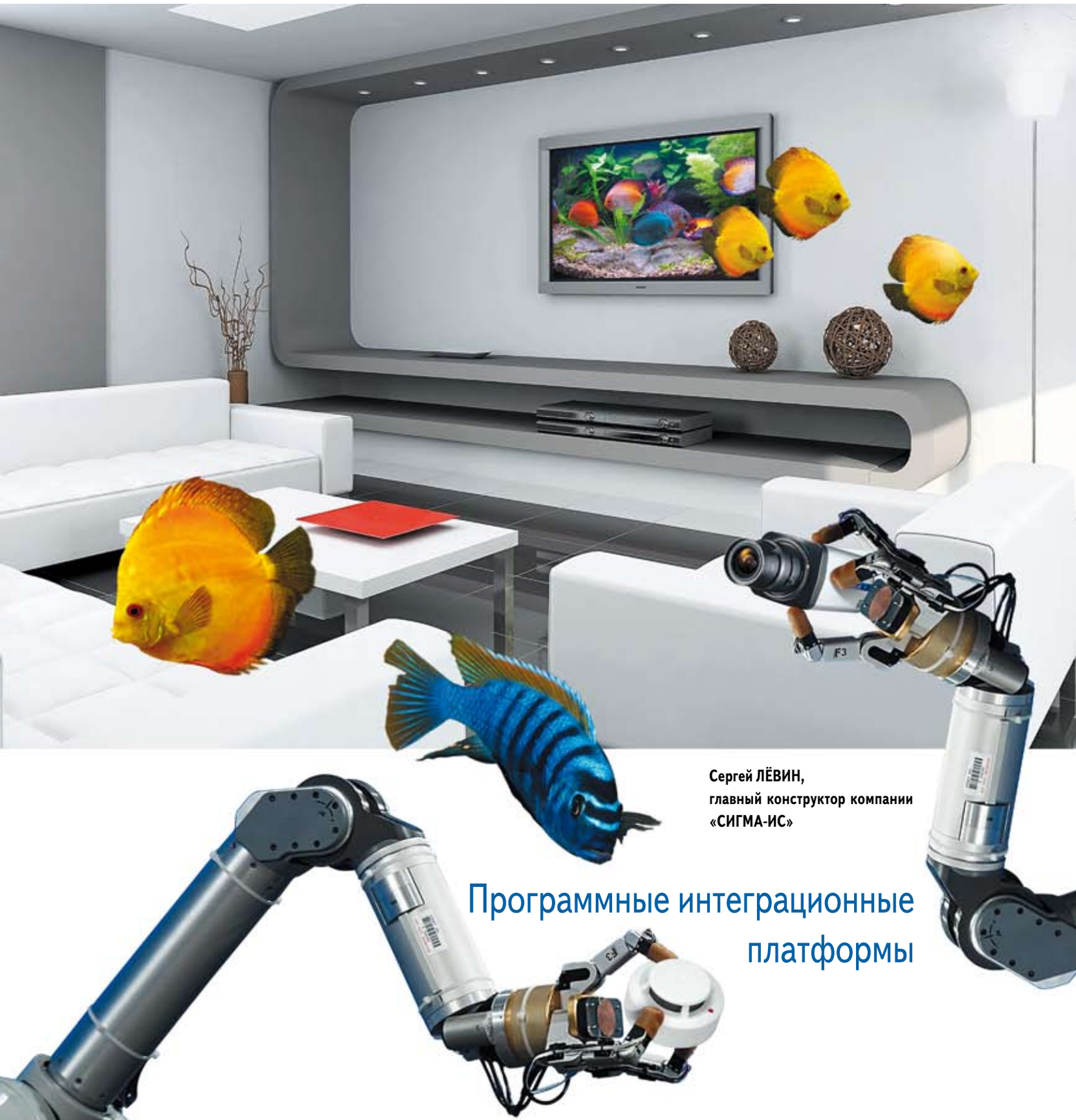

Сергей ЛЁВИН,
главный конструктор компании «СИГМА-ИС»

Программные интеграционные платформы

Современная система безопасности – это сложный программно-аппаратный комплекс, включающий в себя множество подсистем. Типовой набор может выглядеть следующим образом:

– охранная сигнализация: охрана периметра, объектовая охрана, тревожная сигнализация, терминалы для постановки на охрану/снятия с охраны;

– противопожарная защита: пожарная сигнализация, система дымоудаления, автоматическое пожаротушение, оповещение;

– видеонаблюдение: прием видеопотока от камер, отображение и запись видео, видеоаналитка в реальном времени и постпроцессинг;

– система контроля и управления доступом: проходные и тамбур-шлюзы, пункты въезда автотранспорта, точки доступа в помещения, бюро пропусков, учет рабочего времени.

Все это хозяйство должно работать согласованно, безконфликтно и эффективно. Кроме того, часто возникают задачи совместной работы системы безопасности с другими системами объекта. Например, согласованные действия с системами отопления, вентиляции, кондиционирования, освещения. На промышленных объектах нередко нужно интегрировать систему безопасности с АСУТП предприятия.

Для чего нужны интегрированные решения? Ведь можно спроектировать, смонтировать и пользоваться каждой подсистемой отдельно, независимо от других. Раньше так и делали. Ответ предельно прост и, в общем-то, ожидаем: деньги. Интегрированные решения в конечном счете дешевле и эффективнее. Ну и помимо всего прочего гораздо функциональнее.

Что дает интеграция? Ответ на этот вопрос напрямую зависит от способа и глубины решения задачи. Как минимум интеграция должна позволять создавать единые рабочие места службы охраны, для того чтобы оператор мог получать сообщения от всех подсистем в едином формате. Также должна быть возможность реализации реакций в подсистеме А на события в подсистеме Б. Например, в случае возгорания на объекте пожарная сигнализация должна сообщить системе контроля доступа, что нужно разблокировать двери для беспрепятственной эвакуации людей. Или при срабатывании охранной сигнализации на монитор оператора автоматически выводится изображение от видеокамер, в поле зрения которых попадает место происшествия. Более полная интеграция даст возможность объединения подсистем не только на рабочем месте оператора, но и построения единого пространства администрирования системы. Это дает дополнительную степень свободы при конфигурировании и настройке алгоритмов работы. Еще большие возможности дает интеграция на аппаратном уровне, когда взаимодействие осуществляется на уровне контроллеров подсистем. Но об этом мы поговорим в следующий раз, а здесь остановимся на особенностях программной интеграции.

Кому это нужно? Прежде все такие решения интересны интеграторам при реализации сложных проектов, где применяется разнородное оборудование от разных производителей. Дело осложняется еще тем, что в системах безопасности, в отличие, например, от систем автоматизации, крайне плохо решен вопрос со стандартизацией в области коммуникации разного оборудования между собой. Недавнее создание международных альянсов по стандартизации в области IP-видеооборудования ONVIF и PSIA скорее исключение из правил, так как по остальным направлениям (прежде всего ОПС и СКУД) такого праздника пока не предвидится.

Как это делается? Если говорить о программной интеграции, то существует особый класс программных продуктов, в той или иной мере призванных решать эту задачу. В России более менее устоялось определение «интегрирующее ПО». Если говорить об остальном мире, то это называется Physical Security Information Management (PSIM), т. е. управление данными в системах физической защиты. В целом в подобном ПО можно выделить три основных уровня.

Уровень подключения к оборудованию: это набор неких драйверов, позволяющих подключать к системе самое разное оборудование системы безопасности. В минимальном случае ПО должно позволять получать события от оборудования, в идеале управлять этим оборудованием и конфигурировать его.

Уровень представления и хранения данных, а также блок бизнес-логики. На этом уровне информация, полученная от оборудования, должна быть преобразована во внутреннее унифицированное представление системы. Здесь же реализуются алгоритмы обработки информации, получаемой от оборудования. Алгоритмы могут быть как встроенные в ПО, как правило, параметризуемые с помощью механизма настроек, так и пользовательские. Для реализации пользовательских алгоритмов часто используется встроенный язык программирования, оригинальный или стандартный, например C#, Java. Наличие этой функции крайне важно для тонкой и качественной настройки поведения системы в целом.

Уровень пользовательского интерфейса (GUI). Основная задача этого уровня – представление различных данных от различного оборудования для оператора системы в унифицированном и понятном ему виде.

Кто это делает? Исходя из задач, решаемых интегрирующим ПО, можно сделать вывод, что продукты такого класса весьма сложны в реализации. И это действительно так! Универсальность, многообразие поддерживаемого оборудования, его разнородность многократно усложняют задачу разработчикам по сравнению с прикладными решениями, предназначенными для работы с каким-то одним конкретным типом оборудования. Самое сложное заключается в том, что на этапе разработки неизвестно, какое оборудование впоследствии нужно будет интегрировать в систему. Поэтому действительно успешных примеров реализации программных интеграционных платформ не так уж и много. Можно выделить три типа компаний, которые занимаются разработкой интегрирующего ПО.

Крупные производители оборудования для систем безопасности, которые выпускают широкую номенклатуру собственного оборудования. Основной минус таких компаний в том, что обычно они закладываются прежде всего на поддержку собственной продукции. И это естественно, ведь, как правило, они сами предлагают оборудование для решения всех задач в области защиты объектов ну или, по крайней мере, думают, что всех.

Интеграторы, реализующие сложные комплексные проекты. Как правило, разработка интегрирующего ПО в этом случае начинается достаточно стихийно. На каком то конкретном объекте нужно что-то с чем-то состыковать, увязать. Стандартными средствами не получается. Нанимаются программисты, пишется интегрирующее ПО.

Решается частная задача, дальше производится наращивание функционала, подключение все нового и нового оборудования. В результате получается продукт, раздираемый внутренними противоречиями, так как изначально не было сформулировано всеобъемлющих требований к задаче. Более того, единого ПО, как такового, и нет, есть множество веток одного проекта, произрастающих каждая из своего объекта применения. То есть ПО, установленное на крупном заводе два года назад, это не совсем то, а, скорее всего, оно даже несовместимо с той версией, что в прошлом году устанавливали на подобном же предприятии. Причем справиться с установкой и поддержкой такого ПО никто, кроме разработчиков, не сможет. Поддерживать современный уровень разработки ПО такие компании тоже, как правило, не могут в силу непрофильности этого направления для них. Таким образом, такое решение надежным, серийно выпускаемым и уж тем более «коробочным» назвать нельзя. Здесь стоит отметить, что серийность выпуска и надежность продукта прямо пропорциональны.

Софтверные компании. Так как изначально разработка ПО является основным бизнесом компании, а не побочным продуктом, как часто бывает в первых двух случаях, то здесь разумно ожидать самого что ни на есть профессионального подхода к делу. Что касается самой технологии разработки ПО, его производства и поддержки, то здесь, скорее всего, вопросов не будет. Однако есть и обратная сторона медали: довольно часто продукт портит «программистский» подход к задаче. В результате страдает качество сопряжения ПО с оборудованием, так как вникнуть во все тонкости предметной области без реального опыта работы с тем или иным «железом» бывает очень сложно, а иногда и невозможно. Пользователи часто высказывают претензии по поводу излишней сложности пользовательского интерфейса, как административного, так и операторского. Компании, специализирующиеся на разработке ПО, хотят сделать свой продукт как можно более массовым, коробочным решением, что вполне естественно. Главному же потребителю — интегратору — больше нужен индивидуальный подход, так как задачи и, главное, способы их решения у всех разные. Таким образом, разработчики и потребители входят в некоторый конфликт.

Что же делать? Получается, что все подходы имеют свои плюсы и минусы. И, в общем-то, это нормально, но что выбрать в качестве оптимума? На мой взгляд, идеальным может считаться вариант, когда компания — разработчик программной интеграционной платформы является также и разработчиком оборудования для систем безопасности, но в то же время считает разработку ПО одним из своих основных направлений. То есть имеет профессиональный коллектив программистов, владеет современными технологиями разработки. При этом в силу сложности продукта и его относительной малотиражности (крупных объектов не так уж и много) разработчик не пытается сделать из него коробочный продукт, а стремится заключить прямые договоры с основными потребителями — интеграторами. Что это даст? Во-первых, каждый интегратор получает не продукт, «как у всех», а в чем-то оригинальное решение, учитывающее его методы работы, идеологию реализации крупных проектов. В конце концов, даже внешний вид ПО, т. е. пользовательские интерфейсы могут иметь фирменный вид, характерный только для этого интегратора, что, согласитесь, немаловажно для получения дополнительных конкурентных преимуществ. При этом варианты реализации для разных интеграторов будут базироваться на одной общей, хорошо отработанной платформе, что является гарантией стабильной и надежной работы. Во-вторых, интегратор сможет сконцентрировать все свои усилия непосредственно на своем бизнесе: на разработке и реализации проектов. Будучи при этом уверенным, что в случае, если очередной проект потребует доработки ПО, он с полным правом сможет обратиться к разработчику и потребовать выполнение этих работ в соответствии с договорными обязательствами. С коробочным ПО, согласитесь, такое зачастую просто невозможно. Софтверные компании тоже не останутся в накладе, так как в рамках интеграционной платформы есть масса прикладных, отдельно стоящих задач, не затрагивающих идеологию продукта в целом. Например, реализация различных алгоритмов в области видеоаналитики. Получается, что всем есть место на этом рынке, нужно лишь сделать так, чтобы каждый занимался своим делом.

Хотя, как мне кажется, такая проблема в нашей стране имеется не только в отрасли систем безопасности.