СТАНДАРТИЗАЦИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОСНОВЫ ПОСТРОЕНИЯ СИСТЕМЫ СТАНДАРТОВ ИТ

18.1. Понятие открытых систем

Пользователи вычислительной техники неоднократно сталкивались с ситуацией, когда программное обеспечение, отлично работающее на одном компьютере, не желает работать на другом. Или системные блоки одного вычислительного устройства не стыкуются с аппаратной частью другого. Или информационная система другой компании упорно не желает обрабатывать данные, которые пользователь подготовил в информационной системе у себя на рабочем месте, хотя были выполнены все необходимые требования по подготовке данных. Или при загрузке разработанной странички на «чужом» браузере на экране вместо понятного текста возникает бессмысленный набор символов. Эта проблема, которая возникла в ходе бурного развития производства вычислительной и телекоммуникационной техники и разработки программного обеспечения, получила название *проблемы совместимости* вычислительных, телекоммуникационных и информационных устройств.

Развитие систем и средств вычислительной техники, повсеместное их внедрение в сферы управления, науки, техники, экономики и бизнеса привели к необходимости объединения конкретных вычислительных устройств и реализованных на их основе информационных систем в единые информационно-вычислительные системы и среды, к формированию *единого информационного пространства*. Такое пространство можно определить как совокупность баз данных, хранилищ знаний, систем управления ими, информационно-коммуникационных систем и сетей, методологий и технологий их разработки, ведения и использования на основе единых принципов и общих правил, обеспечивающих информационное взаимодействие для удовлетворения потребностей пользователей.

Единое информационное пространство складывается из следующих основных составляющих:

• информационные ресурсы, содержащие данные, сведения, информацию и знания, собранные, структурированные по некоторым правилам, подготовленные для доставки заинтересованному пользователю, защищенные и архивированные на соответствующих носителях;

• организационные структуры, обеспечивающие функционирование и развитие единого информационного пространства: поиск, сбор, обработку, хранение, защиту и передачу информации;

• средства информационного взаимодействия, в том числе программно-аппаратные средства и пользовательские интерфейсы, правовые и организационно-нормативные документы, обеспечивающие доступ к информационным ресурсам на основе соответствующих информационно-коммуникационных технологий.

При формировании единого информационного пространства менеджеры, проектировщики и разработчики программно-аппаратных средств столкнулись с рядом проблем. Например, разнородность технических средств вычислительной техники с точки зрения организации вычислительного процесса, архитектуры, систем команд, разрядности процессоров и шины данных потребовала создания *физических интерфейсов,* реализующих взаимную совместимость компьютерных устройств. При увеличении числа типов интегрируемых устройств сложность организации физического взаимодействия между ними существенно возрастала, что приводило к проблемам в управлении такими системами.

Разнородность программируемых сред, реализуемых в конкретных вычислительных устройствах и системах, с точки зрения многообразия операционных систем, различия в разрядности и прочих особенностей привели к созданию *программных интерфейсов*. Разнородность физических и программных интерфейсов в системе

«пользователь — компьютерное устройство — программное обеспечение» требовала постоянного согласования программно-аппаратного обеспечения и переобучения кадров.

История концепции открытых систем начинается с того момента, когда возникла проблема *переносимости* (мобильности) программ и данных между компьютерами с различной архитектурой [9]. Одним из первых шагов в этом направлении, оказавшим влияние на развитие отечественной вычислительной техники, явилось создание компьютеров серии IBM 360, обладающих единым набором команд и способных работать с одной и той же операционной системой. Корпорация «IBM», кроме того, предоставляла лицензии на свою ОС пользователям, которые предпочли купить компьютеры той же архитектуры у других производителей.

Частичное решение проблемы мобильности для программ обеспечили ранние стандарты языков, например ФОРТРАН и КОБОЛ. Язык позволяли создавать переносимые программы, хотя часто ограничивали функциональные возможности. Мобильность обеспечивалась также за счет того, что эти стандарты были приняты многими производителями различных платформ. Когда языки программирования приобрели статус стандарта де-факто, их разработкой и сопровождением начали заниматься национальные и международные организации по стандартизации. В результате языки развивались уже независимо от своих создателей. Достижение мобильности уже на этом уровне было первым примером истинных возможностей открытых систем.

Следующий этап в развитии концепции открытости — вторая половина 1970-х гг. Он связан с областью интерактивной обработки и увеличением объема продуктов, для которых требуется переносимость (пакеты для инженерной графики, системы автоматизации проектирования, базы данных, управление распределенными базами данных). Компания «DIGITAL» начала выпуск мини-ЭВМ VAX, работающих под управлением операционной системы VMS. Машины этой серии имели уже 32-разрядную архитектуру, что обеспечило значительную эффективность программного кода и сократило издержки на работу с виртуальной памятью. Программисты получили возможность напрямую использовать адресное пространство объемом до 4 Гбайт, что практически снимало все ограничения на размеры решаемых задач. Машины этого типа надолго стали стандартной платформой для систем проектирования, сбора и обработки данных, управления экспериментом и т.п. Именно они стимулировали создание наиболее мощных САПР, систем управления базами данных и машинной графики, которые широко используются до настоящего времени.

Конец 1970-х гг. характеризуется массовым применением сетевых технологий. Компания «DIGITAL» интенсивно внедряла свою архитектуру DECnet. Сети, использующие протоколы Интернет (TCP/IP), первоначально реализованные Агентством по перспективным исследованиям Министерства обороны США (DARPA), начали широко применяться для объединения различных систем — как военных, так и академических организаций США. Фирма «IBM» применяла собственную сетевую архитектуру SNA (System Network Architecture), которая стала основой для предложенной Международной организацией по стандартизацииISO архитектуры Open Systems Interconnection (OSI).

Когда сетевая обработка стала реальностью и насущной необходимостью для решения большого числа технических, технологических, научных экономических задач, пользователи начали обращать внимание на совместимость и возможность интеграции вычислительных средств как на необходимые атрибуты открытости систем. Организация ISO в 1977—1978 гг. развернула интенсивные работы по созданию стандартов взаимосвязи в сетях открытых систем. Тогда же впервые было введено определение открытой информационной системы.
Таким образом, решение проблем совместимости и мобильности привело к разработке большого числа международных стандартов и соглашений в сфере применения информационных технологий и разработки информационных систем. Основополагающим, базовым понятием при использовании стандартов стало понятие «открытая система».

Существует достаточное число определений, даваемых различными организациями по стандартизации и отдельными фирмами. Например, Ассоциация французских

пользователей UNIX и открытых систем (AFUU) дает следующее определение: «Открытая система — это система, состоящая из элементов, которые взаимодействуют друг с другом через стандартные интерфейсы».

Производитель средств вычислительной техники — компания Hewlett Packard: «Открытая система — это совокупность разнородных компьютеров, объединенных сетью, которые могут работать как единое интегрированное целое независимо от того, как в них представлена информация, где они расположены, кем они изготовлены, под управлением какой операционной системы они работают».

Определение Национального института стандартов и технологий США (NIST): «Открытая система — это система, которая способна взаимодействовать с другой системой посредством реализации международных стандартных протоколов. Открытыми системами являются как конечные, так и промежуточные системы. Однако открытая система не обязательно может быть доступна другим открытым системам. Эта изоляция может быть обеспечена или путем физического отделения, или путем использования технических возможностей, основанных на защите информации в компьютерах и средствах коммуникаций».

Другие определения в той или иной мере повторяют основное содержание определений, приведенных выше. (Анализируя их, можно выделить некоторые общие черты, присущие открытым системам:

• технические средства, на которых реализована информационная система, объединяются сетью или сетями различного уровня — от локальной до глобальной;

• реализация открытости осуществляется на основе функциональных стандартов (профилей) в области информационных технологий;

• информационные системы, обладающие свойством открытости, могут выполняться на любых технических средствах, которые входят в единую среду открытых систем;

• открытые системы предполагают использование унифицированных интерфейсов в процессах взаимодействия в системе «человек — компьютер»;

• применение положений открытости предполагает некоторую избыточность при разработке программно-аппаратных комплексов.

Открытую систему сегодня определяют как исчерпывающий и согласованный набор международных стандартов на информационные технологии и профили функциональных стандартов, которые реализуют открытые спецификации на интерфейсы, службы и поддерживающие их форматы, чтобы обеспечить взаимодействие (интероперабельность) и мобильность программных приложений, данных и персонала.

Это определение, сформулированное специалистами Комитета IEEE POSIX 1003.0 Института инженеров по электротехнике и электронике (IEEE), унифицирует содержание среды, которую предоставляет открытая система для широкого использования. Базовым в этом определении является термин «открытая спецификация», имеющий следующее.толкование: «это общедоступная спецификация, которая поддерживается открытым, гласным, согласительным процессом, направленным на постоянную адаптацию новой технологии, и которая соответствует стандартам». Таким образом, под открытыми системами следует понимать системы, обладающие стандартизованными интерфейсами. Решение проблемы открытости систем основывается на стандартизации интерфейсов систем и протоколов взаимодействия между их компонентами.

В качестве примеров использования технологии открытых систем можно привести технологии фирм «Intel» Plug&Play и USB, а также операционные системы UNIX и (частично) ее основного конкурента — Windows NT. Многие новые продукты сразу разрабатываются в соответствии с требованиями открытых систем, примером тому может служить широко используемый в настоящее время язык программирования Java фирмы «Sun Microsystems».

Общие свойства открытых информационных систем можно сформулировать следующим образом:

• *взаимодействие/интероперабелъностъ* — способность к взаимодействию с другими прикладными системами на локальных и (или) удаленных платформах (технические средства, на которых реализована информационная система, объединяются сетью или сетями различного уровня — от локальной до глобальной);

• *стандартизуемостъ* — ИС проектируются и разрабатываются на основе согласованных международных стандартов и предложений, реализация открытости осуществляется на базе функциональных стандартов (профилей) в области информационных технологий;

• *расширяемость/масштабируемость* — возможность перемещения прикладных программ и передачи данных в системах и средах, которые обладают различными характеристиками производительности и различными функциональными возможностями, возможность добавления новых функций ИС или изменения некоторых уже имеющихся при неизменных остальных функциональных частях ИС;

• *мобильность/переносимость* — обеспечение возможности переноса прикладных программ и данных при модернизации или замене аппаратных платформ ИС и возможности работы с ними специалистов пользующихся ИТ, без их специальной переподготовки при изменениях ИС;

• *дружественность к пользователю* — развитые унифицированные интерфейсы в процессах взаимодействия в системе «пользователь -компьютерное устройство — программное обеспечение», позволяющие работать пользователю, не имеющему специальной системной подготовки.

Все эти общепринятые свойства современных открытых систем, взятые по отдельности, были характерны и для предыдущих поколений ИС и средств вычислительной техники. Новый взгляд на открытые системы определяется тем, что эти черты рассматриваются в совокупности, как взаимосвязанные, и реализуются в комплексе. Это естественно, поскольку все указанные выше свойства дополняют друг друга. Только в такой совокупности возможности открытых систем позволяют решать проблемы проектирования, разработки, внедрения, эксплуатации и развития современных информационных систем.

Проиллюстрируем важность такого подхода на примере важнейшего свойства — *интероперабелъности* (Interoperability) [5]. Ниже перечислены обстоятельства, которые отражают насущные потребности развития областей применения информационных технологий и мотивируют переход к интероперабельным информационным системам и разработке соответствующих стандартов и технических средств.

Функционирование систем в условиях информационной и реализационной неоднородности, распределенности и автономности информационных ресурсов системы. Информационная неоднородность ресурсов заключается в разнообразии их прикладных контекстов (понятий, словарей, семантических правил, отображаемых реальных объектов, видов данных, способов их сбора и обработки, интерфейсов пользователей и т.д.). Реализационная неоднородности источников проявляется в использовании разнообразных компьютерных платформ, средств управления базами данных, моделей данных и знаний, средств программирования и тестирования, операционных систем и т.п.

Интеграция систем. Системы эволюционируют от простых, автономных подсистем к более сложным, интегрированным системам, основанным на требовании взаимодействия компонентов.

Реинжиниринг систем. Эволюция бизнес-процессов — непрерывный процесс, который является неотъемлемой составляющей деятельности организаций. Соответственно, создание системы и ее реконструкция (реинжиниринг) — непрерывный процесс формирования, уточнения требований и проектирования. Система должна быть спроектирована так, чтобы ее ключевые составляющие могли быть реконструированы при сохранении целостности и работоспособности системы.

Трансформация унаследованных систем. Практически любая система после создания и внедрения противодействует изменениям и имеет тенденцию быстрого превращения в бремя организации. Унаследованные системы (Legacy Systems), построенные на «уходящих»

технологиях, архитектурах, платформах, а также программное и информационное обеспечение, при проектировании которых не были предусмотрены нужные меры для их постепенного перерастания в новые системы, требуют перестройки (Legacy Transformation) в соответствии с новыми требованиями бизнес-процессов и технологий. В процессе трансформации необходимо, чтобы новые модули системы и оставшиеся компоненты унаследованных систем сохраняли способность к взаимодействию.

Повторное использование неоднородных информационных ресурсов. Технология разработки информационных систем должна позволять крупномасштабно применять технологию повторного использования информационных ресурсов, которые могут быть «соединены» (т.е. образованы их «интероперабельные сообщества») для производства серий стандартизованных продуктов в определенной прикладной области.

Продление жизненного цикла систем. В условиях исключительно быстрого технологического развития требуются специальные меры, обеспечивающие необходимую продолжительность жизненного цикла продукта, включающего в себя постоянное улучшение его потребительских свойств. При этом новые версии продукта должны поддерживать заявленные функциональности предыдущих версий.

Свойство интероперабельности информационных ресурсов является необходимой предпосылкой удовлетворения перечисленных выше требований.

Таким образом, основной принцип формирования открытых систем состоит в *создании среды,* включающей в себя программные и аппаратурные средства, службы связи, интерфейсы, форматы данных и протоколы. Такая среда в основе имеет развивающиеся доступные и общепризнанные стандарты и обеспечивает значительную степень *взаимодействия* (Inter-operability), *переносимости* (Portability) и *масштабирования* (Scalability) приложений и данных.

Благодаря этим свойствам минимизируются затраты на достижение преемственности и повторного использования накопленного программно-информационного задела при переходе на более совершенные компьютерные платформы, а также интеграция систем и ресурсов в распределенные системы. Экономическая рентабельность реализации на практике концепции открытых систем основывается на том, что переход к открытым технологиям создает наилучшие предпосылки для инвестиций в ИТ, так как благодаря свойствам открытости систем ИТ существенно повышается конечная эффективность их использования.

Принципы создания и использования открытых систем применяются в настоящее время при построении большинства классов систем:вычислительных, информационных, телекоммуникационных, систем управления в реальном масштабе времени, встроенных микропроцессорных систем. В условиях широкого использования интегрированных вычислительно-телекоммуникационных систем принципы открытости составляют основу технологии интеграции, В развитии и применении открытых систем заинтересованы все участники процесса информатизации: пользователи, проектировщики систем и системные интеграторы, производители технических и программных средств вычислительной техники и телекоммуникации. В частности, по встроенным микропроцессорным системам (МПС) в рамках программы ESPRIT существует проект OMI (Open Microprocessor Initiative), направленный на создание коллективной пользовательской библиотеки МПС в соответствии с принципами открытых систем.

В условиях перехода к информационному обществу, когда государственное управление и большинство секторов экономики становятся активными потребителями информационных технологий, а сектор производителей средств и услуг информационных технологий непрерывно растет, проблема развития и применения открытых систем составляет для каждой страны национальную проблему. Так, администрация Клинтона еще в 1993 г. объявила о программе создания Национальной информационной инфраструктуры на принципах открытых систем (National Information Infrastructure Initiative), вкладывала в эту программу большие деньги и содействовала инвестициям со стороны частного сектора. Совет Европы в 1994 г. в своих рекомендациях о путях перехода к информационному обществу (Bangemann Report) подчеркнул, что стандарты открытых систем должны играть

важнейшую роль при создании информационной инфраструктуры общества. Ведется работа по созданию глобальной информационной инфраструктуры, также основанной на принципах открытых систем.

Таким образом, в условиях перехода к информационному обществу технология открытых систем становится основным направлением информационных технологий.

18.2. Международные структуры в области стандартизации информационных технологий

Значение принципа взаимосвязи открытых систем стало осознаваться, когда глобализация экономики и бизнеса в рамках единого экономического пространства Европы привела к необходимости унификации применяемых информационных систем и технологий. Вначале каждая страна или компании развивали свои программные и сетевые концепции и технические средства, которые часто оказывались несовместимыми. Различные концептуальные направления имели свои системы форматов данных и обмена данными, например система SWIFT в банковской сфере, EDIFAST в торговле, промышленности, на транспорте. Из-за различий в протоколах системы были несовместимы и не могли быть интегрированы в единое целое. Таким образом, подобные ситуации дали толчок развитию международной стандартизации в области ИТ.

Область ИТ очень динамична, она характеризуется быстрыми темпами развития. При этом определяющую роль в формировании стратегических ориентиров процесса развития играют глобальные концепции. К важнейшим глобальным концепциям, прежде всего относятся концепция открытых систем и концепция Глобальной информационной инфраструктуры (Global Information Infractracture — GII), которые для практического воплощения требуют не только развитой научно-методической базы и всеобъемлющей системы стандартов, но и сами могут рассматриваться как вехи процесса стандартизации ИТ. Отсюда следует, что процесс стандартизации ИТ также приобрел глобальный характер. Его целью является полномасштабная комплексная стандартизация ИТ.

Интенсивность усилий в области стандартизации ИТ в мировом масштабе обеспечила развитие соответствующей системы стандартов до такого уровня, когда она становится главным носителем научно-методических основ области ИТ. На этом пути получены фундаментальные нормативно-методические решения, в частности созданы стандарты, определяющие:

- глобальные концепции развития области ИТ;
- концептуальный базис и эталонные модели построения основных разделов ИТ;
- функции, протоколы взаимодействия, интерфейсы и другие аспекты ИТ;
- языки программирования, языки спецификации информационных ресурсов, языки управления базами данных;
- модели технологических процессов создания и использования систем ИТ, а также языки описания таких моделей;
- методы тестирования соответствия (конформности) систем ИТ исходным стандартам и профилям;
- методы и процедуры функционирования собственно системы стандартов ИТ;
- метаязыки и нотации для описания стандартов ИТ;
- общесистемные функции ИТ, например безопасность, администрирование, интернационализация, качество сервисов; и пр.

Состояние и развитие стандартизации в области информационных технологий характеризуются в настоящее время следующими особенностями:

- несколько сотен международных и национальных стандартов не полностью и неравномерно удовлетворяют потребности в стандартизации объектов и процессов создания и применения сложных ИС;

• длительные сроки разработки, согласования и утверждения международных и национальных стандартов (3—5 лет) приводят к их консерватизму и хроническому отставанию от современных технологий создания сложных ИС;

• стандарты современных ИС должны учитывать необходимость построения ИС как открытых систем, обеспечивать их расширяемость при наращивании или изменении выполняемых функций (переносимость программного обеспечения и возможность взаимодействия с другими ИС);

• в области ИС функциональными стандартами поддержаны и регламентированы только самые простые объекты и рутинные, массовые процессы (телекоммуникация, программирование, документирование программ и данных);

• наиболее сложные и творческие процессы создания и развития крупных распределенных ИС (системный анализ и проектирование, интеграция компонентов и систем, испытания и сертификация ИС и т.п.) почти не поддержаны требованиями и рекомендациями стандартов из-за разнообразия содержания, трудности их формализации и унификации;

• пробелы и задержки в подготовке и издании стандартов высокого ранга, а также текущая потребность в унификации и регламентировании современных объектов и процессов в области ИС приводят к созданию многочисленных нормативных и методических документов отраслевого, ведомственного или фирменного уровней;

• последующие селекция, совершенствование и согласование нормативных и методических документов в ряде случаев позволяют создать на их основе национальные и международные стандарты.

Приведенный список проблем в общем виде очерчивает поле деятельности в области международной стандартизации.

В определении среды открытых систем, приведенном выше, следует обратить внимание на то, что среда в своей основе имеет развивающиеся, доступные и общепризнанные стандарты. Это означает, что очень важен механизм выработки стандартов их согласование или гармонизация. Вопросами разработки стандартов и спецификаций в области информационных технологий занимаются во всем мире более 300 организаций, которые можно разделить на три категории: аккредитованные организации по стандартизации, производители и группы пользователей. Внутри каждой из этих категорий организации объединяются между собой, в том числе в различные ассоциации и консорциумы, организации всех этих категорий участвуют в сложном и дорогостоящем процессе выработки стандартов по принципам рабочих групп (Workshop).

На рис. 18.1 представлена система авторитетных международных организаций (ISO — Международная организация стандартизации, IEC — Международная электротехническая комиссия), играющие значительную роль в решении задач стандартизации ИТ.

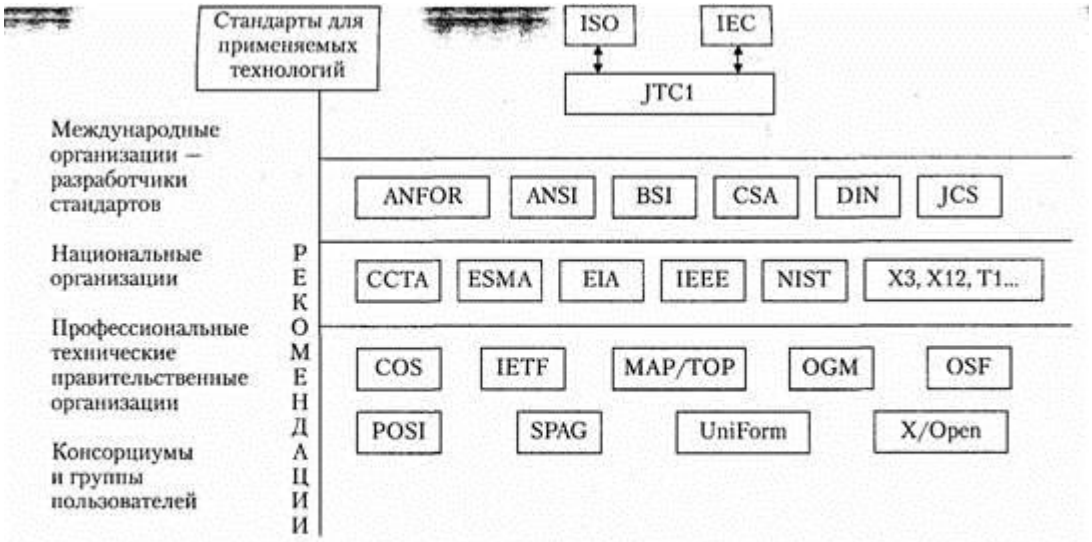

Рис. 18.1. **Международные организации и консорциумы – разработчики стандартов**

В этой деятельности участвуют также многие специализированные профессиональные организации:

• IEEE (Институт инженеров по электротехнике и электронике — международная организация — разработчик ряда важных международных стандартов в области ИТ);

• CEN (Европейский комитет стандартизации широкого спектра товаров, услуг и технологий, в том числе связанных с областью разработки ИТ, аналогISO);

• CENELEC (Европейский комитет стандартизации решений в электротехнике, в частности стандартизации коммуникационных кабелей, волоконной оптики и электронных приборов — аналог IEC);

• ETSI (Европейский институт стандартизации в области сетевой инфраструктуры — аналог ITU-T);

• OMG (Группа объектно-ориентрованного управления — крупнейший международный консорциум, осуществляющий разработку стандартов для создания унифицированного распределенного объектного программного обеспечения, включающий в себя свыше 600 компаний — производителей программного продукта, разработчиков прикладных систем и конечных пользователей);

• ECMA (Европейская ассоциация производителей вычислительных машин — международная ассоциация, целью которой служит промышленная стандартизация информационных и коммуникационных систем); и др.

В 1987 г. ISO и IEC объединили свою деятельность в области стандартизации ИТ, создав единый орган JTC1 (Joint Technical Committee 1 — Объединенный технический комитет 1), предназначенный для формирования всеобъемлющей системы базовых стандартов в области ИТ и их расширений для конкретных сфер деятельности.

К основным целям Комитета JTC1 относятся разработка, поддержание, продвижение стандартов ИТ, являющихся необходимыми для глобального рынка, удовлетворяющих требованиям бизнеса и пользователей и имеющих отношение:

• к проектированию и разработке систем и средств ИТ;
• производительности и качеству продуктов и систем ИТ;
• безопасности систем ИТ и информации;
• переносимости прикладных программ;
• интероперабельности продуктов и систем ИТ;
• унифицированным средствам и окружениям;
• гармонизированному словарю понятий области ИТ;

• дружеским и эргономичным пользовательским интерфейсам. Работа над стандартами ИТ в JTC1 тематически распределена по

подкомитетам (Subcommittees — SC). Ниже показаны подкомитеты и группы JTC1, связанные с разработкой стандартов ИТ, относящихся к окружению открытых систем OSE (Open Systems Environment).

• C2 — символьные наборы и кодирование информации;

• SC6 — телекоммуникация и информационный обмен между системами;

• SC7 — разработка программного обеспечения и системная документация;

• SC18 — текстовые и офисные системы;

• SC21 — открытая распределенная обработка (Open Distributed Processing — ODP), управление данными (Data Management — DM) и взаимосвязь открытых систем (Open System Interconnection — OSI);

• SC22 — языки программирования, их окружения и интерфейсы системного программного обеспечения;

• SC24 — компьютерная графика;

• SC27 — общие методы безопасности для ИТ-приложений;

• SGFS — специальная группа по функциональным стандартам. Результатом целенаправленной деятельности по стандартизации

явилось создание развитой системы стандартов, охватывающей весь спектр основных направлений ИТ. Диапазон этих решений включает в себя методические руководства, глобальные концепции развития области ИТ, основополагающие эталонные модели, а также спецификации типовых аспектов разработки, тестирования, функционирования, использования систем ИТ.

Характерная особенность стандартов ИТ состоит в том, что они содержат определения основных понятий и терминов области ИТ, описания моделей, сценариев, функций, правил поведения и представления информации. По существу, в стандартах ИТ свойства систем ИТ представляются в виде концептуальных, функциональных, информационных моделей объектов стандартизации.

18.3. Методологический базис открытых систем

Процесс стандартизации информационных технологий должен иметь методологическое основание, которое позволило бы обоснованно определять методы и объекты стандартизации. При этом понятие «информационные технологии» трактуется следующим образом: «Информационные технологии включают в себя спецификацию, проектирование и разработку систем и средств, имеющих дело со сбором, представлением, обработкой, безопасностью, передачей, организацией, хранением и поиском информации, а также обменом и управлением информацией».

Такое толкование и единая методологическая база связаны с общими принципами построения информационных систем и применяемыми средствами анализа и разработки. Она реализована в виде методологического базиса открытых систем [7].

Методологический базис информационных технологий представляет собой основу для создания наиболее экономически рентабельных технологий и систем, удовлетворяющих свойствам открытости. Наиболее значительными результатами в становлении методологического базиса открытых систем сегодня являются:

• создание системы специализированных международных организаций по целостной разработке и стандартизации открытых систем;

• разработка эталонных моделей и соответствующих им базовых спецификаций для важнейших разделов области ИТ, что позволило сформировать концептуальный и функциональный базис пространства ИТ/ИС;

• разработка и широкое использование концепции профиля, предоставляющей аппарат для спецификации и документирования сложных и многопрофильных открытых ИТ/ИС, задающих функциональности базовых спецификаций и (или) профилей;

• разработка таксономии профилей, представляющей собой классификационную систему ИТ/ИС и обеспечивающую систематическую идентификацию профилей в пространстве ИТ/ИС;

• разработка концепции и методологии соответствия реализаций ИТ/ИС тем спецификациям, которые ими реализуются.

Методологический базис информационных технологий, основную часть которого составляют спецификации ИТ различных уровней абстракции, формируется на основе иерархического подхода, что способствует анализу его структуры с помощью некоторой многоуровневой модели. На рис. 18.2 показана, модель, представляющая собой достаточно полную классификационную схему спецификаций ИТ.

В данной модели выделены следующие уровни спецификаций информационных технологий:

• *концептуальный уровень (уровень метазнаний)* — состоит из архитектурных спецификаций, называемых эталонными моделями (Reference Model), которые предназначены для структуризации спецификаций функций, определяющих семантику конкретных областей информационных технологий;

• *функциональный уровень, или уровень базовых спецификаций (базовых стандартов),* — включает в себя также PAS и предназначен для определения индивидуальных функций или наборов функций, описанных в эталонных моделях;

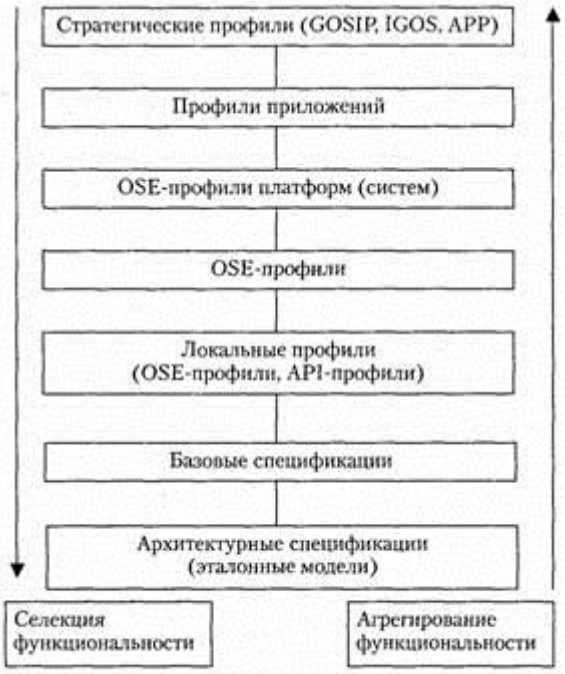

Рис. 18.2. **Многоуровневая модель пространства спецификаций**

• *предметные, или локальные, профили* ЯГ (например, OSI-профили, API-профили), т.е. профили, разрабатываемые на основе использования базовых спецификаций, которые относятся к предметной области, описанной одной эталонной моделью (возможно вместе с профилями форматов данных, т.е. F-профилями);

• *OSE-профили* — спецификации поведения открытых систем на их границах (интерфейсах), объединяющие базовые спецификации и (или) профили, базирующиеся на различных эталонных моделях, в Целевые комплексы;

• *полные OSE-профили открытых платформ и систем* — спецификации, предназначенные для описания поведения ИТ-систем на всех их интерфейсах;

• OSE-профили прикладных технологий — полная спецификация окружений прикладных технологий обработки данных (например, банковских систем, распределенных

офисных приложений и т.п.), построенных на принципах открытости, т.е. удовлетворяющих условиям переносимости, интероперабельности, масштабируемости;

• *стратегические профили (например, International Standardized Profiles* — IPS, Government Open System Interconnection Profile — GOSIP), т.е. профили, рассматриваемые в данном случае не как спецификации одной технологии, а как наборы стандартов, определяющих техническую политику в области телекоммуникации и открытых технологий крупной организации или даже государства.

18.4. Архитектурные спецификации (эталонные модели)

Метод архитектурных спецификаций применяется для формирования концептуального базиса и определения семантической структуры важнейших разделов ИТ. Как правило, базис реализуется посредством разработки эталонных моделей, образующих методологическое ядро ИТ. Эти модели определяют структуризацию конкретных разделов ИТ, задавая тем самым контекст разработки соответствующих этим разделам стандартов. Эталонные модели могут рассматриваться в качестве фундаментальных моделей (законов) в пространстве ИТ (информационной материи).

Эталонные модели определяют архитектуру наиболее важных и достаточно независимых разделов ИТ. Таким образом, каждая эталонная модель представляет собой концептуальный и методологический базис конкретного раздела ИТ, определяя структуру множества базовых спецификаций, соответствующих данному разделу. Наиболее известными эталонными моделями являются (в квадратных скобках приведена ссылка на соответствующий стандарт, описывающий эталонную модель) [6]:

1. Базовая эталонная модель взаимосвязи открытых систем (Basic Reference Model for Open Systems Interconnection — RM-OSI) [ISO 7498:1984, Information processing systems — Open Systems Interconnection, Basic Reference Model, ITU-T Rec. X.200 (1994)].

2. Руководство по окружению открытых систем POSIX (Portable Operating System Interface for Computer Environments — RM API) [ISO/IEC DTR 14252, Portable Operating System Interface for Computer Environments - POSIX-IEEE, P1003.0, Draft Guide to the POSIX Open System Environment, February 1995].

3. Эталонная модель для открытой распределенной обработки (Reference Model for Open Distributed Processing - RM-ODP) [ITU-T Rec. 902|ISO/IEC 10746-2:1995, Reference Model for Open Distributed Processing].

4. Эталонная модель управления данными (Reference Model for Data Management- RM DF) [DIS 9075:1992, Information technology -Reference Model for Data Management].

5. Эталонная модель компьютерной графики (Reference Model of Computer Graphics- RM CG) [ISO/IEC 11072:1992, Information Technology. Computer Graphics — Computer Graphics Reference Model].

6. Эталонная модель текстовых и офисных систем (Text and Office Systems Reference Model) [ISO/IEC TRTOSM-1, Information technology. Text and office systems reference model — Part 1. Basic reference model].

7. Общая модель распределенных офисных приложений [ISO/IEC 10031/1:1991, Information technology— Text communication — Distributed-office-applications model — Part 1. General model].

В процессе разработки находятся следующие эталонные модели:

• модель конформности (Coformality — соответствия, подобия) и методы тестирования конформности, называемые также методами аттестационного тестирования;

• модель основ общей безопасности (Generic Security Frameworks);

• модель качества OSI-сервиса (Quality of Service for OSI).

18.5. Эталонная модель взаимосвязи открытых систем

18.5.1. Эталонная модель среды открытых систем (модель OSE)

Требование совместимости и взаимодействия прикладных программ привело к разработке системы стандартов «Интерфейс переносимой операционной системы» (свод POSIX-стандартов) и стандартов коммуникаций. Однако эти стандарты не охватывают требуемый спектр потребностей даже в рамках установленной для них области распространения. Дальнейшее развитие стандартизации в области информационных технологий и формирования принципа открытых систем нашло выражение в создании функциональной среды открытых систем (Open Systems Environment — OSE) и построении соответствующей модели, которая охватывала бы стандарты и спецификации по обеспечению возможностей ИТ [2].

Модель ориентирована на руководителей ИТ-служб и менеджеров проектов, ответственных за приобретение, внедрение, эксплуатацию и развитие информационных систем, состоящих их неоднородных программно-аппаратных и коммуникационных средств. Прикладные программы в среде OSE могут включать в себя:

• системы реального времени (Real Time System — RTS) и встроенные системы (Embedded System — ES);

• системы обработки транзакций (Transaction Processing System — TPS);

• системы управления базами данных (DataBase Management System - DBM);

• разнообразные системы поддержки принятия решения (Decision Support System - DSS);

• управленческие информационные системы административного (Executive Information System — EIS) и производственного (Enterprise Resource Planning — ERP) назначения;

• географические информационные системы (Geographic Information System — GIS);

• другие системы, в которых могут применяться рекомендуемые международными организациями спецификации.

С точки зрения производителей и пользователей OSE является достаточно универсальной функциональной инфраструктурой, регламентирующей и облегчающей разработку или приобретение, эксплуатацию и сопровождение прикладных защищенных систем, которые:

• выполняются на любой используемой платформе поставщика или пользователя;

• используют любую ОС;

• обеспечивают доступ к базе данных и управление данными;

• обмениваются данными и взаимодействуют через сети любых поставщиков и в локальных сетях потребителей;

• взаимодействуют с пользователями через стандартные интерфейсы в системе общего интерфейса «пользователь — компьютер».

Таким образом, OSE поддерживает переносимые, масштабируемые и взаимодействующие прикладные компьютерные программы через стандартные функциональности, интерфейсы, форматы данных, протоколы обмена и доступа. Стандартами могут быть международные, национальные и другие общедоступные спецификации и соглашения. Эти стандарты и спецификации доступны любому разработчику, поставщику и пользователю вычислительного и коммуникационного программного обеспечения и оборудования при построении систем и средств, удовлетворяющих критериям OSE.

Прикладные программы и средства OSE *переносимы,* если они реализованы на стандартных платформах и написаны на стандартизованных языках программирования. Они работают со стандартными интерфейсами, которые связывают их с вычислительной средой,

читают и создают данные в стандартных форматах и передают их в соответствии со стандартными протоколами, выполняющимися в различных вычислительных средах.

Прикладные программы и средства OSE *масштабируемы* в среде различных платформ и сетевых конфигураций — от персональных компьютеров до мощных серверов, от локальных систем распараллеленных вычислений до крупных GRID-систем. Разницу в объемах вычислительных ресурсов на любой платформе пользователь может заметить по некоторым косвенным признакам, например по скорости выполнения прикладной программы, но никогда — по отказам работы системы.

Прикладные программы и средства OSE *взаимодействуют* друг с другом, если они предоставляют услуги пользователю, используя стандартные протоколы, форматы обмена данными и интерфейсы систем совместной или распределенной обработки данных для целенаправленного использования информации. Процесс передачи информации с одной платформы на другую через локальную вычислительную сеть или комбинацию любых сетей (вплоть до глобальных) должен быть абсолютно прозрачен для прикладных программ и пользователей и не вызывать технических трудностей при использовании. При этом местонахождение и расположение других платформ, операционных систем, баз данных, программ и пользователей не должно иметь значения для используемого прикладного средства.

Рабочая группа 1003.0 POSIX IEEE разработала эталонную модель OSE (Open Systems Environment / Reference Model — OSE/RM). Эта модель описана на международном уровне в техническом отчете TR 14250 комитета JTC1 (рис. 18.3).

В ее описании используется два типа элементов:

• логические объекты, включающие в себя ППО, прикладные платформы и внешнюю функциональную среду;

• интерфейсы, содержащие интерфейс прикладной системы и интерфейс обмена с внешней средой.

Логические объекты представлены тремя классами, интерфейсы — двумя.

В контексте эталонной модели OSE прикладное программное обеспечение включает в себя непосредственно коды программ, данные, документацию, тестирующие, вспомогательные и обучающие средства.

Прикладная платформа состоит из совокупности программно-аппаратных компонентов, реализующих системные услуги, которые используются ППО.

Внешняя среда платформ состоит из элементов, внешних по отношению к ППО и прикладной платформе (например, внешние периферийные устройства, услуги других платформ, операционных систем или сетевых устройств).

Интерфейс прикладной программы (Application Program Interface — API) является интерфейсом между ППО и прикладной платформой. Основная функция API — поддержка переносимости ППО. Классификация API производится в зависимости от типа реализуемых услуг: взаимодействие в системе «пользователь — компьютер», обмен информацией между приложениями, внутренние услуги системы, коммуникационные услуги.

Интерфейс обмена с внешней средой (External Environment Interface — EEI) обеспечивает передачу информации между прикладной платформой и внешней средой, а также между прикладными программами, которые выполняются на одной платформе.

Рис. 18.3. **Эталонная модель среды открытых систем**

Эталонная модель OSE/RM реализует и регулирует взаимоотношения «поставщик — пользователь». Логические объекты прикладной платформы и внешней среды являются поставщиком услуг, ППО — пользователем. Среда OSE обеспечивает функционирование ППО, используя определенные правила, компоненты, методы сопряжения элементов системы (Plug Compatibility) и модульный подход к разработке программных и информационных систем. Достоинствами модели являются выделение внешней среды в самостоятельный элемент, имеющий определенные функции и соответствующий интерфейс, и возможность ее применения для описания систем, построенных на основе архитектуры «клиент-сервер». Относительный недостаток — еще не все требуемые спецификации представлены на уровне международных гармонизированных стандартов.

18.5.2. Базовая эталонная модель взаимосвязи открытых систем (модель OSI)

Обобщенная структура любой программной или информационной системы может быть представлена, как было отмечено выше, двумя взаимодействующими частями:

1) функциональной части, включающей в себя прикладные программы, которые реализуют функции прикладной области;

2) среды или системной части, обеспечивающей исполнение прикладных программ.

С этим разделением и обеспечением взаимосвязи тесно связаны две группы вопросов стандартизации:

1) стандарты интерфейсов взаимодействия прикладных программ со средой ИС, прикладной программный интерфейс (Application Program Interface — API);

2) стандарты интерфейсов взаимодействия самой ИС с внешней для нее средой (External Environment Interface — EEI).

Эти две группы интерфейсов определяют спецификации внешнего описания среды ИС — архитектуру, с точки зрения конечного пользователя, проектировщика ИС, прикладного программиста, разрабатывающего функциональные части ИС.

Спецификации внешних интерфейсов среды ИС и интерфейсов взаимодействия между компонентами самой среды — это точные описания всех необходимых функций, служб и форматов определенного интерфейса. Совокупность таких описаний составляет эталонную модель взаимосвязи открытых систем.

Эта модель используется более 20 лет, она «выросла» из сетевой архитектуры SNA (System Network Architecture), предложенной компанией «IBM». Модель взаимосвязи открытых систем OSI (Open Systems Interconnection) используется в качестве основы для разработки многих стандартов ISO в области ИТ. Публикация этого стандарта

подвела итог многолетней работы многих известных телекоммуникационных компаний и стандартизующих организаций.

В 1984 г. модель получила статус международного стандарта ISO 7498, а в 1993 г. вышло расширенное и дополненное издание ISO 7498-1-93. Стандарт имеет составной заголовок «Информационно-вычислительные системы — Взаимосвязь (взаимодействие) открытых систем — Эталонная модель». Краткое название — «Эталонная модель взаимосвязи (взаимодействия) открытых систем» (Open Systems Interconnection / Basic Reference Model — OSI/BRM).

Модель основана на разбиении вычислительной среды на семь уровней, взаимодействие между которыми описывается соответствующими стандартами и обеспечивает связь уровней вне зависимости от внутреннего построения уровня в каждой конкретной реализации (рис. 18.4). Основным достоинством этой модели является детальное описание связей в среде с точки зрения технических устройств и коммуникационных взаимодействий. Вместе с тем она не принимает в расчет взаимосвязь с учетом мобильности прикладного программного обеспечения.

Рис. 18.4. **Семиуровневая модель взаимодействия (взаимосвязи) ИС**

Преимущества «слоистой» организации модели взаимодействия заключаются в том, что она обеспечивает независимую разработку уровневых стандартов, модульность разработок аппаратуры и программного обеспечения информационно-вычислительных систем и способствует тем самым техническому прогрессу в этой области.

В соответствии с ISO 7498 выделяют семь уровней (слоев) информационного взаимодействия, которые отделены друг от друга стандартными интерфейсами:

1) уровень приложения (прикладной уровень);
2) уровень представления;
3) сеансовый (уровень сессии);
4) транспортный;
5) сетевой;
6) канальный;
7) физический.

Таким образом, информационное взаимодействие двух или более систем представляет собой совокупность информационных взаимодействий уровневых подсистем, причем каждый слой локальной информационной системы взаимодействует, как правило, с соответствующим слоем удаленной системы.

Протоколом является набор алгоритмов (правил) взаимодействия объектов одноименных уровней различных систем.

Интерфейс — это совокупность правил, в соответствии с которыми осуществляется взаимодействие с объектом данного или другого уровня. Стандартный интерфейс в некоторых спецификациях может называться услугой.

Инкапсуляция — это процесс помещения фрагментированных блоков данных одного уровня в блоки данных другого уровня.

При разбиении среды на уровни соблюдались следующие принципы:

• не создавать слишком много мелких разбиений, так как это усложняет описание системы взаимодействий;

• формировать уровень из легко локализуемых функций — это в случае необходимости позволяет быстро перестраивать уровень и существенно изменить его протоколы для использования новых решений в области архитектуры, программно-аппаратных средств, языков программирования, сетевых структур, не изменяя при этом стандартные интерфейсы взаимодействия и доступа;

• располагать на одном уровне аналогичные функции;

• создавать отдельные уровни для выполнения таких функций, которые явно различаются по реализующим их действиям или техническим решениям;

• проводить границу между уровнями в таком месте, где описание услуг является наименьшим, а число операций взаимодействий через границу (пересечение границы) сведено к минимуму;

• проводить границу между уровнями в таком месте, где в определенный момент должен существовать соответствующий стандартный интерфейс.

Каждый уровень имеет *протокольную спецификацию,* т.е. набор правил, управляющих взаимодействием равноправных процессов одного и того же уровня, и *перечень услуг,* которые описывают стандартный интерфейс с расположенным выше уровнем. Каждый уровень использует услуги расположенного ниже уровня, каждый расположенный ниже предоставляет услуги расположенному выше. Приведем краткую характеристику каждого уровня.

Уровень 1 — уровень приложения (прикладной уровень). Этот уровень связан с прикладными процессами. Протоколы предназначены для обеспечения доступа к ресурсам сети и программам-приложениям пользователя. На данном уровне определяется интерфейс с коммуникационной частью приложений. В качестве примера можно привести протокол Telnet, который обеспечивает доступ пользователя к хосту (глазному вычислительному устройству, одному из основных Элементов в многомашинной системе, или любому устройству, подключенному к сети и использующему протоколы TCP/IP) в режиме удаленного терминала.

Уровень 2 — уровень представления. На этом уровне информация преобразуется к такому виду, в каком это требуется для выполнения прикладных процессов. Например, выполняются алгоритмы преобразования формата представления данных — ASC II или КОИ-8. Если для представления данных используется дисплей, то эти данные по заданному алгоритму формируются в виде страницы, которая выводится на экран.

Уровень 3 — сеансовый уровень (уровень сессии). На данном уровне устанавливаются, обслуживаются и прекращаются сессии между представительными объектами приложений (прикладными процессами). В качестве примера протокола сеансового уровня можно рассмотреть протокол RPC (Remote Procedure Call). Как следует из названия, данный протокол предназначен для отображения результатов выполнения процедуры на удаленном хосте. В процессе выполнения этой процедуры между приложениями устанавливается сеансовое соединение. Назначением данного соединения является обслуживание запросов, которые возникают, например, при взаимодействии приложения-сервера с приложением-клиентом.

Уровень 4 — транспортный уровень. Этот уровень предназначен для управления потоками сообщений и сигналов. Управление потоком является важной функцией

транспортных протоколов, поскольку этот механизм позволяет надежно обеспечивать передачу данных по сетям с разнородной структурой, при этом в описание маршрута включаются все компоненты коммуникационной системы, обеспечивающие передачу данных на всем пути от устройств отправителя до приемных устройств получателя. Управление потоком заключается в обязательном ожидании передатчиком подтверждения приема обусловленного числа сегментов приемником. Число сегментов, которое передатчик может отправить без подтверждения их получения от приемника, называется окном.

Существует два типа протоколов транспортного уровня: сегментирующие и дейтаграммные. *Сегментирующие* протоколы транспортного уровня разбивают исходное сообщение на блоки данных транспортного уровня — сегменты. Основной функцией таких протоколов является обеспечение доставки этих сегментов до объекта назначения и восстановление сообщения. *Дейтаграммные* протоколы не сегментируют сообщение, они отправляют его одним пакетом вместе с адресной информацией. Пакет данных — дейтаграмма (Datagram) маршрутизируется в сетях с переключением адресов или передается по локальной сети прикладной программе или пользователю.

Уровень 5 — сетевой уровень. Основной задачей протоколов сетевого уровня является определение пути, который будет использован для доставки пакетов данных при работе протоколов верхних уровней. Для того чтобы пакет был доставлен до какого-либо хоста, этому хосту должен быть поставлен в соответствие известный передатчику сетевой адрес. Группы хостов, объединенные по территориальному принципу, образуют сети. Для упрощения задачи маршрутизации сетевой адрес хоста составляется из двух частей: адреса сети и адреса хоста. Таким образом, задача маршрутизации распадается на две: поиск сети и поиск хоста в этой сети.

Уровень 6 — канальный уровень (уровень звена данных). Назначением протоколов канального уровня является обеспечение передачи данных в среде передачи по физическому носителю. В канале формируется стартовый сигнал передачи данных, организуется начало передачи, производится сама передача, проводится проверка правильности процесса, осуществляется отключение канала при сбоях и восстановление после ликвидации неисправности, формирование сигнала на окончание передачи и перевода канала в ждущий режим.

На канальном уровне данные передаются в виде блоков, которые называются *кадрами.* Тип используемой среды передачи и ее топология во многом определяют вид кадра протокола транспортного уровня, который должен быть использован. При использовании топологии «общая шина» и Point-to-Multipointсредства протокола канального уровня задают физические адреса, с помощью которых будет производиться обмен данными в среде передачи и процедура доступа к этой среде. Примерами таких протоколов являются протоколы Ethernet (в соответствующей части) и HDLC. Протоколы транспортного уровня, которые предназначены для работы в среде типа «точка-точка», не определяют физических адресов и имеют упрощенную процедуру доступа. Примером протокола такого типа является протокол PPP.

Уровень 7 — физический уровень. Протоколы этого уровня обеспечивают непосредственный доступ к среде передачи данных для протоколов канального и последующих уровней. Данные передаются с помощью протоколов данного уровня в виде последовательностей битов (для последовательных протоколов) или групп битов (для параллельных протоколов). На этом уровне определяются набор сигналов, которыми обмениваются системы, параметры этих сигналов (временные и электрические) и последовательность формирования сигналов при выполнении процедуры передачи данных. Кроме того, на данном уровне формулируются требования к электрическим, физическим и механическим характеристикам среды передачи, передающих и соединительных устройств.

Таким образом, эталонная модель взаимосвязи (взаимодействия) открытых систем описывает и реализует стандартизованную систему ^взаимодействия в процессах обмена информацией и данными между прикладными программами и системами в вычислительных

сетях. Стандартизация интерфейсов обеспечивает полную прозрачность взаимодействия вне зависимости от того, каким образом устроены уровни в конкретных реализациях модели.

18.6. Базовые спецификации

Базовые спецификации являются основными строительными блоками, из которых конструируются конкретные открытые технологии, и относятся к понятию «общедоступные спецификации» (Publicly Available Specifications — PAS). Система PAS охватывает стандарты де-факто, которые не являются международными стандартами. Однако сейчас интенсивно осуществляется процесс принятия наиболее распространенных и сопровождаемых PAS в качестве международных стандартов, что открывает возможность использования PAS в качестве элементов стандартизованных профилей ИТ.

Системный подход к проектированию профилей опирается на классификацию базовых спецификаций и PAS, в основе которой используется по существу ортогональный набор эталонных моделей. В частности, ниже приводится возможная классификация базовых спецификаций [8].

Базовые функции ОС: определяются стандартами по окружению открытых систем POSIX (Portable Operating System Interface for Computer Environments) [ISO/IEC 9945/1:1990, (IEEE Std 1003.1 -1990), Information technology. Portable Operating System Interface(POSIX) - Part 1: System Application Program Interface (API) [C Language]].

Функции управления базами данных:

• язык баз данных SQL (Structured Query Language);

• информационно-справочная система IRDS (Information Resource Dictionary System);

• протокол распределенных операций RDA (Remote Database Access);

• PAS Microsoft на открытый прикладной интерфейс доступа к базам данных ODBC API.

Функции пользовательского интерфейса, которые включают в себя следующие стандарты ИТ:

• MOTIF из OSF для графического пользовательского интерфейса

• стандарт OPEN LOOK;

• X Window вместе с GUI и телекоммуникациями;

• стандарты для виртуального терминала (Virtual Terminal — VT), включая процедуры работы VT в символьном режиме через TCP/IP;

• стандарты машинной графики GKS (Graphical Kernel System);

• GKS-3D (Graphical Kernel System - 3 Dimentional);

• PHIGS (Programmers Hierarchical Interactive Graphics System);

• CGI (Computer Graphics Interface).

Функции взаимосвязи открытых систем, включающие в себя:

• спецификации сервиса и протоколов, разработанные в соответствии с моделью OSI (рекомендации серии X.200);

• стандарты для локальных сетей (IEEE 802) [IEEE Std 802-1990];

• спецификации сети Интернет [Transmission Control Protocol (TCP) - RFC 793, User Datagram Protocol (UDP) - RFC 768, Internet Protocol (IP)-RFC 791].

Функции распределенной обработки, включая следующие базовые спецификации OSI:

• вызов удаленной процедуры RPC (Remote Procedure Call);

• фиксация, параллельность и восстановление CCR (Commitment, Concurrency and Recovery);

• протокол надежной передачи (RT);

• обработка распределенной транзакции DTP (Distributed Transaction Processing);

• управление файлами, доступ к файлам и передача файлов FTAM (File Transfer, Access and Management);

• управление открытыми системами (OSI Management);

• API для доступа к сервису Object Request Broker (ORB) в архитектуре CORBA и API, определяющий базовые возможности такого сервиса (Commom ObjectServices — COS);

• язык спецификации интерфейсов объектов IDL (Interface Definition Language) и его проекции на объектно-ориентированные языки.

Распределенные приложения: спецификации специальных сервисных элементов прикладного уровня модели OSI, стандартов Internet OMG, X/Open. Как, например:

• система обработки сообщений MHS (Message Handling System -X.400)],

• служба справочника (The Directory — X.500);

• спецификации распределенных приложений с архитектурой «клиент-сервер» и распределенных объектных приложений.

Структуры данных и документов, форматы данных.

• средства языка ASN. 1 (Abstract Syntax Notation One), предназначенного для спецификации прикладных структур данных — абстрактного синтаксиса прикладных объектов;

• форматы метафайла для представления и передачи графической информации CGM (Computer Graphics Metafile);

• спецификация сообщений и электронных данных для электронного обмена в управлении, коммерции и транспорте EDIFACT (Electronic Data Interchangefor Administration, Commence and Trade);

• спецификации документов — спецификации структур учрежденческих документов ODA (Open Document Architecture);

• спецификации структур документов для производства, например SGML (Standard Generalized Markup Language);

• языки описания документов гипермедиа и мультимедиа, например: HTML (Hypertext Markup Language); HyTime, SMDI»(Standard Music Description Language), SMSL (Standard Multimedia/Hypermedia Scripting Language), SPDS (Standard Page Description Language), DSSSL (Document Style Semantics and Specification Language);

• спецификация форматов графических данных, например форматов JPEG, JBIG и MPEG.

Спецификации инструментальных окружений (в частности, языков реализации и их библиотек) и CASE-окружений (например, ISO/IEC DIS 13719, ECMA Portable Common Tool Environment).

Кроме базовых в настоящее время существуют сотни различных типовых и конкретных спецификаций, разработанных и разрабатываемых в десятках организаций, занимающихся стандартизацией ИТ. Для оценки пригодности и актуальности той или иной спецификации разработана *система оценки спецификаций,* которая предназначена для поставщиков и пользователей. В рамках этой системы каждая спецификация оценивается с позиции ее соответствия некоторым выделенным критериям: степени согласованности, полноте, зрелости, стабильности, степени актуализации, доступности. Например, низкая оценка по степени *согласованности* назначается тем спецификациям, которые являются частной (корпоративной) принадлежностью и используются ограниченной группой поставщиков и пользователей. Напротив, высоко оцениваются спецификации, ставшие общепризнанными национальными или международными стандартами.

Параметр *полноты* оценивает степень, в которой спецификация описывает основные свойства системы, необходимые для обеспечения функциональной области услуг. *Зрелость* указывает на уровень развития соответствующей технологии — высокую оценку получают спецификации, разработанные для хорошо изученных и активно применяемых технологий: эталонная модель достаточно проработана, существуют развитые формализованные математические модели, технологические принципы широко используются на практике. Высокая оценка *стабильности* означает, что спецификация «устоялась» и никаких существенных изменений в ближайшие годы не предвидится. Также высоко оцениваются спецификации, ориентированные на широкий диапазон систем и программных продуктов, *доступных* широкому кругу разработчиков и поставщиков с различными прикладными платформами.

Средние оценки по указанным показателям присваиваются тем спецификациям, которые требуют некоторых дополнительных функций для обеспечения более высокой эффективности применения в предназначенной для них среде. Расширения функционального поля и повышения уровня оценки можно достичь разработкой соответствующего стандарта или включением этой спецификации в состав другой, более «продвинутой» спецификации.

Идентификация спецификаций производится по следующим элементам: имя (наименование) спецификации, дата публикации (дата, когда спецификация стала доступной для общего использования), организация-спонсор (организация, ответственная за разработку и/или поддержание, и/или существование данной спецификации), применимость, степень согласованности, доступность изделия, полнота, зрелость, стабильность, проблемы/ограничения, аттестационное тестирование, привязки, дальнейшие возможности развития, альтернативные спецификации [2].

Анализ базовых спецификаций ИТ показывает, что современная методологическая база открытых систем представляет собой сложную систему концептуальных, структурных, функциональных, поведенческих и лингвистических моделей, взаимосвязанных между собой, а также вспомогательных процедур и средств. При этом следует отметить Динамичность развития всей этой системы, поддерживаемого целенаправленной деятельностью развитой инфраструктуры специализированных международных институтов.

Приведенный обзор базовых спецификаций ИТ является достаточно общим и возможны другие подходы к классификации и анализу спецификаций ИТ. Однако область спецификаций ИТ, несмотря на свою обширность и техническую сложность, легко систематизируется, что важно при использовании спецификаций в процессе разработки новых открытых систем и технологий, например посредством аппарата функциональной стандартизации — профилирования.

Глава 19
ИНСТРУМЕНТЫ ФУНКЦИОНАЛЬНОЙ СТАНДАРТИЗАЦИИ

19.1. Понятие профиля открытой системы

При создании и развитии сложных, распределенных, тиражируемых программных и информационных систем требуется гибкое формирование и применение согласованных (гармонизированных) совокупностей базовых стандартов и нормативных документов разного уровня, выделение в них требований и рекомендаций, необходимых для реализации заданных функций ИС. Для унификации и регламентирования такие совокупности базовых стандартов должны адаптироваться и конкретизироваться применительно к определенным классам проектов, процессов функций и компонентов разрабатываемых систем. В связи с этой потребностью выделилось и сформировалось понятие профиля как основного инструмента функциональной стандартизации.

Профиль — это совокупность нескольких (или подмножество одного) базовых стандартов с четко определенными и гармонизированными подмножествами обязательных и рекомендуемых возможностей, предназначенная для реализации заданной функции или группы функций ИТ/ИС в конкретной функциональной среде. Функциональная характеристика объекта стандартизации является исходной позицией для формирования и применения профиля этого объекта или процесса [4].

Примерами такой среды могут быть среда рабочей станции, управления встроенными вычислительными устройствами, распределенная среда передачи и обработки данных, среда офисного документооборота и т.д. Если все программно-аппаратные и коммуникационные средства, поставляемые различными производителями для использования в рамках целостной ИС, соответствуют профилю, т.е. выполнены в соответствии с необходимыми стандартами, то они будут работать в
единой среде, в которой обеспечена переносимость приложений, масштабирование, взаимодействие и функциональная расширяемость.

Профиль не может противоречить использованным в нем базовым стандартам и нормативным документам. На базе одной совокупности базовых стандартов могут формироваться и утверждаться различные профили для разных проектов разработки программных или информационных систем и сфер их применения. Эти ограничения базовых документов профиля и их гармонизация, проведенная разработчиками профиля, должны обеспечивать качество, совместимость и корректное взаимодействие компонентов системы, соответствующих профилю, в заданной области его применения.

Базовые стандарты и профили могут использоваться как непосредственные директивные, руководящие или рекомендательные документы, а также как нормативная база, необходимая при выборе или разработке средств автоматизации технологических этапов или процессов создания, сопровождения и развития ИС.

Основными целями применения профилей при создании и использовании ИС являются:

• снижение трудоемкости и повышение связности проектов ИС;

• обеспечение переносимости ППО;

• обеспечение расширяемости ИС по набору прикладных функций и масштабируемости;

• предоставление возможности функциональной интеграции в ИС задач, которые раньше решались раздельно и менее эффективно;

• повышение качества компонентов ИС.

Выбор стандартов и документов для формирования конкретных профилей ИС зависит от того, какие из этих целей определены приоритетными.

В качестве методологической базы построения и применения профилей сложных, распределенных ИС предлагается использовать технический отчет ИСО/МЭК'ТО 10000. Части 1.и 2 этого документа введены в России в качестве стандарта ГОСТ Р. Часть 3, определяющую основы и таксономию профилей среды открытых систем, предлагается задействовать при построении и использовании профилей И С как документ прямого применения.

В связи с этим заметим, что международными органами стандартизации ИТ принята жесткая трактовка понятия профиля. На этом уровне считается, что основой профиля могут быть только международные, региональные и национальные утвержденные стандарты — не допускается использование стандартов де-факто и нормативных документов фирм. Подобное понятие профиля активно используется в совокупности международных функциональных стандартов, конкретизирующих и регламентирующих основные процессы и объекты взаимосвязи открытых систем (ВОС), в которых возможна и целесообразна жесткая формализация профилей (например, функциональные стандарты ИСО/МЭК 10607-10613 и соответствующие им ГОСТы Р). Однако при таком подходе невозможны унификация, регламентирование и параметризация множества конкретных функций и характеристик сложных объектов архитектуры и структуры современных развивающихся ИС.

Новый, прагматический подход к разработке и применению профилей ИС состоит в использовании совокупности адаптированных и параметризованных базовых международных и национальных стандартов и открытых спецификаций, отвечающих стандартам де-факто и рекомендациям международных консорциумов.

19.2. Классификация профилей

Существующие базовые профили имеют достаточно жесткую смысловую и иерархическую структуру. По широте охвата области стандартизации, степени признания и области функционального применения профили можно разделить на следующие виды: стратегические (ISP, GOSIP), OSE-профили прикладных технологий, полные OSE-профили (профили платформ, систем), OSE-профили (специализация поведения открытых систем), локальные (OSI-профили).

На верхнем уровне находятся *международные стандартизованные профили* (International Standardized Profiles — ISP), признанные соответствующим комитетом ИСО. В области международной стандартизации ИТ профили ISP имеют такой же статус, что и международные базовые стандарты, и предназначены для широкой области применения.

Определение профиля включает в себя следующие элементы:

• область действия функции, для которой определяется профиль;

• иллюстрация сценария, показывающего пример применения профиля, при этом желательно использование диаграммного представления ИТ-системы, самого приложения и имеющих место интерфейсов;

• нормативные ссылки на набор базовых стандартов или ISP, содержащие точную идентификацию актуальных текстов базовых спецификаций, а также охватывающие принятые дополнения и исправления;

• спецификации применения каждого цитируемого базового стандарта или ISP, устанавливающие выбор классов, подмножеств, опций, диапазонов значений параметров, а также ссылки на регистрируемые объекты;

• раздел, определяющий требования на соответствие данному профилю реализующих его ИТ-систем;

• ссылка на спецификацию аттестационных тестов для реализации данного профиля, если таковые имеют место;

• информативные ссылки на любые полезные, желательно актуализированные документы.

Типовая структура описания ISR FOREWORD // Предисловие. INTRODUCTION//Введение.

1. SCOPE // Область применения + Сценарии.
2. NORMATIVE REFERENCES // Нормативные ссылки.
3. DEFINITIONS//Определения.
4. ABBREVIATIONS // Сокращения.
5. CONFORMANCE // Соответствие.
6. Requirements specifications related to each base standard // Спецификации требований для каждого базового стандарта.

NORMATIVE ANNEXES // Нормативные приложения, задающие требования соответствия профиля в табличном представлении. Л4. INFORMATIVE ANNEXES // Объяснения и руководства, если это требуется.

Требования к содержанию и формату ISR

1) профили непосредственно связаны с базовыми стандартами, и аттестация на соответствие профилю подразумевает аттестацию на соответствие этим базовым стандартам;

2) ISP должен удовлетворять правилам ISO/IEC для представления проектов и самих международных стандартов;

3) ISP должен быть компактным документом, не повторяющим текста документов, на которые он ссылается;

4) определение одного профиля может включать в себя ссылки на определение других;

5) многие профили документируются и публикуются в виде отдельных ISP. Однако для тесно связанных между собой профилей может быть использован более подходящий для такого случая» механизм многокомпонентных ISP (Multipart ISPs). Многокомпонентные ISP позволяют избежать копирования общего текста для связанных профилей;

6) для каждого профиля должна обеспечиваться спецификация тестирования профиля (Profile Test Specification), которая определяется или как часть ISP, или как отдельный самостоятельный ISP. В последнем случае в исходном ISP используется ссылка на этот документ.

В дополнении к ГОСТ Р ISO/IEC TR-10000-1 приводятся правила составления каждого из элементов ISP, соответствующие правилам ISO/IEC. (В случае разбиения ISP на части каждая часть должна удовлетворять этой структуре.)

Ступенькой ниже в иерархии следуют *национальные профили,* в соответствии с которыми должна строиться национальная система ИТ-стандартизации. Несмотря на то что инициатива разработки концепции таких профилей принадлежит Великобритании, примерами наиболее «влиятельных» могут служить профиль переносимости приложений APP (Application Portability Profile — APP), разработанный по заказу Правительства США, а также входящий в него государственный профиль взаимосвязи открытых систем GOSIP (Government Open System Interconnection Profile) (рис. 19.1). Мощным фактором, усилившим престиж GOSIP США, стало то, что в 1990 г. он получил статус федерального стандарта по обработке информации FIPS (Federal Information Processing Standard) и стал обязательным стандартом при разработке и применении соответствующих технологий. Из рисунка видно, что GOSIP строится на базе семиуровневой модели.

В мае 1993 г. Национальным институтом стандартов и технологий США был выпущен документ «Application Portability Profile APP. The U.S. Government's Open System Evironment Profile OSE/1 Version 2.0». Этот документ определяет рекомендуемые для федеральных учреждений США спецификации в области информационных технологий, обеспечивающие мобильность персонала, системных и прикладных программных средств.

Профиль APP строится на основе модели OSE/RM, описанной выше, как профиль открытой среды, предназначенный для использования Правительством США. Он охватывает широкую область прикладных систем, представляющих интерес для многих федеральных агентств. Индивидуальные стандарты и спецификации, входящие в APP, определяют форматы данных, интерфейсы, протоколы и (или) их комбинации.

Все виды функционального обслуживания в рамках APP могут быть представлены следующими семью функциональными областями:

1) функции, реализуемые операционной системой;
2) функции, реализующие человекомашинные интерфейсы;
3) поддержка разработки программного обеспечения;
4) управление данными;
5) обмен данными;
6) компьютерная графика;
7) сетевые функции.

На рис. 19.1 была приведена модель OSE/RM, на которой представлены эти функциональные области и их отношение к элементам модели.

Область функций операционной системы. Функции, реализуемые ОС, обеспечивают управление прикладной платформой, а также обслуживают интерфейсы для взаимодействия прикладных программ и платформы. Область действия этих функций включает в себя:

Функции ядра ОС — функции нижнего уровня, применяются для создания и управления процессами исполнения программ, генерации и передачи сигналов операционной системы, генерации и обработки сигналов системного времени, управления файловой системой и каталогами, управления и обработкой запросов ввода/вывода и обслуживанием внешних устройств.

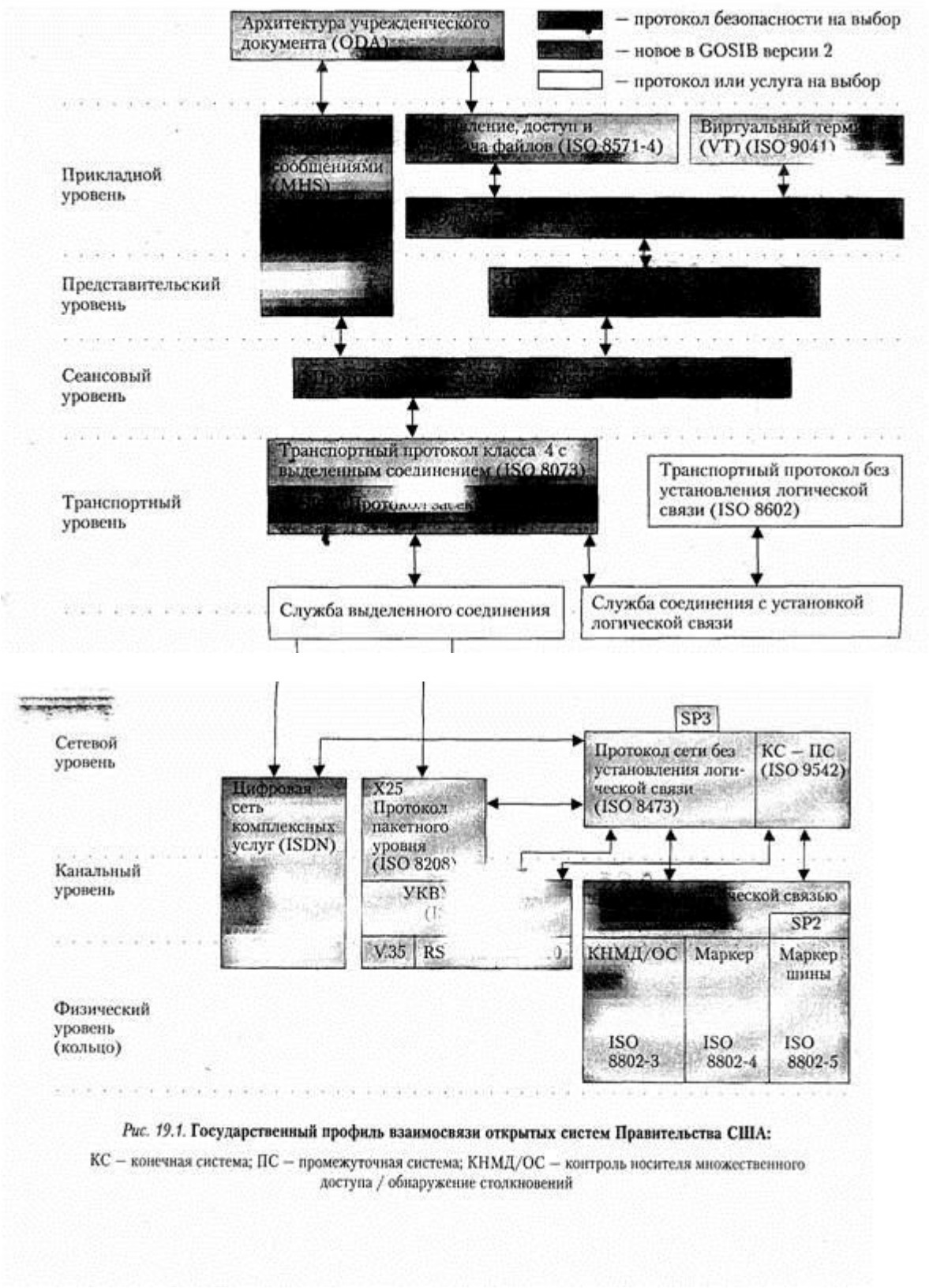

Рис. 19.1. **Государственный профиль взаимосвязи открытых систем Правительства США:**

КС – конечная система; ПС – промежуточная система; КНМД/ОС – контроль носителя множественного доступа / обнаружение столкновений

Команды и утилиты — механизмы для исполнения функций уровня оператора, такие как сравнение, печать и отображение содержимого файлов, редактирование файлов; поиск образцов; регистрация сообщений; перемещение файлов из каталога в каталог; сортировка данных; исполнение командных строк и доступ к служебной информации системы.

Расширение реального времени — функции, реализующие прикладные и системные интерфейсы, которые используются в прикладных областях, требующих детерминированного исполнения, обработки и реакции. Расширения этого типа определяют

прикладные интерфейсы к базовым функциям ОС: ввода/вывода, доступа к файловой системе и управления процессами.

Управление системой — функции, позволяющие создавать и управлять ресурсами, предоставляемыми пользователю, правами доступа к подсистемам и устройствам, управление конфигурацией и производительностью устройств, файловой системы, административными процессами, авторизацией доступа, поддержкой живучести системы.

Человекомашинные интерфейсы. Такие интерфейсы определяют методы, с помощью которых пользователи могут общаться с прикладными системами. В зависимости от различных условий, которые могут определяться как пользователями, так и прикладными системами, интерфейсы этого типа могут обеспечивать следующие функции:

Операции типа «клиент-сервер» — определяют взаимоотношения между процессом «клиент» и процессом «сервер» в сети, в частности между процессами, имеющими место при отображении с помощью графического пользовательского интерфейса. В этом случае программа, которая управляет каждым дисплейным устройством, реализует процесс-сервер, в то время как пользовательская программа представляет процесс-клиент, который запрашивает обслуживания сервером.

Определение объектов и управление — включает в себя спецификации, с помощью которых задаются характеристики отображаемых элементов: цвет, форма, размеры, движение, графические характеристики, взаимодействие между отдельными элементами и т.д.

Параметры окон — спецификации, которые позволяют определить, как создаются окна, передвигаются, сохраняются, восстанавливаются, удаляются и взаимодействуют друг с другом.

Поддержка диалога — спецификации, с помощью которых устанавливаются взаимоотношение между тем, что отображено на экране (в том числе движение курсора, данные, введенные с клавиатуры и дополнительных устройств), и тем, как меняется изображение в зависимости от вводимых данных.

Спецификации мультимедиа, включая API, — спецификации, определения функций и форматов данных, которые поддерживают манипуляции различными формами цифровой и аналоговой аудиовизуальной информации в рамках единой прикладной системы.

Пользовательские интерфейсы являются едва ли не самой сложной областью в разработке и эксплуатации. В течение последних нескольких лет в области технологии пользовательских интерфейсов получены заметные результаты как в части связанной с пользователями, так и в области создания инструментальных средств для построения систем.

Функциональная область поддержки разработки программного обеспечения (программная инженерия). Цель, которую преследует технология открытых систем, — это создание и применение мобильных, гибких, способных настраиваться на различные конфигурации аппаратных платформ, интероперабельных программных средств. Функциональная область программной инженерии обеспечивает для этого необходимую инфраструктуру, в которую входят как языки программирования, так и интегрированные инструментальные системы для поддержки разработки программного обеспечения. В этой функциональной области можно выделить следующие средства:

Стандартные языки программирования вместе со средой поддержки, в которую входят отладчики, средства настройки и оптимизации кода, редакторы.

Интегрированные среды и инструментальные системы для разработки программного обеспечения, включающие в себя системы и программы его автоматизированного создания и поддержки. В состав таких систем входят средства выбора спецификаций и анализа прикладных систем на этапе проектирования, создания и тестирования программного кода, документации и средств поддержки коллективных проектов для групп разработчиков. Интерфейсы, входящие в состав этих средств, обеспечивают хранение, выборку и обмен информацией между программами среды.

Функциональная область управления данными. Центральной задачей большинства систем является управление данными. Системы управления данными реализуют следующие функции.

Обслуживание доступа к словарям и каталогам данных, которые обеспечивают программистам и пользователям доступ к информации о данных (метаданным). Метаданные могут включать в себя внутренние и внешние форматы, правила, обеспечивающие сохранность и секретность, и располагаться в распределенных системах.

Системы управления базами данных обеспечивают управление доступом к структурированным данным и позволяют обращаться к комбинациям данных, расположенным в различных схемах. Базы данных доступны через интерфейсы, входящие в стандарты языков программирования или интерактивные интерфейсы языков четвертого поколения. Также СУБД обычно реализуют особые функции создания, размножения, перемещения, резервного копирования/восстановления и архивации баз данных, хотя некоторые из них предоставляются и файловой системой, относящейся к функциональной области операционных систем.

Функции распределенного доступа обеспечивают обращение к данным, хранящимся в удаленных базах.

Функциональная область обмена данными. Эта область обеспечивает поддержку специфических аспектов обмена информацией, включая форматы данных и их семантику, как для программ, работающих на одной платформе, так и на различных (неоднородных) платформах. Эта область содержит спецификации для функций обмена следующих типов:

Документы — спецификации для кодирования данных (текст, рисунки, числа, специальные символы и т.д.) и как логические, так и визуальные структуры электронных документов.

Графические данные — независимые от устройств определения элементов рисунков.

Производственные данные — спецификации, описывающие технические рисунки, документацию, данные другого типа, необходимые в производстве и проектировании продукции, включая геометрические или негеометрические данные, характеристики форм, допуски, свойства материалов и т.п.

Существуют различные уровни сложности представления данных, используемые в процессе обмена.

Уровень 1 (формат данных) обеспечивает возможность представить данные, участвующие в обмене, явным образом, путем указания формата либо ссылки на язык программирования.

Уровень 2 (объект единого контекста) отображает содержание одиночного объекта. Примерами спецификаций такого типа могут быть тексты, растровые изображения или аудиоинформация.

Уровень 3 (комплексный объект) включает в себя спецификации для представления сложных объектов, состоящих из элементарных объектов, соответствующих уровню 2.

Уровень 4 (семантика и синтаксис языков) — это уровень языка представления данных.

Уровень 5 (прикладной) — уровень приложений, который может использовать любые из нижних уровней для обмена с другими прикладными программами.

Область графических функций. Эта область используется для создания и манипуляций с отображаемыми изображениями. Функции такого рода обеспечивают определение и поддержку отображаемого элемента и его атрибутов. Функции этой области содержатся в спецификациях многомерных графических объектов и изображений в форме, независимой от конкретных устройств. Доступность и целостность как функций, поддерживающих разработку ПО для изображений и графики, а также самих графических данных обеспечивается за счет средств, обеспечивающих безопасность для данной области.

Функциональная область сетевой поддержки. Сетевые функции предоставляют механизмы для поддержки прикладных систем, которым требуется интероперабельный доступ к данным и программам в неоднородной сетевой среде. В этой области имеются следующие механизмы:

Собственно коммуникации, которые включают в себя API и спецификации протоколов для прозрачной и надежной передачи данных через коммуникационные сети.

Прозрачный доступ к файлам, расположенным в любом месте неоднородной сети. *Поддержка* персональных ЭВМ и микро-ЭВМ для обеспечения интероперабельности с системами, базирующимися на различных ОС, в частности микрокомпьютерных, которые могут оказаться не соответствующими международным и национальным стандартам.

Дистанционное обращение к процедурам — спецификации для обращения к процедурам, расположенным во внешней распределенной среде.

Интегрально поддерживаемые функциональные области. К этой области относятся функции, которые интегрируются внутри уже описанных областей, и в рамках данной референтной модели затруднительно выделить в самостоятельные элементы, поскольку эти функции в каждом конкретном случае имманентно связаны с функциями каждой из рассмотренных функциональных областей. К этим функциям относятся функции, обеспечивающие защиту в компьютерной среде и функции системной поддержки и администрации.

Национальные профили GOSIP имеют Великобритания, Франция, Швеция, Япония, Австралия, Гонконг (Сянган). В январе 2000 г. государственный профиль взаимосвязи открытых систем России был утвержден Госстандартом (ГОСТ Р 50.1.22—2000). Этот профиль разработан на основе базовых и функциональных стандартов семиуровневой эталонной модели взаимосвязи открытых систем (ВОС ИСО/ МЭК) с учетом опыта по разработке и применениюGOSIP указанных стран. Вследствие общего отставания России в области разработки ИТ, состояния и развития стандартизации в этой области, уровня применения ИТ/ИС на федеральном уровне государственный профиль ВОС России имеет некоторые заметные отличия от GOSIP других стран. Однако, несмотря на некоторые различия между национальными и региональными версиями GOSIP, их объединяет функциональная идентичность по следующим обстоятельствам:

• проведение единой политики федеральных органов и правительственных департаментов по приобретению, внедрению и эксплуатации вычислительного и коммуникационного оборудования для реализации максимального информационного взаимодействия;

• уменьшение зависимости от поставщиков;

• обеспечение разработчиков, поставщиков и пользователей на федеральном уровне однозначно понимаемыми спецификациями, на основе которых должна формироваться стратегия разработки вычислительных и информационных систем, сетей и систем связи.

Основное преимущество института GOSIP заключается в том, что все протоколы, на которых основаны GOSIP, обладают общими характеристиками, такими, например, как:

• широкая применимость (активное использование не только соответствующими службами отдельных стран, но и международными организациями);

• доступность (реализации уже существуют либо имеются пилотные выпуски);

• стабильность (не планируется внесение существенных изменений в ближайшие 3—4 года);

• эффективность (протоколы удовлетворяют общим потребностям федеральных органов и правительственных учреждений).

Профили следующего уровня — отраслевые или корпоративные. Для каждой отрасли может и должен быть построен свой профиль, например профили банковской деятельности, военного назначения, научных исследований, образования и т.д.

19.3. Основные свойства и назначение профилей

Эталонная модель среды открытых систем (OSE/RM) определяет разделение любой информационной системы на приложения (прикладные программы и программные комплексы) и среду, в которой эти приложения функционируют. Между приложениями и средой определяются стандартизованные интерфейсы, которые являются необходимой частью профилей любой открытой системы. Кроме того, в профилях ИС могут быть

определены унифицированные интерфейсы взаимодействия функциональных частей друг с другом и интерфейсы взаимодействия между компонентами среды ИС.

Классификация интерфейсов открытых систем вводит следующие четыре основных типа интерфейсов OSE:

1) API (Application Program Interface — интерфейс прикладной программы);

2) CSI (Communication Services Interface — интерфейс коммуникационных услуг);

3) HCI (Human/Computer Interface — человекомашинный интерфейс);

4) ISI (Information Services Interface — интерфейс информационных услуг).

Могут быть определены и другие типы интерфейсов, например интерфейс управляемых объектов.

Под API понимается интерфейс между ППО и поставщиком необходимого для функционирования этого программного обеспечения сервиса, т.е. прикладной платформой.

Интерфейс CSI обеспечивает реализацию взаимодействия с внешними системами, которая осуществляется с помощью протоколов (процедур обмена). Стандартизация этих протоколов вместе со стандартизацией форматов обмениваемых данных в них является основой обеспечения интероперабельности систем.

Через интерфейс HCI осуществляется физическое взаимодействие пользователя и системы ИТ. Примерами такого интерфейса служат клавиатуры для ввода информации и оконные системы взаимодействия с пользователем.

Интерфейс ISI рассматривается как граница взаимодействия с внешней памятью долговременного хранения данных, для переносимости и интероперабельности которых необходима стандартизация форматов и синтаксиса представления данных.

Таким образом, определяемая профилем OSE функциональность в общем случае может рассматриваться как композиция функций, или сервисов, реализуемых на интерфейсах определенных выше классов. Функциональность профиля специфицируется в терминах вызовов функций, протоколов взаимодействия, форматов данных. Естественным требованием к профилю является согласованность используемых им спецификаций, относящихся к интерфейсам различных классов.

Полный OSE-профиль — это профиль, который специфицирует все поведение ИТ-системы или часть ее поведения на одном или большем числе интерфейсов OSE. Он состоит из выбранного набора открытых, общедоступных, согласованных стандартов и спецификаций, определяющих различные услуги в среде эталонной модели OSE/RM.

Профиль OSI — конкретный (локальный) профиль, составленный из базовых стандартов, соответствующих модели OSI (Open System Interconnection), и (или) базовых стандартов представления форматов и данных, т.е. F-профилей.

На основании этих определений можно сформулировать следующие общие свойства профилей:

1. Профиль только ограничивает функциональность базовой спецификации благодаря выбору его опций и значений параметров. Таким образом, функциональность профилей вытекает из функциональности выбранных в них базовых стандартов.

2. Профили не могут содержать никаких требований, противоречивых с базовым стандартом, они лишь осуществляют выбор соответствующих опций и диапазонов значений параметров.

. Профиль может содержать дополнительные, более специальные или ограничительные аттестационные требования. Таким образом, аттестация на соответствие профилю подразумевает аттестацию на соответствие всему набору составляющих его спецификаций, в частности базовых стандартов, на которые он ссылается.

Основными целями OSE- и OSI-профилей является реализация основных свойств открытости проектируемой, внедряемой, эксплуатируемой или развиваемой системы. В связи с этим формируемый OSE-профиль должен обеспечивать [6]:

1. Переносимость ППО и многократную используемость ПО на уровне исходного кода и стандартных библиотек (Application Software Portability and SoftwareReuse at the Source Code Level). Именно пере-

носимость между различными платформами исходного текста ПО считается одной из основных практически достижимых задач, решение которой позволяет организациям защитить себя от необходимости дополнительного инвестирования в существующее ПО для его перепроектирования при переходе на новые прикладные платформы. Если под переносимостью приложений понимается перенос всего соответствующего данному приложению ПО на другие платформы, то под его переиспользумостью, как правило, понимается перенос в новые приложения некоторой части работающих программ, что также имеет большое практическое значение и непосредственно относится к целям открытости систем.

Переносимость данных (Data Portability). Не менее важной целью открытых систем является переносимость на новые прикладные платформы данных, хранящихся во внешней памяти существующих систем ИТ, что обеспечивается разработкой OSE на основе стандартов и ISP, строго регламентирующих форматы и способы представления данных.

3. Интероперабельность прикладного программного обеспечения (Application Software Interoperability). Здесь имеется в виду возможность обмена данными между сущностями ПО, в том числе между сущностями, реализуемыми на разнородных прикладных платформах, а также возможность совместного использования ими обмениваемых данных. Данное свойство на нижнем уровне обеспечивается построением стандартизованных коммуникационных интерфейсов, т.е. CSI-интерфейсов, систем на основе стандартов сетевых протоколов, в частности OSI-профилей. Реализация его в полном объеме приводит к необходимости решения проблемы семантической интероперабельности, т.е. понимания разнородными платформами семантики данных, которыми они обмениваются друг с другом.

4. Интероперабельность управления и безопасности (Management and Security Interoperability). Для целей интеграции и совместного использования разнородных платформ в рамках распределенных систем ИТ необходима унификация и концептуальная целостность средств административного управления и управления информационной безопасностью систем ИТ независимо от реализационных окружений. Поэтому для обеспечения бесшовной интеграции систем их средства административного управления и средства защиты должны строиться в соответствии с международными стандартами.

5. Переносимость пользователей (User Portability). Под переносимостью пользователей понимается отсутствие необходимости в их повторном обучении при переносе ППО на другие платформы, что также является одной из важных целей концепции открытых систем.

6. Использование существующих стандартов и аккомодацию к стандартам перспективных технологий (Accommodation of Standards). Профили OSE являются эффективным средством продвижения существующих стандартов в практику. В то же время они являются объектами, способными эволюционировать с учетом изменения стандартов, технологий и пользовательских требований, прежде всего потому, что они конструируются посредством ссылок на базовые стандарты. Таким образом, на основе понятия OSE-профиля поддерживается такое свойство открытых систем, как адаптируемость к изменению стандартов.

7. Легкую настраиваемость на новые технологии создания информационных систем (Accommodation of New Information System Technology). Профили OSE, являясь исходным материалом при построении открытых систем, не связаны непосредственно с нижележащими технологиями. Однако развитие таких технологий влечет развитие сие- . темы стандартов. Гибкость аппарата OSE-профилей позволяет учитывать тенденции перехода к новым стандартам и соответственно к новым технологиям.

8. Масштабируемость прикладных платформ и распределенных систем (Application Platform Scalability). Масштабируемость относится к важнейшим свойствам открытости систем ИТ. Применительно к прикладной платформе оно означает возможность разных типов реализаций некоторого OSE-профиля, отличающихся техническими и

ресурсными характеристиками (например, суперкомпьютеры и рабочие станции), поддерживать одну и ту же функциональность, т.е. один и тот же набор сервисов.

9. Прозрачность реализаций процессов (Implementation Transparency). Данное свойство поддерживается благодаря систематическому использованию через аппарат OSE-профилей стандартизованных спецификаций (стандартов и ISPs), одним из принципов разработки которых является независимость от конкретных реализаций. Таким образом, все особенности реализации OSE-профилей скрываются за интерфейсами открытых систем, что и обеспечивает свойство прозрачности реализаций для конечных пользователей систем ИТ.

10. Поддержку пользовательских требований (Support Clear Statement of User Requirements). Важным свойством открытых систем является точная спецификация пользовательских требований, определенных в виде наборов сервисов, предоставляемых открытыми системами на их интерфейсах. Это свойство адекватно поддерживается применением аппарата OSE-профилей.

При практическом формировании и применении профилей, как было сказано выше, можно использовать региональные, национальные стандарты, стандарты де-факто и ведомственные нормативные документы. В процессе применения стандартов и профилей могут быть выявлены пробелы в положениях некоторых стандартов и необходимость модификации или дополнения требований, определенных в них. Некоторые функции, не формализованные стандартами, но важные для унификации построения или взаимодействия компонентов конкретной технологии или ИС, могут определяться нормативными документами ведомства или фирмы, обязательными для конкретного профиля и проекта. Для эффективного использования конкретного профиля необходимо:

• выделить объединенные логической связью проблемно-ориентированные области функционирования, где могут применяться стандарты, общие для одной организации или их группы;

• идентифицировать стандарты и нормативные документы, варианты их использования и параметры, которые необходимо включить в профиль;

• документально зафиксировать участки конкретного профиля, где требуется создание новых стандартов или нормативных документов, и идентифицировать характеристики, которые могут оказаться важными для разработки недостающих стандартов и нормативных документов этого профиля;

• формализовать профиль в соответствии с его категорией, включая стандарты, различные варианты нормативных документов и дополнительные параметры, которые непосредственно связаны с профилем;

• опубликовать профиль и (или) продвигать его по формальным инстанциям для дальнейшего распространения.

При использовании OSE- и OSI-профилей для создания ИС следует обеспечить проверку корректности их применения путем тестирования, испытаний и сертификации, для чего должны быть создана технология контроля и тестирования в процессе применения профиля. Она должна поддерживаться совокупностью методик, инструментальных средств, составом и содержанием оформляемых документов на каждом этапе обеспечения и контроля корректности применения соответствующей версии и положений профиля.

Использование профилей способствует унификации при разработке тестов, проверяющих качество и взаимодействие компонентов проектируемой ИС. Профили должны определяться таким образом, чтобы тестирование их реализации можно было проводить наиболее полно по стандартизованной методике. Некоторые тесты для проверки соответствия применяемых компонентов международным стандартам могут быть использованы готовыми, так как международные стандарты и профили являются основой при создании международных признанных аттестационных тестов.

19.4. Пример компоновки функционального профиля

Наиболее актуальными в настоящее время представляются открытые распределенные ИС с архитектурой «клиент-сервер». Рассмотрим подходы к построению функциональных профилей таких систем [4].

Профиль среды ИС должен определять ее архитектуру в соответствии с выбранной моделью распределенной обработки данных, например DCE (DistributedComputing Environment) или CORBA (Common Object Request Broker Architecture). В первом случае модель определяется стандартами Консорциума OSF, в частности механизма удаленного вызова процедур RPC (Remote Procedure Call) с учетом стандартов де-факто, которые специфицируют применяемые мониторы транзакций (например, монитор транзакций Tuxedo) [3]. Во втором случае модель определяется стандартами консорциума OMG, в частности спецификацией брокера объектных запросов ORB (Object Request Broker). Стандарты интерфейсов приложений со средой ИС (Application Program Interface — API) должны быть определены по функциональным областям профилей ИС. Декомпозиция структуры среды функционирования ИС на составные части, выполняемая на стадии эскизного проектирования, позволяет детализировать профиль среды ИС по функциональным областям эталонной модели OSE/RM:

• графического пользовательского интерфейса (Motif консорциума «OSF» или стандарт X Window IEEE);

• реляционных или объектно-ориентированных СУБД (например, стандарт языка SQL-92 и спецификации доступа к разным базам данных);

• операционных систем с учетом сетевых функций, выполняемых на уровне ОС (например, набора стандартов POSIX — ISO и IEEE);

• телекоммуникационной среды в части услуг и сервисов прикладного уровня: электронной почты (по рекомендациям ITU-T X.400, X.500), доступа к удаленным базам данных RDA (по стандарту ISO ; 9594-1.2), передачи файлов, доступа к файлам и управления файлами (по стандарту ISO10607 - 1, 2, 3, 4, 5, 6).

Профиль среды распределенной ИС должен включать в себя стандарты протоколов транспортного уровня (по ISO OSI или стандарт де-факто протокола TCP/IP), стандарты локальных сетей (например, стандарт Ethernet IEEE 802.3 или стандарт Fast Ethernet IEEE 802.3 u), а также стандарты средств сопряжения проектируемой ИС с сетями передачи данных общего назначения (например, по рекомендациям ITU-T X.25,X.3,X.29 и др.).

Выбор аппаратных платформ ИС связан с определением их параметров: вычислительной мощности серверов и рабочих станций в соответствии с проектными решениями по разделению функций между клиентами и серверами; степени масштабируемости аппаратных платформ; надежности. Профиль среды И С содержит стандарты, определяющие параметры технических средств и способы их измерения (например, стандартные тесты измерения производительности).

Профиль защиты информации в ИС должен обеспечивать реализацию политики информационной безопасности, разрабатываемой в соответствии с требуемой категорией безопасности и критериями безопасности, заданными в техническом задании на систему [8]. Построение профиля защиты информации в распределенных системах «клиент-сервер» методически связано с точным определением компонентов системы, ответственных за те или иные функции, сервисы и услуги, и средств защиты информации, встроенных в эти компоненты. Функциональная область защиты информации включает в себя функции защиты, реализуемые разными компонентами ИС, а именно функции:

• защиты, реализуемые ОС;

• защиты от несанкционированного доступа, реализуемые на уровне ЦР промежуточного слоя;

• управления данными, реализуемые СУБД;

• защиты программных средств, включая средства защиты от вирусов;

• защиты информации при обмене данными в распределенных системах, включая криптографические функции;

• администрирования средств безопасности. Основополагающим документом в области защиты информации в распределенных системах являются рекомендации X.800, принятые МККТТ (сейчас ITU-T) в 1991 г. Подмножество указанных рекомендаций должно составлять профиль защиты информации в ИС с учетом распределения функций защиты информации по уровням концептуальной модели ИС и взаимосвязи функций и применяемых механизмов защиты информации. При использовании профиля защиты информации при проектировании, разработке и сопровождении ИС целесообразно учитывать методические рекомендации, изложенные в интерпретации «Оранжевой книги» национального центра компьютерной безопасности США для сетевых конфигураций. Профиль защиты информации должен включать в себя указания на методы и средства обнаружения в применяемых аппаратных и программных средствах недекларированных возможностей («закладных» элементов и вирусов). Профиль должен также содержать указания на методы и средства резервного копирования информации и восстановления информации при отказах и сбоях аппаратуры системы.

Профиль инструментальных средств, встроенных в ИС, отображает решения по выбору методологии и технологии создания, сопровождения и развития конкретной ИС, описание которых должно быть выполнено на стадии эскизного проектирования системы. Состав встроенных инструментальных средств определяется на основании решений и нормативных документов об организации сопровождения и развития ИС. При этом должны быть учтены правила и порядок, регламентирующие внесение изменений в действующие системы. Функциональная область профиля инструментальных средств, встроенных в ИС, охватывает функции централизованного управления и администрирования, связанные:

• с контролем производительности и корректности функционирования системы в целом;

• преобразованием конфигурации ППО, тиражированием версий;

• управлением доступом пользователей к ресурсам системы и конфигурацией ресурсов;

• перенастройкой приложений в связи с изменениями прикладных функций ИС;

• настройкой пользовательских интерфейсов (генерация экранных форм и отчетов);

• ведением баз данных системы;

• восстановлением работоспособности системы после сбоев и аварий.

Дополнительные ресурсы, необходимые для функционирования встроенных инструментальных средств (минимальный и рекомендуемый объем оперативной памяти, размеры требуемого пространства на дисковых накопителях и т.д.), учитываются в разделе проекта, относящемся к среде ИС. Выбор инструментальных средств, встроенных в ИС, производится в соответствии с требованиями профиля среды ИС. Ссылки на соответствующие стандарты, входящие в профиль среды, должны быть указаны и в профиле инструментальных средств, встроенных в ИС. В этом профиле также предусмотрены ссылки на требования к средствам тестирования, которые необходимы для процессов сопровождения и развития системы. К встроенным в ИС средствам тестирования относятся средства функционального контроля приложений, проверки интерфейсов, системного тестирования и диагностирования серверов/клиентов при максимальной нагрузке.

Рассмотренный пример еще раз демонстрирует исключительную по важности роль профилей в реализации принципов открытых систем. Построение профиля на самых ранних стадиях создания программной или информационной системы и следование ему на всех дальнейших стадиях жизненного цикла системы позволяет пользователю (заказчику) составлять спецификацию на приобретаемые и разрабатываемые аппаратные и программные средства, как системные, так и прикладные, обеспечивать независимость от конкретного поставщика при создании и модернизации системы. Разработчики приложений, в свою очередь, следуя профилю, создают условия для повторного применения разработанных

приложений при смене платформ, а поставщики средств вычислительной техники — для расширения рынка сбыта.

На основе профиля должно проводиться тестирование и сертификация приложений на соответствие требованиям открытости. Основу большинства применяемых в настоящее время профилей составляют стандарты серии POSIX. Общий их перечень состоит из более 45 наименований. Перечень российских стандартов в области реализации открытых систем составляет 92 стандарта [1].

Раздел VII
БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ СИСТЕМ

Глава 20
ЗАЩИЩЕННАЯ ИНФОРМАЦИОННАЯ СИСТЕМА

20.1. Определение защищенной информационной системы

В отличие от корпоративных сетей, подключенных к Интернет, где обычные средства безопасности в большой степени решают проблемы защиты внутренних сегментов сети от злоумышленников, распределенные корпоративные информационные системы, системы электронной коммерции и предоставления услуг пользователям Интернет предъявляют повышенные требования в плане обеспечения информационной безопасности.

Межсетевые экраны, системы обнаружения атак, сканеры для выявления уязвимостей в узлах сети, ОС и СУБД, фильтры пакетов данных на маршрутизаторах — достаточно ли всего этого мощного арсенала (так называемого «жесткого периметра») для обеспечения безопасности критически важных информационных систем, работающих в Интернет и Интранет? Практика и накопленный к настоящему времени опыт показывают — чаще всего нет!

Концепция «Защищенные информационные системы» включает в себя ряд законодательных инициатив, научных, технических и технологических решений, готовность государственных организаций и компаний использовать их для того, чтобы люди, используя устройства на базе компьютеров и ПО, чувствовали себя так же комфортно и безопасно (табл. 20.1).

В «Оранжевой книге» надежная информационная система определяется как «система, использующая достаточные аппаратные и программные средства, чтобы обеспечить одновременную достоверную обработку информации разной степени секретности различными пользователями или группами пользователей без нарушения прав доступа, целостности и конфиденциальности данных и информации, и поддерживающая свою работоспособность в условиях воздействия на нее совокупности внешних и внутренних угроз» [8].

Таблица 20.1

Трехуровневая модель проблемы защищенности ИС

Система целей	Средства	Исполнение
Общая цель: защищенные информационные системы *Цели:* • безопасность • безотказность • деловое взаимодействие	*Установки:* • защищенность • конфиденциальность • целостность • готовность к работе • точность • управляемость • безотказность • прозрачность • удобство пользования *Подтверждения:* • внутренняя оценка • аккредитация • внешний аудит	*Обеспечение:* • законы, нормы • характер ведения бизнеса • контракты, обязательства • внутренние принципы • международные, отраслевые, и внутренние стандарты *Реализация:* • методы взаимодействия с внешней и внутренней средой • методы работ • анализ рисков • методы разработки, внедрения, эксплуатации и сопровождения • обучение

Это качественное определение содержит необходимое достаточное условие безопасности. При этом не обусловливается, какие механизмы и каким образом реализуют безопасность — практическая реализация зависит от многих факторов: вида и размера бизнеса, предметной области деятельности компании, типа информационной системы, степени ее распределенности и сложности, топологии сетей, используемого ПО и т.д. В общем случае можно говорить о степени доверия, или надежности систем, оцениваемых по двум основным критериям.

1. *Наличие и полнота политики безопасности* — набор внешних и корпоративных стандартов, правил и норм поведения, отвечающих законодательным актам страны и регламентирующих, сбор, обработку, распространение и защиту информации. В частности, стандарты и правила определяют, в каких случаях и каким образом пользователь имеет право оперировать с конкретными наборами данных. В политике сформулированы права и ответственности пользователей и персонала отделов информационной безопасности (ИБ), позволяющие выбирать механизмы защиты безопасности системы. Политика безопасности — это активный компонент защиты, включающий в себя анализ возможных угроз и рисков, выбор мер противодействия и методологию их применения. Чем больше информационная система и чем больше она имеет «входов» и «выходов» (распределенная система), тем «строже», детализированнее и многообразнее должна быть политика безопасности.

2. Гарантированностъ безопасности — мера доверия, которая может быть оказана архитектуре, инфраструктуре, программно-аппаратной реализации системы и методам управления ее конфигурацией и целостностью. Гарантированность может проистекать как из

тестирования и верификации, так и из проверки (системной или эксплуатационной) общего замысла и исполнения системы в целом и ее компонентов. Гарантированность показывает, насколько корректны механизмы, отвечающие за проведение в жизнь политики безопасности. Гарантированность является пассивным, но очень важным компонентом защиты, реализованным качеством разработки, внедрения, эксплуатации и сопровождения информационной системы и заложенных принципов безопасности.

Концепция гарантированности является центральной при оценке степени защиты, с которой информационную систему можно считать надежной. Надежность обеспечивается всей совокупностью защитных механизмов системы в целом и надежностью вычислительной базы (ядра системы), отвечающих за проведение в жизнь политики безопасности. Надежность вычислительной базы определяется ее реализацией и корректностью исходных данных, вводимых административным и операционным персоналом. Компоненты вычислительной базы могут не быть абсолютно надежными, однако это не должно влиять на безопасность системы в целом.

Основное назначение надежной вычислительной базы — выполнять функции монитора обращений и действий, т.е. контролировать допустимость выполнения пользователями определенных операций над объектами. Монитор проверяет каждое обращение к программам или данным на предмет их согласованности со списком допустимых действий. Таким образом, важным средством обеспечения безопасности является механизм подотчетности или протоколирования. Надежная система должна фиксировать все события, касающиеся безопасности, а ведение протоколов дополняется аудитом — анализом регистрационной информации.

Эти общие положения являются основой для проектирования и реализации безопасности открытых информационных систем [4].

20.2. Методология анализа защищенности информационной системы

При разработке архитектуры и создании инфраструктуры корпоративной ИС неизбежно встает вопрос о ее защищенности от угроз. Решение вопроса состоит в подробном анализе таких взаимно пересекающихся видов работ, как реализация ИС и аттестация, аудит и обследование безопасности ИС [1].

Основой формального описания систем защиты традиционно считается модель системы защиты с полным перекрытием, в которой рассматривается взаимодействие области угроз, защищаемой области и системы защиты. Таким образом, модель может быть представлена в виде трех множеств: T= *{ti}* —множество угроз безопасности, *O = {oj}* — множество объектов (ресурсов) защищенной системы, M= *{mk}* — множество механизмов безопасности.

Элементы этих множеств находятся между собой в определенных отношениях, собственно и представляющих систему защиты. Для описания системы защиты обычно используется графовая модель. Множество отношений «угроза — объект» образует двухдольный граф *{T, O}*. Цель защиты состоит в том, чтобы перекрыть все возможные ребра в графе. Это достигается введением третьего набора M; в результате получается трехдольный граф {T, *M, O}*.

Развитие модели предполагает введение еще двух элементов. Пусть *V*— набор уязвимых мест, определяемый подмножеством декартова произведения *{TxO}: vr = <ti, oj>*. Под уязвимостью системы защиты понимают возможность осуществления угрозы *T* в отношении объекта *O*. (На практике под уязвимостью системы защиты обычно понимают те свойства системы, которые либо способствуют успешному осуществлению угрозы, либо могут быть использованы злоумышленником для ее осуществления.)

Определим *B* как набор барьеров, определяемый декартовым произведением *{VxM}:* b_1 *= < ti, oj, mk >*, представляющих собой пути осуществления угроз безопасности, перекрытые

средствами защиты. В результате получаем систему, состоящую из пяти элементов: $<T, O, M, V, B>$, описывающую систему защиты с учетом наличия уязвимостей.

Для системы с полным перекрытием для любой уязвимости имеется устраняющий ее барьер. Иными словами, в подобной системе защиты для всех возможных угроз безопасности существуют механизмы защиты, препятствующие осуществлению этих угроз. Данное условие является первым фактором, определяющим защищенность ИС, второй фактор — «прочность» и надежность механизмов защиты.

В идеале каждый механизм защиты должен исключать соответствующий путь реализации угрозы. В действительности же механизмы защиты обеспечивают лишь определенную степень сопротивляемости угрозам безопасности, поэтому в качестве характеристик элемента набора барьеров $b_1 = <ti, oj, mk>$ может рассматриваться набор $<P_1, L_1, R_1>$, где P_1 — вероятность появления угрозы; L_1 — величина ущерба при удачном осуществлении угрозы в отношении защищаемых объектов (уровень серьезности угрозы); а R_1 — степень сопротивляемости механизма защиты m_k, характеризующаяся вероятностью его преодоления.

Надежность барьера $b_1 = <ti, oj, mk>$ характеризуется величиной остаточного риска $Risk_1$, связанного с возможностью осуществления угрозы t_i в отношении объекта информационной системы o_j при использовании механизма защиты m_k. Эта величина определяется по следующей формуле: $Risk_1 = P_k L_k (1 — R_k)$. Для нахождения примерной величины защищенности S можно использовать следующую простую формулу: $S = 1/Risk_0$, где $Risk_0$ является суммой всех остаточных рисков, $(0 < [P_h L_k] < 1)$, $(0 \le R_k < 1)$.

Суммарная величина остаточных рисков характеризует приблизительную совокупную уязвимость системы защиты, а защищенность определяется как величина, обратная уязвимости. При отсутствии в системе барьеров by «перекрывающих» выявленные уязвимости, степень сопротивляемости механизма защиты R_k принимается равной нулю.

На практике получение точных значений приведенных характеристик барьеров затруднено, поскольку понятия угрозы, ущерба и сопротивляемости механизма защиты трудно формализовать. Так, оценку ущерба в результате несанкционированного доступа к информации политического и военного характера точно определить вообще невозможно, а определение вероятности осуществления угрозы не может базироваться на статистическом анализе. Построение моделей системы защиты и анализ их свойств составляют предмет теории безопасных систем, еще только оформляющейся в качестве самостоятельного направления.

Для защиты информации экономического характера, допускающей оценку ущерба, разработаны стоимостные методы оценки эффективности средств защиты. Для этих методов набор характеристик барьера дополняет величина C_1 затраты на построение средства защиты барьера b_1. В этом случае выбор оптимального набора средств защиты связан с минимизацией суммарных затрат $W\{w_1\}$, состоящих из затрат $C=\{c_1\}$ на создание средств защиты и возможных затрат в результате успешного осуществления угроз $N=\{n_1\}$.

Формальные подходы к решению задачи оценки защищенности из-за трудностей, связанных с формализацией, широкого практического распространения не получили. Значительно более действенным является использование неформальных классификационных подходов. Для этого применяют категорирование: нарушителей (по целям, квалификации и доступным вычислительным ресурсам); информации (по уровням критичности и конфиденциальности); средств защиты (по функциональности и гарантированности реализуемых возможностей); эффективности и рентабельности средств защиты; и т.п.

20.3. Требования к архитектуре информационных систем для обеспечения безопасности ее функционирования

Идеология открытых систем (см. гл. 19) существенно отразилась на методологических аспектах и направлении развития сложных ИС.

Она базируется на строгом соблюдении совокупности профилей, протоколов и стандартов де-факто и де-юре. Программные и аппаратные компоненты по этой идеологии должны отвечать важнейшим требованиям переносимости и возможности согласованной, совместной работы с другими удаленными компонентами. Это позволяет обеспечить совместимость компонент различных информационных систем, а также средств передачи данных. Задача сводится к максимально возможному повторному использованию разработанных и апробированных программных и информационных компонент при изменении вычислительных аппаратных платформ, ОС и процессов взаимодействия [4, 5, 10].

При создании сложных, распределенных информационных систем, проектировании их архитектуры, инфраструктуры, выборе компонент и связей между ними следует учитывать помимо общих (открытость, масштабируемость, переносимость, мобильность, защита инвестиций и т.п.) ряд специфических концептуальных требований, направленных на обеспечение безопасности функционирования:

• архитектура системы должна быть достаточно гибкой, т.е. должна допускать относительно простое, без коренных структурных изменений, развитие инфраструктуры и изменение конфигурации используемых средств, наращивание функций и ресурсов ИС в соответствии с расширением сфер и задач ее применения;

• должны быть обеспечены безопасность функционирования системы при различных видах угроз и надежная защита данных от ошибок проектирования, разрушения или потери информации, а также авторизация пользователей, управление рабочей загрузкой, резервированием данных и вычислительных ресурсов, максимально быстрым восстановлением функционирования ИС;

• следует обеспечить комфортный, максимально упрощенный доступ пользователей к сервисам и результатам функционирования ИС на основе современных графических средств, мнемосхем и наглядных пользовательских интерфейсов;

• систему должна сопровождать актуализированная, комплектная документация, обеспечивающая квалифицированную эксплуатацию и возможность развития ИС.

Подчеркнем, что системы безопасности, какими бы мощными они ни были, сами по себе не могут гарантировать надежность программно-технического уровня защиты. Только проверенная архитектура способна сделать эффективным объединение сервисов, обеспечить управляемость информационной системы, ее способность развиваться и противостоять новым угрозам при сохранении таких свойств, как высокая производительность, простота и удобство использования.

С практической точки зрения обеспечения безопасности наиболее важными являются следующие принципы построения архитектуры ИС:

• проектирование И С на принципах открытых систем, следование признанным стандартам, использование апробированных решений, иерархическая организация ИС с небольшим числом сущностей на каждом уровне — все это способствует прозрачности и хорошей управляемости ИС;

• непрерывность защиты в пространстве и времени, невозможность преодолеть защитные средства, исключение спонтанного или вызванного перехода в небезопасное состояние — при любых обстоятельствах, в том числе нештатных, защитное средство либо полностью выполняет свои функции, либо полностью блокирует доступ в систему или ее часть;

• усиление самого слабого звена, минимизация привилегий доступа, разделение функций обслуживающих сервисов и обязанностей персонала. Предполагается такое распределение ролей и ответственности, чтобы один человек не мог нарушить критически важный для организации процесс или создать брешь в защите по неведению или заказу злоумышленников. Применительно к программно-техническому уровню принцип минимизации привилегий предписывает выделять пользователям и администраторам только те права доступа, которые необходимы им для выполнения служебных обязанностей. Это

позволяет уменьшить ущерб от случайных или умышленных некорректных действий пользователей и администраторов;

• эшелонирование обороны, разнообразие защитных средств, простота и управляемость информационной системы и системой ее безопасности. Принцип эшелонирования обороны предписывает не полагаться на один защитный рубеж, каким бы надежным он ни казался. За средствами физической защиты должны следовать программно-технические средства, за идентификацией и аутентификацией — управление доступом, протоколирование и аудит. Эшелонированная оборона способна не только не пропустить злоумышленника, но и в некоторых случаях идентифицировать его благодаря протоколированию и аудиту. Принцип разнообразия защитных средств предполагает создание различных по своему характеру оборонительных рубежей, чтобы от потенциального злоумышленника требовалось овладение разнообразными и, по возможности, несовместимыми между собой навыками.

Очень важен общий принцип простоты и управляемости ИС в целом и защитных средств в особенности. Только в простой и управляемой системе можно проверить согласованность конфигурации различных компонентов и осуществлять централизованное администрирование. В этой связи важно отметить интегрирующую роль Web-сервиса, скрывающего разнообразие обслуживаемых объектов и предоставляющего единый, наглядный интерфейс. Соответственно, если объекты некоторого вида (например, таблицы базы данных) доступны через Интернет, необходимо заблокировать прямой доступ к ним, поскольку в противном случае система будет уязвимой, сложной и плохо управляемой.

Продуманная и упорядоченная структура программных средств и баз данных, топология внутренних и внешних сетей непосредственно отражается на достигаемом качестве и безопасности ИС, а также на трудоемкости их разработки. При строгом соблюдении правил структурного построения значительно облегчается достижение высоких показателей качества и безопасности, так как сокращается число возможных ошибок в реализующих программах, отказов и сбоев оборудования, упрощается их диагностика и локализация. В хорошо структурированной системе с четко выделенными компонентами (клиент, сервер приложений, ресурсный сервер) контрольные точки выделяются достаточно четко, что решает задачу доказательства достаточности применяемых средств защиты и обеспечения невозможности обхода этих средств потенциальным нарушителем.

Высокие требования, предъявляемые к формированию архитектуры и инфраструктуры на стадии проектирования ИС, определяются тем, что именно на этой стадии можно в значительной степени минимизировать число уязвимостей, связанных с непредумышленными дестабилизирующими факторами, которые влияют на безопасность программных средств, баз данных и систем коммуникации.

Анализ безопасности И С при отсутствии злоумышленных факторов базируется на модели взаимодействия основных компонент ИС (рис. 20.1).

В качестве объектов уязвимости рассматриваются:

• динамический вычислительный процесс обработки данных, автоматизированной подготовки решений и выработки управляющих воздействий;

• объектный код программ, исполняемых вычислительными средствами в процессе функционирования ИС;

• данные и информация, накопленная в базах данных;

• информация, выдаваемая потребителям и на исполнительные механизмы.

Полное устранение перечисленных угроз принципиально невозможно. Задача состоит в выявлении факторов, от которых они зависят, в создании методов и средств уменьшения их влияния на безопасность ИС, а также в эффективном распределении ресурсов для обеспечения защиты, равнопрочной по отношению ко всем негативным воздействиям.

Рис. 20.1. **Модель анализа безопасности ИС при отсутствии злоумышленных угроз**

20.4. Этапы построения системы безопасности ИС

Концепция информационнной безопасности определяет этапы построения системы информационной безопасности в соответствии со стандартизованным жизненным циклом ИС: аудит безопасности (обследование) существующей системы защиты ИС, анализ рисков, формирование требований и выработка первоочередных мер защиты, проектирование, внедрение и аттестация, сопровождение системы [7]. Рассмотрим кратко содержание отдельных этапов.

Аудит безопасности. Аудит безопасности может включать в себя, по крайней мере, четыре различных группы работ.

К первой группе относятся так называемые тестовые взломы ИС. Эти тесты применяются, как правило, на начальных стадиях обследования защищенности ИС. Причина малой эффективности тестовых взломов скрывается в самой постановке задачи. Действительно, основной задачей взломщика является обнаружение нескольких уязвимостей и их использование для доступа в систему. Если тест оказался успешным, то, предотвратив потенциальное развитие возможных сценариев взлома, работу надо начинать сначала и искать следующие. Неуспех взлома может означать в равной мере как защищенность системы, так и недостаточность тестов.

Вторая группа — экспресс-обследование. В рамках этой, обычно непродолжительной работы оценивается общее состояние механизмов безопасности в обследуемой ИС на основе стандартизованных проверок. Экспресс-обследование обычно проводится в случае, когда необходимо определить приоритетные направления, позволяющие обеспечить минимальный уровень защиты информационных ресурсов. Основу для него составляют списки

контрольных вопросов, заполняемые в результате как интервьюирования, так и тестовой работы автоматизированных сканеров защищенности.

Третья группа работ по аудиту — аттестация систем на соответствие требованиям защищенности информационных ресурсов. При этом происходит формальная проверка набора требований как организационного, так и технического аспектов, рассматриваются полнота и достаточность реализации механизмов безопасности. Типовая методика анализа корпоративной информационной защищенности состоит их совокупности следующих методов:

• изучение исходных данных по структуре, архитектуре, инфраструктуре и конфигурации ИС на момент обследования;

• предварительная оценка рисков, связанных с осуществлением угроз безопасности в отношении технических и информационных ресурсов;

• анализ механизмов безопасности организационного уровня, политики безопасности организации и организационно-распорядительной документации по обеспечению режима ИБ и оценка их соответствия требованиям существующих стандартов и нормативных документов, а также их адекватности существующим рискам;

• анализ конфигурационных файлов маршрутизаторов и Proxy-серверов, почтовых и DNS-серверов (Domain Name System), шлюзов виртуальных частных сетей (VPN), других критических элементов сетевой инфраструктуры;

• сканирование внешних сетевых адресов локальной сети;

• сканирование ресурсов локальной сети изнутри;

• анализ конфигурации серверов и рабочих станций при помощи специализированных программных агентов.

Перечисленные технические методы предполагают применение как активного, так и пассивного тестирования системы защиты. *Активное* тестирование заключается в моделировании действий потенциального злоумышленника; а *пассивное* — предполагает анализ конфигурации ОС и приложений по шаблонам с использованием списков проверки. Тестирование может производиться вручную или с использованием специализированных программных средств.

При анализе конфигурации средств защиты для внешнего периметра локальной сети и управления межсетевыми взаимодействиями особое внимание обращается на следующие аспекты:

• настройка правил разграничения доступа (фильтрация сетевых пакетов);

• используемые схемы и настройка параметров аутентификации;

• настройка параметров системы регистрации событий;

• использование механизмов, обеспечивающих сокрытие топологии защищаемой сети (например, трансляция сетевых адресов);

• настройка механизмов оповещения об атаках и реагирования;

• наличие и работоспособность средств контроля целостности;

• версии используемого ПО и установленные обновления.

Анализ конфигурации предполагает проверку правильности установки сотен различных параметров. Для автоматизации этого процесса могут использоваться специализированные программные средства анализа защищенности, выбор которых в настоящее время достаточно широк.

Один из современных и быстро развивающихся методов автоматизации процессов анализа и контроля защищенности распределенных компьютерных систем — использование технологии интеллектуальных программных агентов. На каждую из контролируемых систем устанавливается программный агент, который выполняет соответствующие настройки ПО, проверяет их правильность, контролирует целостность файлов, своевременность установки обновлений, а также решает другие задачи по контролю защищенности ИС. Управление агентами осуществляет по сети программа-менеджер. Такие менеджеры, являющиеся центральными компонентами подобных систем, посылают управляющие команды всем агентам контролируемого ими домена и сохраняют все полученные от агентов данные в

центральной БД. Администратор управляет менеджерами при помощи графической консоли, позволяющей выбирать, настраивать и создавать политики безопасности, анализировать изменения состояния системы, осуществлять ранжирование уязвимостей и т.п. Все взаимодействия между агентами, менеджерами и управляющей консолью осуществляются по защищенному клиент-серверному протоколу. Такой подход, например, использован при построении комплексной системы управления безопасностью организации ESM (производитель — компания «Symantec Enterprise Security Manager»).

Четвертая группа — предпроектное обследование — самый трудоемкий вариант аудита. Такой аудит предполагает анализ организационной структуры предприятия в приложении к ИР, правила доступа сотрудников к тем или иным приложениям. Затем выполняется анализ самих приложений. После этого должны учитываться конкретные службы доступа с одного уровня на другой, а также службы, необходимые для информационного обмена. Затем картина дополняется встроенными механизмами безопасности, что в сочетании с оценками потерь в случае нарушения ИБ дает основания для ранжирования рисков, существующих в ИС, и выработки адекватных контрмер. Успешное проведение предпроектного обследования, последующего анализа рисков и формирования требований определяют, насколько принятые меры будут адекватны угрозам, эффективны и экономически оправданы.

Проектирование системы. В настоящее время сложились два подхода к построению системы ИБ: продуктовый и проектный. В рамках *продуктового*подхода выбирается набор средств физической, технической и программной защиты (готовое решение), анализируются их функции, а на основе анализа функций определяется политика доступа в рабочие и технологические помещения, к информационным ресурсам. Можно поступать наоборот: вначале прорабатывается политика доступа, на основе которой определяются функции, необходимые для ее реализации, и производится выбор средств и продуктов, обеспечивающих выполнение этих функций. Выбор методов зависит от конкретных условий деятельности организации, ее местонахождения, расположения помещений, состава подсистем ИС, совокупности решаемых задач, требований к системе защиты и т.д. Продуктовый подход более дешев с точки зрения затрат на проектирование. Кроме того, в некоторых случаях он является единственно возможным в условиях дефицита решений или жестких требований нормативных документов на государственном уровне (например, для криптографической защиты в сетях специального назначения и правительственных телефонных сетях применяется только такой подход). *Проектный* подход заведомо более полон, и решения, построенные на его основе, более оптимизированы и проще аттестуемы. Он предпочтительнее и при создании больших гетерогенных распределенных систем, поскольку в отличие от продуктового подхода не связан изначально с той или иной платформой. Кроме того, он обеспечивает более «долгоживущие» решения, поскольку допускает проведение замены продуктов и решений без изменения политики доступа. Это, в свою очередь, обеспечивает хороший показатель возврата инвестиций (ROI) при развитии ИС и системы ИБ.

Объекты или приложения? При проектировании архитектуры системы информационной безопасности применяются объектный, прикладной или смешанный подходы.

Объектный подход строит защиту информации на основании физической структуры того или иного объекта (здания, подразделения, предприятия). Применение объектного подхода предполагает использование набора универсальных решений для обеспечения механизмов безопасности, поддерживающих однородный набор организационных мер. Классическим примером такого подхода является построение защищенных инфраструктур внешнего информационного обмена, локальной сети, системы телекоммуникаций и т.д. К его недостаткам относятся очевидная неполнота универсальных механизмов, особенно для организаций с большим набором сложно связанных между собой приложений.

Прикладной подход «привязывает» механизмы безопасности к конкретному приложению. Пример такого подхода — защита подсистемы либо отдельных зон

автоматизации (бухгалтерия, склад, кадры, проектное бюро, аналитический отдел, отделы маркетинга и продаж и т.д.). При большей полноте защитных мер такого подхода у него имеются и недостатки, а именно: необходимо увязывать различные по функциональным возможностям средства безопасности для минимизации затрат на администрирование и эксплуатацию, а также задействовать уже существующие средства защиты информации для сохранения инвестиций.

Возможна комбинация двух описанных подходов. В *смешанном* подходе ИС представляется как совокупность объектов, для каждого из которых применен объектный подход, а для совокупности взаимосвязанных объектов — прикладной. Такая методика оказывается более трудоемкой на стадии проектирования, однако часто дает хорошую экономию средств при внедрении, эксплуатации и сопровождении системы защиты информации.

Службы и механизмы безопасности. Стратегию защиты можно реализовать двумя методами: ресурсным и сервисным. Первый метод рассматривает ИС как набор ресурсов, которые «привязываются» к конкретным компонентам системы ИБ. Этот метод хорош для небольших ИС с ограниченным набором задач. При расширении круга задач и разрастании ИС приходится во многом дублировать элементы защиты для однотипных ресурсов, что часто приводит к неоправданным затратам. Сервисный подход трактует ИС как набор служб, программных и телекоммуникационных сервисов для оказания услуг пользователям. В этом случае один и тот же элемент защиты можно использовать для различных сервисов, построенных на одном и том же ПО или техническом устройстве. Сегодня сервисный подход представляется предпочтительным, поскольку он предполагает строгий функциональный анализ существующих многочисленных служб, обеспечивающих функционирование ИС, и позволяет исключить широкий класс угроз при помощи отказа от «лишних» служб и оптимизации работы оставшихся, делая структуру системы ИБ логически обоснованной. Именно сервисный подход лежит в основе современных стандартов по безопасности, в частности ISO 15408.

Внедрение и аттестация. Этап внедрения включает в себя комплекс последовательно проводимых мероприятий, в том числе установку и конфигурирование средств защиты, обучение персонала работе со средствами защиты, проведение предварительных испытаний и сдачу в опытную эксплуатацию. Опытная эксплуатация позволяет выявить и устранить возможные недостатки функционирования подсистемы информационной безопасности, прежде чем запустить систему в «боевой» режим. Если в процессе опытной эксплуатации выявлены факты некорректной работы компонентов, проводят корректировку настроек средств защиты, режимов их функционирования и т.п. По результатам опытной эксплуатации вносят корректировки (при необходимости) и уточняют настройки средств защиты. Далее следует проведение приемо-сдаточных испытаний, ввод в штатную эксплуатацию и оказание технической поддержки и сопровождения.

Подтверждение функциональной полноты системы безопасности и обеспечения требуемого уровня защищенности ИС обеспечивается проведением аттестации системы И Б соответствующим аккредитованным центром Гостехкомиссии России или зарубежной независимой лабораторией. Аттестация предусматривает комплексную проверку защищаемого объекта в реальных условиях эксплуатации для оценки соответствия применяемого комплекса мер и средств защиты требуемому уровню безопасности. Аттестация проводится в соответствии со схемой, составляемой на подготовительном этапе исходя из следующего перечня работ:

• анализ исходных данных, предварительное ознакомление с аттестуемым объектом и информатизации;

• экспертное обследование объекта информатизации и анализ документации по защите информации на предмет соответствия требованиям;

• испытания отдельных средств и систем защиты информации на аттестуемом объекте с помощью специальной контрольной аппаратуры и тестовых средств;

• испытания отдельных средств и систем защиты информации в испытательных центрах (лабораториях);

• комплексные аттестационные испытания объекта информатизации в реальных условиях эксплуатации;

• анализ результатов экспертного обследования и аттестационных испытаний и утверждение заключения по результатам аттестации объекта информатизации.

По результатам испытаний готовится отчетная документация, проводится оценка результатов испытаний и выдается аттестат соответствия установленного образца. Его наличие дает право обработки информации со степенью конфиденциальности и на период времени, установленными в аттестате.

Техническая поддержка и сопровождение. Для поддержания работоспособности подсистемы информационной безопасности и бесперебойного выполнения ей своих функций необходимо предусмотреть комплекс мероприятий по технической поддержке и сопровождению программного и аппаратного обеспечения подсистемы информационной безопасности, включая текущее администрирование, работы, проводимые в экстренных случаях, а также периодически проводимые профилактические работы. Данный комплекс мероприятий включает в себя:

• администрирование штатных средств защиты и их техническое обслуживание;

• контроль состояния системы, профилактическое обследование конфигурации, выявление потенциальных проблем;

• мониторинг и установку выпускаемых обновлений и программных коррекций средств защиты, а также используемых ОС, СУБД и приложений;

• регулярный поиск и анализ уязвимостей в защищаемой системе с использованием специальных средств сканирования;

• диагностику неисправностей и проведение восстановительных работ при возникновении аварийных и нештатных ситуаций;

• периодическое тестирование системы информационной безопасности и оценка эффективности защиты.

Техническая поддержка и сопровождение системы информационной безопасности требует наличия у обслуживающего персонала определенных знаний и навыков и может осуществляться как штатными сотрудниками организации — владельца ИС, ответственными за информационную безопасность, так и сотрудниками специализированных организаций.

20.5. Стандартизация подходов к обеспечению информационной безопасности

Специалистам в области ИБ сегодня практически невозможно обойтись без знаний соответствующих стандартов и спецификаций. На то имеется несколько веских причин. Формальная причина состоит в том, что необходимость следования некоторым стандартам (например, криптографическим и Руководящим документам Гостехкомиссии России) закреплена законодательно. Убедительны и содержательные причины. Во-первых, стандарты и спецификации — одна из форм накопления знаний, прежде всего о процедурном и программно-техническом уровнях ИБ и ИС. В них зафиксированы апробированные, высококачественные решения и методологии, разработанные наиболее квалифицированными компаниями в области разработки ПО и безопасности программных средств. Во-вторых, и те и другие являются основным средством обеспечения взаимной совместимости аппаратно-программных систем и их компонентов, причем в Интернет-сообществе это средство работает весьма эффективно.

На верхнем уровне можно выделить две существенно отличающиеся друг от друга группы стандартов и спецификаций:

1) оценочные стандарты, предназначенные для оценки и классификации ИС и средств защиты по требованиям безопасности;

2) спецификации, регламентирующие различные аспекты реализации и использования средств и методов защиты.

Эти группы дополняют друг друга. Оценочные стандарты описывают важнейшие с точки зрения ИБ понятия и аспекты ИС, играя роль организационных и архитектурных спецификаций. Специализированные стандарты и спецификации определяют, как именно строить И С предписанной архитектуры и выполнять организационные требования.

Из числа оценочных необходимо выделить стандарт Министерства обороны США «Критерии оценки доверенных компьютерных систем» и его интерпретацию для сетевых конфигураций, «Гармонизированные критерии Европейских стран», международный стандарт «Критерии оценки безопасности информационных технологий» и конечно, Руководящие документы Гостехкомиссии России. К этой же группе относится и Федеральный стандарт США «Требования безопасности для криптографических модулей», регламентирующий конкретный, но очень важный и сложный аспект информационной безопасности.

Технические спецификации, применимые к современным распределенным ИС, создаются главным образом «Тематической группой по технологии Интернет» (Internet Engineering Task Force — IETF) и ее подразделением — рабочей группой по безопасности. Ядром технических спецификаций служат документы по безопасности на IP-уровне (IPsec). Кроме этого, анализируется защита на транспортном уровне (Transport Layer Security — TLS), а также на уровне приложений (спецификации GSS-API, Kerberos). Интернет-сообщество уделяет должное внимание административному и процедурному уровням безопасности, создав серию руководств и рекомендаций: «Руководство по информационной безопасности предприятия», «Как выбирать поставщика Интернет-услуг», «Как реагировать на нарушения информационной безопасности» и др.

В вопросах сетевой безопасности невозможно обойтись без спецификаций X.800 «Архитектура безопасности для взаимодействия открытых систем», X.500 «Служба директорий: обзор концепций, моделей и сервисов» и X.509 «Служба директорий: каркасы сертификатов открытых ключей и атрибутов».

Критерии оценки механизмов безопасности программно-технического уровня представлены в международном стандарте ISO 15408— 1999 «Общие критерии оценки безопасности информационных технологий» («The Common Criteria for Information Technology Security Evaluation»), принятом £ 1999 г. «Общие критерии» («ОК») определяют функциональные требования безопасности (Security Functional Requirements) и требования к адекватности реализации функций безопасности (Security Assurance Requirements).

«Общие критерии» содержат два основных вида требований безопасности:

1) функциональные, соответствующие активному аспекту защиты, предъявляемые к функциям (сервисам) безопасности и реализующим их механизмам;

2) требования доверия, соответствующие пассивному аспекту; они предъявляются к технологии и процессу разработки и эксплуатации.

Требования безопасности формулируются, и их выполнение проверяется для определенного объекта оценки — аппаратно-программного продукта или ИС. Безопасность в «ОК» рассматривается не статично, а в соответствии с жизненным циклом объекта оценки. Кроме того, обследуемый объект предстает не изолированно, а в «среде безопасности», характеризующейся определенными уязвимостями и угрозами. «Общие критерии» целесообразно использовать для оценки уровня защищенности с точки зрения полноты реализованных в ней функций безопасности и надежности реализации этих функций. Хотя применимость «ОК» ограничивается механизмами безопасности программно-технического уровня, в них содержится определенный набор требований к механизмам безопасности организационного уровня и требований по физической защите, которые непосредственно связаны с описываемыми функциями безопасности.

Британский стандарт BS 7799 «Управление информационной безопасностью. Практические правила» без сколько-нибудь существенных изменений воспроизведен в международном стандарте ISO/IEC 17799—2000 «Практические правила управления информационной безопасностью» («Code of practice for Information security

management»). В этом стандарте обобщены правила по управлению ИБ, они могут быть использоваться в качестве критериев оценки механизмов безопасности организационного уровня, включая административные, процедурные и физические меры защиты. Практические правила разбиты на десять разделов.

1. Политика безопасности.
2. Организация защиты.
3. Классификация ресурсов и их контроль.
4. Безопасность персонала.
5. Физическая безопасность.
6. Администрирование компьютерных систем и сетей.
7. Управление доступом.
8. Разработка и сопровождение информационных систем.
9. Планирование бесперебойной работы организации.
10. Контроль выполнения требований политики безопасности.

В этих разделах содержится описание механизмов организационного уровня, реализуемых в настоящее время в государственных и коммерческих организациях во многих странах.

Ключевые средства контроля (механизмы управления ИБ), предлагаемых в ISO 17799—2000, считаются особенно важными. При использовании некоторых из средств контроля, например шифрования, могут потребоваться советы специалистов по безопасности и оценка рисков. Для обеспечения защиты особенно ценных ресурсов или оказания противодействия особенно серьезным угрозам безопасности в ряде случаев могут потребоваться более сильные средства контроля, которые выходят за рамки ISO 17799—2000. Процедура аудита безопасности ИС по стандарту ISO 17799 включает в себя проверку наличия перечисленных ключевых средств контроля, оценку полноты и правильности их реализации, а также анализ их адекватности рискам, существующим в данной среде функционирования. Составной частью работ по аудиту также является анализ и управление рисками.

На нижнем уровне разработаны в разных странах сотни отраслевых стандартов, нормативных документов и спецификаций по обеспечению ИБ, которые применяются национальными компаниями при разработке программных средств, ИС и обеспечении качества и безопасности их функционирования.

20.6. Обеспечение интегральной безопасности информационных систем

Наряду с системной и функциональной интеграцией ИС в последнее время стала активно развиваться *интегральная информационная безопасность* (ИИБ), под которой понимается такое состояние условий функционирования человека, объектов, технических средств и систем, при котором они надежно защищены от всех возможных видов угроз в ходе непрерывного процесса подготовки, хранения, передачи и обработки информации.

Интегральная безопасность информационных систем включает в себя следующие составляющие:

- физическая безопасность (защита зданий, помещений, подвижных средств, людей, а также аппаратных средств — компьютеров, носителей информации, сетевого оборудования, кабельного хозяйства, поддерживающей инфраструктуры);
- безопасность сетей и телекоммуникационных устройств (защита каналов связи от воздействий любого рода);
- безопасность ПО (защита от вирусов, логических бомб, несанкционированного изменения конфигурации и программного кода);
- безопасность данных (обеспечение конфиденциальности, целостности и доступности данных).

Задача обеспечения ИБ появилась вместе с проблемой передачи и хранения информации. На современном этапе можно выделить три подхода к ее решению:

1) *частный* — основывается на решении частных задач обеспечения ИБ. Этот подход является малоэффективным, но достаточно часто используется, так как не требует больших финансовых и интеллектуальных затрат;

2) *комплексный* — реализуется решением совокупности частных задач по единой программе. Этот подход в настоящее время применяется наиболее часто;

3) *интегральный* — основан на объединении различных вычислительных подсистем ИС, подсистем связи, подсистем обеспечения безопасности в единую информационную систему с общими техническими средствами, каналами связи, ПО и базами данных.

Третий подход направлен на достижение ИИБ, что предполагает обязательную непрерывность процесса обеспечения безопасности как во времени (в течение всей «жизни» ИС), так и в пространстве (по всему технологическому циклу деятельности) с обязательным учетом всех возможных видов угроз (несанкционированный доступ, съем информации, терроризм, пожар, стихийные бедствия и т.п.). В какой бы форме ни применялся интегральный подход, он связан с решением ряда сложных разноплановых частных задач в их тесной взаимосвязи. Наиболее очевидными из них являются задачи разграничения доступа к информации, ее технического и криптографического «закрытия», устранение паразитных излучений технических средств, технической и физической укрепленности объектов, охраны и оснащения их тревожной сигнализацией.

На рис. 20.2 представлена блок-схема интегрального комплекса физической защиты объекта, обеспечивающего функционирование всех рассмотренных выше систем, а на рис 20.3 — соотношение эффективности современных электронных средств контроля физического доступа.

Рис. 20.2. **Блок-схема интегрального комплекса физической защиты ИС**

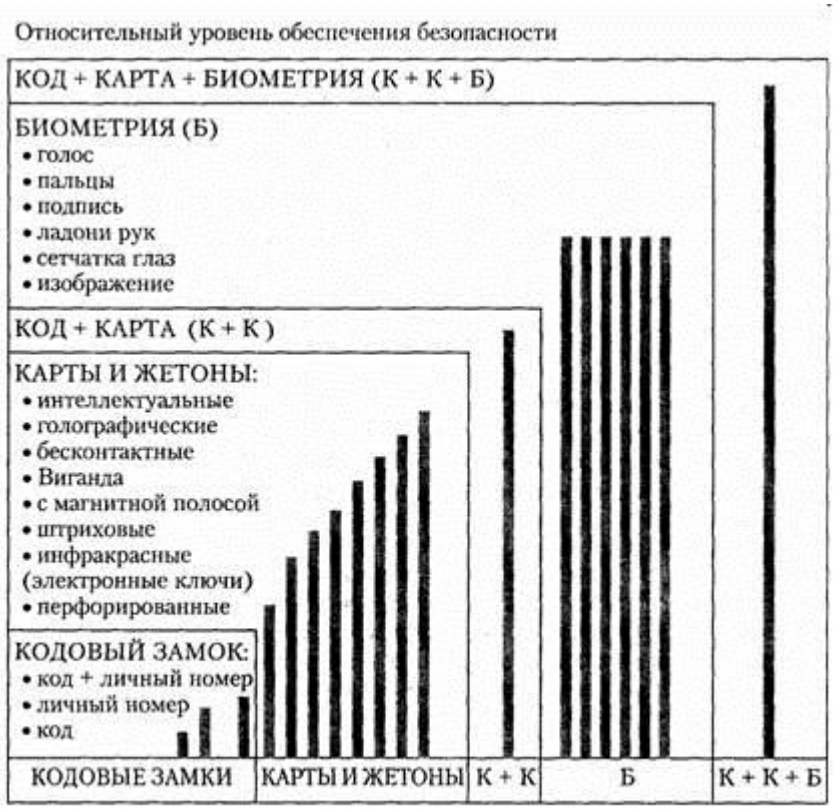

Рис. 20.3. **Сравнение эффективности современных электронных средств контроля доступа**

Стандартный набор средств защиты информации в составе современной ИС обычно содержит следующие компоненты:

• средства обеспечения надежного хранения информации с использованием технологии защиты на файловом уровне (File Encryption System — FES);

• средства авторизации и разграничения доступа к информационным ресурсам, а также защита от несанкционированного доступа к информации с использованием технологии токенов (смарт-карты, touch-memory, ключи для USB-портов и т.п.);

• средства защиты от внешних угроз при подключении к общедоступным сетям связи (Интернет), а также средства управления доступом из Интернет с использованием технологии межсетевых экранов (FireWall) и содержательной фильтрации (Content Inspection);

• средства защиты от вирусов с использованием специализированных комплексов антивирусной профилактики;

• средства обеспечения конфиденциальности, целостности, доступности и подлинности информации, передаваемой по открытым каналам связи с использованием технологии защищенных виртуальных частных сетей (Virtual Private Net — VPN);

• средства обеспечения активного исследования защищенности информационных ресурсов с использованием технологии обнаружения атак (IntrusionDetection);

• средства обеспечения централизованного управления системой ИБ в соответствии с согласованной и утвержденной политикой безопасности.

Защита информации на файловом уровне. Эти технологии позволяют скрыть конфиденциальную информацию пользователя на жестком диске компьютера или сетевых дисках путем кодирования содержимого файлов, каталогов и дисков. Доступ к данной информации осуществляется по предъявлению ключа, который может вводиться с клавиатуры, Храниться и предоставляться со смарт-карты, HASP-ключей или USB-ключей и прочих токенов. Помимо перечисленных выше функций указанные средства позволяют мгновенно «уничтожить» информацию при подаче сигнала «тревога» и при «входе под принуждением», а также блокировать компьютер в перерывах между сеансами работы.

Технологии токенов (смарт-карты, touch-memory, ключи для USB-портов). Электронные ключи-жетоны (Token) являются средством повышения надежности защиты данных на основе гарантированной идентификации пользователя. Токены являются «контейнерами» для хранения персональных данных пользователя системы и некоторых его паролей. Основное преимущество токена заключается в том, что персональная информация всегда находится на носителе (смарт-карте, ключе и т.д.) и предъявляется только во время доступа к системе или компьютеру. Эта система находит все новых и новых приверженцев, так как позволяет унифицировать правила доступа и поместить на одном персональном электронном носителе систему паролей для доступа на различные устройства и системы кодирования и декодирования информации. В настоящее время получают распространение токены с системой персональной аутентификации на базе биометрической информации, которая считывается с руки пользователя. Таким «ключом» может воспользоваться только тот пользователь, на которого настроен этот ключ.

Межсетевые экраны. Использование технологии межсетевых экранов предлагается для решения таких задач, как:

• безопасное взаимодействие пользователей и информационных ресурсов, расположенных в Экстранет- и Интранет-сетях, с внешними сетями;

• создание технологически единого комплекса мер защиты для распределенных и сегментированных локальных сетей подразделений предприятия;

• построение иерархической системы защиты, предоставляющей адекватные средства обеспечения безопасности для различных по степени закрытости сегментов корпоративной сети.

В зависимости от масштабов организации и установленной политики безопасности рекомендуются межсетевые экраны (FireWall), отличающиеся по степени функциональности и по стоимости [межсетевые экраны Checkpoint Fire Wall-1, Private Internet Exchange (PIX) компании «Cisco» и др.]. Устройства содержательной фильтрации (Content Inspection) устанавливаются, как правило, на входы почтовых серверов для отсечения большого объема неопасной, но практически бесполезной информации, обычно рекламного характера (Spam), принудительно рассылаемой большому числу абонентов электронной почты.

Антивирусные средства. Лавинообразное распространением вирусов («червей», «троянских коней») действительно стало большой проблемой для большинства компаний и государственных учреждений. В настоящее время известно более 45 000 компьютерных вирусов и каждый месяц появляется более 300 новых разновидностей. При этом считается, что основной путь «заражения» компьютеров — через Интернет, поэтому наилучшее решение, по мнению многих руководителей, — отключить корпоративную сеть от Интернет. Часто говорят: «Есть Интернет — есть проблемы, нет Интернет — нет проблем». При этом не учитывается, что существует множество других путей проникновения вирусов на конкретный компьютер, например при использовании чужих дискет и дисков, пиратское программное обеспечение или персональные компьютеры «общего пользования» (например, опасность представляют домашние или студенческие компьютеры, если на них работает более одного человека). Системное применение лицензионных антивирусных средств (например, Лаборатории Касперского) существенно уменьшает опасность вирусного заражения.

Защищенные виртуальные частные сети. Для защиты информации, передаваемой по открытым каналам связи, поддерживающим протоколы TCP/IP, существует ряд программных продуктов, предназначенных для построения VPN на основе международных стандартов IPSec. Виртуальные сети создаются чаще всего на базе арендуемых и коммутируемых каналов связи в сетях общего пользования (Интернет). Для небольших и средних компаний они являются хорошей альтернативой изолированным корпоративным сетям, так как обладают очевидными преимуществами: высокая гарантированная надежность, изменяемая топология, простота конфигурирования, легкость масштабирования, контроль всех событий и действий в сети, относительно
невысокая стоимость аренды каналов и коммуникационного оборудования.

Продукты работают в операционных системах Windows 98/2000/ NT и Solaris и обеспечивают:

• защиту (конфиденциальность, подлинность и целостность) передаваемой по сетям информации;

• контроль доступа в защищаемый периметр сети;

• идентификацию и аутентификацию пользователей сетевых объектов;

• централизованное управление политикой корпоративной сетевой безопасности.

Системы шифрования с открытым криптографическим интерфейсом позволяют использовать различные реализации криптоалгоритмов. Это дает возможность использовать продукты в любой стране мира в соответствии с принятыми национальными стандартами. Наличие разнообразных модификаций (линейка продуктов включает в *** себя до десятка наименований для клиентских, серверных платформ, сети масштаба офиса, генерации ключевой информации) позволяет подбирать оптимальное по стоимости и надежности решение с возможностью постепенного наращивания мощности системы защиты.

Технологии обнаружения атак (Intrusion Detection). Постоянное изменение сети (появление новых рабочих станций, реконфигурация программных средств и т.п.) может привести к появлению новых уязвимых мест, угроз и возможностей атак на ИР и саму систему защиты. В связи с этим особенно важно своевременное их выявление и внесение изменений в соответствующие настройки информационного комплекса и его подсистем, и в том числе в подсистему защиты. Это означает, что рабочее место администратора системы должно быть укомплектовано специализированными программными средствами обследования сетей и выявления уязвимых мест (наличия «дыр») для проведения атак «извне» и «снаружи», а также комплексной оценки степени защищенности от атак нарушителей. Например, в состав продуктов ЭЛВИС+, Net Pro VPN входят наиболее мощные среди обширного семейства коммерческих пакетов продукты компании «Internet Security Systems» (Internet Scanner и System Security Scanner), а также продукты компании «Cisco»: система обнаружения несанкционированного доступа NetRanger и сканер уязвимости системы безопасности NetSonar.

Инфраструктура открытых ключей (Public Key Infrastruture — PKI). Основными функциями РК1 являются поддержка жизненного цикла цифровых ключей и сертификатов (т.е. их генерация, распределение, отзыв и пр.), поддержка процесса идентификации и аутентификации пользователей и реализация механизма интеграции существующих приложений и всех компонент подсистемы безопасности. Несмотря на существующие международные стандарты, определяющие функционирование системы PKI и способствующие ее взаимодействию с различными средствами защиты информации, к сожалению, не каждое средство информационной защиты, даже если его производитель декларирует соответствие стандартам, может работать с любой системой PKI. В нашей стране только начинают появляться компании, предоставляющие услуги по анализу, проектированию и разработке инфраструктуры открытых ключей. Поскольку при возрастающих масштабах ведомственных и корпоративных сетей VPN-продукты не смогут работать без PKI, только у разработчиков и поставщиков VPN есть опыт работы в этой области.

В зависимости от масштаба деятельности компании методы и средства обеспечения ИБ могут различаться, но любой «продвинутый» СЮ или специалистIT-службы скажет, что любая проблема в области ИБ не решается односторонне — всегда требуется комплексный, интегральный подход. В настоящее время с сожалением приходится констатировать, что в российском бизнесе многие высшие менеджеры компаний и руководители крупных государственных организаций считают, что все проблемы в сфере ИБ можно решить, не прилагая особых организационных, технических и финансовых усилий.

Нередко со стороны людей, позиционирующих себя в качестве IT-специалистов, приходится слышать высказывания: «Проблемы информационной безопасности в нашей компании мы уже решили — установили межсетевой экран и купили лицензию на средства антивирусной защиты». Такой подход свидетельствует, что существование проблемы уже

признается, но сильно недооценивается масштаб и сложность необходимых срочных мероприятий по ее решению. В тех компаниях, где руководство и специалисты всерьез задумались над тем, как обезопасить свой бизнес и избежать финансовых потерь, признано, что одними локальными мерами или «подручными» средствами уже не обойтись, а нужно применять именно комплексный подход.