

УДК 004.414.22

DOI: 10.15827/0236-235X.116.027-035

Дата подачи статьи: 01.08.16

2016. Т. 29. № 4. С. 27–35

МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ УПРАВЛЕНИЯ ЖИЗНЕННЫМ ЦИКЛОМ ИНФОРМАЦИОННОЙ СИСТЕМЫ НА ОСНОВЕ ИНСТРУМЕНТОВ ФУНКЦИОНАЛЬНОЙ СТАНДАРТИЗАЦИИ

*О.В. Лукинова, д.т.н., ведущий научный сотрудник, lobars@mail.ru
(Институт проблем управления им. В.А. Трапезникова РАН,
ул. Профсоюзная, 65, г. Москва, 117342, Россия)*

В работе представлена методология управления полным жизненным циклом информационной системы, синтезирующая инструменты и модели функциональной стандартизации, теории открытых систем, управленческого консалтинга, базовых стандартов в области управления жизненным циклом информационной системы. Описаны основные положения и показана концептуальная модель, отражающая взаимосвязь задач, решаемых в рамках методологии, а именно: построение модели автоматизируемых бизнес-процессов предприятия средствами управленческого консалтинга, разработка модели информационной системы, а также ее стандартизация в соответствии с моделью OSE/RM (Open System Environment/Reference Model) при помощи функционального профиля, выбор варианта жизненного цикла, разработка его модели и профилирования. Исследованы различные варианты жизненных циклов, особое внимание уделено жизненному циклу покупной системы, а также системы, включающей, помимо локального компонента (развернутого на предприятии заказчика), совокупность интегрированных публичных сервисов. Показано (в терминах модели открытой среды OSE/RM) структурное представление не только целевой компоненты проектируемой информационной системы, но и системы ее защиты как единого создаваемого объекта. Приведена концептуальная модель механизмов защиты, совокупность которых составляет функциональную структуру системы защиты.

Ключевые слова: информационная система, функциональный и технологический профиль, модель открытой системы, свойства открытых систем, функциональная стандартизация.

Современные интегрированные информационно-телекоммуникационные системы представляют собой наиболее сложный класс *информационных систем* (ИС) с точки зрения методов и средств, используемых в ходе проработки и реализации их *жизненных циклов* (ЖЦ). К этим системам предъявляются весьма высокие требования, связанные с ответственностью их функционирования, необходимостью адаптации к непрерывно изменяющимся условиям бизнеса в рыночной экономике, информационного взаимодействия со смежными ИС. Кроме того, противоречие между возрастающей сложностью ИС и сокращением сроков и средств, отпускаемых на проектирование и сопровождение системы, требует внедрения в процесс управления ЖЦ определенных подходов.

Вопросам управления ЖЦ ИС посвящены многие публикации, например [1–3]. Данную работу отличают следующие особенности:

- создаваемая система проектируется как открытая на основе стандартизованного структурированного представления;
- проблема стандартизации и самой ИС, и ее ЖЦ, связанная с необходимостью применения открытых спецификаций, решается в рамках систематизированного подхода;
- предлагается новый способ описания видов деятельности ЖЦ, которые зачастую имеют неоднозначную интерпретацию в силу их существующей смысловой разницы;
- объекты *бизнес-процесс – приложение ИС – платформа* рассматриваются как пирамида услуг (сервисов), оказываемых по иерархии и в конечном счете ориентированных на пользователя.

Таким образом, в статье описан методологический подход к управлению ЖЦ сложных ИС, основанный на объединении современных инструментов *функциональной стандартизации* (ФС), моделей теории открытых систем, базовых стандартов и моделей управления ЖЦ, предпроектных консалтинговых исследований, что позволит повысить качество выходного продукта, процесса разработки, а также управляемость этим процессом.

Концептуальная модель методологии

Методология представляет собой инструмент для эффективного выполнения проектов по созданию, сопровождению и модификации ИС (причем под созданием здесь понимаются как разработка системы под заказ, так и внедрение покупного тиражируемого варианта). Она разработана для управления полным ЖЦ существования ИС на предприятии [1] и представлена следующими аспектами.

- Проведение консалтинговых исследований и построение модели бизнес-процессов, подлежащих автоматизации. Указанные модели интерпретируются как совокупность сервисов, требующихся предприятию для ведения своей деятельности.
- Представление ИС, приложения которой являются средой реализации бизнес-процессов, базируется на функциональных референсных моделях открытых систем, разработанных Комитетом IEEE POSIX (IEEE Std 1003.0-1005) и получивших развитие в [4, 5]. Модели не только являются основой для применения ФС, но и позволяют выстроить

требования системных интерфейсов между приложениями и платформенными сервисами.

- Моделирование различных вариантов ЖЦ ИС.
- Стандартизация ИС и ее ЖЦ с использованием инструментов методологии ФС, позволяющей описывать на нормативном уровне сложные объекты по частям в рамках систематизированного представления.

Концептуальная модель, отражающая взаимосвязь положений и задач, решаемых в рамках указанных аспектов, приведена на рисунке 1.

Как видно из рисунка, инициацией применения методологии является наличие у заказчика потребности в автоматизации своих бизнес-процессов. Поэтому первая задача, которую необходимо решить, – это проведение консалтинговых работ, в ходе которых строится бизнес-модель автоматизированных процессов предприятия. Бизнес-модель имеет самостоятельное значение для предприятия и, что крайне важно в контексте данной работы, является основой для формирования бизнес-требований к будущей ИС [6, 7].

Центральное место на схеме отведено объекту «Создание ИС», являющемуся, с одной стороны, средой реализации бизнес-процессов в виде своих приложений, а с другой – результатом (продуктом)

проекта по созданию ИС. Этот объект является водоразделом между сферой заказчика в части выставления бизнес-требований к ИС и сферой разработчика в части удовлетворения этих требований. Для получения качественного результата будущая ИС (или ее модификация) должна быть разработана как открытая система в соответствии с концепцией и моделью открытых систем *OSE/RM* (Open System Environment/Reference Model) [4], а также стандартизована как сложный объект (сложность в данном случае предполагает невозможность описания объекта в рамках одного стандарта) на основе инструментов ФС.

Верхняя часть схемы (рис. 1) представляет задачи, связанные с ЖЦ корпоративной ИС. Как видно из рисунка, необходимо построить модель варианта ЖЦ в соответствии с тем или иным базовым стандартом, наполнить ее соответствующими видами деятельности и стандартизовать те виды деятельности, которые будут выбраны и выстроены на ЖЦ (здесь также используются инструменты ФС).

Модель ИС как открытой системы

Важнейшим аспектом предлагаемой методологии является использование концептуальных поло-

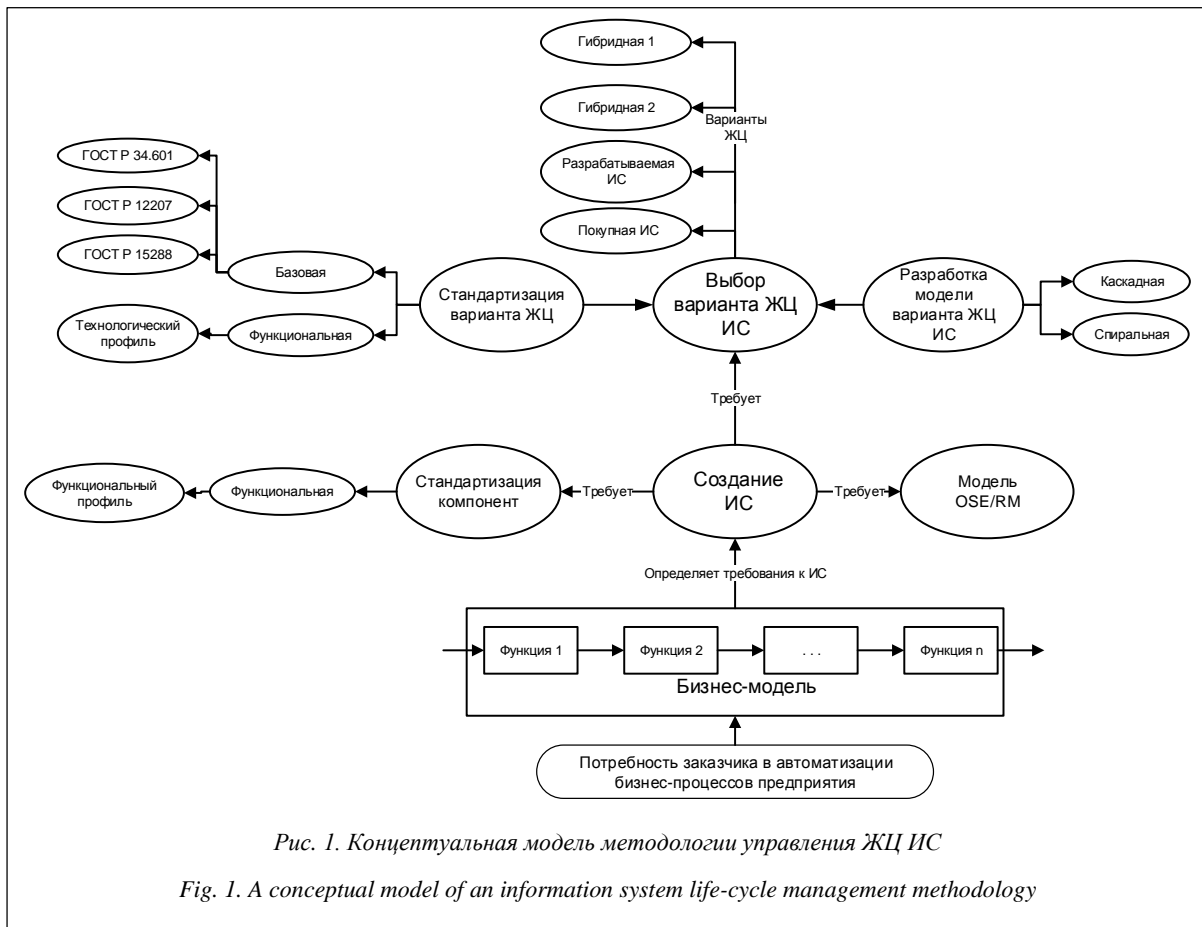


Рис. 1. Концептуальная модель методологии управления ЖЦ ИС

Fig. 1. A conceptual model of an information system life-cycle management methodology

жений теории открытой среды, в рамках которой ИС характеризуется определенными свойствами, а именно:

- расширяемость (*extensibility*) – возможность добавления новых прикладных функций ИС или изменения некоторых функций из числа уже реализованных без изменения при этом остальных функциональных частей (подсистем) ИС;

- мобильность приложений и данных (*portability*) – возможность перевода ИС на более совершенные аппаратно-программные платформы при их модернизации или замене с минимальными затратами, сохраняя вложенные инвестиции (обеспечивается соблюдением принятых стандартизованных программных интерфейсов (*API-Application Program Interface*) между приложениями и функциональной средой открытых систем);

- мобильность пользователей (*user portability*), обеспечиваемая дружественным пользовательским интерфейсом (поддерживается стандартизованными *API* среды по функциям пользовательского интерфейса);

- интероперабельность (*interoperability*), означающая возможность взаимодействия данной ИС с другими системами при необходимости обращения к информационным ресурсам;

- масштабируемость (*scalability*) – возможность изменения количественных характеристик (размерности решаемых задач, числа обслуживаемых пользователей и т.д.) путем настройки параметров, а не перепроектирования и программирования заново; рост количественных системных характеристик при добавлении определенных вычислительных ресурсов (например, процессоров, модулей оперативной и дисковой памяти в конфигурациях серверов);

- способность к интеграции – возможность объединения нескольких ИС различного назначения в единую интегрированную многофункциональную ИС.

Для представления ИС и реализации свойств открытости используется модель открытой среды *OSE/RM*, которая синтезирует как прикладную функциональность, так и другие аспекты системы – системы администрирования, защиты и пр. – на референсном (эталонном, справочном) уровне.

Модель (рис. 2) представляет ИС прежде всего как сочетание прикладной (*App*) и платформенной (*Platf*) компонент, которые структурируются на три уровня и четыре функциональные группы в каждом:

- компоненты служб и сервисов промежуточного, системно-прикладного слоя (*MW*);
- компоненты операционных систем или операционного слоя (*OW*);
- аппаратный слой (*HW*).

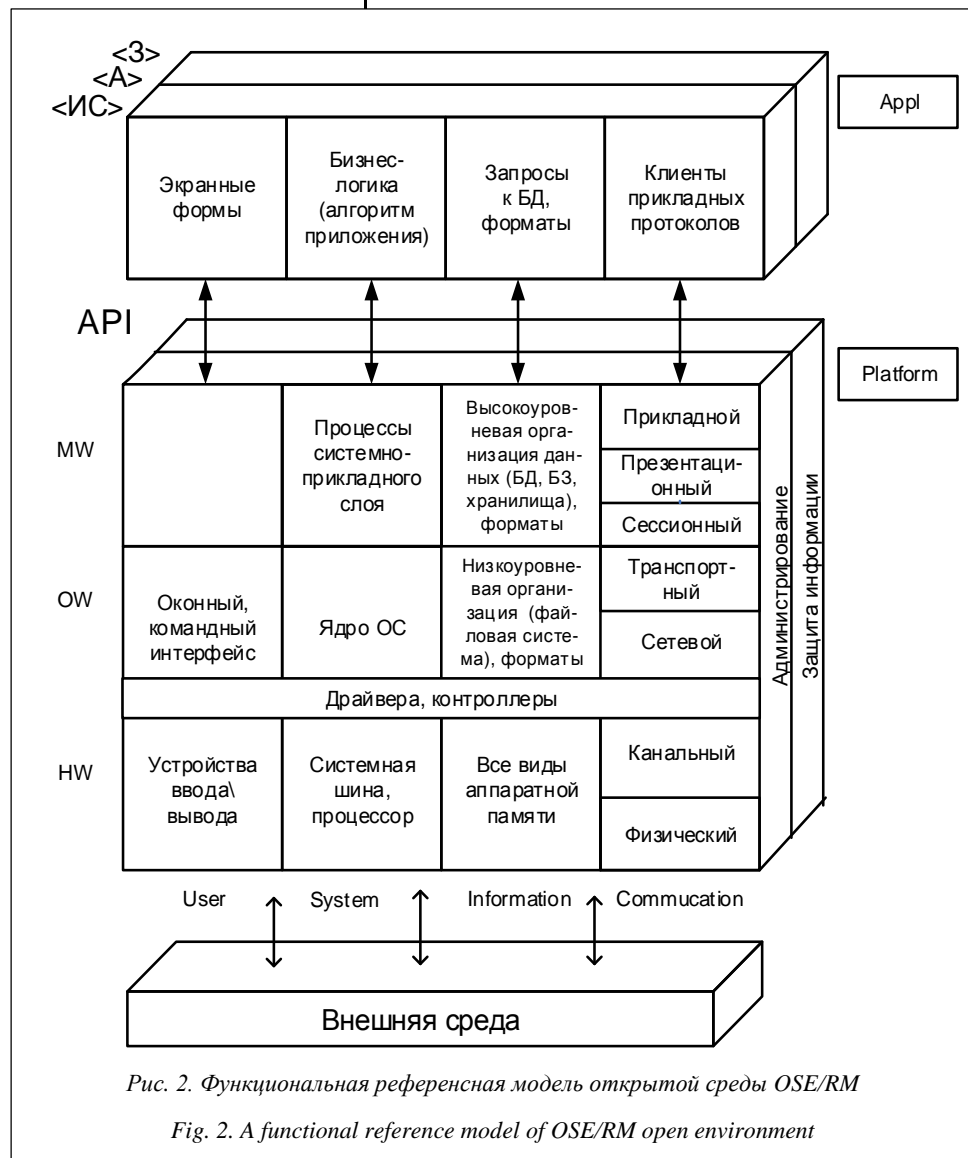


Рис. 2. Функциональная референсная модель открытой среды OSE/RM

Fig. 2. A functional reference model of OSE/RM open environment

Функциональные группы компонентов в данной модели составляют компоненты, обеспечивающие интерфейс с пользователем (*User*), все необходимые процессы в системе (*System*), организацию, представление, доступ и хранение данных (*Information*), а также компоненты телекоммуникационной среды, обеспечивающие взаимосвязь ИС (*Communication*) (данный уровень представляет собой модель взаимосвязи открытых систем (*OSI/RM* – *Open System Interconnection/Reference Model*)).

Кроме того, модель трехмерна. Она имеет дополнительные плоскости, в частности, администрирования и защиты, которые в зависимости от контекста аккумулируют сервисы, обслуживающие соответствующие клетки целевой плоскости.

Следует заметить, что при любых вариантах ЖЦ, отмеченных на рисунке 1, ИС может быть представлена моделью *OSE/RM*. При этом возможны два варианта: а) создание системы с нуля, которое заключается в разработке/покупке/аренде, интеграции реализаций всех компонент (клеток) модели; б) модификация существующей системы – внесение изменений в некоторые компоненты без изменения остальных.

Если ИС создаваемая, то ее частями (в соответствии с плоскостями модели *OSE/RM*) будут целевая компонента – передняя плоскость модели <ИС>, система администрирования – плоскость администрирования <А> и система безопасности – плоскость защиты <З>, обеспечивающая заданный уровень свойств конфиденциальности, целостности, доступности и др. относительно ресурсов бизнес-процесса.

Естественно, эти части желательно создавать параллельно в одном проекте, хотя на практике система безопасности зачастую достраивается в процессе эксплуатации целевого компонента.

Для модифицируемой ИС результатом проекта может быть изменение существующих конкретизаций некоторых клеток модели или разработка и добавление новых:

- приложение;
- программное средство платформы:
 - системная компонента операционного слоя;
 - изменение или разработка СУБД, СУБЗ, хранилища данных и т.п.;
 - телекоммуникационная компонента системы, обеспечивающая взаимосвязь систем (она представлена клетками столбца *Communication* модели *OSE/RM*, хотя возможна и разработка нового приложения);
 - интеграционная компонента системы, обеспечивающая взаимодействие систем в части данных, БД, приложений (строится на основе телекоммуникационной компоненты и соответствующих средств клетки «Процессы системно-прикладного слоя»);
- аппаратное средство.

Аналогичный перечень может использоваться для всех трех плоскостей модели, то есть и для це-

левой системы, и для систем безопасности и администрирования.

Плоскость защиты <З> представляется в двух аспектах.

Первый аспект. Клетки плоскости интегрируют совокупности механизмов, обеспечивающих защиту реализаций соответствующих клеток базовой плоскости <ИС>, плоскости <А> и саму себя – межкатегорийный аспект. При этом защитные механизмы (*Mx*) – достаточно сложные по своему содержанию сущности. Для дальнейшей структуризации механизмов по плоскости необходимо представить внутреннюю структуру *Mx* в виде некоторой модели. Для этой цели были использованы семантические модели в виде онтологий. На рисунке 3 представлена онтология класса механизмов защиты {*Mx*}.

Каждый *Mx* класса референсный и представляет собой иерархию механизмов-подклассов, действие которых направлено на разные объекты ИС. Назовем механизмы-подклассы механизмами первой очереди *Mx*¹, второй очереди *Mx*² и т.д. в зависимости от номера иерархического слоя. Например, механизм идентификации и аутентификации имеет иерархию вложенности в две очереди. Действие конечных механизмов таксономии направлено на обеспечение безопасности различных клеток модели *OSE/RM*.

Каждый *Mx* в иерархии может быть осуществлен каким-либо {Методом}, а каждый метод представляет собой программную или аппаратную реализацию некоторого {Алгоритма}. Именно алгоритмы обеспечивают тот или иной уровень безопасности.

Участниками *Mx* являются *субъект, операции* над *объектом*, которые он осуществляет. Каждый из этих участников обладает некоторыми атрибутами. Поэтому концепт {*Mx*} имеет набор атрибутов: {Режим}, {АБ}, {Операции}, {Субъект}, {Объект}. Режим определяет то, как будет использоваться механизм, например, значения режима могут быть «периодический» или «кзбирательный». Атрибут {АБ} определяет атрибуты безопасности, присущие каждому участнику *Mx*. Для механизмов разных очередей перечни атрибутов или их значения могут быть различными. Атрибут {Операции} задает действия, которые осуществляются над объектом или субъектом, а значения атрибутов {Субъект}, {Объект} определяют конкретные субъекты и объекты, над которыми *Mx* осуществляет свои операции. {Совокупная стойкость/Уровень критерия} определяет уровень свойств безопасности, необходимый для защиты соответствующей клетки модели *OSE/RM*. Указанные сущности для разных механизмов различны. Конечные механизмы таксономий (листья онтологий) уже можно соотносить с клетками *OSE/RM*.

Второй аспект. Реализация *Mx* плоскости <З> в виде ИС требует ее структуризации по аналогии с

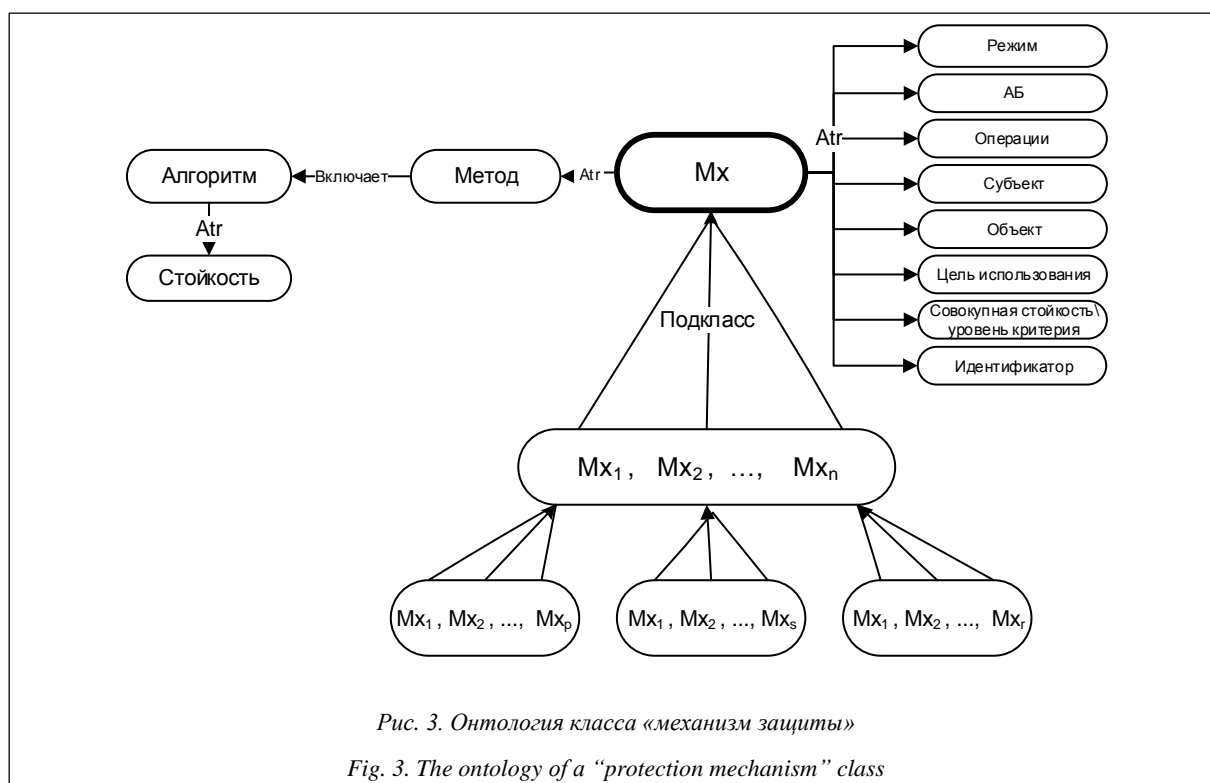


Рис. 3. Онтология класса «механизм защиты»

Fig. 3. The ontology of a "protection mechanism" class

базовым представлением <ИС> в виде прикладной и платформенной компонент модели OSE/RM с поправкой на контекст. Прикладная компонента включает приложения, представляющие собой программную реализацию тех Mx, которые реализуют межкатегорийное представление каждой клетки OSE/RM.

Уровни платформенной компоненты имеют следующую специфику:

- к HW-слою, помимо стандартной аппаратуры, относятся также аппаратные защитные средства (AMx), которые используются в соответствующих клетках;
- операционный уровень может представляться как стандартной операционной системой, обслуживающей запросы приложений плоскостей <ИС>, <A> и приложений <З>, так и специфической, работающей только на приложения системы защиты;
- системно-прикладной слой в данном случае обслуживает и организует выполнение приложений защиты.

Описание вариантов ЖЦ ИС

Способ приобретения предприятием КИС целиком и компонент ИС может быть различным. При этом ЖЦ во всех случаях строится по-разному и требует различных наборов работ. Поэтому методология предусматривает следующие варианты создания ИС.

1. В заказном варианте, когда предприятие заказывает разработку системы под свою специфику,

под реализацию конкретных бизнес-процессов. ЖЦ такой системы традиционен. На рисунке 1 этот вариант идентифицируется как «разрабатываемая ИС». Это может быть разработка с нуля, когда предприятие автоматизируется впервые, а может быть модификация уже существующей системы, но в любом случае это разработка под заказ.

2. Покупная система, когда компоненты системы приобретаются в готовом виде, зачастую у различных вендоров. ЖЦ покупной системы отличается от традиционного (рис. 4). Здесь речь идет о приобретении заказчиком экземпляра системы у производителя, поэтому на рисунке изображено представление ЖЦ как со стороны производителя, так и со стороны заказчика, поскольку, как правило, их отношения фактом покупки не ограничиваются, а продолжаются на стадии эксплуатации посредством договоров аутсорсинга на сопровождение системы.

Для заказчика важно не просто определиться с бизнес-требованиями, предъявляемыми к ИС, но и сопоставить их с тем, что предлагается на рынке, то есть произвести валидацию своих требований. Кроме того, в этом случае особую значимость приобретает процесс интеграции покупных компонент. Для этого необходимо наличие разработанных моделей OSE/RM и их нормативных спецификаций, которые фактически определяют перечень необходимых компонент системы и их характеристики. Кроме того, компоненты и их интерфейсы должны быть выполнены в соответствии с USIC-структуризацией открытых систем (рис. 2), что позволит без труда интегрировать их между собой.



3. ЖЦ гибридных ИС. В последнее время активно стали конфигурироваться так называемые гибридные КИС, когда для реализации бизнес-процесса заказчика к локальному компоненту добавляется ряд облачных сервисов. Это означает, что часть функций бизнес-процесса, то есть часть функциональных требований заказчика, реализуются за счет собственных вычислительных ресурсов, а другая часть – за счет публичных. При этом локальная часть может быть поставлена как в заказном, так и в тиражируемом варианте. ЖЦ этих вариантов имеет специфику и их следует рассматривать отдельно.

В качестве локального компонента рассматривается ИС, установленная и функционирующая непосредственно на предприятии заказчика. Под облачным сервисом понимается *стандартизованная* ИТ-услуга, которую *любой* провайдер может поставить на рынок и которой может воспользоваться *любой* пользователь. Это определение обуславливает следующие особенности, свойственные облачному публичному сервису (не путать с ИТ-услугами (сервисами) частных «облаков»).

- Функциональное содержание сервиса, реализованное любым поставщиком, должно отражать стандарты или иные законодательные акты, принятые в данной предметной области. Потребитель должен быть уверен: если он воспользовался, например, сервисом подготовки отчета в ПФР, этот отчет будет выполнен в соответствии с нормативными требованиями ведомства.

- Поскольку такой сервис должен быть доступным для любого пользователя, в Сети необходимо наличие общеизвестного ресурс-реестра, куда провайдер должен занести описание функци-

онального содержания предлагаемой услуги и интерфейса, по которому пользователь сможет к этой услуге обратиться. Сама же услуга (ее код) находится на ресурсах провайдера-поставщика.

Этим свойством *multitanancy* (разделяемости провайдеров и разделяемости пользователей) отличаются ИТ-услуги, предлагаемые в частных облаках: услуги, предлагаемые на ресурсах корпоративных ЦОД, эксплуатируются на принципах функционального взаимодействия, когда разработчики и потребители всегда могут напрямую договориться о способах обращения к услуге и ее требуемом содержимом. ИТ-услуга публичного облака фактически базируется на идеологии web-сервисов.

- Третья особенность использования публичных сервисов заключается в том, что очень часто потребителю приходится «докручивать» сервис под свои нужды. (В частных облаках, кстати, такой проблемы нет.) Тогда возникает потребность введения в число участников процесса (предприятия и поставщика сервисов) провайдера-интегратора, задача которого состоит в том, чтобы осуществлять доводку облачных сервисов в соответствии с бизнес-процессом заказчика.

ЖЦ гибридной КИС, состоящей из покупного локального компонента и совокупности облачных сервисов, представлен на рисунке 5.

Схема отражает ЖЦ компонент гибридной системы с точки зрения четырех участников процесса:

- заказчика гибридной системы, являющегося покупателем тиражируемого экземпляра коробочной системы и арендатором облачных сервисов (средний фрагмент схемы);

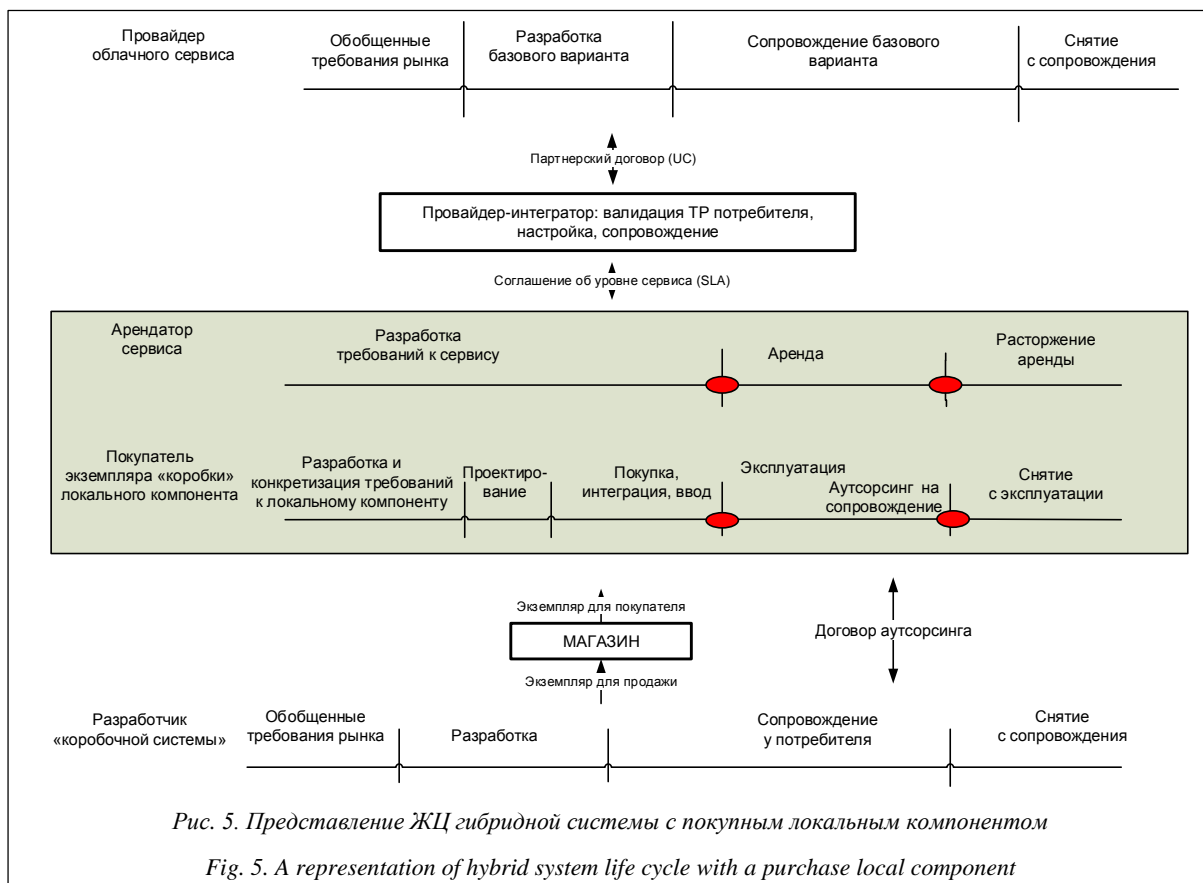


Рис. 5. Представление ЖЦ гибридной системы с покупным локальным компонентом

Fig. 5. A representation of hybrid system life cycle with a purchase local component

– разработчика тиражируемой системы, осуществляющего продажу системы через магазин и, как правило, сопровождение на условиях аутсорсинга;

– провайдера облачных сервисов, осуществляющего разработку базового варианта сервиса на основе обобщенных требований рынка, сопровождение базового варианта сервиса и снятие его с сопровождения в случае отсутствия требований на аренду данного сервиса;

– провайдер-интегратора облачных сервисов, в рамках соглашений об уровне сервиса осуществляющего валидацию требований заказчика с предложениями рынка, настройку нужных сервисов, их интеграцию и сопровождение сервиса под требования заказчика.

ЖЦ гибридной системы с разрабатываемым локальным компонентом аналогичен соответствующим поправкам.

Профилирование ИС и ЖЦ

При создании и развитии сложных распределенных ИС требуются гибкое формирование и применение гармонизированных совокупностей базовых стандартов и нормативных документов для унификации и регламентирования заданных функций ИС. Для этого используется методология ФС. Структуризация функциональности ИС в виде модели *OSE/RM* создает основу для применения мето-

дологии ФС, основным инструментом которой является понятие профиля.

По ГОСТ Р ИСО/МЭК ТО 10000-1-99, *профиль* – это совокупность нескольких (или подмножество одного) базовых стандартов (и других нормативных документов) с четко определенными и гармонизированными подмножествами обязательных и факультативных возможностей, предназначенная для реализации заданной функции или группы функций.

Для стандартизации ИС используются функциональные профили, регламентирующие архитектуру и структуру системы и ее компонентов в соответствии с компонентами и плоскостями модели *OSE/RM*. Следовательно, для любой ИС должны быть определены профили

- приложений;
- среды ИС, включающие в себя спецификации программных интерфейсов между приложениями и средой;
- интерфейсов между ИС и внешней для нее средой;
- администрирования целевой системы;
- защиты информации (включает профили межкатегорийного представления и базовой функциональности системы безопасности) [8].

Кроме набора функций, эти профили должны описывать интерфейсы взаимодействия соответствующих компонентов системы как с другими компонентами того же горизонтального уровня мо-

дели, так и с компонентами выше- и нижележащего уровней.

Таким образом, проектирование ИС в значительной степени сводится к ее компоновке из стандартизованных узлов. Этот подход позволяет осуществлять развитие и модернизацию ИС путем добавлений или замены отдельных узлов без изменения других частей системы.

Кроме функциональных профилей, в составе профилей ИС должны рассматриваться *вспомогательные профили*, регламентирующие процессы создания, сопровождения и развития ИС, то есть профили процессов ЖЦ. Такие профили в [9] называются технологическими. Прежде чем строить технологический профиль, необходимо для выбранного варианта ЖЦ разработать модель, основанную на нормативно-правовой базе и соответствующим образом структурированную.

Из существующих в настоящее время моделей наиболее распространены две: каскадная и спиральная. Они принципиально различаются самим подходом к моделированию ЖЦ ИС и ее ПО. Каскадная модель более универсальна, то есть применима к производству разных изделий и позволяет достичь хороших результатов, когда в самом начале разработки можно достаточно точно и полно сформулировать все требования, которые в процессе разработки изменяются незначительно. Спиральная модель более ориентирована именно на ИС, особенно на программные продукты, поэтому при разработке ИС и их ПО она предпочтительнее каскадной. Что касается стандартов и методологических документов, относящихся к ЖЦ систем и их ПО, действующих в РФ, то базовыми являются ГОСТ 34.601-90, ISO/IEC 12207 и ISO/IEC 15288 и их российские аналоги ГОСТ Р ИСО/МЭК 12207-2010 и ГОСТ Р ИСО МЭК 15288-2005. С точки зрения методологических подходов к управлению ЖЦ систем указанные документы имеют принципиальные различия. Так, ГОСТ 34.601-90 достаточно жестко устанавливает стадии и этапы создания ИС и отражает идеологию каскадной модели ЖЦ. Международные стандарты ISO/IEC 12207, ISO/IEC 15288 и соответствующие российские аналоги концептуально более развитые. В них заложена принципиально иная идея управления проектированием ИС.

В отличие от жесткой каскадной схемы ЖЦ, регламентируемой ГОСТ 34.601-90, указанные стандарты интерпретируются как библиотеки процессов, которые, вообще говоря, покрывают все пространство задач, возникающих при создании и сопровождении ИС. Но какие конкретно задачи надо решать при разработке конкретной ИС, то есть какие процессы выбирать из библиотеки, в какой последовательности их выстраивать, это дело исключительно экспертного суждения.

Профиль ЖЦ представляет собой набор стандартов, упорядоченный на основе выбранной мо-

дели ЖЦ, которой могут быть каскадная модель, каскадная модель с промежуточным контролем (эволюционная, итерационная) или спиральная модель. В [10] подробно описан алгоритм построения технологического профиля ЖЦ ИС в соответствии с нормативной базой.

Набор функциональных и технологических профилей составляет *полный профиль ИС*.

Профиль ЖЦ «закрывает» еще одну проблему указанных стандартов: все они имеют трехуровневую иерархию видов деятельности: стадия/процесс, этап/подпроцесс, работа/задача (здесь до «/» указаны термины по ГОСТ 34, после – по ГОСТ 12207), и если для первого уровня деятельности стандарты предусматривают достаточно содержательную интерпретацию (хотя бывают исключения), то для деятельностей нижних уровней семантика зачастую размывается. И это понятно, суть выполняемых работ здесь слишком вариативна, зависит от многих факторов: технологии разработки, специфики разрабатываемой ИС, вариантов компоновки ЖЦ и т.п.

В заключение отметим, что результаты работы целесообразно использовать в следующих ситуациях. Во-первых, при реализации проектов по разработке сложных систем, включая систему безопасности, с целью представления результата проекта – открытой ИС, а также объекта управления проекта – ЖЦ ИС – в стандартизованном виде, а также понимания взаимосвязи между бизнес-процессами предприятия, ИС и защитой, выстраиваемой исходя из соображений безопасности бизнес-процесса. Во-вторых, даже если система состоит из покупных компонент, перечисленные цели не теряют своей актуальности при дальнейшем сопровождении ИС у заказчика.

Кроме того, изложенный подход позволяет заказчику сохранить уже сделанные инвестиции при изменении требований к ОИС и ее развитию, использовать информационные ресурсы, эксплуатируемые в других системах, сократить затраты на обучение персонала при переходе на новые версии ИС, не зависеть от одного поставщика технических и программных средств, повторно использовать готовые приложения, а также облегчает решение проблемы «унаследованных» систем и т.п.

Литература

1. Батоврин В.К., Бахтурин Д.А. Управление жизненным циклом технических систем: Сер. докл. СПб, 2012. Вып. 1. 59 с.
2. Вендров А.М. Проектирование программного обеспечения. М.: Финансы и статистика, 2005. 544 с.
3. Позин Б.А. Ввод в действие информационных систем и сопровождение их программного обеспечения // Новые технологии. 2010. № 4. С. 1–32.
4. Бойченко А.В., Кондратьев В.К., Филинов Е.Н. Основы открытых информационных систем. М.: Изд-во Евразийского открытого ин-та, 2004. 128 с.
5. Лукинова О.В. Методология проектирования систем защиты, построенных на основе референсной модели POSIX OSE/RM // Системы высокой доступности. 2012. № 3. С. 38–45.

6. Калянов Г.Н. Модели и методы теории бизнес-процессов (обзор) // Открытое образование. 2015. № 6. С. 4–9.
7. Васильев Р.Б., Калянов Г.Н., Левочкина Г.А. Направления стратегического ИТ-консалтинга // Автоматизация в промышленности. 2009. № 11. С. 3–8.
8. Лукинова О.В., Пугачев А.В. Особенности построения профилей систем безопасности ИС // Открытое образование.

2015. № 4. С. 80–87.

9. Липаев В.В., Филинов Е.Н. Формирование и применение профилей открытых информационных систем // Открытые системы. 1997. № 5. С. 3–11.

10. Бойченко А.В., Лукинова О.В. Управление жизненным циклом информационной системы на основе профилей // Системы высокой доступности. 2015. Т. 11. № 3. С. 64–68.

Software & Systems

DOI: 10.15827/0236-235X.116.027-035

Received 01.08.16

2016, vol. 29, no. 4, pp. 27–35

METHODOLOGICAL ASPECTS OF INFORMATION SYSTEM LIFE CYCLE MANAGEMENT BASED ON FUNCTIONAL STANDARDIZATION TOOLS

O.V. Lukinova¹, Dr.Sc. (Engineering), Leading Researcher, lobars@mail.ru

¹ V.A. Trapeznikov Institute of Control Sciences of RAS, Profsoyuznaya St. 65, 117997, Russian Federation

Abstract. The paper presents a methodology for managing the life cycle of information systems. It synthesizes tools and models of functional standardization, open system theories, management consulting, and basic standards in management of information system life cycle.

The paper also describes the basics and a conceptual model showing the relationship between problems to be solved using the methodology framework. The problems are the following: construction of a business process model using management consulting; information system model development in accordance with the OSE/RM model (Open System Environment / Reference Model) and its standardization using a functional profile; choosing a life cycle type, development of its respective model and profiling.

The paper investigates different types of a life cycle, focusing on a purchase system life cycle including a set of integrated public services in addition to a local component (deployed at a customer site). The structural representation of the information system target components together with its security system is shown in terms of the OSE/RM model. The article also describes a conceptual model of a protection mechanism (which is a base element of a protection system functional structure).

Keywords: information system, functional and technological profile, open system model, open system properties, functional standardization.

References

1. Batovrin V.K., Bakhturin D.A. *Upravlenie zhiznennym tsiklom tekhnicheskikh sistem: ser. dokl.* [Technical System Life Cycle Management: report series]. St. Petersburg, 2012, iss. 1, 59 p.
2. Vendrov A.M. *Proektirovanie programmnoy obespecheniya* [Software Design]. Moscow, Finansy i statistika Publ., 2005.
3. Pozin B.A. Implementation of information systems and their software support. *Novye tekhnologii* [New Technologies]. 2010, no. 4, pp. 1–32 (in Russ.).
4. Boychenko A.V., Kondratev V.K., Filinov E.N. *Osnovy otkrytykh informatsionnykh sistem* [The Basics of Opened Information Systems]. Kondratev V.K. (Ed.). Moscow, EAOI Publ., 2004, 128 p.
5. Lukinova O.V. Methodology of design of systems of the security, constructed on the basis of the reference POSIX OSE/RM model. *Sistemy vysokoy dostupnosti* [Highly Available Systems]. 2012, no. 3, pp. 38–45 (in Russ.).
6. Kalyanov G.N. Models and methods of a business process theory (review). *Otkrytoe obrazovanie* [Open Edukation]. 2015, no. 6, pp. 4–9 (in Russ.).
7. Vasilev R.B., Kalyanov G.N., Levochkina G.A. Directions of strategic IT consulting. *Avtomatizatsiya v promyshlennosti* [Automation in Industry]. 2009, no. 11, pp. 3–8 (in Russ.).
8. Lukinova O.V., Pugachev A.V. Construction features of IP security systems' sections. *Otkrytoe obrazovanie* [Open Edukation]. 2015, no. 4, pp. 80–87 (in Russ.).
9. Lipaev V.V., Filinov E.N. Formation and application of open information system profiles. *Otkrytye sistemy* [Open Systems]. 1997, no. 5, pp. 3–11 (in Russ.).
10. Boychenko A.V., Lukinova O.V. Management of life cycle of information system on the basis of profiles. *Sistemy vysokoy dostupnosti* [Highly Available Systems]. 2015, no. 3, iss. 11, pp. 58–63 (in Russ.).