

О.В. Пешкова

АРХИТЕКТУРА ПРЕДПРИЯТИЯ

Учебное пособие

Министерство науки и высшего образования Российской Федерации
Байкальский государственный университет

О.В. Пешкова

АРХИТЕКТУРА ПРЕДПРИЯТИЯ

Учебное пособие

Иркутск
Издательство БГУ
2019

УДК 004.275.1:658(075.8)

ББК 32.97:65.291.216я7

ПЗ1

Печатается по решению редакционно-издательского совета
Байкальского государственного университета

Рецензенты канд. экон. наук, доц. Т.И. Хитрова
канд. техн. наук, доц. А.В. Родионов

Пешкова О.В.

ПЗ1 Архитектура предприятия [Электронный ресурс] : учеб. пособие /
О.В. Пешкова. — Иркутск : Изд-во БГУ, 2019. — 105 с.— Режим доступа:
<http://lib-catalog.bgu.ru>.

Рассмотрено понятие предприятия как объекта системного моделирования, дано определение и описаны составляющие архитектуры предприятия как базы для эффективного построения информационных систем организации, представлены методологии построения архитектуры предприятия, а также модели и методики разработки архитектур предприятия, используемые в современном мире.

Рекомендовано студентам бакалавриата, обучающимся по направлениям подготовки 09.03.03 Прикладная информатика, 38.03.05 Бизнес-информатика.

УДК 004.275.1:658(075.8)

ББК 32.97:65.291.216я7

© Пешкова О.В., 2019

© Издательство БГУ, 2019

Оглавление

Предисловие	4
1. Понятие архитектуры предприятия.....	6
1.1. Предприятие как объект моделирования.....	6
1.2. Предприятие как система.....	16
1.3. Архитектурный подход как основа управления развитием информационных систем.....	22
1.4. Понятие архитектуры и значение архитектуры предприятия в современных условиях.....	25
1.5. Эволюция представлений об архитектуре предприятия.....	28
2. Построение архитектуры предприятия	34
2.1. Процесс и принципы выстраивания архитектуры	34
2.2. Архитектурные слои и их наполнение.....	35
2.3. Бизнес-архитектура: процессная модель	41
2.4. Системная архитектура	45
2.5. Традиционные инструменты описания архитектуры.....	52
3. Модели представления, методы и инструментальные среды разработки архитектуры предприятия.....	61
3.1. Модель Захмана — базовая методология построения архитектуры предприятия	61
3.2. Обзор моделей и методик построения архитектуры предприятия	67
3.3. Метод планирования архитектуры организации EAP.....	79
3.4. Построение архитектуры предприятия с использованием методологии ARIS	89
Список использованной и рекомендуемой литературы.....	104

Предисловие

Современная экономика немыслима без эффективного управления. Успех управления во многом определяется эффективностью принятия решений, которые учитывают самые разносторонние факторы и тенденции развития объекта управления. Все знают, что мир меняется все быстрее и что информационные системы должны успевать за этими изменениями. Но известно и другое. Внедренную и реально используемую систему трудно (долго, дорого, часто даже невозможно) изменять. В результате информационная система может стать или становится тормозом развития предприятия. Причины могут быть разные, но истоки проблем — всегда в трансформации предприятия.

Информационная управляющая система — это не только программы, данные и коммуникации, но также и люди (заказчики, пользователи, аналитики, конструкторы и реализаторы), организационные структуры, планы-графики работы, а также цели и стимулы предприятия и отдельных людей. Все это должны быть понятным и непротиворечивым образом соединены в одну систему. Всевозрастающая сложность и многоаспектность предприятия определяют растущую сложность согласования его частей и аспектов работы. Основная идея такого согласования: его надо начинать с самых главных характеристик предприятия, рассматривая самые главные содержательные аспекты. И проводить его не мысленно и не на словах, а на явно изложенных описаниях предприятия, позволяющих видеть все существенные взаимосвязи, а это значит — на его моделях.

Отсутствие решений в области информационных технологий и автоматизированных информационных систем или бессистемное их принятие на практике приводят к появлению большого разнообразия аппаратных средств и приложений, напоминающих спонтанную застройку в условиях отсутствия градостроительных планов со всеми вытекающими последствиями. Для решения этих проблем и появилась концепция «архитектура системы» и «архитектура предприятия».

Архитектура системы по определению является бесконечно сложным, глубоким и неявным понятием. Только часть этого общего понятия, которая в принципе доступна для восприятия архитекторами, может быть переведена в явную документируемую форму — модель или набор моделей с неизбежными упрощениями, ограничениями и субъективными искажениями. Такая проекция и представляет собой «описание архитектуры». Поэтому использование подобных описаний при проектировании систем всегда подразумевает их неизбежную неполноту. При этом совершенно уместным является замечание о том, что все модели являются, в общем-то, неверными, но некоторые из них при этом являются полезными.

В настоящее время все больше внимания уделяют новому направлению, которое стали называть архитектурой предприятия. Это направление изначально предназначалось для решения двух основных проблем в сфере информационных технологий (ИТ), выявленных 20 лет назад. Первая проблема заключается в постоянном увеличении сложности ИТ-систем, что приводит к увеличению затрат на их построение. Вторая вызвана тем, что со временем получить реальную от-

дачу от ИТ-систем становится труднее, т.е. несмотря на всевозрастающую стоимость ИТ-систем, организациям с большим трудом удавалось поддерживать их соответствие требованиям бизнеса.

Очевидно, эти проблемы взаимосвязаны. Чем сложнее система, тем труднее ею управлять. В итоге — высокие затраты, низкая эффективность. Эти проблемы постепенно достигли критической точки. Стоимость и сложность ИТ-систем выросли экспоненциально, а реальная польза от них резко снизилась. Чем лучше удается справиться со сложностью системы, тем выше вероятность получения от нее реальной выгоды. Для создания простой однопользовательской информационной системы понятие «архитектура» не нужно. Но в процессе создания корпоративной, распределенной системы представление об архитектуре все более усложняется.

Следует отметить, что архитектура предприятия существует независимо от предпринимаемых в организации проектов по ее описанию, упорядочиванию и развитию. Представление об архитектуре предприятия имеет свои корни в дисциплине, которая получила название «системное мышление», так как любое предприятие — это сложная система и для выстраивания эффективной системы управления необходим системный подход.

Принципы построения архитектуры предприятия, которые 20 лет назад представлялись странными и далекими от действительности, в настоящее время становятся все более востребованными. За прошедший период был разработан ряд методологий построения архитектуры предприятия, описанию которых посвящено данное учебное пособие.

1. Понятие архитектуры предприятия

1.1. Предприятие как объект моделирования

Прежде чем рассматривать понятие «архитектура предприятия» необходимо уточнить что представляет собой предприятие как система и как объект моделирования, поскольку архитектура предприятия в целом может рассматриваться как его системная модель, предназначенная для изучения его поведения в изменяющейся среде и принятия решения о проведении организационных изменений, чаще всего, с использованием информационных технологий (ИТ). В российской и зарубежной литературе предприятие определяют как организацию (хозяйствующий субъект), созданную для производства продукции, выполнения работ и оказания услуг в целях удовлетворения общественных потребностей и получения прибыли. Согласно более общему определению, предприятие представляет собой комплексную систему культурных, технологических и процессных компонент, организованных для достижения целей организации.

В данном учебном пособии под термином «предприятие» будем понимать формальное объединение, не обязательно связанное с коммерческой деятельностью. Это может быть и государственная организация, и общественная, в том числе неформальное объединение участников, связанных общей целью.

Как объект моделирования предприятие рассматривается в рамках бизнес-моделирования, представляющего собой процесс разработки и внедрения бизнес-моделей организации (миссия и стратегии, бизнес-процессы, организационно-функциональная структура, качество, ключевые показатели деятельности и др.) с целью формализации и оптимизации ее деятельности. Философский энциклопедический словарь определяет модель как образ некоторой системы, аналог определенного фрагмента природной или социальной реальности, «заместитель» оригинала в познании и практике. В широком смысле модель — любой образ, аналог (мысленный или условный: изображение, описание, схема, чертеж, график и т.п.) какого-либо объекта, процесса или явления (оригинала данной модели).

Определение бизнес-модели зависит от точек зрения с которых дается это понятие. С точки зрения менеджмента можно выделить два подхода к определению бизнес-модели: ориентация на ценность/клиента и ориентация на внутренние бизнес-процессы/роли. С позиций первого подхода бизнес-модель логически описывает каким образом организация создает, поставляет клиентам и приобретает стоимость — экономическую, социальную и другие формы стоимости. Второй подход определяет бизнес-модель как структуру бизнес-процессов и связей между ними, которые используются для планирования, контроля и коррекции деятельности предприятия.

С точки зрения системного подхода бизнес-модель — это описание предприятия, как сложной системы, с заданной точностью, в рамках которой отображаются все объекты (сущности), процессы, правила выполнения операций, существующая стратегия развития, а также критерии оценки эффективности функционирования системы. С описательной точки зрения бизнес-модель предприя-

тия — это совокупность графических и текстовых описаний, позволяющих понимать процессы, протекающие на предприятии (процессы управления, производственные процессы, процессы взаимоотношений с клиентами и т.п.) а в случае использования электронных средств динамического моделирования имитировать эти процессы.

Очевидно, что для сложных систем практически невозможно получить единственное описание, отвечающее на все вопросы с точки зрения управления. Например, одна и та же организация может быть представлена как:

- инфраструктура (территория, здания, сооружения, технологии, коммуникации и т.д.);
- организационная структура;
- совокупность источников и потоков информации и типов данных;
- сеть процессов, с помощью которых организация выполняет свою миссию.

На рис. 1 представлена схема взаимосвязи базовых компонент бизнес-модели организации. При создании моделей сначала описывают миссию и фундаментальные цели предприятия, затем дают характеристику структурных элементов и/или процессов, которые позволяют этих целей достигнуть.

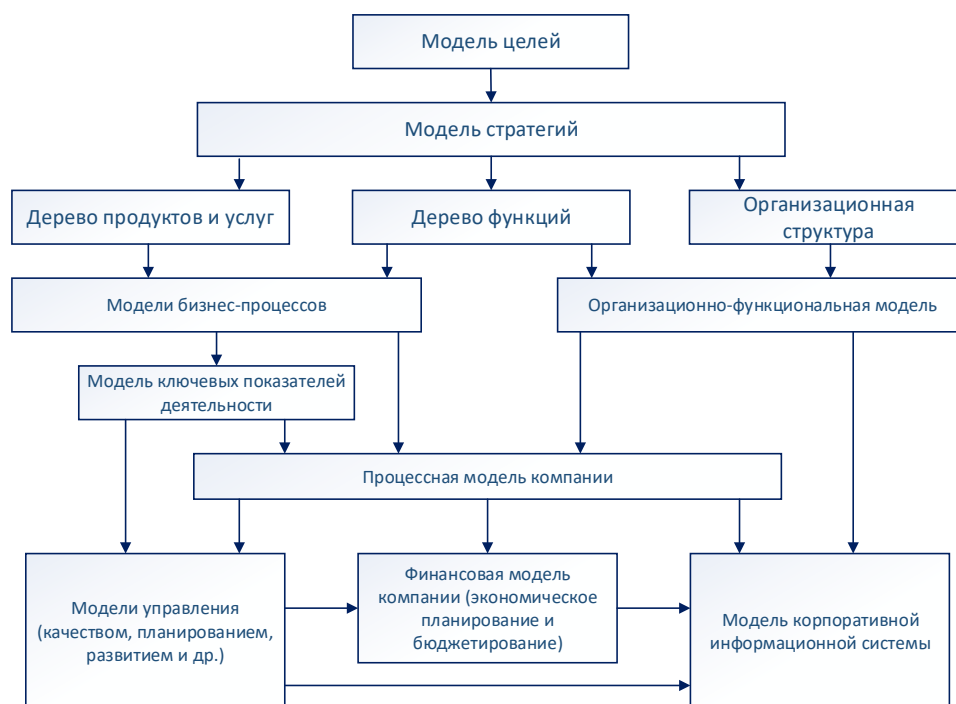


Рис. 1. Схема моделей предприятия и их взаимосвязи

В основе бизнес-модели всегда лежат бизнес-цели предприятия, сформулированные на базе стратегических целей и миссии организации, которые полностью определяют состав всех базовых компонентов бизнес-модели:

- бизнес-функции, описывающие, ЧТО делает бизнес;
- бизнес-процессы, описывающие, КАК предприятие выполняет свои бизнес-функции;
- организационная структура, определяющая, ГДЕ исполняются бизнес-функции и бизнес-процессы;

- фазы, определяющие, КОГДА (в какой последовательности) должны быть внедрены те или иные бизнес-функции;
- роли, определяющие, КТО исполняет бизнес-процессы;
- правила, определяющие связь между ЧТО, КАК, ГДЕ, КОГДА и КТО.

Миссия — это утверждение, раскрывающее смысл существования организации, в котором проявляется отличие данной организации от ей подобных и отражающее интересы всех участников: владельцев бизнеса, акционеров, сотрудников, потребителей. Миссией также называют основную цель организации. Если миссия задает общие ориентиры, направления функционирования организации, выражающие смысл ее существования, то конкретные конечные состояния, к которым стремится организация, фиксируются в виде ее целей, т.е., говоря иначе, цели — это конкретное состояние отдельных характеристик организации, достижение которых является для нее желательным и на достижение которых направлена ее деятельность.

Сформулированные на основе миссии компании цели могут быть классифицированы по разным признакам, например, долгосрочные (стратегические), среднесрочные (тактические) и краткосрочные (оперативные); цели-задания и цели-ориентиры; цели функционирования и цели развития, и т.п. Кроме того, они должны удовлетворять SMART-критериям. Аббревиатура SMART расшифровывается английскими словами Specific — конкретный, Measurable — измеримый, Attainable — достижимый, Relevant — значимый, Time-bounded — обозначенный во времени. Стратегия как модель организации бизнес-процессов предприятия для достижения долгосрочных целей формулирует общие направления развития деятельности предприятия, в первую очередь касающиеся производимой продукции и каналов ее продвижения. При этом стратегия должна обеспечить концентрацию усилий в той области, где будут иметь место устойчивые конкурентные преимущества. Разработка корпоративной стратегии позволяет перейти от управления организацией, зависящего от воздействия случайно возникающих внешних и внутренних факторов, к планомерной деятельности по достижению определенных результатов с возможностью оценки их достижимости по определенным критериям и применения адекватных управляющих воздействий [2], подробнее о миссии, целях и стратегии см. [7].

Далее строятся продуктовая, функциональная и организационная модели в виде иерархии соответственно продуктов и услуг, выполняемых функций и организационных звеньев. В практике хозяйствования в зависимости от масштаба деятельности предприятия, его производственно-технических особенностей, стратегических и текущих задач может применяться несколько типов организационных структур:

- линейная;
- функциональная;
- линейно-функциональная;
- дивизиональная;
- проектная;
- матричная.

Линейная организационная структура управления — наиболее простая иерархическая структура, в которой каждый подчиненный имеет только одного руководителя, выполняющего все административные и специальные функции в соответствующем структурном подразделении. Управление организовано по линейной схеме (директор → начальник цеха → мастер). Преимущества организационной структуры управления линейного типа: четкость взаимоотношений, единоначалие, однозначность команд, оперативность подготовки и реализации управленческих решений, надежный контроль. Недостаток структуры заключается в том, что руководитель при этом должен быть высококвалифицированным универсалом, способным решать любые стратегические и текущие вопросы деятельности подчиненных ему подразделений.

Основой *функциональной организационной структуры управления* является разделение функций управления между отдельными подразделениями аппарата управления. При этом функциональный руководитель может отдавать распоряжения не только сотрудникам своего функционального подразделения. Основные преимущества функциональной структуры — прямое воздействие специалистов на производство, высокий уровень специализации управления, глубокая разработка и обоснование принимаемых решений. Основные недостатки — нарушение принципа единоначалия, сложность и неэкономичность, обусловленные большим количеством подразделений, а, следовательно, и каналов управления.

На практике чаще всего используется *линейно-функциональная структура*, которая характерна для организации малого и среднего размера, а также для крупных организаций, выпускающих монопродукт или ограниченную группу продуктов (рис. 2). В линейно-функциональной структуре выделяются руководители линейных (производственных) подразделений, и руководители функциональных подразделений (отдел стратегического планирования, ИТ-отдел, финансовое управление и т.п.). Такой тип организационной структуры позволяет организовать управление по линейной схеме, а функциональные отделы аппарата управления предприятия помогают линейным руководителям решать управленческие задачи. При этом линейные руководители не подчинены руководителям функциональных отделов аппарата управления, функциональные связи между сотрудниками разных подразделений осуществляются через вышестоящее руководство.

Преимущество линейно-функциональной структуры состоит в быстрой реализации управленческих решений вследствие иерархичности ее построения, снятие большей части нагрузки с высшего уровня управления, что способствует повышению уровня специализации и эффективности работы функциональных служб. К ее достоинствам также можно отнести: стабильность (наиболее эффективны в стабильной среде); экономию на управленческих расходах; быстрое решение простых проблем, находящихся в компетенции одной функциональной службы.

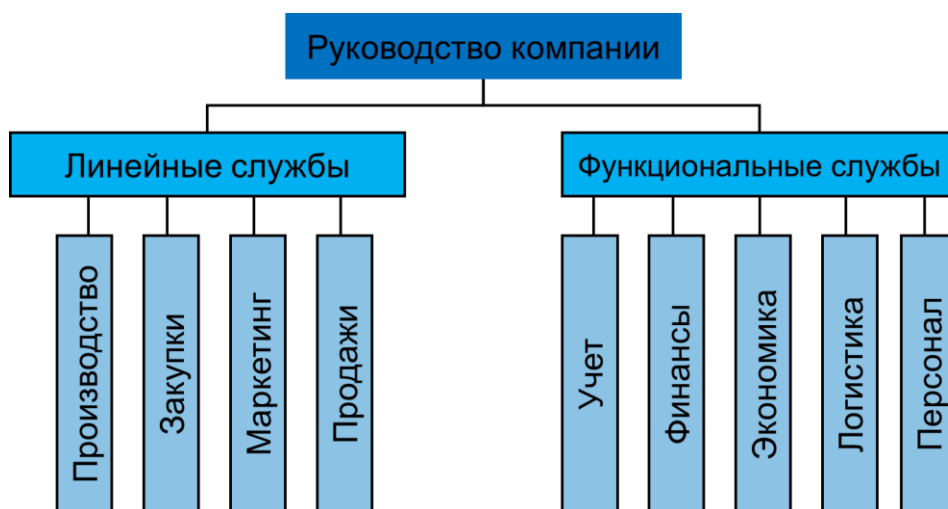


Рис. 2. Линейно-функциональная организационная структура

Недостаток состоит в значительном усложнении связей внутри предприятия и в появлении большого количества новых информационных каналов, это затрудняет координацию деятельности организации, дает возможность переноса ответственности за неудачи на сотрудников других подразделений. Кроме того, следует отметить, что поскольку у руководителей среднего звена появляются возможности влиять на решения вышестоящих менеджеров в собственных интересах или в интересах своей функциональной службы, линейно-функциональная структура порождает «ведомственность» внутри предприятия. Узкая специализация работников, которая сужает горизонт их профессионального видения, снижает общеорганизационные цели и задачи до функциональных. Ограничение возможности профессионального развития функциональных и особенно линейных руководителей (последние освобождаются от специализированных управленческих функций, сосредотачивая свое внимание на проблемах собственно производства).

В настоящее время крупные компании используют *дивизиональную организационную структуру*, которая базируется на углублении разделения управленческого труда. Схема дивизиональной структуры приведена на рис. 3. Для такой структуры характерна децентрализация оперативных функций управления, осуществляемых производственными звеньями, и централизации общекорпоративных функций (принятия стратегических решений, маркетинговых исследований, финансовой деятельности и др.), которые концентрируются в высших звеньях администрации интегрированных предпринимательских структур. Это позволяет производственным подразделениям решать самостоятельные задачи, связанные с разработкой, производством и сбытом собственной продукции. Высшее руководство предприятия может при этом сосредоточиться на постановке и решении стратегических задач.

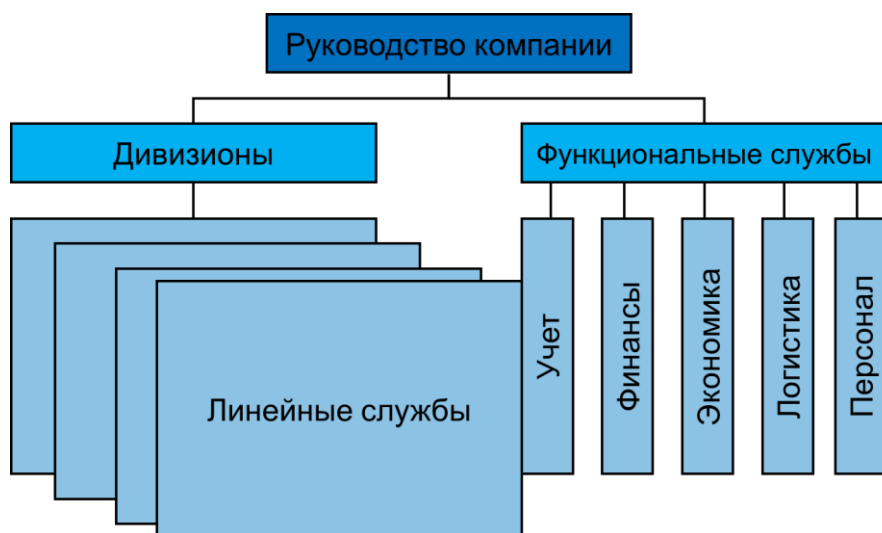


Рис. 3. Дивизиональная организационная структура

При дивизиональной организационной структуре возможны три способа группирования производственных подразделений:

- продуктовый (изготовление определенного продукта);
- по группам потребителей (удовлетворение потребностей определенной группы потребителей);
- территориальный (размещение в определенном географическом районе).

Преимущества дивизиональной структуры:

- обеспечивает управление многопрофильными предприятиями с общей численностью сотрудников порядка сотен тысяч и территориально удаленными подразделениями;
 - корпоративное руководство освобождается от оперативных функций и имеет возможность сконцентрироваться на стратегии;
 - по сравнению с линейной и линейно-функциональной структурой более гибкое реагирование на изменения во внешней среде и более быстрое принятие управленческих решений;
 - уменьшение сложности управления, с которой сталкиваются управляющие высшего звена;
 - при расширении границ самостоятельности дивизионов они становятся «центрами получения прибыли», активно работая на повышение эффективности и качества производства;
 - более тесная связь производства с потребителями;
 - отделение оперативного управления от стратегического, в результате чего высшее руководство компании концентрируется на стратегическом планировании и управлении;
 - перенесение ответственности за прибыль на уровень дивизионов, децентрализацию принятия оперативных управленческих решений, такая структура помогает приблизить руководство к проблемам рынка;
 - улучшение коммуникаций;
 - развитие широты мышления, гибкости восприятия и предприимчивости руководителей отделений (дивизионов).

Использование дивизиональных структур позволяет компании уделять конкретному продукту, потребителю или географическому региону столько же внимания, сколько уделяет небольшая специализированная компания, в результате чего возможно быстрее реагировать на изменения, происходящие во внешней среде, адаптироваться к изменяющимся условиям. Этот вид структуры управления ориентирует на достижение конечных результатов деятельности компании (производство конкретных видов продукции, удовлетворение потребностей определенного потребителя, насыщение товарами конкретного регионального рынка).

Недостатки дивизиональной структуры:

- большое количество уровней управления, что влечет за собой необходимость увеличения при такой структуре численности аппарата управления и затрат на его содержание;
- разобщенность дивизионов от централизованных функциональных служб;
- дивизиональные структуры управления привели к росту иерархичности, т.е. вертикали управления. Они потребовали формирования промежуточных уровней менеджмента для координации работы отделений, группы и т.п.;
- противопоставление целей отделений общим целям развития компании, несовпадение интересов «верхов» и «низов» в многоуровневой иерархии;
- возможность возникновения межотделенческих конфликтов, в частности, в случае дефицита централизованно распределяемых ключевых ресурсов;
- невысокая координация деятельности отделений (дивизионов), штабные службы разобщены, горизонтальные связи ослаблены;
- неэффективное использование ресурсов, невозможность их использовать в полной мере в связи с закреплением ресурсов за конкретным подразделением;
- увеличение затрат на содержание управленческого аппарата вследствие дублирования одних и тех же функций в подразделениях и соответствующего увеличения численности персонала;
- затруднение осуществления контроля сверху донизу;
- многоуровневая иерархия и в рамках самих отделений (дивизионов), действие в них всех недостатков линейно-функциональных структур;
- возможное ограничение профессионального развития специалистов подразделений, поскольку их коллективы не столь велики, как в случае применения линейно-функциональных структур на уровне компаний.

Широкий спектр недостатков дивизиональной и линейно-функциональной структур привело к созданию адаптивных структур. Для адаптивных (гибких, органических) организационных структур характерно отсутствие бюрократической регламентации деятельности органов управления, отсутствие детального разделения труда по видам работ, размытость уровней управления и небольшое их количество, гибкость структуры управления, децентрализация принятия решений, индивидуальная ответственность каждого работника за общие результаты деятельности. Кроме того, адаптивные организационные структуры, как правило, характеризуются следующими признаками:

- способностью сравнительно легко менять свою форму, приспосабливаться к изменяющимся условиям;
- ориентацией на ускоренную реализацию сложных проектов, комплексных программ, решение сложных проблем;
- ограниченным действием во времени, т.е. формированием на временной основе на период решения проблемы, выполнения проекта, программы;
- созданием временных органов управления.

Матричная структура (рис. 4) отражает закрепление в организационном построении фирмы двух направлений руководства, двух организационных альтернатив. Вертикальное направление — управление функциональными и линейными структурными подразделениями компании. Горизонтальное — управление отдельными проектами, программами, продуктами, для реализации которых привлекаются человеческие и иные ресурсы различных подразделений компании. Матричная структура чаще всего представляет собой наложение проектной структуры на постоянную для данной компании линейно-функциональную структуру управления.

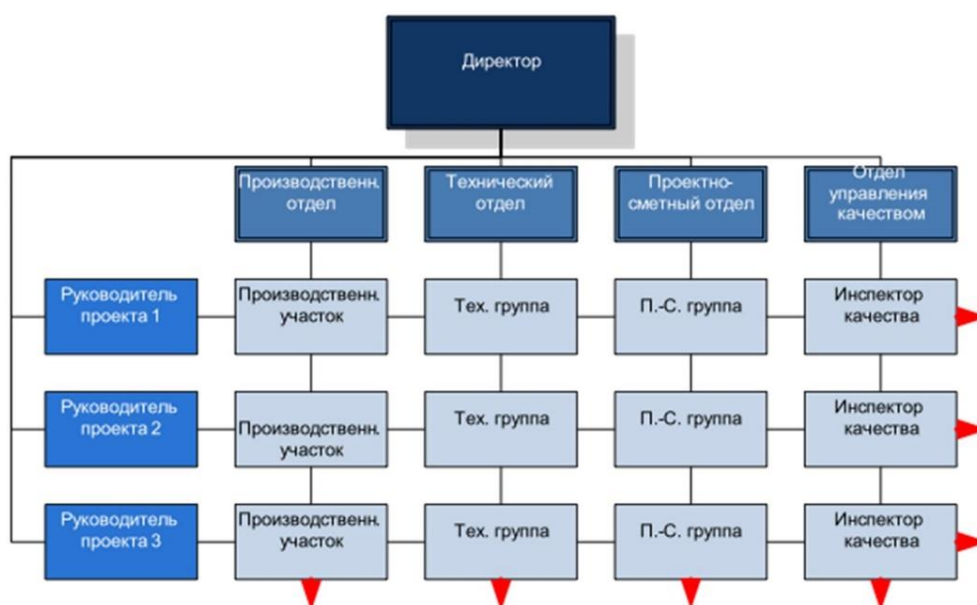


Рис. 4. Матричная организационная структура

Преимущества матричной структуры:

- интеграция различных видов деятельности компании в рамках реализуемых проектов, программ;
- получение высококачественных результатов по большому количеству проектов, программ, продуктов;
- значительная активизация деятельности руководителей и работников управленческого аппарата в результате формирования проектных (программных) команд, активно взаимодействующих с функциональными подразделениями, усиление взаимосвязи между ними;

- вовлечение руководителей всех уровней и специалистов в сферу активной творческой деятельности по реализации организационных проектов и, прежде всего, по ускоренному техническому совершенствованию производства;
- сокращение нагрузки на руководителей высшего уровня управления путем передачи полномочий принятия решений на средний уровень при сохранении единства координации и контроля над ключевыми решениями на высшем уровне;

- усиление личной ответственности конкретного руководителя как за проект (программу) в целом, так и за его элементы;

- достижение большей гибкости и скоординированности работ, чем в линейно-функциональных и дивизиональных организационных структурах управления, т.е. лучшее и более быстрое реагирование матричной структуры на изменение внешней среды;

- преодоление внутриорганизационных барьеров, не мешая при этом развитию функциональной специализации

Недостатки матричной структуры:

- сложность матричной структуры для практической реализации, для ее внедрения необходима длительная подготовка работников и соответствующая организационная культура;

- структура сложна, громоздка и дорога не только во внедрении, но и в эксплуатации;

- она является трудной и порой непонятной формой организации;

- в связи с системой двойного подчинения подрывается принцип единоначалия, что часто приводит к конфликтам; в рамках этой структуры порождается двусмысленность роли исполнителя и его руководителей, что создает напряжение в отношениях между членами трудового коллектива компании;

- в рамках матричной структуры наблюдается тенденция к анархии, в условиях ее действия нечетко распределены права и ответственность между ее элементами;

- для этой структуры характерна борьба за власть, т. к. в ее рамках четко не определены властные полномочия;

- для данной структуры характерны чрезмерные накладные расходы в связи с тем, что требуется больше средств для содержания большего количества руководителей, а также порой на разрешение конфликтных ситуаций;

- мешает достижению высококачественных результатов двусмысленность и потеря ответственности;

- при использовании матричной структуры возникают трудности с перспективным использованием специалистов в данной компании;

- наблюдается частичное дублирование функций;

- несвоевременно принимаются управленческие решения; как правило, характерно групповое принятие решений;

- отмечается конформизм в принятии групповых решений;

- нарушается традиционная система взаимосвязей между подразделениями;

- в условиях матричной структуры затрудняется и практически отсутствует полноценный контроль по уровням управления;
- структура считается абсолютно неэффективной в кризисные периоды.

Проектные структуры — это структуры управления комплексными видами деятельности, которые из-за их решающего значения для компании требуют обеспечения непрерывного координирующего и интегрирующего воздействия при жестких ограничениях по затратам, срокам и качеству работ. Под проектной структурой управления понимается временная структура, создаваемая для решения конкретной комплексной задачи (разработки проекта и его реализации). Смысл проектной структуры управления состоит в том, чтобы собрать в одну команду самых квалифицированных сотрудников разных профессий для осуществления сложного проекта в установленные сроки с заданным уровнем качества и в рамках выделенных для этой цели материальных, финансовых и трудовых ресурсов. Проектная структура управления предполагает обеспечение централизованного управления всем ходом работ по каждому крупному проекту.

На базе продуктовой, функциональной и организационной моделей заполняется матрица организационных проекций (рис. 5), представляющая собой организационно-функциональную модель. Матрица организационных проекций позволяет не только создать эффективную организационную структуру, но и в компактной форме зафиксировать информацию о том, кто и что делает в подразделении и более того, помогает в решении многих задач менеджмента, в том числе в управлении персоналом. В связи с этим любые изменения в структуре или перераспределение функций должны немедленно фиксироваться.

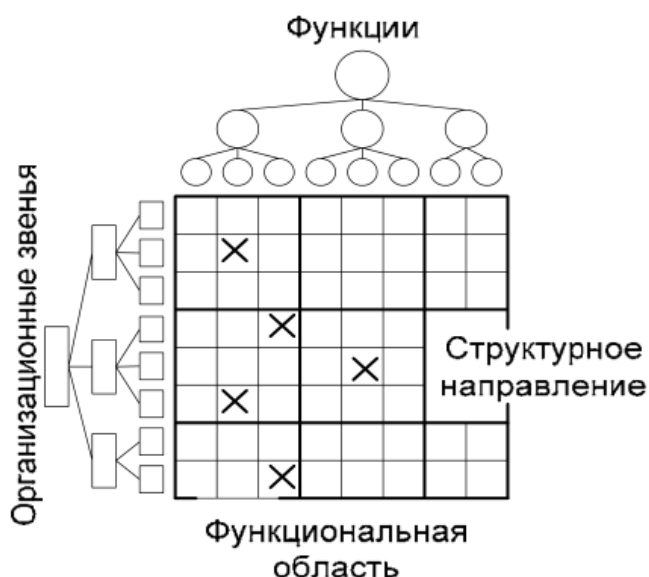


Рис. 5. Организационно-функциональная модель предприятия

Построение организационно-функциональной модели предприятия обеспечивает прозрачность и предсказуемость бизнеса, что повышает эффективность управления предприятием. Помимо корпоративной организационно-функциональной модели предприятия могут быть построены частные модели для отдельных структурных направлений (подразделений) или функциональных областей,

например, для отдела маркетинга или рекламного бюро, входящего в отдел маркетинга, и т.д. Значительно упрощает задачу использование для построения названных моделей специальных программных сред, описание которых приведено в гл. 3 пособия.

1.2. Предприятие как система

Наряду с классическим подходом, основанным на «механистических» представлениях об организации, подавляющее большинство исследователей выделяют отдельное направление в теории организации — системный подход, в основе которого лежит понятие «система». В связи с этим, в процессе бизнес-моделирования так же должен использоваться системный подход к описанию целей, структуры, механизмов и регламентов деятельности организации и их связей между собой, направленных на достижение стратегических целей.

Рассмотрим понятие системы, определяемое разными исследователями. Согласно стандарта ISO/IEC 15288:2002, система — это «совокупность взаимодействующих элементов, упорядоченная для достижения одной или нескольких поставленных целей». Это определение распространяется на самые разные по характеру и масштабу системы — от локального устройства (например, мобильного телефона) до целой отрасли (включая ее работников).

Рассел Л. Аккофф, американский ученый в областях исследования операций, теории систем и менеджмента определяет систему как множество, состоящее из двух или более элементов, которое удовлетворяет следующим трем условиям:

- поведение каждого элемента воздействует на поведение целого (например, организм человека: каждая его составляющая — сердце, легкие, желудок и т.д. воздействует на функционирование организма в целом);
- поведение элементов и их воздействия на целое взаимны (это условие означает, что поведение каждого элемента и его воздействие на целое зависит от того, как ведет себя по крайней мере еще один другой элемент);
- какие бы подгруппы элементов ни образовывались, каждый элемент воздействует на поведение целого, но ни один из них не имеет самостоятельного воздействия на систему в целом (другими словами, элементы системы соединены таким образом, что образование независимых подгрупп невозможно).

Например, в торговой компании деятельность отдела маркетинга оказывает существенное влияние на деятельность отдела продаж. В свою очередь отдел продаж диктует свои требования к отделу закупок, те все три подсистемы: маркетинг, продажи, закупки не могут функционировать отдельно, и функционирование каждой из них воздействует на функционирование всей организации. Если по каким-либо причинам будет упразднен отдел маркетинга, то это будет уже другая организация. Любой отдел организации имеет свои цели и задачи, которые, естественно согласованы с целями и задачами предприятия. Если мы отделим отдел от организации, то его цели и задачи изменятся и это будет уже другая система.

Эберхард Речтин (основатель понятия «системное мышление») отмечает: «система, когда целое составляет нечто большее, чем механическая сумма составляющих ее элементов, т.е. система обладает свойствами, которые отсутствуют у составляющих ее элементов». Каждая часть системы обладает свойствами, которые она теряет в случае отделения от системы. Каждая система обладает определенными (существенными) свойствами, которыми не обладает ни одна из ее частей. Существенные свойства системы в целом вытекают из взаимодействия ее частей, а не от их действий самих по себе. Если систему разбить на части, она утратит свои существенные свойства. Следовательно, поведение системы невозможно изучить только с помощью анализа, нужен еще и синтез. Анализ фокусируется на структуре системы, он дает описание системы с точки зрения ее частей и знание о ней через понимание функционирования ее элементов. Синтез фокусируется на функционировании системы в целом, он дает объяснение поведения всей системы, т.е. через синтез мы понимаем суть системы, получаем представление о деятельности системы, и приходим к пониманию процессов, которые протекают внутри системы.

Отличие системного подхода к изучению системы от механистического состоит в следующем. При механистическом подходе анализ предшествует синтезу и содержит три этапа:

- разложение на части того, что необходимо объяснить;
- объяснение поведения или свойств отдельных частей;
- составление из полученных объяснений целостной трактовки.

Первые два этапа — анализ системы, третий — синтез системы. Системный подход переворачивает трехэтапный порядок механистического подхода:

- идентификация целого (системы), частью которого является предмет, который необходимо объяснить;
- объяснение поведения или свойств целого;
- объяснение поведения или свойств предмета по его роли (ролям) или функции (функциям) в содержащем его целом.

Здесь первые два этапа — синтез системы, третий — анализ системы.

Механистический подход к изучению бизнес-системы не является эффективным, т.к. современные бизнес-системы достаточно быстро изменяются, распадаются, подсистемы собираются в новые системы и т.п. Системный подход подразумевает: сначала идентификацию целого (синтез системы), т.е. выделение параметров системы, по которым мы ее отличаем от других систем (например, по сфере деятельности организационной системы), потом с помощью анализа изучаем из каких частей (подразделений) она состоит, какие показатели деятельности она имеет.

Русским ученым, ставшим основоположником системного подхода в теории организации является А.А. Богданов (Малиновский), автор книги «Тектология» о всеобщей организационной науке и принципах равновесия. Суть его взглядов на организацию как систему выражается в следующем. Во-первых, Богданов определяет организационную систему (или комплекс) как «процесс или поток процессов производства составляющих, связанных циклами развития и деградации». Во-вторых, он проводит четкое различие между организацией и

структурой: «организация — сеть процессов производства ее составляющих, структура — особый пространственно-временной образ (паттерн) произведенных составляющих». Далее отмечается, что организационная система должна рассматриваться не как конечное состояние, нечто застывшее, а как «процесс постоянных преобразований, связанных с непрерывной сменой состояний равновесия». Здесь следует пояснить, что любая система обладает некой устойчивостью (имеет баланс), т.е. если система существует, то в ней есть некий баланс. Но состояние равновесия постоянно качественно изменяется. И если мы управляем этим процессом, то мы управляем системой, независимо от того, на каком качественном этапе своего развития она находится. Также автор констатирует, что сохранность организационной системы обеспечивается только активным использованием внешней среды.

Уточним понятие организационной системы: это определенным образом *организованная* социально-экономическая система, назначением которой является согласование действий целеустремленных частей (социальных групп и личностей, входящих в систему), а также нецелеустремленных средств и предметов деятельности (ресурсов), ориентированных на достижение определенных целей.

Основными элементами организационной системы (а значит и объектами организационного управления) выступают: производство, маркетинг и сбыт, финансы, информация, персонал, человеческие ресурсы. Человеческие ресурсы обладают системообразующим качеством, от них зависит эффективность использования всех остальных ресурсов. Эти элементы являются основными объектами организационного управления. Но у организационной системы есть и другие стороны:

- *Люди*. Задача менеджера в том, чтобы способствовать координации и интеграции человеческой деятельности.

- *Цели и задачи*. Организационная цель — есть идеальный проект будущего состояния организации. Эта цель способствует объединению усилий людей и их ресурсов. Цели формируются на основе общих интересов, поэтому организация- инструмент для достижения целей.

- *Организационная структура* — есть способ объединения различных частей организации в определенную целостность.

- *Специализация и разделение труда*. Это тоже объект управления. Дробление сложных производственных процессов, операций и задач на составляющие, которые предполагают специализацию человеческого труда.

- *Организационная власть* — это право, способность (знания + навыки) и готовность (воля) руководителя проводить свою линию при подготовке, принятии и реализации управленческих решений. Каждый из этих компонентов необходим для реализации власти. Власть — есть взаимодействие. Функцию координации и интеграции деятельности людей бессильный и неэффективный менеджер организовать не может. Организационная власть — не только субъект, но и объект управления.

- *Организационная культура* — присущая организации система традиций, верований, ценностей, символов, ритуалов, мифов, норм общения между людьми. Организационная культура придает организации индивидуальность,

собственное лицо. Что немаловажно, она объединяет людей, создает организационную целостность.

- *Организационные границы* — это материальные и нематериальные ограничители, которые фиксируют обособленность данной организации от других объектов, находящихся во внешней среде организации. Менеджер должен обладать способностью расширять (в меру) границы собственной организации.

Организационные системы можно разделить на закрытые и открытые. Закрытая организационная система — это та система, которая не имеет связи со своей внешней средой (т.е. ее действия не зависят от внешней среды). Пример — натуральное хозяйство, которое не обменивается с внешней средой продуктами, услугами, товарами и др. Открытая организационная система имеет связи с внешней средой, т.е. другими организациями.

Организация как система представляет собой совокупность взаимосвязанных элементов, образующих целостность (т.е. внутреннее единство, неразрывность, взаимную связь). Любая организация является открытой системой, т.к. взаимодействует с внешней средой. Она получает из окружающей среды ресурсы в виде капитала, сырья, энергии, информации, людей, оборудования т.п., которые становятся элементами ее внутренней среды и являются входами. Часть ресурсов с помощью определенных технологий перерабатывается, преобразуется в продукты и услуги, которые затем передаются во внешнюю среду и являются выходами системы. Другими словами, организационная система в динамике представляется в виде трех процессов: вход — преобразование — выход (рис. 6). Слева — «входы» в систему, т.е. ресурсы, которая она преобразует в «выходы». Сверху — правила, определяющие существование и функционирование системы. Снизу — ресурсы, с помощью которых входы преобразуются в выход, справа — выходы системы. Таким образом, мы не должны рассматривать организацию в отрыве от внешней среды, которая задает правила существования этой системы, влияет на ее целеполагание, поставляет в организацию ресурсы. Также внешняя среда является потребителем результатов существования системы.

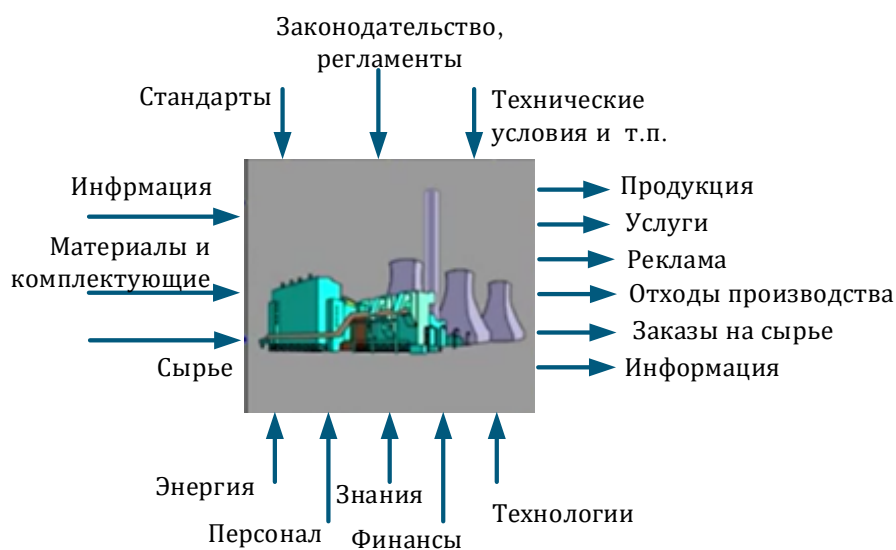


Рис. 6. Модель организационной системы

В практике менеджмента во внешней среде организации разделяют на среду прямого и среду косвенного воздействия. Среда прямого воздействия включает факторы, которые непосредственно влияют на операции организации и испытывают на себе прямое влияние операций организации. Среда косвенного воздействия — это факторы, не оказывающие прямого воздействия на операции организации, но все же сказывающиеся на них опосредственно. Среда косвенного воздействия обычно сложнее, чем среда прямого воздействия.

На рис. 7 представлена схема взаимодействия организационной системы с внешней средой прямого воздействия.

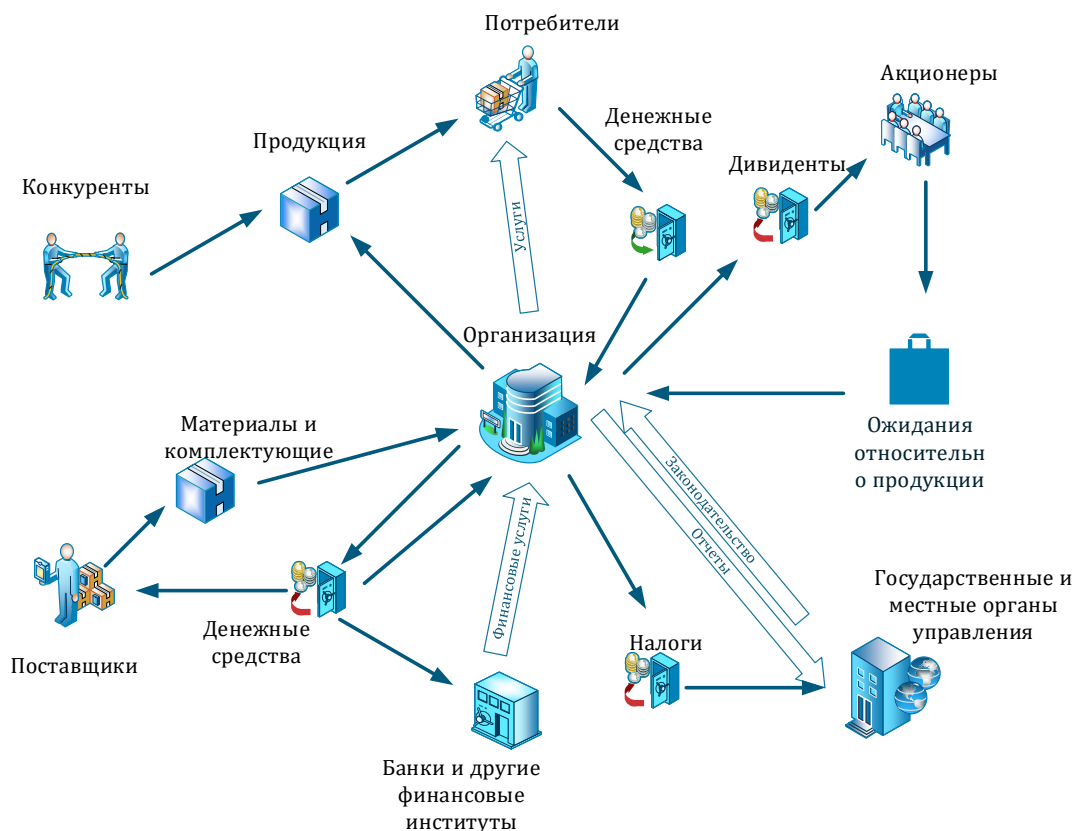


Рис. 7. Связи организационной системы с внешней средой

Применяя системный подход к изучению объекта, нельзя не сказать о свойствах организации как системы. Свойства организационной системы рассматриваются как состояния, характеризующие организацию.

У организации всегда есть некие цели. Наличие целей определяет не только существование организации, но и ее поведение, направленное на достижение целей. В менеджменте важное значение имеют измеримые показатели деятельности, по которым определяется достижимость целей.

Сложность системы определяется тем сколько и каких элементов входит в систему: люди, документы, система управления, информационная система и т.п.



Рис. 8. Свойства системы, рассматриваемые как некое «состояние»

Свойство делимости используется для преодоления сложностей в изучении системы. Определяют признаки, по которым можно выделить подсистемы. Для организационных систем признак чаще всего определяется продуктом или услугой, которые появляются в качестве результатов функционирования подсистемы и результатов процессов, протекающих в подсистеме.

Важный момент — это целостность изучения системы. Нельзя изучать систему, описывая ее отдельные части. Всегда нужно подразумевать совокупный результат действий.

Многообразие не столько по количеству, сколько по природе элементов, их характеристикам. Например, документы — формальная природа элементов; люди — навыки, компетенции, личные цели; оборудование — сложная природа функционирования и управления: закупки, учет, амортизация, обновление, утилизация. Нужно уметь адекватно использовать систему и управлять ей не нарушая природу элементов и их многообразие.

Структурированность — это некая степень взаимосвязанности элементов системы, их реальное взаимодействие. Показывает степень взаимозависимости и соответствие элементов системы. Это очень важный пункт, т.к. качество любой организации зависит от слаженности действий, адекватности одних частей другим частям. Если элементы не будут адекватно соответствовать системе, определенному уровню ее развития, то какими бы хорошими они не были, для системы это будет не нужно. То есть они должны соответствовать определенным целям системы, определенным этапам. Например, специалисты, обладающие обширными компетенциями, хорошими знаниями и опытом, могут быть не востребованы системой. И наоборот, если системы выставляет слишком высокие требования без учета своего текущего состояния, текущих целей, это тоже приводит к

дисбалансу. Например, перестраховка при поиске сотрудников на работу. Система документов — часто тоже бывает излишней, особенно в госучреждениях.

1.3. Архитектурный подход как основа управления развитием информационных систем

Современная экономика немыслима без эффективного управления. Успех управления во многом определяется эффективностью принятия интегрированных решений, которые учитывают самые разносторонние факторы и тенденции динамики их развития. Отправной точкой в формировании архитектурного подхода к управлению стало развитие понятия «архитектура информационных систем». В компьютерном деле понятие «архитектура» начали использовать в середине XX века в применении к чисто аппаратным средствам, в 60-е годы — к программно-аппаратным компонентам и комплексам, затем, к концу 70-х годов в целях построения эффективной системы управления человеко-машинным комплексом стали выстраивать архитектуры таких комплексов. При этом применялся системный подход и отмечались следующие характеристики человеко-машинных комплексов:

- имеет цели и поведение, но может их менять, изменяя при этом внешнюю среду, другие системы и/или саму себя;
- обладает социальным устройством своей внутренней среды;
- может иметь весьма приблизительную границу между внутренней и внешней средой.

Архитектура предприятия возникла как концепция комплексного подхода к человеко-машинным системам, где важны свойства людей, машин, их взаимосвязи, цели систем и их поведение. Причины введения в оборот современных толкований предприятия и его архитектуры лежат в тех изменениях в жизни предприятий, которые начали происходить еще в 70-х и 80-х годах. Среди главных изменений — начало перехода от рынка продавца к рынку потребителя, TQM (Total Quality Management) и конкуренция японских компаний как в мире, так и на внутреннем рынке США. Это побудило Эдвардса Деминга преобразовать методы повышения качества в подход CPI (Continuous Process Improvement — непрерывное совершенствование процесса), который проник в практику предприятий США. В единый комплекс были сведены цели и задачи бизнеса в конкурентной среде, особая ценность и роль людей, свойства используемых машин и технологий, аналитический подход к поиску причин имеющих место потерь и способов повышения конкурентоспособности, к опоре на измеримые показатели деятельности. Такое целостное представление было зафиксировано в четырнадцати принципах Деминга, которые в явном виде задавали как философию управления и культуру работников предприятия, так и постоянные изменения его процессов, в первую очередь за счет совершенствования базовых технологий и процессов.

В остро конкурентной среде организация бизнес-деятельности оформилась в виде комплексной дисциплины процессного управления предприятием. Учи-

тывалась необходимость преобразования всех компонентов предприятия: целеполагания, организационных структур, ценностных установок отдельных работников, базовых технологий и т. п. Параллельно с этим процессный подход к совершенствованию предприятия стал выделяться в более явной форме и в конце 80-х годов вместе с идеями СРІ проник в стандарты СММ (Capability Maturity Model).

С другой стороны, в 80-х годах много говорилось о явно недостаточной отдаче, получаемой бизнесом от ИТ. Поэтому возросла значимость методов согласования ИТ-систем с реальными потребностями бизнеса — то, что и сегодня часто выделяется в отдельное направление «business & IT alignment (выравнивание)». Так, в IBM длительное время развивалась методика BSP (Business Systems Planning), направленная на то, чтобы архитектура ИС выводилась из потребностей бизнеса и их приоритетов.

Появление концепции архитектуры предприятия связано с тем, что многие организации испытывают постоянные трудности и находятся в перманентном поиске синхронизации целей и задач бизнеса с процессами развития своих информационных систем. Существует как бы «облако неопределенности» между организацией и обеспечивающей ее деятельность ИТ-инфраструктурой. [5]. Раскрыть это облако неопределенности и позволяет архитектура предприятия в целом и ее подмножество — архитектура информационных технологий, которые являются основными механизмами интерпретации и реализации целей организации через адекватные ИТ-инфраструктуру и ИТ-системы. Это достигается за счет создания определенного количества взаимосвязанных архитектурных представлений.

Значение архитектуры организации в современных условиях постоянно увеличивается за счет обеспечения возможностей эффективного использования существующих технологий и эволюционного перехода к новейшим технологиям. Фактически, архитектура организации является одним из главных средств управления изменениями, обеспечивая при этом:

- оказание помощи менеджерам при анализе потенциальных изменений и их реализации;
- предоставление основы для совместной работы бизнес-менеджеров и ИТ-менеджеров над целями, бизнес-процессами и выстраиванием организации в целом;
- предоставление единого хранилища всей информации об организации;
- обеспечение менеджерам поддержки в принятии решений: они могут обобщать отношения, задавать вопросы, идентифицировать проблемы, выполнять моделирование и т. д.

Таким образом, концепция корпоративной архитектуры напоминает градостроительство в области ИТ — составление общего плана интеграции различных объектов в рамках всей организации, определение порядка их использования и путей построения необходимых для этого механизмов. Ее суть заключается в том, чтобы разработать план использования ИТ-ресурсов бизнес-процессами, а также совокупность принципов управления, позволяющих выразить стратегию бизнеса через ИТ.

Непосредственно архитектура организации не описывает конкретные технические решения отдельных информационных систем, но позволяет получить существенную выгоду для бизнеса организации в целом. Основные аспекты связаны с повышением эффективности эксплуатации информационных систем, снижением рисков инвестиций в ИТ, а также с повышением гибкости или возможности относительно простой адаптации под изменяющиеся внешние условия и требования бизнеса.

Наличие в организации разработанной архитектуры обеспечивает:

- поддержку принятия решений и управление в условиях сложных бизнес-процессов и информационных технологий;
- план развития и изменений;
- основу для назначения приоритетов при формировании ИТ-бюджетов;
- основу для управления портфелем ИТ- проектов;
- соответствие принятым корпоративным стандартам;
- поддержку разработки новых систем.

Архитектура в конечном итоге обеспечивает более эффективное использование ИТ-систем за счет:

- снижения стоимости разработки, внедрения и поддержки, в том числе и уменьшения излишних и необоснованных расходов на ИТ (предприятия лучше понимают, какими ИТ-активами они владеют, что уменьшает риск принятия решений о покупке или разработке систем, имеющих аналогичную функциональность, а также для их консолидации и уменьшения общего количества);
- упрощения процессов управления системами;
- повторного и многократного использования технологий;
- оптимизации функциональности и процессов внедрения новых ИТ-систем, а также проведение доработок используемых ИТ-систем;
- оценки внедрения по времени и результатам;
- обеспечения взаимопонимания между всеми участниками ИТ-деятельности предприятия.

Архитектура является средством снижения рисков и увеличения отдачи от инвестиций в ИТ. Причина в том, что она четко определяет структуру как существующих, так и будущих ИТ-систем, что приводит к снижению их сложности. А наличие ясной стратегии будущих закупок, выбора поставщиков технологий и планируемых изменений позволяет упростить и ускорить все процессы, связанные с закупками, при одновременном обеспечении совместимости и взаимодействия компонент ИТ-систем организации.

Наконец, необходимая гибкость развития бизнеса и структурных изменений обеспечивается за счет простоты доступа к интегрированным информационным ресурсам в масштабе организации. Ускорение выхода новых продуктов на рынок может осуществляться за счет быстрого внедрения новых или модификации существующих приложений. Существенный выигрыш может быть получен при проведении слияний и поглощений, связанных с реинжинирингом процессов или объединением ИТ-систем и служб.

Таким образом, имеется три причины использования архитектурного подхода:

- рост масштаба и сложности ИТ, рост их стоимости и рисков в проектах их создания и внедрения;
- включение ИТ в основную деятельность, рост требований к эффективности инвестиций в ИТ;
- переход к процессному подходу, интегрирующему деятельность подразделений, рост требований к эффективному взаимодействию ИТ-систем между собой.

В результате его использования обеспечивается:

1. Информационная поддержка работ по сопровождению и развитию ИТ-инфраструктуры, включает:

- выявление бизнес-процессов, требующих первоочередной автоматизации;
- выявление первоочередных направлений совершенствования каналов связи;
- анализ ИТ-систем и их взаимодействия, оценка степени покрытия бизнес-процессов и информационных потоков существующими системами;
- оптимизация обработки информации во взаимодействующих системах (избавление от дублирующих систем и данных, согласование справочников и классификаторов, используемых в различных системах и т.п.);
- выявление, согласование, формализация и документирование требований к перспективным ИТ-системам, контроль внедрения новых систем на предмет соответствия согласованным требованиям в части покрытия информационных потоков;
- анализ альтернативных вариантов совершенствования ИТ-инфраструктуры.

2. Информационная поддержка работ по совершенствованию бизнес-процессов организации, включая:

- выявление бизнес-процессов, требующих совершенствования;
- избавление от дублирующих действий в различных сценариях (ввод одних и тех же сведений в различные системы);
- анализ альтернативных вариантов совершенствования бизнес-процессов.

3. Информационная поддержка всех заинтересованных лиц, включая сотрудников организации, использующих ИТ-системы в силу своих должностных обязанностей, а также разработчиков и сопровождающих различных систем, используемых организацией (включая обеспечение всех заинтересованных лиц единым языком базовых представлений, наглядным и интуитивно понятным).

1.4. Понятие архитектуры и значение архитектуры предприятия в современных условиях

Современные подходы к формулировке понятия архитектуры являются попытками предоставить язык, понятный и полезный одновременно для бизнес-руководства и для специалистов в области ИТ. По мнению Gartner Group¹, подход

¹ Gartner Group — Исследовательская и консалтинговая компания, специализирующаяся на рынках информационных технологий.

к формулировке архитектуры ИТ должен основываться на анализе общекорпоративных процессов и переоценке своих бизнес-процессов и поддерживающих их приложений. Гуру в области бизнеса отмечают, что архитектура предприятия является одним из инструментов организационных изменений предприятия в целом с использованием ИТ и выделяют два основных подхода к организационным изменениям. Первый подход связан с реорганизацией, реинжинирингом бизнес-процессов, второй — с управлением знаниями. В рамках первого подхода возможность оптимизации бизнес-процессов обеспечивается включением в архитектуру предприятия представлений о бизнес-архитектуре. Архитектура предприятия частично затрагивает и процессы ИТ-управления в организации. В этом плане она дополняет достаточно эффективные методики организации и реорганизации процессов внутри ИТ-службы, такие как ITIL², COBIT³ и другие.

Как отмечает компания Giga Group⁴ в индустрии ИТ нет одного, единственно правильного стандарта на определение архитектуры ИТ и архитектуры предприятия, поэтому общие соглашения внутри организации важнее теоретической точности. Важна не столько академическая точность определения того, что такое архитектура, сколько реальный процесс использования архитектурных принципов. Введя понятие корпоративной архитектуры ИТ как общего видения, принципов и стандартов, которыми организации руководствуются при разработке и внедрении технологий, Giga Group констатирует, что это инвестиция в стандарты процессов, технологий и интерфейсов в целях улучшения возможностей организаций и уменьшения стоимости разработки и сопровождения информационных систем. Преимущества инвестиций в архитектуру распространяются на несколько проектов сразу, но не все эти проекты могут быть известны в момент разработки архитектуры.

Специалисты Gartner Group⁵, определяя понятие архитектура, считают, что это абстрактное описание системы, ее структуры, компонентов и их взаимосвязей, а также общий план или концепция, используемая для создания системы, такой как здание или информационная система. Они дают следующее определение архитектуры предприятия: «Семейство руководящих принципов, концепций, правил, шаблонов, интерфейсов и стандартов, требований, используемых при построении совокупности информационных технологий предприятия».

В глоссарии Фонда поддержки системного проектирования, стандартизации и управления проектами (ФОСТАС) также отмечается наличие в модели архитектуры предприятия описания структуры существующей системы как совокупности ее компонентов и их взаимосвязей, а архитектура ИТ-системы определяется как многоаспектное описание или план задуманной или развиваемой си-

² ITIL — IT Infrastructure Library.

³ COBIT — Control Objectives for Information and Related Technologies.

⁴ Giga Group — исследовательская и консалтинговая компания, специализирующаяся на рынках информационных технологий.

⁵ Gartner Group — исследовательская и консалтинговая компания, специализирующаяся на рынках информационных технологий.

стемы на уровне ее компонентов, детализированное в достаточной мере для руководства ее воплощением, а также принципы и руководящие материалы, определяющие руководство конструированием и развитием системы во времени.

Анализируя модели архитектуры предприятия, предлагаемые различными компаниями, которые будут описаны во второй главе, можно выделить, по крайней мере, три различных «измерения» в представлении архитектуры:

- иерархия архитектур различных организационных систем;
- соотношения между объективной реальностью и субъективным восприятием;
- соотношения между общесистемной архитектурой и частными архитектурами.

Кроме этого следует выделить уровни принятия архитектурных решений. Точно так же, как и в строительстве, существуют различные уровни архитектуры (план города, план застройки района, планы отдельных зданий), требуется дальнейшая детализация высокоуровневых определений и классификация архитектуры бизнеса и информационных технологий на различных уровнях. Мы можем говорить об архитектуре предприятия в целом, архитектуре уровня отдельных проектов или семейства продуктов, можем говорить об архитектуре отдельной прикладной системы. И в первом, и во втором, и в третьем случае — это все архитектуры. Вопрос заключается в декомпозиции сложных систем и в том, на каком уровне принимаются те или иные архитектурные решения.

Архитектура предприятия определяет общую структуру и функции бизнес-систем и ИТ-систем в рамках всей организации в целом (включая партнеров и другие организации, формирующие так называемое «расширенное предприятие») и обеспечивает общую рамочную модель (framework), стандарты и руководства для архитектуры уровня отдельных проектов. Общее видение, обеспечиваемое архитектурой предприятия, создает возможность единого проектирования систем, адекватных, с точки зрения обеспечения потребностей организации, и способных к взаимодействию и интеграции там, где это необходимо.

Архитектура уровня отдельных проектов определяет структуру и функции систем (бизнес и ИТ) на уровне проектов и программ (совокупностей проектов), но в контексте всей организации в целом, т.е. не в изолированном рассмотрении индивидуальных систем. Архитектура уровня отдельных проектов соответствует архитектуре предприятия, детализирует ее и существует в ее рамках.

Архитектура прикладных систем определяет структуру и функции приложений, которые разрабатываются с целью обеспечения требуемой функциональности. Некоторые элементы этой архитектуры могут быть определены на уровне архитектуры предприятия или архитектуры отдельных проектов (в форме стандартов, руководств и требований) в целях использования лучшей практики и ответственности принципам всей архитектуры в целом.

Каждая информационная система представляет собой сложный, комплексный объект, который динамически изменяется во времени. Существует необходимость обеспечения достаточного уровня определенности для оценки его такого объекта, планирования его развития, и сравнения различных систем. Для этого выделяются наиболее существенные характеристики объекта, которые и

образуют архитектуру системы, понимаемую как компонентный состав системы и связи между ними. В то же время конкретная реализация информационной системы будет, наряду с архитектурой, включать все многообразие экземпляров данных, физическое расположение компонент, фактическую реализацию процессов управления и т.п. Таким образом, можно выделить два понятия:

- собственно Архитектура информационной системы — объективная реальность, включающая существующие компоненты и их связи;
- описание Архитектуры — отражение объективной или планируемой реальности в какой-либо документированной форме, представляющее собой некоторую модель реальной системы, которая динамически изменяется, сохраняя соответствие оригиналу.

Отличительной характеристикой решений, принимаемых в отношении архитектуры, является то, что эти решения должны приниматься с учетом системной перспективы. Любое решение, которое может быть принято локально, не является архитектурным для системы в целом. Это позволяет делать различие между детальным проектированием и принятием решений по поводу практической реализации системы, с одной стороны, и архитектурными решениями — с другой. Первые решения имеют локальные влияния, а вторые — систематическое. Поэтому для проектных решений нужна соответствующая более широкая перспектива, позволяющая учесть системное влияние решений более высокого уровня, что обеспечивает должный уровень качества системы в целом. Кроме этого, каждому уровню соответствуют различные лица, принимающие архитектурные решения. Если предметом рассмотрения является прикладная программная система, то возможность принятия решений на уровне ее отдельных компонент или модулей должна быть предоставлена разработчикам этих подсистем. Архитектор прикладной системы должен рассматривать вопросы, которые важны для системы в целом. Если же предметом рассмотрения является архитектура проекта, то решения по поводу архитектуры отдельных прикладных систем должны принимать разработчики этих систем. На уровне архитектуры проекта должны рассматриваться только те вопросы, которые имеют систематическое значение или важны для проекта в целом.

Определяющей характеристикой, которая отличает архитектуру предприятия от других типов архитектур является соответствующий корпоративный масштаб и охват. Она пересекает и пронизывает все внутренние организационные границы: границы различных бизнес-подразделений и границы отдельных функций.

1.5. Эволюция представлений об архитектуре предприятия

Рассмотрим эволюцию представлений об архитектуре предприятия. Принято считать, что направление корпоративных архитектур появилось в 1987 году после публикации в IBM Systems Journal статьи Захмана «A framework for information systems architecture» («Инфраструктура для архитектуры информационных систем») [1]. В дальнейшем Захман переименовал инфраструктуру «информационных систем» в инфраструктуру «корпоративной архитектуры». Сегодня

эта инфраструктура известна как инфраструктура Захмана. В последующем Захман внес вклад в разработку информационной системы Министерства Обороны США и в 1994 году была представлена архитектура технического обеспечения для управления информацией (Technical Architecture Framework for Information Management, TAFIM) [4]. Архитектура TAFIM позиционировалась как новый стандарт корпоративной архитектуры для всех оборонных работ и прошла через ряд итераций, пока не была отменена в 1998-2000 годах.

В 1996 году развитие корпоративных архитектур ускорилось благодаря принятию Конгрессом США акта Клингера — Коэна, называемого также Законом о реформе управления информационными технологиями (Information Technology Management Reform Act, ITMRA). Этот закон предоставляет Административно-бюджетному управлению США (OMB) широкие права «анализа, отслеживания и оценки рисков и результатов всех крупных капиталовложений в информационные системы, совершаемых органами исполнительной власти». Акт Клингера — Коэна требует от всех органов исполнительной власти назначать директоров по ИТ, которые, в частности, несут ответственность за «разработку, сопровождение и упрощение внедрения интегрированной инфраструктуры, предназначенной для развития или сопровождения существующих ИТ-продуктов и приобретения новых ИТ-продуктов, необходимых для достижения стратегических целей соответствующего учреждения и управления информационными ресурсами». Таким образом, фактическим результатом принятия акта Клингера — Коэна стало то, что все работы, относящиеся к ИТ и проводимые правительственными учреждениями США или для правительственных учреждений США, должны (по крайней мере, теоретически) выполняться в рамках единой, общей корпоративной архитектуры. В соответствии с этим законом всем федеральным агентствам было предписано принять меры по повышению эффективности инвестиций в ИТ. Для надзора за выполнением закона был сформирован совет директоров по информационным технологиям, в который вошли директора по информационным технологиям из всех основных правительственных органов.

В апреле 1998 г. Совет директоров по информационным технологиям начал работу над первым крупным проектом — структурой архитектуры федеральной организации. Версия 1.1 данной структуры была выпущена в начале 1999 г., а к концу года была разработана универсальная инфраструктура корпоративной архитектуры для правительства США в целом. Эта инфраструктура получила название FEAF (Federal Enterprise Architectural Framework) [4]. Через некоторое время полномочия совета директоров по информационным технологиям по архитектуре федеральной организации были переданы Административно-бюджетному управлению. В 2002 г. Административно-бюджетное управление запустило проект по переработке методологии FEAF и переименовало ее в архитектуру федеральной организации (Federal Enterprise Architecture — FEA), который полностью завершился в 2006 г.

Параллельно с этим в 2003 г. Министерством Обороны США была разработана DoDAF (Department of Defence Architecture Framework), а консорциумом The Open Group — методология TOGAF (The Open Group Architecture Framework — структура архитектуры The Open Group). Хотя архитектура TAFIM

больше не используется, некоторые ее принципы нашли применение в инфраструктуре архитектуры Open Group (TOGAF). TOGAF — это инфраструктура «с открытым кодом», контролируемая консорциумом Open Group. В настоящее время используется версия 8.1 [1]. По всей видимости, TOGAF — это наиболее популярная в начале двухтысячных инфраструктура корпоративной архитектуры из используемых в частном секторе. Несколько реже используется модель Захмана.

В 2002–2003 гг., примерно в то же время, когда Административно-бюджетное управление США стало доминировать в области разработки архитектуры предприятия в государственном секторе экономики, в частном секторе экономики стала доминировать другая организация — группа Gartner. Методология Gartner не является ни таксономией⁶ (как модель Захмана), ни процессом (как TOGAF), ни полной методологией (как FEA). Эта методология представляет собой набор практических рекомендаций по построению архитектуры предприятия от одной из наиболее известных в мире исследовательских и консалтинговых ИТ-организаций — компании Gartner.

Компания Gartner считает, что архитектура предприятия должна начинаться с того, что организация собирается достичь, а не с текущего положения дел, т.е. базироваться на системно-целевом подходе. После того как в организации сформировано единое представление о будущем, определяется влияние этого представления на архитектуру бизнеса, технологическую архитектуру, информационную архитектуру и архитектуру решений.

На основе этого подхода в 2006 г. сформулированы рекомендации, касающиеся разработки архитектуры в виде последовательности шагов и задач участников, которые, однако, не детализированы до уровня моделей процесса разработки архитектуры, а затем разработана Методика описания архитектуры Gartner, которая представляет собой трехмерную структуру: горизонтальные слои — бизнес-архитектура, вертикальные домены — информационная архитектура (приложения, данные, интеграция, доступ), вертикальные элементы — техническая архитектура (инфраструктура, системное управление, безопасность).

К 2005 г. компания Gartner стала одной из наиболее влиятельных организаций, занимающихся консалтингом на уровне директоров по информационным технологиям. Однако в области разработки архитектуры предприятия ведущее место заняла консалтинговая компания Meta Group. В 2005 г. компания Meta Group предложила Методику Meta Group, отличительной особенностью которой является более детальное и формализованное описание именно процесса разработки архитектуры и всех его составляющих. Meta Group отмечает, что архитектура реализуется на практике через процесс управления ИТ-программами и проектами.

Компания Gartner пыталась создать рекомендации по разработке архитектуры предприятия, однако ей не удалось превзойти Meta Group. В 2006 г. компания Gartner решила, что раз ей не удастся конкурировать с Meta Group, можно

⁶ Таксономия — это теория систематизации сложноорганизованных областей действительности и знания, имеющих иерархическое строение.

поступить иначе: компания Gartner приобрела компанию Meta Group. После приобретения Meta Group компания Gartner/Meta потратила год на то, чтобы разобраться, какой вклад внесла каждая из компаний в методологию разработки архитектуры предприятия и как наилучшим образом объединить свои заметно различающиеся подходы. В конце концов, был применен достаточно простой алгоритм: если компания Meta Group одобряла какой-либо пункт, он включался в методологию, если не одобряла, соответствующий пункт исключался. Так, компании Gartner нравились архитектурные структуры, компании Meta Group нравился архитектурный процесс. Поэтому с учетом принятого принципа структуры были исключены, а процессы включены.

Рассматривая современные зарубежные представления об архитектуре предприятия можно отметить следующее.

1. Согласно стандарту по формальному описанию архитектуры предприятия ISO 15704 [6], предложенному рабочей группой IFAC/IFIP (International Federation of Automatic Control / International Federation for Information Processing)⁷, архитектура организации должна включать роль людей, описание процессов (функции и поведение) и представление всех вспомогательных технологий на протяжении всего жизненного цикла организации

2. Как сформулировал The Chief Information Officers Council⁸ FEAF архитектура организации является стратегической информационной основой, определяющей следующие компоненты:

- структуру бизнеса;
- информацию, необходимую для ведения бизнеса;
- технологии, применяемые для поддержания бизнес-операций;
- процессы преобразования, развития и перехода, необходимые для реализации новых технологий в ответ на изменение/появление новых бизнес-потребностей.

3. Стандарт IEEE 1471 [9], разработанный Institute of Electrical and Electronics Engineers⁹, предоставляет метамодель для определения архитектуры предприятия — Conceptual Framework. Стандарт определяет такие абстрактные элементы архитектуры, как представления, системы, среды, обоснования, заинтересованные стороны и т. д. Согласно этой модели, называемой еще «рамочной» Система обладает некой Архитектурой, которая может быть определенным образом представлена в виде некоторого Описания. Описание включает различные Точки зрения в зависимости от Интересов, которые имеют Участники, рассматривающие архитектуру системы. Каждой Точке зрения на архитектуру системы соответствует определенное Представление, основу которого составляет некоторый набор Моделей, из которых состоит полное описание Архитектуры (рис. 9). Од-

⁷ Международная федерация автоматизации управления / Международная федерация по обработке информации.

⁸ Совет директоров по информационным технологиям (США).

⁹ Институт инженеров-электриков и электронщиков — международная организация, занимающаяся в частности изданием рекомендаций (международных стандартов). Член организаций ANSI и ISO.

нако стандарт IEEE 1471 не определяет структуру собственно архитектуры предприятия. Например, в стандарте указывается на необходимость иметь различные представления архитектуры, но при этом не указывается, какие именно.

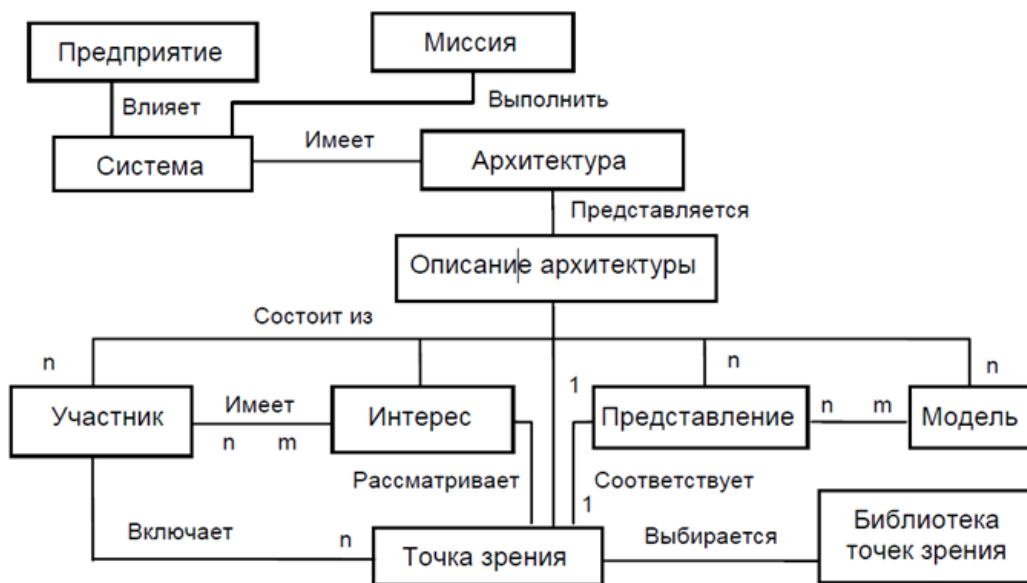


Рис. 9. Мета-модель для определения архитектуры предприятия стандарта IEEE 1471

Среди представлений российских исследователей в области архитектуры предприятия можно выделить высказывания В.И. Галактионова и Ю.Б. Гриценко, Р.Б. Васильева и Г.Н. Калянова, А.В. Данилина и А.И. Слюсаренко.

В.И. Галактионов и Ю.Б. Гриценко отмечают цикличность процесса развития архитектуры, как это показано на рис. 10 и предлагают рассматривать АП в двух аспектах: статическом — в некоторый фиксированный момент времени со стороны трех основных компонентов: миссии, бизнес-архитектуры и системной архитектуры и динамическом — как процесс перехода от текущего состояния к некоторому желаемому состоянию.

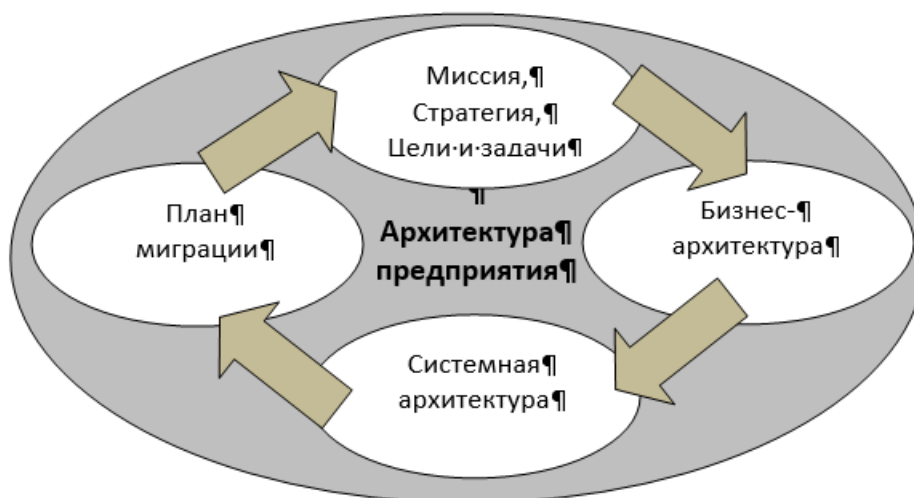


Рис. 10. Цикличность выстраивания архитектуры предприятия

Р.Б. Васильев и Г.Н. Калянов считают, что в общем виде под архитектурой организации понимается всестороннее и исчерпывающее описание (модель) всех ее ключевых элементов и межэлементных отношений.

А.В. Данилин и А.И. Слюсаренко отмечают:

- трехуровневую схему определения архитектуры;
- два аспекта: как объективная реальность, так и ее отражение;
- элементы архитектуры предприятия: цели и задачи, бизнес-модели, модели информации, модели прикладных систем, модели ИТ-сервисов, ИТ-услуги, ИТ-инфраструктура (рис. 11);
- уровни описания архитектуры: концептуальная, логическая, физическая.

Охарактеризуем названные уровни описания архитектуры.

Концептуальная архитектура определяет компоненты системы и их назначение, обычно, в неформальном виде. Это представление используется в обсуждении с нетехническими специалистами, такими как руководство, бизнес-менеджеры и конечные пользователи, вопросов о функциональных требованиях к системе.

Логическая архитектура выделяет, прежде всего, вопросы взаимодействия компонент системы, интерфейсы и используемые протоколы. Это представление упрощает процессы проектирования системы и позволяет эффективнее организовывать ее разработку.

Физическая реализация описывает привязку к конкретным узлам размещения, типам оборудования, характеристикам окружения и т.п.

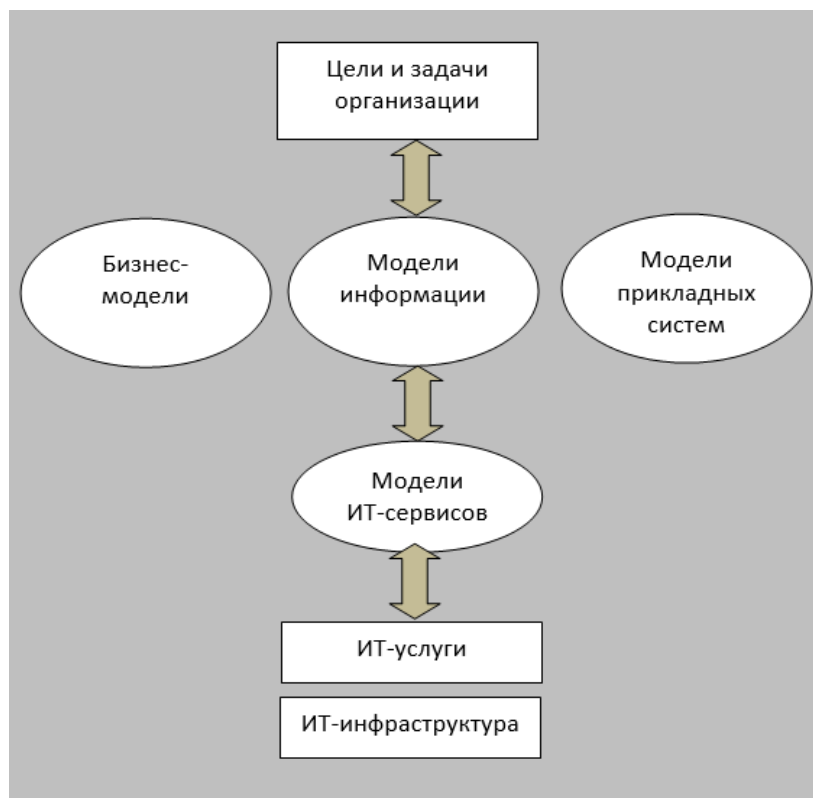


Рис. 11. Элементы архитектуры предприятия

2. Построение архитектуры предприятия

2.1. Процесс и принципы выстраивания архитектуры

К настоящему моменту в мировой практике накоплен значительный опыт в области построения архитектуры организации. Существующие подходы и методы во многом базируются на использовании, обобщении и интеграции имеющихся результатов в области бизнес-моделирования, системного анализа и проектирования (прежде всего, в части проектирования информационных систем). Цикл выстраивания архитектуры организации основными участниками процесса приведен на рис. 12.



Рис. 12. Процесс встраивания архитектуры предприятия

Следует отметить, что архитектурная концепция привела к появлению принципиально новой роли в составе основных участников процесса — роли архитектора. Названная роль характеризуется высоким статусом, отражающим степень важности наличия у организации архитектуры, архитектор, как правило, является главным заместителем ИТ-директора и является постановщиком задач как для аналитиков бизнес-процессов, так и для системных аналитиков.

Основными этапами процесса построения архитектуры организации являются следующие:

- осознание необходимости построения архитектуры;
- формирование рабочей группы;
- выбор среды моделирования, средств моделирования и репозитория;
- наполнение среды фактическим материалом (формирование архитектуры);
- использование;
- расширение и сопровождение.

При этом на этапе формирования архитектуры, как наиболее трудоемком, решаются следующие задачи, собственно, относящиеся к моделированию:

- определение бизнес-целей и требований;
- моделирование бизнеса с позиции менеджера;
- моделирование бизнес-процессов;
- моделирование бизнес-функций;
- моделирование оргструктуры, включая логические схемы принятия решений;
- моделирование ресурсов;
- преобразование бизнес-моделей в модели приложений и технологической архитектуры.

Отметим, что задача моделирования бизнеса с позиции менеджера, фактически, впервые идентифицируется в качестве самостоятельного элемента методологии. И хотя ряд инструментов «доархитектурного» периода обеспечивал построение так называемых «презентационных диаграмм», данный этап не декларировался практически ни одной из известных методологий бизнес-моделирования.

2.2. Архитектурные слои и их наполнение

В целом архитектура предприятия, рассматриваемая в статике, традиционно представляется в виде трех базовых слоев (рис. 13):

- миссия и стратегия, стратегические цели и задачи;
- бизнес-архитектура;
- системная архитектура.

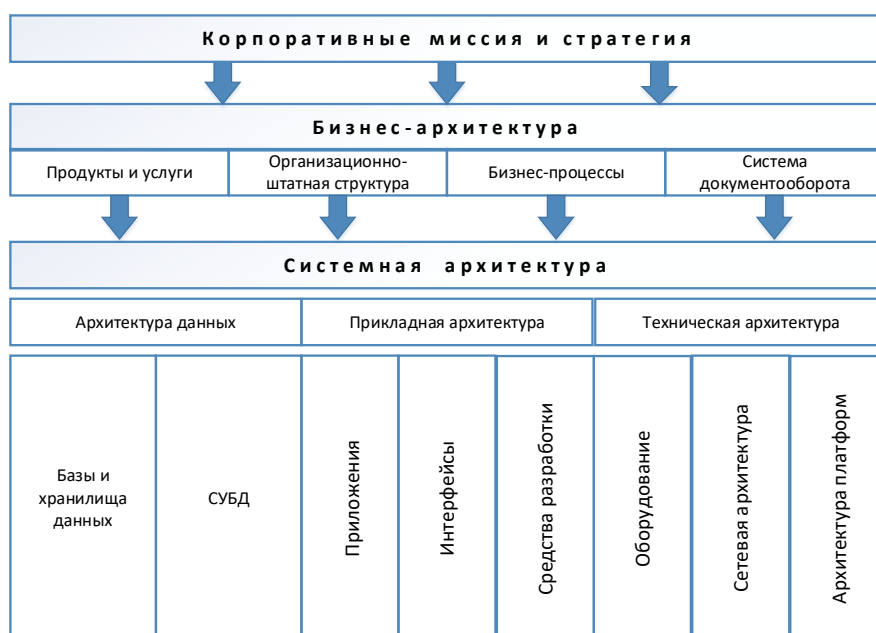


Рис. 13. Слои архитектуры предприятия

Верхний слой архитектуры предприятия описывает корпоративные миссию и стратегию, которые определяют основные направления развития организации и ставят долгосрочные цели и. *Миссия* — одно из основополагающих понятий стратегического управления. Разные исследователи дают различные формулировки миссии, но в целом можно выделить два подхода к пониманию мис-

сии. В широком смысле миссия — это философия и предназначение, смысл существования организации. *Философия* организации определяет ценности и принципы, в соответствии с которыми организация намеривается осуществлять свою деятельность. *Предназначение* определяет действия, которые организация намерена выполнять и цели, которые она намерена достичь. Философия организации определяется на этапе ее становления и редко меняется, тогда как предназначение может меняться в процессе изменений как внутренней, так и внешней среды организации. В узком понимании миссия — это глобальная цель предприятия и причины, оправдывающие ее существование. В таком понимании миссия должна раскрывать смысл существования организации, в котором проявляется отличие данной организации от подобных ей.

Назначение миссии заключается в том, что она способствует формированию имиджа организации во внешнем мире. Внутренняя задача миссии заключается в поддержке и развитии корпоративного духа, поскольку она проясняет сотрудникам общую цель существования организации, выражает принципы и ценностные ориентиры организации, облегчает сотрудникам осознание своего места и роли в системе деловых отношений, что в конечном итоге способствует созданию благоприятной атмосферы.

Конкретизация миссии осуществляется посредством формирования основных (стратегических) целей организации и последующей разработки корпоративной стратегии и корпоративной философии (культуры). При этом вырабатываемые на основе миссии цели, отвечающие на вопрос «Чего мы хотим достичь?», служат критериями всего последующего процесса принятия управленческих решений — если основные цели неизвестны, то у руководства нет точки отсчета для выбора наилучших по тем или иным параметрам решений.

Стратегия в широком толковании представляет собой искусство руководства общественными процессам, общий план руководства. Применительно к предприятию стратегия — это модель поведения, которой следует организация для достижения своих долгосрочных целей.

Под *корпоративной стратегией* понимается долгосрочное направление развития организации, следование которому приведет к достижению стратегических целей.

Стратегия формулирует общие направления развития организации, в первую очередь касающиеся производимой продукции и каналов ее продвижения. При этом стратегия должна обеспечить концентрацию усилий в той области, где будут иметься устойчивые конкурентные преимущества. Разработка корпоративной стратегии позволяет перейти от управления организацией, зависящего от воздействия случайно возникающих внешних и внутренних факторов, к планомерной деятельности по достижению определенных результатов с возможностью оценки их достижимости по определенным критериям и применения адекватных управляющих воздействий.

Корпоративная культура направлена на формирование общих для всех сотрудников организации целей, ценностей и принципов поведения, она должна способствовать развитию организации и достижению ею своих бизнес-целей.

Основным принципом формирования корпоративной культуры является ее соответствие всем элементам системы управления. На практике этот принцип означает, что при разработке или внедрении изменений в стратегии, структуре и в других элементах системы управления необходимо оценивать степень их реализуемости в рамках существующей культуры и, при необходимости, предпринимать шаги по ее изменению. При этом нужно учитывать, что культура по своей природе более инертна, чем остальные элементы системы управления. Поэтому действия по ее изменению должны опережать все остальные преобразования, необходимо понимать, что результаты будут видны не сразу.

Четкое определение миссии, стратегии и бизнес-целей предприятия позволяет не только сформулировать основные направления его развития и поставить долгосрочные цели и задачи, но и выработать требования к следующему слою архитектуры предприятия — Бизнес-архитектуре.

Бизнес-архитектура на основании миссии, стратегии развития и долгосрочных бизнес-целей определяет необходимые информационные и материальные потоки для производства и реализации продуктов и услуг; функциональную модель предприятия, описывающую направленные на реализацию текущих задач и перспективных целей бизнес-процессы, а также поддерживающую их организационно-штатную структуру. Также бизнес-архитектура формализует документооборот компании, в случае необходимости в нее включают описание каналов дистрибуции и логические структуры данных.

Бизнес-архитектура является областью деятельности высших руководителей, отвечающих за основные функции (бизнес) организации, и, как правило, содержит утверждения критических факторов успеха (КФУ), бизнес-правил, описания функций, а также структур и процессов, необходимых для их реализации.

Продуктовая модель содержит иерархический список продуктов и услуг компании и сопровождается схемой материальных потоков, отражающей движение материальных ресурсов, комплектующих и готовых изделий. *Организационная модель* включает в себя описание структуры управления компанией, организационно-штатную структуру, организационно-функциональную структуру, о которых рассказывалось в первой главе. *Система документооборота* описывается в виде схем документооборота; альбома форм документов и их реквизитов; лиц, ответственных за формирование, заполнение и утверждение документа. Наиболее объемной частью бизнес-архитектуры является процессная модель компании, которая описывает деятельность предприятия с точки зрения его ключевых бизнес-процессов. Существующая или планируемая бизнес-архитектура предприятия определяет требования к системной архитектуре.

Системная архитектура (ИТ-архитектура, архитектура ИС) определяет совокупность методологических, технологических и технических решений для обеспечения информационной поддержки работы предприятия в соответствии с правилами и концепциями, определенными бизнес-архитектурой.

Архитектура данных определяет, какие данные необходимы для поддержки бизнес-процессов [2] и включает в себя:

- базы данных, хранилища данных, витрины данных;
- системы управления базами данных или хранилищами данных;

- правила и средства санкционирования доступа к данным.

Архитектура приложений определяет, какие приложения используются и должны использоваться для управления данными и поддержки бизнес-процессов, включает в себя:

- собственно прикладные системы, поддерживающие исполнение бизнес-процессов (приложения);
- интерфейсы взаимодействия прикладных систем между собой и с внешними системами и источниками или потребителями данных;
- средства и методы разработки и сопровождения приложений.

Технологическая архитектура (ИТ-инфраструктура) определяет, какие обеспечивающие технологии (аппаратура и системное ПО, сети и коммуникации) необходимы для создания среды работы приложений, включает в себя оборудование, сетевую архитектуру и архитектуру платформ.

Сетевая архитектура в свою очередь включает в себя:

- локальные и территориальные вычислительные сети;
- используемые в сетях коммуникационные протоколы, сервисы и системы адресации;
- аварийные планы по обеспечению бесперебойной работы сетей в условиях чрезвычайных обстоятельств.

Архитектура платформ включает в себя:

- аппаратные средства вычислительной техники - серверы, рабочие станции, накопители и другое компьютерное оборудование;
- операционные и управляющие системы, утилиты и офисные программные системы;
- аварийные планы по обеспечению бесперебойной работы аппаратуры (главным образом серверов) и баз данных в условиях чрезвычайных обстоятельств.

Описанные архитектурные слои характеризуют архитектуру в статическом аспекте. Статистический аспект архитектуры предприятия описывает ее в текущем состоянии (модель AS-IS — как есть) в слоях бизнес-архитектуры и системной архитектуры. В динамике архитектура предприятия представляет собой план действий и портфель проектов, реализация которых приведет к преобразованию сложившейся архитектуры предприятия к новому, целевому состоянию. Целевая архитектура базируется на стратегических целях компании, текущих и планируемых бизнес-задачах и бизнес-процессах. Таким образом, динамический аспект архитектуры предприятия в общем случае подразумевает наличие модели TO-BE — как должно быть. Кроме того, обязательно наличие планов мероприятий и проектов по переходу из текущего состояния в планируемое (планами миграции).

Иногда модели AS-IS и TO-BE различаются очень сильно, так что переход от начального к конечному состоянию становится неочевидным. В этом случае необходима третья модель, описывающая процесс перехода от начального состояния системы к конечному, поскольку такой переход — это тоже бизнес-процесс.

Планы миграции определяют *сценарий* перехода предприятия от текущего состояния к будущему, и *преобразования* как бизнес-архитектуры, так и системной архитектуры. При поэтапной миграции для целей формализации промежу-

точных результатов разрабатываются один или несколько промежуточных (миграционных) элементов архитектуры. Планы миграции в соответствии с принятой на предприятии методологией управления проектами формализуются в виде отдельных проектов, включающих, в частности [4]:

- определение проекта как совокупности задач и работ;
- фазы и сроки реализации проекта в целом и составляющих проект задач и работ;
- анализ конкурентной среды и рисков, связанных с реализацией проекта;
- определение бюджета проекта и состава статей расходов;
- критерии успешности реализации проекта и ожидаемый экономический эффект.

Выполнение плана миграции не означает замораживания развития бизнес- и системной архитектуры. Таким образом, планируемая архитектура предприятия является архитектурой ТО-ВЕ только на определенном витке развития предприятия. Одновременно возврат к стратегическому уровню миссии и стратегических целей и задач не означает необходимость пересмотра миссии и стратегии, но в конце каждого цикла обязательно проводится анализ эффективности проведенных мероприятий и выполненных проектов. В случае необходимости на следующей итерации корректируются бизнес-архитектура, системная архитектура, реализуются новые планы миграции. Цикл развития архитектуры не обязательно затрагивает все предприятие в целом, он может касаться отдельных направлений деятельности, отдельных вопросов бизнеса и может быть зафиксирован в виде отдельного проекта. Технология проектирования информационных систем подразумевает сначала создание модели AS-IS, затем анализ и улучшение бизнес-процессов, т. е. создание модели ТО-ВЕ, и только на основе модели ТО-ВЕ строится модель данных, прототип и затем окончательный вариант модернизированной информационной системы.

Представление архитектуры предприятия в статическом аспекте как совокупности трех основных слоев определяет позиционирование следующих архитектурных сред.

- фронт-офис (Front-Office);
- мидл-офис (Middle-office);
- бэк-офис (Back-office);
- учет (Accounting);
- информационное хранилище (Data Warehouse);
- отчетность (Reporting).

Первые четыре присутствуют как в бизнес-архитектуре, так и в системной архитектуре. Два последних слоя относятся только к системной архитектуре.

Фронт-офис в бизнес-архитектуре — это совокупность бизнес-процессов, процедур, нормативных документов (регламентов), справочников, печатных форм, организационно-штатных подразделений, обеспечивающих со стороны предприятия взаимодействие с клиентом:

- получение и ввод для последующей обработки первичных документов; печать и предоставление клиенту информации и документов;
- рассылку клиентам информационных сообщений;

- «обзвон» клиентов;
- прием входящих телефонных звонков клиентов;
- прием запросов и предоставление информации.

Примеры подразделений фронт-офиса: Call-центр, подразделение операционного обслуживания, касса (отдельные бизнес-процессы).

Фронт-офис в системной архитектуре — это совокупность информационных систем, баз данных (БД) и справочников, направленных на автоматизацию бизнес-процессов взаимодействия с клиентом. Примеры информационных систем фронт-офиса: интернет-банк, информационная система Call-центра, система управления взаимоотношениями с клиентами (CRM — Customer Relationship Management System).

Мидл-офис в бизнес-архитектуре — это совокупность бизнес-процессов, процедур, нормативных документов (регламентов), справочников, печатных форм, организационно-штатных подразделений, обеспечивающих подготовку и принятие решений. Примеры подразделений мидл-офиса: подразделение проверки заемщиков в службе безопасности, подразделение управления рисками.

Мидл-офис в системной архитектуре — это совокупность информационных систем, баз данных и справочников, направленных на автоматизацию бизнес-процессов, связанных с подготовкой и принятием решений. Примеры информационных систем мидл-офиса: система ведения позиционного учета, система проверки заемщика в бюро кредитных историй, система расчета скорингового¹⁰ балла по кредитной заявке.

Бэк-офис в бизнес-архитектуре — это совокупность бизнес-процессов, процедур, нормативных документов (регламентов), справочников, печатных форм, организационно-штатных подразделений, реализующих журнальный (регистрационный) учет операций, совершенных клиентом. Как правило, регистрационный учет представляет собой журнал операций клиентов. Этот учет не связан с бухгалтерскими счетами, не является двухсторонним. Примером подразделения бэк-офиса является подразделение розничного кредитования.

Бэк-офис в системной архитектуре — это совокупность информационных систем, баз данных и справочников, реализующих журнальный (регистрационный) учет операций, совершенных клиентом. В современной системной архитектуре крупных корпоративных информационных систем данный класс систем представлен недостаточно широко. К данному классу относится большинство систем учета финансов для личного использования.

Учет в бизнес-архитектуре — это совокупность бизнес-процессов, процедур, нормативных документов (регламентов), справочников, печатных форм, организационно-штатных подразделений, бизнес-процессов, реализующих ведение бухгалтерского учета и отчетности по Российским правилам бухгалтерского учета (РПБУ) и Международным стандартам финансовой отчетности (МСФО),

¹⁰ Скоринговый балл — количественная оценка кредитоспособности потенциального заемщика (чем выше балл, тем выше кредитоспособность). В зависимости от балла определяется либо вероятность дефолта заемщика, либо принадлежность к определенному классу.

ведение баланса предприятия. На данном уровне часто реализован также налоговый и управленческий учет. Формирование проводок бухгалтерского учета происходит на основании журнала операций бэк-офиса путем разнесения операций в соответствии со справочником котировок.

Учет в системной архитектуре — это совокупность информационных систем, баз данных и справочников, реализующих ведение бухгалтерского учета и отчетности по РПБУ и МСФО, ведение баланса предприятия. Данный класс систем часто реализует также налоговый учет. Формирование проводок бухгалтерского учета происходит на основании журнала операций бэк-офиса путем разнесения операций в соответствии со справочником котировок.

Информационное хранилище в системной архитектуре представляет собой совокупность информационных систем, баз данных и справочников, реализующих функциональность по описанию метаданных, сбору, очистке, обогащению, консолидации первичной информации из транзакционных систем, а также построению витрин данных.

Отчетность в системной архитектуре — это совокупность информационных систем, баз данных и справочников, автоматизирующая построение отчетности на основе данных из информационного хранилища. Примеры систем отчетности: система управленческой отчетности, система аналитической отчетности, система ключевых показателей эффективности подразделений предприятия, система формирования показателей для расчета скорингового балла по кредитной заявке.

2.3. Бизнес-архитектура: процессная модель

Бизнес-архитектура как составной элемент архитектуры предприятия определяет необходимые организационную структуру и бизнес-процессы, описываемые функциональной моделью предприятия и используемые в процессе разработки и реализации продуктов. Бизнес-архитектура в расширенном варианте включает:

- предлагаемые и планируемые к реализации продукты и услуги (включая индивидуальные схемы их производства), формализованные в виде единого реестра продуктов и услуг;
- каналы продажи продуктов и услуг, построенные как на базе структурных и территориальных подразделений предприятия, так и на базе современных информационных технологий (ИТ);
- функции и процессы по реализации внешних и внутренних продуктов и услуг, образующие деревья функций и процессов (бизнес-функции и бизнес-процессы);
- финансовые и распорядительные документы (как в бумажном, так и в электронном виде), формализованные в виде единого реестра (альбома форм) документов предприятия;
- документопотоки, определяемые нормативными актами по внутреннему и внешнему документообороту;

- организационную структуру предприятия, включающую штатное расписание предприятий и его территориальных подразделений, являющихся самостоятельными хозяйствующими единицами (юридическими лицами), комитеты, рабочие группы и ролевые функции отдельных сотрудников, должностные инструкции, положения о подразделениях и рабочих органах и другие документы, регламентирующие взаимоотношения и распределение ответственности между сотрудниками, а также между структурными подразделениями.

Одним из основных элементов бизнес-архитектуры, помимо организационной структуры, являются направленные на реализацию текущих задач и перспективных целей бизнес-процессы, описываемые функциональной моделью. Таким образом, функциональная модель предприятия представляет собой набор регламентов бизнес-процессов. Построение функциональной (процессной) модели предприятия — одна из самых важных задач для любого руководителя. С помощью функциональной модели можно полноценно строить техпроцесс, распределять ответственность за функции среди персонала, управлять контрольными точками, устанавливать категории и очередность процессов, собирать функции для формирования должностных инструкций и положений о структурных подразделениях. В функциональной модели должно быть отражено как внутреннее взаимодействие бизнес-процессов, так и взаимодействие предприятия с внешней средой

Ключевым понятием в процессной модели предприятия является бизнес-процесс, т.е. регулярно повторяющаяся последовательность действий, направленных на получение заданного результата, ценного для организации. Приведем определения бизнес-процесса, предлагаемые разными авторами. Согласно стандарту ISO 9001:2008 Бизнес-процесс — это совокупность взаимосвязанных или взаимодействующих видов деятельности (работ), преобразующих входы в выходы. Внешне бизнес-процесс характеризуется своими входами и выходами, потребляемыми ресурсами (предметы труда и средства труда), участниками, владельцем, управляющим воздействием. Входы процесса — это преобразуемые ресурсы: сырье и материалы, комплектующие, информация, документы. В качестве выходов процесса выступают продукты и услуги, новая информация, документы. Используемые долговременные ресурсы — это здания и сооружения, оборудование, технологии, среда, персонал, методы, знания и навыки. Управляющее воздействие на выполнение процесса оказывают законодательство, стандарты, приказы и инструкции и другие нормативные документы.

М. Хаммер и Дж. Чампи определяют бизнес-процесс как «совокупность различных видов деятельности, в рамках которой «на входе» используется один или более видов ресурсов, и в результате этой деятельности на «выходе» создается продукт, представляющий ценность для потребителя» [8]. Российские исследователи в области процессного подхода к управлению Репин и Елиферов характеризуют бизнес-процесс как устойчивую целенаправленную совокупность взаимосвязанных видов деятельности, которая по определенной технологии преобразует входы в выходы, представляющие ценность для потребителя. Костров, Александров, рассматривающие процессную модель организации как основу для проектирования информационной системы предприятия определяют бизнес-

процесс как последовательность действий по преобразованию информации и принятию решений для производства и реализации продукции (услуг).

Бизнес-процесс — одна или несколько связанных работ или процедур, в совокупности реализующих некоторую цель производственной и непроизводственной деятельности в рамках определенной организационной структуры

Бизнес-процесс представляет собой систему последовательных, целенаправленных и регламентированных видов деятельности (операций), достигающих значимых для организации результатов (при этом входы процесса посредством управляющего воздействия преобразуются в выходы т.е. результаты процесса, представляющие ценность для потребителей).

Описание процессной модели начинается с выделения и классификации бизнес-процессов. Различают основные и вспомогательные, внешние и внутренние бизнес-процессы. Основной бизнес-процесс непосредственно относится к производству продукции. Вспомогательный бизнес-процесс обслуживает основной бизнес-процесс или снижает затраты ресурсов для его исполнения. Внешним называется процесс, имеющий вход и/или выход вне фирмы. Внутренним называется процесс, находящийся целиком в рамках одной организации. Существует также понятие главный бизнес-процесс, subprocess, сквозной процесс. В модели бизнес-процессов главный процесс декомпозируется на subprocessы с достаточной для анализа степенью детализации. Декомпозиция осуществляется до уровня отдельных процедур или работ. Сквозными называются процессы, которые пронизывают несколько подразделений организации.

Основные бизнес-процессы создают добавленную стоимость компанией конечному продукту компании, представляющему ценность для внешнего клиента и за который клиент готов платить деньги. Результатом основного процесса являются продукты (услуги) для конечного потребителя. Основные процессы располагаются по цепочке создания основного продукта для потребителя. Пример: закупка, производство, хранение, реализация, транспортировка, послепродажный сервис.

Вспомогательные процессы предназначены для обеспечения деятельности основных процессов, могут иметь потребителей только внутри организации (подготовка кадров, ИТ-ресурсы, бухгалтерия), или работать на всю организацию в целом (PR-деятельность, экология). Вспомогательные процессы характеризуются большим количеством взаимодействий с основными процессами и подразделяются на обеспечивающие бизнес-процессы, бизнес-процессы управления и бизнес-процессы развития.

Обеспечивающие бизнес-процессы — это процессы, которые создают и поддерживают инфраструктуру организации, их целью является обеспечение бесперебойного выполнения основных бизнес-процессов. Бизнес-процессы управления обеспечивают выживание и развитие организации, регулируют ее текущую деятельность, прямой целью которых является повышение эффективности деятельности организации. Бизнес-процессы развития — это процессы совершенствования и развития деятельности организации, прямой целью которых является получение прибыли в долгосрочной перспективе.

При выделении и классификации бизнес-процессов следует придерживаться ряд правил, обеспечивающих их управляемость:

1. Для управления каждым процессом необходимо назначить владельцев процессов, то есть должностных лиц или коллегиальные органы и предоставить в их распоряжение все необходимые ресурсы. В понятие ресурсы входят: персонал, оборудование, среда, финансовые ресурсы, связь, программное обеспечение и т.д. То есть владелец процесса должен иметь право распоряжения ресурсами, выделенными в его распоряжение.

2. Необходимо создать систему планирования и переноса стратегических планов организации в планы процесса. Размер процесса должен быть не меньше, чем величина объекта управления (подразделения), для которого составляется документированный план.

3. Необходимо создание полноценного комплекта документации. В комплект документации входят: регламент процесса, должностная инструкция владельца процесса и, как минимум, документация по выполнению технологии процесса исполнителями.

4. Количество процессов, находящихся в подчинении у одного владельца не должно превышать типовые нормы управляемости. У одного руководителя в подчинении может быть не более чем 7 ± 2 процесса. При этом, если процессы в организации построены по принципу вложенности (декомпозиции) сверху вниз, то для руководителей верхнего уровня количество процессов должно быть меньше 7, так сложность управления объектами растет с их размерами.

5. Для управления процессом необходимо создать систему управленческого учета, хотя бы в простейшем виде. Система должна учитывать результаты процесса (показатели продукта), эффективность процесса (показатели расхода ресурсов на единицу продукта или времени) и показатели удовлетворенности клиентов результатами процесса.

6. Контроль исполнения процесса выполняется с учетом технологической цепочки создания продукта. Для системы управленческого учета удобнее, когда непрерывная цепочка разбита на конечное число отрезков, каждый из которых завершается созданием законченного или промежуточного продукта (полуфабриката), для которого можно подсчитать затраты на его создание на этом отрезке. Размер процесса определяется разбиением сквозной цепочки создания продукта на промежуточные отрезки (процессы, подпроцессы, функции).

7. Максимальная ответственность при выборе владельца процесса. При выделении процессов, как объектов управления придется выбирать владельца процесса в ситуации, когда в создании продукта на выделенном отрезке принимают участие несколько подразделений с различными руководителями. Для решения этого вопроса необходимо оценить несколько факторов: а) кто отвечает за передачу полуфабриката или конечного продукта на следующий этап или клиенту? б) кто отвечает за наибольшую (наиболее значимую или ресурсоемкую) часть работ по созданию данного конечного или промежуточного продукта? Если ответы на эти вопросы различаются, то придется делать выбор в пользу одного из руководителей.

2.4. Системная архитектура

В системную архитектуру включены следующие основные элементы: архитектура данных, в некоторых источниках называемая архитектурой информации; прикладная архитектура, называемая еще архитектурой приложений и технологическая (инфраструктурная) архитектура. Рассмотрим более подробно вопросы представления системной архитектуры, а именно архитектуры данных и архитектуры приложений. Вопросы технологической архитектуры лежат за пределами этого пособия.

Архитектура данных. В настоящее время организации должны искать эффективные способы работы с информацией, которая поступает из самых разных источников и должна быть доступна там, где это нужно и тогда, когда это необходимо. Ситуация осложняется тем, что различные формы информации требуют специфических технологий и методов работы с ней. Различают следующие формы представления информации: структурированная (реляционные и объектные модели); неструктурированная информация (текст, графики, образы), сопровождаемая определенными описательными данными (метаданные, каталоги); полуструктурированная информация (представленная в XML-форматах и ему подобных).

Архитектура информации в целом описывает видение, принципы, модели и стандарты, которые обеспечивают процессы создания, использования и поддержания информации, относящейся к деятельности предприятия. Описание архитектуры информации служит созданию возможностей быстрого принятия решений и распространения информации как внутри организации, так и за ее пределами посредством использования информационных технологий. Можно сказать, что архитектура информации является «зеркальным отражением» бизнес-архитектуры [Данилин]. Если бизнес-архитектура отвечает на вопрос *«кто и что будет делать с учетом общего видения, целей и стратегий?»*, то архитектура информации — на вопрос *«какая информация должна быть предоставлена для осуществления этих процессов исполнителями?»*. Архитектура информации включает в себя:

- основные информационные объекты, связанные с бизнес-событиями;
- информационные потоки;
- модели, описывающие процессы обработки информации (information value chain);
- принципы управления информацией.

Архитектура должна описывать следующие виды данных:

- операционные данные, требуемые для выполнения процессов;
- аналитические данные;
- «контент», публикуемый в веб-приложениях.

Следует отметить, что разработка архитектуры информации состоит не в создании структур баз данных или моделей всех данных и что в контексте архитектуры предприятия более правильно говорить об архитектуре и моделях информации, а не данных, хотя эти понятия и пересекаются. Суть процесса разработки архитектуры информации заключается в организации общего описания

информации, необходимой для ведения бизнеса, а также формализации политики и правил работы с информацией. Модели архитектуры информации являются более абстрактными, они используют язык бизнеса и обеспечивают контекст, требуемый для моделирования данных. Модели же данных предполагают четкие описания структуры объектов, атрибутов, отношений между сущностями. Таким образом, понятие «архитектура информации» является расширением понятия «архитектура данных». В целом, архитектура информации позволяет управлять процессами организации и представления значимой для бизнес-пользователей информации в интуитивно понятной форме с использованием соответствующих средств каталогизации, навигации, пользовательского интерфейса. Этот аспект понятия архитектуры предприятия отражает место хранимой и обрабатываемой информации как стратегического корпоративного ресурса. Поэтому, описание архитектуры информации должно дополнительно включать средства для оценки качества и востребованности данных, учета стоимости данных как нематериального актива и т.п.

В ходе разработки архитектуры информации решаются следующие задачи [5]:

- идентификация и инвентаризация существующих данных, включая определение их источников, процедур изменения и использования, ответственности, оценку качества;
- сокращение избыточности и фрагментарности данных с целью уменьшения их стоимости за счет снижения затрат на устройства хранения и их обслуживание, а также повышения качества данных за счет исключения неоднозначности и противоречивости различных экземпляров;
- исключение ненужных перемещений или копирования данных, особенно связанных с наличием большого количества унаследованных или устаревших приложений;
- формирование интегрированных представлений данных, таких как витрины и хранилища;
- обеспечение доступности данных в режиме, приближенном к режиму реального времени, за счет использования средств обмена сообщениями, интеграционных брокеров и шлюзов;
- интеграция метаданных, позволяющая обеспечить целостное представление данных из различных источников;
- сокращение числа используемых технологий и продуктов, ведущее к снижению расходов на обслуживание и появлению возможности получения дополнительных скидок от поставщиков применяемых продуктов;
- улучшение качества данных, прежде всего, за счет привлечения пользователей к управлению данными и их определению;
- улучшение защиты данных на основе использования последовательных и согласованных мер, обеспечивающих, с одной стороны, защиту от несанкционированного доступа, а с другой — доступность данных для их использования на практике.

Результатом процессов разработки моделей информации и моделей данных является графическое представление информационных потоков и информации, удовлетворяющей информационные потребности как в целом организации,

так и потребностям отдельных бизнес-процессов. Разработка таких моделей является основой для реорганизации существующих бизнес-процессов и конструирования новых, которые будут использоваться для создания информационных систем. Анализ моделей информации проводится на различных уровнях абстракции: концептуальном, логическом и физическом.

На концептуальном уровне рассматриваются информационные потоки между функциональными подразделениями организации в обобщенном виде без описания практической реализации. Эти потоки не связаны с какой-либо автоматизированной системой и не описывают методы доступа или хранения информации. *На логическом уровне* описываются требования к информации в форме и терминах, понятных бизнес-пользователям. Этот уровень анализа позволяет идентифицировать общие элементы данных, которые используются разными подразделениями и бизнес-процессами, благодаря чему уменьшаются пересечения и сокращается возможность конфликтов между элементами данных. Назначение процесса моделирования на логическом уровне состоит в обеспечении средствами обнаружения, анализа, определения, стандартизации и нормализации отношений между бизнес-процессами и прикладными системами, идентификации потоков информации и соответствующих элементов данных, необходимых организации. Однако данный процесс не описывает способы хранения информации в базе данных. *На физическом уровне* задается описание способов реализации логики бизнес-процессов соответствующей автоматизированной системы, приводится необходимый набор информационных объектов и их элементов данных. Физическая модель данных предназначена для представления того, как данные, приведенные в логической модели, будут храниться в системе управления базами данных.

На концептуальном уровне абстракции архитектура информации должна описывать аспекты, связанные с получением, хранением, трансформацией, презентацией, анализом и обработкой информации. Это включает в себя следующие процессы управления информацией:

- получение данных из внутренних и внешних источников;
- классификация данных (по типам, формам представления и т.п.);
- хранение и извлечение данных;
- редактирование / обновление данных;
- контроль качества (удаление или исправление некорректных данных);
- презентация (трансформирование данных для разных категорий потребителей);
- распространение информации для различных групп потребителей;
- оценка информации (полезность, соотношение цена/качество, и др.);
- обеспечение безопасности информации (аутентификация данных от различных источников, назначение адекватного уровня доступа, обеспечение механизмов резервного хранения и восстановления).

Основные шаги создания архитектуры информации:

- создание словаря данных и депозитария метаданных;
- документирование и описание существующих источников данных;

- выбор системы записи информации о каждом элементе данных и построение модели данных;
- описание существующих и планируемых потоков информации;
- описание интерфейсов, алгоритмов преобразования или консолидации данных, составление соглашения о уровне сервиса, связанного с передачей и предоставлением данных;
- описание решений по организации хранения данных: от общих каталогов до витрин и хранилищ данных;
- выбор технологий и средств для преобразования и управления данными;
- определение механизмов доступа к данным;
- выбор программного обеспечения для доступа к данным;
- выбор механизмов для обеспечения безопасности данных.

Потребность в архитектуре информации сейчас очень велика. Например, META Group считает, что при разработке новых систем до 70% времени тратится на решение задач, связанных с идентификацией источников данных, которые должны использоваться прикладной системой, и на написание программного кода для доступа и трансформации данных. Для большинства средних и практически всех крупных предприятий характерно использование нескольких различных СУБД, средств предоставления и преобразования данных. Более того в последнее десятилетие наблюдается тенденция все более широкого использования готовых прикладных систем, каждая из которых имеет свои модели данных. В связи с этим растет количество источников данных, в потоках информации встречаются разветвления и слияния. Одни и те же данные могут обрабатываться разными прикладными системами и храниться в различных базах данных. Все это приводит к фрагментации данных, работе с ними различных подразделений и требует координации в рамках единой архитектуры информации предприятия.

Критическими факторами для обеспечения успеха процесса разработки архитектуры информации являются тщательное планирование и привязка к бизнес-целям предприятия. Рекомендуется проводить анализ данных последовательно для каждого бизнес-процесса, выбирая их в порядке приоритета по важности.

Архитектура приложений обеспечивает идентификацию прикладных систем, необходимых предприятию для выполнения бизнес-процессов и включает такие аспекты как проектирование, разработка (или приобретение) и интеграция прикладных систем. В архитектуре приложений, как правило, выделяют две основные области: портфель прикладных систем предприятия и область разработки прикладных систем (рис. 14).

Портфель прикладных систем представляет собой используемый на предприятии набор прикладных систем, обеспечивающих потребности бизнес-процессов предприятия. Он определяет область ответственности и приоритетность каждого приложения, а также способы достижения необходимой функциональности посредством либо разработки системы, либо покупки готовых приложений, аренды приложений или интеграции и использования возможностей уже имеющихся приложений. Портфель прикладных систем описывает приложения, предназначенные для выполнения функций организации и обмена информацией

между клиентами, поставщиками и партнерами предприятия, каналы возможного взаимодействия пользователей с приложениями: web-браузеры, графический интерфейс «толстого» клиента, мобильные устройства и т.д.

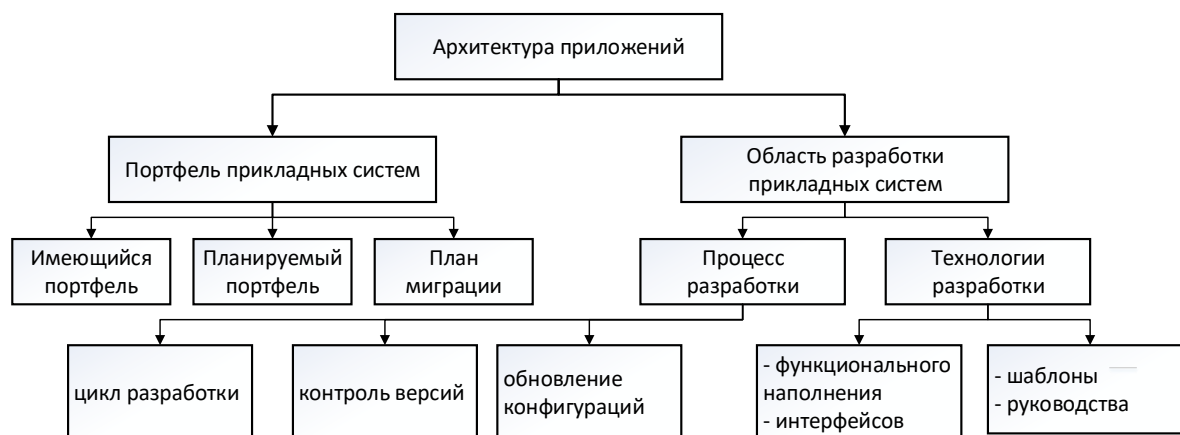


Рис. 14. Структура архитектуры приложений

Портфель прикладных систем формирует целостный взгляд на функциональные компоненты информационных систем, которые обеспечивают потребности бизнес-архитектуры и архитектуры информации и поддерживаются технологической архитектурой. Процессы управления портфелем прикладных систем тесно связаны с процессами управления ИТ-проектами и ИТ-активами в целом. Другими словами, портфель прикладных систем — это интегрированный набор информационных систем предприятия, который обеспечивает потребности бизнеса и включает в себя следующие компоненты [1]:

- имеющийся портфель прикладных систем. Это каталог имеющихся приложений и компонентов, который отражает их связи с поддерживаемыми ими бизнес-процессами, интерфейсы с другими системами, используемую и требуемую информацию, используемые инфраструктурные шаблоны. Чтобы быть реально полезным инструментом, он также должен помогать в идентификации тех элементов портфеля, которые можно использовать повторно и многократно в рамках предприятия, и стимулировать такое повторное использование;
- планируемый портфель прикладных систем, представляющий функциональность, которая требуется для обеспечения желаемого состояния бизнес-архитектуры и архитектуры информации предприятия.;
- план миграции. Процесс перехода от текущего к будущему портфелю прикладных систем в рамках ИТ-проектов. Проекты также могут объединяться в портфели проектов.

Область разработки прикладных систем описывает набор технологий, используемых для построения систем. Данная область выделяется, как правило, в организациях, в которых производится самостоятельная разработка приложений и включает в себя:

- технологии создания функциональных составляющих прикладных систем;
- технологии разработки интерфейсов;

- технологии настройки программных продуктов, а также используемые для этого шаблоны, руководства и т.д.

В данной области определяются процесс разработки и применяемые для него средства (программное обеспечение, средства проектирования). Процесс разработки кроме цикла разработки включает контроль версий и настройку конфигураций. Основной задачей области разработки автоматизированных систем является уменьшение стоимости создания таких систем и повышение их качества за счет обеспечения единых подходов к разработке, что приводит к уменьшению общего количества различных технологических сценариев, связанных с проектированием архитектуры, операционной поддержкой, архитектурной интеграцией систем, обучением персонала. В этой области требуется участие системных архитекторов.

Первым шагом в планировании портфеля прикладных систем является оценка текущего состояния портфеля и степени его соответствия потребностям организации со стратегической и технологической точек зрения. Соответствие бизнес-стратегиям оценивается на основе вклада прикладных систем в достижение бизнес-результатов, что определяется бизнес-архитектурой предприятия. Технологическое соответствие выявляется при сопоставлении прикладных систем, принципов и технологических стандартов, принятых в технологической архитектуре предприятия.

Существуют различные способы оценки портфеля прикладных систем предприятия. В настоящее время исследователями рекомендуются два метода: первый, предложенный компанией Gartner, «Пирамида преимуществ» (рис. 15), второй — «Матрица оценки состояния прикладных информационных систем (Health Grid)» (оценка по двум критериям — ценности с точки зрения бизнеса и техническому состоянию) [1]. Анализ портфеля основан на том факте, что различные прикладные системы играют существенно различные роли в организации, и при выполнении этих ролей возникают различные управленческие проблемы. Преимущества этих методов при принятии решений высшими руководителями организации, не являющимися ИТ-профессионалами, заключаются в простоте, ясности и чувстве уверенности при использовании данного портфеля.

В основу метода Gartner положен принцип ценности приложения для выполнения ключевых функций организации и цели, которые руководство преследует при внедрении соответствующих систем. Используя этот подход, высшие руководители организации могут разделить портфель приложений на три класса в соответствии с относительным вкладом каждого приложения в выполнение ключевых функций и в эффективность деятельности организации.

К первому классу относятся базовые транзакционные (вспомогательные или обслуживающие) приложения. Они играют важную роль с точки зрения обеспечения деятельности организации, но успех в выполнении критически важных задач и лучшие результаты по сравнению с другими организациями создают не они. Хорошими примерами являются приложение для расчета заработной платы или система управления персоналом. Операции, выполняемые этими приложениями, должны проводиться четко и вовремя, но, например, сам факт свое-

временного получения сотрудником зарплаты еще не означает высокую эффективность работы организации в целом. Важными требованиями к таким приложениям являются низкая стоимость, надежность, возможность выполнять большой объем операций при низкой стоимости в расчете на одну транзакцию. В действительности такие приложения в портфеле ИС предприятия составляют большинство.



Рис. 15. Пирамида преимуществ

Второй класс приложений — это информационные приложения, обеспечивающие преимущества бизнесу (предоставление информации для учета, управления, контроля, составления отчетов, анализа, совместной работы). Такими приложениями являются, например, системы предоставления отчета о продажах, аналитические системы. Использование данных приложений благоприятно сказывается на деятельности организации. Примерами преимуществ от использования ИТ являются:

- ускорение цикла выполнения операций (например, принятия решения);
- быстрый вывод на рынок новых продуктов и услуг;
- уменьшение производственного цикла;
- более высокое качество;
- более широкий набор продуктов и услуг;
- более глубокая настройка на потребителя;
- меньшая стоимость выполнения операций.

Третий класс составляют инновационные (стратегические) приложения. В некоторых случаях использование информационных технологий может носить радикально новый, революционный характер с точки зрения влияния на функционирование организаций: способность кардинального изменения самой основы конкуренции и получение преимуществ. Примерами таких систем могут быть система электронной торговли через Интернет или система обслуживания кредитных карт банкоматами, которые в начале жизненного цикла этих технологий обеспечивали рост рынка компаниям, их внедрившим.

2.5. Традиционные инструменты описания архитектуры

Основой традиционных подходов к построению моделей бизнес-слоя и системного слоя архитектуры являются методологии структурного и объектно-ориентированного анализа и проектирования. Обе методологии активно используются в процессах проектирования и разработки программных систем и лежат в основе различных CASE-средств. CASE (computer-aided software engineering) — набор инструментов и методов программной инженерии для проектирования программного обеспечения, который помогает обеспечить высокое качество программ, отсутствие ошибок и простоту в обслуживании программных продуктов.

Структурным анализом принято называть метод исследования системы, которое начинается с ее общего обзора и затем детализируется, приобретая иерархическую структуру.

Объектно-ориентированные методологии, основаны на объектной декомпозиции предметной области, представляемой в виде совокупности объектов, взаимодействующих между собой посредством передачи сообщений.

Архитектурные модели, разработанные с помощью этих методологий, дают полное, точное и адекватное описание системы, имеющее конкретное назначение — цель модели. Целью модели является получение ответов на некоторую совокупность вопросов. Эти вопросы неявно присутствуют (подразумеваются) в процессе анализа и, следовательно, они руководят созданием модели и направляют его. Это означает, что сама модель должна будет дать ответы на эти вопросы с заданной степенью точности. Если модель отвечает не на все вопросы или ее ответы недостаточно точны, то говорят, что модель не достигла своей цели.

Технология структурного анализа и проектирования SADT (Structured Analysis and Design Technique) — одна из самых известных и широко используемых методологий проектирования, представляет собой совокупность методов, правил и процедур, предназначенных для моделирования сложных систем. Основанная на принципах структурного анализа, методология предлагает графический язык проектирования систем, в котором сочетаются декомпозиция и иерархическое упорядочение. Для обозначения составляющих системы используется графическая конструкция, называемая SA-блок. SADT была создана и впервые опробована на практике в период с 1969 по 1973 г. Автором методологии является Дуглас Т. Росс, американский математик и программист. Его разработка и широкое успешное использование ее графического языка превратило SADT в методологию, способную значительно повысить качество результатов работ, проводимых на ранних этапах проектов. Предпосылками появления SADT стало возрастание сложности проектируемых систем и необходимость формализации процесса разработки при создании крупномасштабных систем.

Среди многочисленных преимуществ SADT-методологии можно выделить следующие:

- легко отражает такие системные характеристики как управление, обратная связь и исполнители, так как возникла на базе проектирования систем общего

вида в отличие от других структурных методов, «выросших» из проектирования программного обеспечения.

- имеет развитые процедуры поддержки коллективной работы.
- применяется на ранних стадиях создания системы, что позволяет избежать наиболее дорогостоящих ошибок.
- успешно сочетается с другими структурными методами.

Сущностью структурного подхода отражается в том, что для методов данного класса характерны [2]:

- разбиение на уровни абстракции с ограничением числа элементов на каждом из уровней (обычно от 3 до 6–7);
- система декомпозируется на функциональные подсистемы до нужной степени детализации;
- ограниченный контекст, включающий лишь существенные на каждом уровне детали;
- использование строгих формальных правил записи;
- последовательное приближение к конечному результату.

Все методологии структурного анализа базируются на ряде общих принципов, регламентирующих организацию работ по моделированию. В качестве двух базовых принципов используются принцип «разделяй и властвуй» и принцип иерархического упорядочивания. Принцип «разделяй и властвуй» используется при решении трудных проблем путем разбиения их на множество мелких независимых задач, легких для понимания и решения (так называемых «черных ящиков»). При этом пользователю не требуется знать, каким образом решается данная задача, необходимо лишь выявить входы и выходы «черного ящика», а также его назначение, т. е. функцию, которую он выполняет. В процессе разбиения не возникает одинаковых задач, и каждая задача решается независимо от других. Из решения подзадач komponуется решение исходной задачи. Принцип иерархического упорядочивания облегчает проведение анализа системы при ее разбиении на части и, кроме того, «декларирует что устройство этих частей также существенно для понимания».

Важным понятием методологии является понятие «точка зрения» — это позиция, с которой наблюдается система и создается ее модель. Это позиция человека или объекта, в которую нужно встать, чтобы увидеть систему в действии. В процессе моделирования определяется, что включить в модель, а что исключить из нее. Таким образом, точка зрения диктует выбор нужной информации о субъекте и форму ее подачи. Достижение цели становится критерием окончания моделирования.

SADT — это методология, разработанная специально для того, чтобы облегчить описание и понимание искусственных систем, попадающих в разряд средней сложности, может использоваться для моделирования и новых систем и уже существующих систем и может быть направлена как для описания функций, выполняемых системой, так и на описание объектов, составляющих систему.

В первом случае методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели объекта какой-либо предметной области и направлена на определение

требований и функций, которые используются для разработки системы, удовлетворяющей этим требованиям и реализующей эти функции. Функциональная модель SADT отображает функциональную структуру объекта, т.е. производимые им действия и связи между этими действиями.

Во втором случае методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для описания объектов, входящих в систему, их свойств и взаимосвязей между ними. Для уже существующих систем SADT может быть использована для анализа функций, выполняемых системой, на соответствие требованиям бизнеса а также для указания механизмов, посредством которых они осуществляются.

Военно-воздушные силы США использовали методику SADT в качестве части программы интеграции компьютерных и промышленных технологий (Integrated Computer Aided Manufacturing, ICAM) и назвали ее IDEF0 (Icam Definition). Целью программы ICAM было увеличение эффективности компьютерных технологий в сфере проектирования новых средств вооружений и ведения боевых действий. Одним из результатов этих исследований являлся вывод о том, что описательные языки не эффективны для документирования и моделирования процессов функционирования сложных систем. Подобные описания на естественном языке не обеспечивают требуемого уровня непротиворечивости и полноты, имеющих доминирующее значение при решении задач моделирования.

В рамках программы ICAM было разработано несколько графических языков моделирования, которые получили следующие названия:

- нотация IDEF0 — для документирования процессов производства и отображения информации об использовании ресурсов на каждом из этапов проектирования систем;
- нотация IDEF1 — для документирования информации о производственном окружении систем;
- нотация IDEF2 — для документирования поведения системы во времени;
- нотация IDEF3 — специально для моделирования бизнес-процессов.

Нотация IDEF2 никогда не была полностью реализована. Нотация IDEF1 в 1985 году была расширена и переименована в IDEF1X. Методология IDEF SADT, нашла применение в правительственных и коммерческих организациях, поскольку на тот период времени вполне удовлетворяла различным требованиям, предъявляемым к моделированию широкого класса систем.

В начале 1990 года специально образованная группа пользователей IDEF (IDEF Users Group), в сотрудничестве с Национальным институтом по стандартизации и технологии США (National Institutes for Standards and Technology, NIST), предприняла попытку создания стандарта для IDEF0 и IDEF1X. Эта попытка оказалась успешной и завершилась принятием в 1993 г. стандарта правительства США, известного как FIPS для данных двух технологий IDEF0 и IDEF1X. В течение последующих лет этот стандарт продолжал активно развиваться и послужил основой для реализации в некоторых первых CASE средствах.

В настоящее время методология SADT-IDEF представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели системы какой-либо предметной области. Функциональная модель SADT отображает структуру процессов функционирования системы и ее отдельных подсистем, т. е. выполняемые ими действия и связи между этими действиями. Для этой цели строятся специальные диаграммы, которые позволяют в наглядной форме представить последовательность определенных действий. Диаграммы — главные компоненты модели, все функции системы и интерфейсы на них представлены как блоки и дуги. Функциональная модель кроме диаграмм содержит фрагменты текстов и глоссарий, имеющих ссылки друг на друга. Исходными строительными блоками любой модели IDEF0 процесса являются деятельность (activity) и стрелки (arrows). Перечислим Основные понятия и правила IDEF0:

- IDEF0-модель отображает систему в виде иерархии диаграмм;
- каждая диаграмма содержит блоки (деятельность) и дуги (стрелки);
- диаграмма в виде блока отображает функцию;
- интерфейсы входа/выхода представляются дугами, входящими в блок и выходящими из него;
- интерфейсные дуги показывают взаимодействие блоков друг с другом и выражают ограничения, определяющие, когда и каким образом функции выполняются и управляются.
- диаграмма, лежащая на вершине иерархии, называется контекстной на ней вся система представляется в виде единого функционального блока и дуг, изображающих интерфейсы с функциями вне системы;
- следующей в иерархии является диаграмма декомпозиции контекстной диаграммы, на ней функциональный блок контекстной диаграммы декомпозируется на составляющие его функциональные блоки, соединенные интерфейсными дугами;
- эти блоки представляют основные подфункции исходной функции, каждая функция может быть декомпозирована на подфункции;
- подфункция может содержать только те элементы, которые входят в исходную функцию;
- каждый из этих блоков может иметь свою диаграмму декомпозиции, количество блоков на каждом уровне декомпозиции ограничено (может быть от 3 до 6);
- каждый блок на диаграмме имеет свой номер, диаграммы связаны по номерам блоков;
- метки и наименования блоков уникальны;
- входы и управления разделены по роли данных;
- место соединения дуги с блоком определяет тип интерфейса. Управляющая информация входит в блок сверху, информация, которая подвергается обработке — слева, результаты — выходят из правой стороны. Механизм (человек или автоматизированная система), который осуществляет операцию, входит в блок снизу.

- родительский блок и его интерфейсы обеспечивают контекст, из модели нельзя выбросить какие-либо элементы или добавить их;
- дуги, входящие в блок и выходящие из него на диаграмме верхнего уровня, являются точно теми же самыми, что и дуги, входящие в диаграмму нижнего уровня и выходящие из нее.

Метод моделирования IDEF3, являющийся частью семейства стандартов IDEF, предназначен для таких моделей процессов, в которых важно понять последовательность выполнения действий и взаимозависимости между ними. Хотя IDEF3 и не достиг статуса федерального стандарта США, он приобрел широкое распространение среди аналитиков как дополнение к методу функционального моделирования IDEF0 (модели IDEF3 могут использоваться для детализации функциональных блоков IDEF0, не имеющих диаграмм декомпозиции). Основой модели IDEF3 служит так называемый сценарий процесса, который выделяет последовательность действий и подпроцессов анализируемой системы.

IDEF3 является стандартом документирования технологических процессов, происходящих на предприятии, и предоставляет инструментарий для наглядного исследования и моделирования их сценариев. Сценарием (Scenario) можно назвать описание последовательности изменений свойств объекта в рамках рассматриваемого процесса (например, описание последовательности этапов обработки детали в цеху и изменение ее свойств после прохождения каждого этапа). Исполнение каждого сценария сопровождается соответствующим документооборотом, который состоит из двух основных потоков: документов, определяющих структуру и последовательность процесса. Для эффективного управления любым процессом необходимо иметь детальное представление о его сценарии и структуре сопутствующего документооборота. Средства документирования и моделирования IDEF3 позволяют:

- документировать имеющиеся данные о технологии процесса, выявленные, например, в процессе опроса компетентных сотрудников, отвечающих за организацию исследуемого процесса;
- определять и анализировать точки влияния потоков сопутствующего документооборота на сценарий технологических процессов;
- определять ситуации, в которых требуется принятие решения, влияющего на жизненный цикл процесса, например, изменение конструктивных, технологических или эксплуатационных свойств конечного продукта;
- содействовать принятию оптимальных решений при реорганизации технологических процессов;
- разрабатывать имитационные модели.

Диаграммы потоков данных DFD (Data Flow Diagrams) представляют собой иерархию функциональных процессов, связанных потоками данных. Цель такого представления — продемонстрировать, как каждый процесс преобразует свои входные данные в выходные, а также выявить отношения между этими процессами.

Процессы предназначены для продуцирования выходных потоков из входных в соответствии с действиями, задаваемыми именами процессов. Имя про-

цесса должно содержать глагол в неопределенной форме с последующим дополнением, например «Сформировать личную карточку участника». Кроме того, каждый процесс должен иметь уникальный номер для ссылок на него внутри диаграммы. Этот номер может использоваться совместно с номером диаграммы для получения уникального индекса процесса во всей модели.

Потоки данных — это механизмы, используемые для моделирования передачи информации (или, возможно, физических компонентов) из одной части системы в другую. Потоки на диаграммах обычно изображаются именованными стрелками, ориентация которых указывает направление движения информации.

Для построения DFD традиционно используются две различные нотации, соответствующие методам Йордона-ДеМарко и Гейна-Сэрсона. Эти нотации незначительно отличаются друг от друга графическим изображением символов. В соответствии с данными методами модель системы определяется как иерархия диаграмм потоков данных, описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи потребителю. При этом для описания функционирования процесса в случае отсутствия необходимости детализировать его с помощью DFD используется миниспецификация, фактически представляющая собой алгоритм описания задачи, выполняемой процессом, при этом множество всех миниспецификаций является полной спецификацией системы.

Практически любой класс систем успешно моделируется при помощи DFD-ориентированных методов. Они с самого начала создавались как средство проектирования информационных систем (тогда как SADT как средство моделирования систем вообще) и имеют более богатый набор элементов, адекватно отражающих специфику таких систем (например, хранилища данных являются прообразами файлов или баз данных, внешние сущности отражают взаимодействие моделируемой системы с внешним миром).

Наиболее распространенным средством моделирования отношений между данными (информационного моделирования) является диаграмма «сущность-связь» (Entity-Relationship Diagram — ERD), известная в двух нотациях — Чена и Баркера. Она предназначена для обеспечения стандартного способа определения данных и отношений между ними. С ее помощью документируются информационные аспекты системы, включая идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их связей с другими объектами (отношений). ERD традиционно используется в структурном анализе и проектировании, однако, по существу, представляет собой подмножество объектной модели предметной области. Одна из разновидностей модели «сущность-связь» используется в методе IDEF1/IDEF1X, входящем в семейство стандартов IDEF. Фактически с помощью ERD осуществляется детализация хранилищ данных проектируемой системы и чаще всего используются для разработки моделей данных.

Все вышеперечисленные модели содержат графические и текстовые средства моделирования: первые для удобства отображения основных компонент модели, вторые для обеспечения точного определения ее компонент и связей. Графические нотации используются (в том или ином наборе) практически во всех современных структурных методологиях. Роль этих методологий заключается в

регламентации основ моделирования бизнес-слоя и системного слоя архитектуры предприятия. Они описывают последовательность шагов, модели и подходы, тщательное следование которым приведет к хорошим результатам. Хотя методологии, вообще говоря, не гарантируют качества построенных моделей, тем не менее, они помогают охватить и учесть все важные этапы, шаги и моменты разработки, помогают справиться с проблемами размерности, и в конечном итоге оценить продвижение вперед. Более того, методологии обеспечивают организационную поддержку, позволяющую большим коллективам исполнителей функционировать скоординированным образом.

Другими словами, структурная методология определяет руководящие указания для построения (и дальнейшей) оценки архитектурных моделей, шаги работы, которые должны быть выполнены, их последовательность, а также правила распределения и назначения применяемых при этом операций и методов.

Важнейшей характеристикой структурной методологии является порядок построения модели. В соответствии с ним все методологии классифицируются на два вида: функционально-ориентированные и информационно-ориентированные.

Традиционный функционально-ориентированный подход регламентирует первичность проектирования функциональных компонент по отношению к проектированию структур данных: требования к данным раскрываются через функциональные требования.

При информационно-ориентированном подходе вход и выход являются наиболее важными — структуры данных определяются первыми, а процедурные компоненты являются производными от данных.

Предпочтительное использование функционально-ориентированных подходов связано с тем, что современная организация характеризуется переносом центра тяжести на слой бизнес-правил. Модель процесса является ценным средством для размышлений и совместной работы над перспективами развития организации и системной разработкой, поскольку руководство прекрасно ориентируется в технологиях и бизнес-процессах организации, и функциональные модели (в отличие от информационных) интуитивно понимаемы неспециалистами. Кроме того, информационная модель, как правило, представляет собой единственную диаграмму, возможно содержащую несколько сотен объектов, тогда как функциональная иерархическая модель может включать десятки тысяч объектов. Тем не менее, информационная модель продолжает оставаться важной и соответствующим образом влиять на разрабатываемую функциональную модель. Подтверждением первичности функциональной модели является тот факт, что на Западе, где различные методики реорганизации (BSP, CPI/TQM, BPR) применяются уже длительное время, большинство методологий являются функционально-ориентированными.

Важное место в разработках архитектурных моделей занимают объектно-ориентированные методологии, основанные на объектной декомпозиции предметной области, представляемой в виде совокупности объектов, взаимодействующих между собой посредством передачи сообщений. В качестве объектов предметной области могут рассматриваться конкретные предметы, а также абстракт-

ные или реальные сущности (например, клиент, заказ, организация и т.п.). Каждый объект характеризуется своим состоянием (точнее, набором атрибутов, значения которых определяют состояние), а также набором операций для проверки и изменения этого состояния. Каждый объект является представителем некоторого класса однотипных объектов, определяющего их общие свойства. Все представители (экземпляры) одного и того же класса имеют один и тот же набор операций и могут реагировать на одни и те же сообщения.

Объектно-ориентированный подход заключается в представлении моделируемого процесса в виде совокупности классов и объектов предметной области. При этом иерархический характер сложного процесса отражается с использованием иерархии классов, а его функционирование рассматривается как взаимодействие объектов.

Большинство современных методов объектно-ориентированного подхода основано на использовании унифицированного языка моделирования UML (Unified Modeling Language), фактического приемника наиболее распространенных объектно-ориентированных методов Буча, Рамбо и Якобсона, представляющего собой язык для определения, представления, проектирования и документирования программных систем, организационно-экономических систем, технических систем и других систем различной природы. UML находится в процессе стандартизации, проводимом OMG (Object Management Group), организацией по стандартизации в области объектно-ориентированных методов и технологий, в настоящее время принят в качестве стандартного языка моделирования и получил широкую поддержку в индустрии ПО. Стандарт UML версии 1.1, принятый OMG в 1997 г., содержит следующий набор диаграмм:

1. Структурные (structural) модели:

- диаграммы классов (class diagrams) — для моделирования статической структуры классов системы и связей между ними;
- диаграммы компонентов (component diagrams) — для моделирования иерархии компонентов (подсистем) системы;
- диаграммы размещения (deployment diagrams) — для моделирования физической архитектуры системы.

2. Модели поведения (behavioral):

- диаграммы вариантов использования (use case diagrams) — для моделирования функциональных требований к системе (в виде сценариев взаимодействия пользователей с системой);
- диаграммы взаимодействия (interaction diagrams): диаграммы последовательности (sequence diagrams) и коллаборативные диаграммы (collaboration diagrams) — для моделирования процесса обмена сообщениями между объектами;
- диаграммы состояний (statechart diagrams) — для моделирования поведения объектов системы при переходе из одного состояния в другое;
- диаграммы деятельности (activity diagrams) — для моделирования поведения системы в рамках различных вариантов использования, потоков управления.

UML обладает механизмами расширения для адаптации языка моделирования к конкретным нуждам. Наличие механизмов расширения принципиально отличает UML от таких средств моделирования, как IDEF0, IDEF1, IDEF3, DFD, STD и ERD.

3. Модели представления, методы и инструментальные среды разработки архитектуры предприятия

3.1. Модель Захмана — базовая методология построения архитектуры предприятия

На сегодняшний день существуют различные методы и модели представления архитектуры предприятия, основное назначение которых связать характеристики информационной системы с бизнес-задачами предприятия: методики аналитических компаний Gartner, Giga, META и др., метод Захмана, методика TOGAF; методика POSIX 1003.23i и др. Значительный вклад в развитие концепции архитектуры предприятия был сделан Джоном Захманом (John A. Zachman). Модель Захмана (Zachman Framework for Enterprise Architecture) является наиболее известной обобщенной моделью архитектуры предприятия и давно стала стандартом де-факто. В 1987 году появились первая статья Дж. Захмана в которой была описана первая версия этой модели, расширенная впоследствии в работах 1992–1996 гг. В 1992 году, во второй статье Дж. Захман предложил вариант обобщенной схемы или структуры (framework, или «фреймвок») для описания и анализа архитектуры информационной системы. Формально, по названию это была еще архитектура ИС, но по содержанию уже архитектура предприятия. В 90-х годах она была использована достаточно большим количеством крупных компаний, входящих в список 2000 крупнейших корпораций мира такими, например, как General Motors, Bank of America.

С момента публикации «модель Захмана для описания архитектуры предприятия» прошла определенную эволюцию в своем развитии и стала основой, на базе которой многие организации создают свои собственные методики описания информационной инфраструктуры предприятия. Модель Захмана также послужила основой для создания целого ряда других методик и моделей описания архитектуры предприятия, таких как Федеральная Архитектура США (FEAF — Federal Enterprise Architecture Framework), Методика описания архитектуры Open Group (TOGAF — The Open Group Architecture Framework), Методика описания архитектуры министерства обороны США (DoDAF — Department of Defense Architecture Framework) и др.

В более поздних работах Дж. Захман определил Архитектуру предприятия как «набор описательных представлений (моделей), которые применимы для описания Предприятия в соответствии с требованиями управленческого персонала (качество) и которые могут развиваться в течение определенного периода (динамичность)». Модель Захмана преследует две основные цели — с одной стороны, логически разбить все описание Архитектуры на отдельные разделы для упрощения их формирования и восприятия, с другой — обеспечить возможность рассмотрения целостной Архитектуры с выделенных точек зрения или соответствующих уровней абстракции.

В то время, когда были опубликованы работы Захмана, традиционным подходом при формировании описания системы являлось использование концепции жизненного цикла, включающего такие этапы, как планирование, анализ, проек-

тирование, разработка, документирование, внедрение и промышленная эксплуатация. На каждом из этих этапов рассматриваются вопросы, связанные как с функциями системы, так и с данными. Захман предложил вместо традиционного подхода, связанного с рассмотрением отдельных аспектов работы системы как бы в различные моменты времени, использовать рассмотрение системы с различных перспектив (точек зрения). Основная идея заключается в том, чтобы обеспечить возможность последовательного описания каждого отдельного аспекта системы в координации со всеми остальными. Для любой достаточно сложной системы общее число связей, условий и правил обычно превосходит возможности для одновременного рассмотрения. В то же время отдельное, в отрыве от других, рассмотрение каждого аспекта системы чаще всего приводит к неоптимальным решениям, как в плане производительности, так и стоимости реализации.

Суть метода сводится к формализованному представлению модели предприятия в виде матрицы 6×6 (рис. 16), в которой каждая ячейка задает свой тип описания (модели) свойств предприятия. Вся совокупность ячеек разделена на шесть столбцов матрицы — шесть аспектов деятельности предприятия:

- «ЧТО делается», или объекты/данные;
- «КАК делается», или функции/процессы/действия;
- «ГДЕ делается», — размещение или инфраструктура;
- «КТО делает» — люди, оргединицы;
- «КОГДА делается» — графики событий и работ, время;
- «ЗАЧЕМ делается» — стимулы, мотивы и стратегии деятельности.

	Объекты (Что?)	Действия (Как?)	Дислокация (Где?)	Люди (Кто?)	Время (Когда?)	Мотивы (Почему?)	
От целей бизнеса к проекту	Планировщик						Сфера действия
	Владелец предприятия						Модель предприятия
	Проектировщик						Системная модель
	Разработчик						Техническая модель
	Субподрядчик						Компоненты архитектуры
		Данные	Функции	Сеть	Организация	Расписание	Стратегия
Элементы архитектуры предприятия							

Рис. 16. Модель Захмана

Все аспекты предлагается описывать в шести разных, но связанных представлениях, сгруппированных в строки матрицы. Для строк-представлений Захман применил аналогии с классическим архитектурным делом и строительством. Верхняя строка матрицы фиксировала представление «планировщика застройки» = бизнес-стратег, который рассматривает не одно здание, а все его окружение и то, как в это окружение вписывается здание. Вторая строчка фиксировала представление «владельца дома» = собственник предприятия; третья —

представление дизайнера = проектировщик; четвертая — того, кто будет руководить собственно строительными работами = разработчик; пятая - взгляд тех, кто будет выполнять отдельные работы = субподрядчик, программист; а шестая относилась к эксплуатации дома = работающая система, пользователь. Посредством этой аналогии для АП задавались представления предприятия с позиций бизнеса, аналитиков-проектировщиков ИС, а также их разработчиков. Таким образом, в строках этой матрицы отображаются различные категории специалистов, определенным образом связанных с деятельностью предприятия (планировщик, собственник предприятия, проектировщик, разработчик и субподрядчик).

Такая схема создает контекст для описания различных представлений архитектуры разрабатываемой информационной системы. Эти представления соответствуют тому, как видит систему ее заказчик, проектировщик и разработчик, причем в разрезе трех выбранных аспектов, к которым относятся данные, функции и сетевая структура. В схеме Захмана строке соответствует точка зрения какого-либо участника проекта по созданию системы. Аспекты представлены в схеме колонками. Архитектурное представление — это ячейка таблицы, соответствующая пересечению выбранных столбца и строки.

Захман определяет архитектуру информационной системы как представление ее с точки зрения одного из заинтересованных лиц. В зависимости от того, кем вы являетесь и на каком аспекте фокусируете внимание, вы видите архитектуру системы по-разному. Архитектура ИС пронизывает архитектуру предприятия, которая, в свою очередь, по-разному описывается в зависимости от рассматриваемых аспектов деятельности предприятия.

Перечислим основные правила заполнения таблицы:

- порядок следования колонок несущественен;
- каждая колонка имеет в основе простую, базовую модель;
- базовые модели для каждой из колонок являются уникальными;
- каждая клетка уникальна и содержит соответствующее описание аспекта реализации системы в виде определенной модели или, возможно, простого описания (текстового документа);
- каждая клетка таблицы независима от других, вместе они образуют функционально полное пространство для описания системы («базис»);
- совокупность клеток одного ряда формирует полное описание системы с соответствующей точки зрения (перспективы);
- заполнение клеток должно проводиться последовательно «сверху вниз».

Первые два уровня абстракции, описываемые первой и второй строками таблицы соответствуют наиболее общим представлениям и достаточно широко описывают существующее окружение, планы и цели.

Первая строка соответствует уровню планирования бизнеса в целом (бизнес-модель) и отражает интересы высшего руководства и собрания акционеров. На этом уровне вводятся достаточно общие основные понятия, определяющие бизнес — например, продукты и услуги, клиенты, расположение объектов бизнеса, а также формулируется бизнес-стратегия (колонка 6 — мотивация). Фактически, данная строка определяет контекст всех последующих строк.

Вторая строка (концептуальная модель) предназначена для определения в терминах бизнеса структуры организации, ключевых и вспомогательных бизнес-процессов и соответствует интересам бизнес-менеджеров и владельцев процессов.

Третий уровень (логическая модель) соответствует рассмотрению с точки зрения Системного Архитектора. Здесь бизнес-процессы описываются уже в терминах информационных систем, включая различные типы данных, правила их преобразования и обработки для выполнения определенных на уровне 2 бизнес-функций. На этом уровне происходит организация «командной» работы бизнес-менеджеров, бизнес-аналитиков и менеджеров, отвечающих за разработку ИТ.

Четвертый и последующие уровни описывают детали, представляющие интерес для ИТ-менеджеров, проектировщиков, разработчиков и пользователей ИТ-системы.

На четвертом уровне определяется технологическая модель, включающая физическую модель и детали реализации, т. е. осуществляется привязка данных и операций над ними к выбранным технологиям реализации. Например, здесь может быть определен выбор реляционной СУБД, или средств работы с неструктурированными данными, или объектно-ориентированной среды.

Пятый уровень соответствует детальной реализации системы, включая конкретные модели оборудования, топологию сети, производителя и версию СУБД, средства разработки и собственно готовый программный код. Многие из работ на данном уровне часто выполняются субподрядчиками.

Шестой уровень описывает работающую систему. На этом уровне могут быть введены, в том числе, такие объекты, как инструкции для работы с системой, фактические базы данных, работа службы HelpDesk. Надо заметить, что в исходной работе Захмана содержание этого уровня не детализируется. При развитии модели появляется возможность рассмотрения аспектов функционирования работающей системы с точки зрения, например, конечного пользователя или эксплуатирующих служб.

На каждом уровне участники рассматривают одни и те же категории вопросов, соответствующие столбцам (колонкам) таблицы, только с различным уровнем абстракции и детализации.

Так, первая колонка «**Данные**» отвечает на вопрос «**ЧТО?**» и определяет используемые в системе данные. На верхнем уровне достаточным будет простое перечисление основных объектов, используемых в бизнесе. На втором уровне данные объекты объединяются в семантическую модель высокого уровня и обычно описываются в виде диаграммы "сущности-связи" (ER диаграммы) с отражением основных связей и наиболее существенных бизнес-ограничений. На третьем уровне эта модель приводится к нормализованной форме, определяются все атрибуты и ключи. Четвертый уровень представляет собой физическую модель данных в системе (в объектно-ориентированном подходе — иерархию классов). Следующий уровень содержит описание модели на языке управления данными для формирования таблиц, готовые библиотеки классов, табличные пространства СУБД. Наконец, последний уровень может описывать фактические

наборы данных, в том числе такие характеристики, как журналы доступа, размеры реально занимаемого дискового пространства, статистику обращений и т.п.

Можно отметить определенное несовершенство данной модели при использовании объектно-ориентированного подхода — фактически модель предписывает раздельное рассмотрение данных (свойств) и функций (методов) классов.

Колонка «**Функции**» (ответ на вопрос «**КАК?**») предназначена для последовательной детализации описания того, как миссия предприятия реализуется на уровне отдельных операций. В частности, на первом уровне достаточным будет простое перечисление бизнес-процессов. Второй уровень содержит модель бизнес-процессов, которая впоследствии детализируется *на третьем уровне* в операции над данными и архитектуру приложений; *на четвертом уровне* — в методы классов; *на пятом уровне* содержится программный код и, наконец, исполняемые модули — *на шестом уровне*. При этом, начиная с четвертого уровня, рассмотрение ведется уже не в рамках предприятия в целом, а по отдельным подсистемам или приложениям.

Следующая колонка «**Сеть**» (ответ на вопрос «**ГДЕ?**») определяет пространственное распределение компонент системы и сетевую организацию. *На уровне планирования бизнеса* здесь достаточно определить расположение всех производственных объектов. *На следующем уровне* эти объекты объединяются в модель со связями, характеризующими взаимодействие между собой, — будь то обмен информацией или поставки товаров. *На третьем уровне* системной архитектуры осуществляется привязка компонент информационной системы к узлам сети. *Четвертый уровень* служит для определения физической реализации в терминах аппаратных платформ, системного программного обеспечения, а также средств промежуточного уровня (так называемое *middleware*), используемых для интеграции различных компонент информационной системы между собой. Типичным примером могут являться брокеры запросов или средства обмена сообщениями. *На пятом уровне* определяются используемые протоколы и спецификации каналов связи. *Последний уровень* описывает функционирование реализованной сети.

Колонка «**Организации / Люди**», (ответ на вопрос «**КТО?**»), определяет участников процесса. *На уровне планирования бизнеса* здесь представлен список подразделений предприятия и выполняемые ими функции. *На следующем уровне* приводится полная организационная диаграмма, а также могут быть определены общие требования к информационной безопасности. *Далее* последовательно определяются участники бизнес-процессов и их роли (*уровень 3*) (в RUP используются диаграммы событий и описание вариантов использования); требования к интерфейсам пользователя и правила доступа к отдельным объектам (*уровень 4*), физическая их реализация на уровне кода или операторов определения доступа к таблицам в СУБД (*уровень 5*). *Последний уровень* описывает обученных пользователей системы.

Пятая колонка «**Расписание**» (ответ на вопрос «**КОГДА?**»), определяет временные характеристики бизнес-процессов и работы системы. Детализация так же осуществляется сверху вниз, начиная от списка важных событий (*уровень*

1) и календарного плана (*уровень 2*), с описанием основных параметров, характеризующих выполнение бизнес-процессов, — например, требование ко времени оформления сделки. На *третьем уровне* определяются события, вызывающие изменение состояния информационных объектов и инициацию операций над ними (диаграммы зависимостей, последовательностей). На *следующем уровне* эти события транслируются в программные вызовы (триггеры) или передаваемые сообщения (диаграмма потоков управления). *Пятый уровень* определяет физическую реализацию обработки таких событий (определения интервалов, временные диаграммы). Наконец, на *шестом уровне* — фактическая история функционирования системы.

Последняя колонка «*Стратегии*» (ответ на вопрос «ПОЧЕМУ?» или «ЗАЧЕМ?») служит для определения мотивации и задает порядок перехода от задач бизнеса к требованиям и элементам информационных систем. Исходной точкой является бизнес-стратегия (*уровень 1*), которая затем последовательно транслируется в бизнес-план (*уровень 2*), затем в правила и ограничения для реализации бизнес-процессов (*уровень 3*), на *четвертом уровне* — в соответствующие приложения, необходимые для включения в состав информационных систем и, в дальнейшем, в их физическую реализацию (*уровень 5*).

Основными характеристиками модели Захмана являются:

- простота для понимания как техническими, так и нетехническими специалистами;
- логическая полнота и согласованность;
- целостность в отношении предприятия, то есть каждая проблема может быть соотнесена с предприятием в целом;
- поддержка обсуждений сложных вопросов с использованием относительно небольшого количества нетехнических понятий;
- возможность применения для планирования, позволяющего лучше принимать решения за счет того, что решение никогда не будет выноситься «в пустоте» (в отрыве от остальных аспектов деятельности предприятия);
- применимость для решения задач, то есть возможность работать с абстракциями и сущностями, выделяя и изолируя отдельные параметры системы без потери восприятия Предприятия как целого;
- нейтральность, то есть независимость от каких-либо инструментов; благодаря этому каждый инструмент и методология могут быть отображены на данную модель и могут явно показать, что они делают и чего они не делают;
- модель обеспечивает общий словарь и набор перспектив, или структур, для описания современных сложных корпоративных систем;
- комплексный учет всех архитектурно-существенных факторов, с возможностью концентрироваться на отдельных аспектах архитектуры сохраняя при этом общий взгляд на предприятие как единое целое.

Метод Захмана позволяет:

- концентрироваться на отдельных аспектах предприятия или его конкретной системы и в то же время не терять взгляда на него как на целое;

- использовать одну понятную и бизнес-руководителям, и компьютерным специалистам концептуальную основу для совместных обсуждений и планирования;
- планировать соответствие друг другу описаний-ячеек, обеспечивая тем самым согласование бизнеса и ИТ;
- сохранять при этом независимость от какого-либо программного продукта (инструмента) с его формализмами, особенностями и ограничениями.

Нельзя, конечно, считать, что данная модель лишена недостатков. Один из них заключается в том, что при применении ее на практике возникают определенные трудности, связанные с отсутствием встроенного механизма распространения изменений между элементами таблицы. Действительно, предположим, что изменилась организация процесса поставок в компании (схема логистики). Это потребует отслеживания вручную всех взаимосвязей, проверки актуальности и внесения изменений в модели и другие артефакты во всех потенциально затрагиваемых ячейках. Другим ограничением модели является отсутствие рассмотрения системы в динамике, т.е. не поддерживает представление о динамике развития организации и ее ИС. Действительно, каждый элемент таблицы может содержать как описание существующего состояния «как есть», так и целевого, а также всех промежуточных состояний. При этом сама модель не содержит средств для четкого разделения этих различных временных срезов.

3.2. Обзор моделей и методик построения архитектуры предприятия

Обобщение подхода Захмана было предложено в работах Е.Б. Зиндера. Основная идея заключается в обеспечении возможностей отражения постоянного развития предприятия и его информационных систем как непрерывной последовательности трансформаций. Вместо традиционной двумерной таблицы было предложено ввести трехмерную схему, добавив к плоским схемам ось стратегического времени. На этой оси располагаются отрезки времени осуществления различных проектов и стадий развития информационных систем и всего предприятия. Таким образом, была создана «объемная» схема архитектуры предприятия или модель «3D-предприятие», которая строится в трех измерениях с учетом временного пространства.

При этом первые два измерения аналогичны используемым Захманом, но не совпадают с оригиналом по содержанию и трактовке. Третья ось позволяет явно определять изменения, которые происходили и будут происходить с предприятием, его существующими информационными системами, а также с различными проектами развития и трансформации. Как элементы базовой классификации сущностей в плоскости Развития предприятия на этой оси рассматриваются мероприятия, позволяющие ответить на вопросы:

- каковы текущие потребности предприятия;
- как должно функционировать предприятие сегодня;
- как внедряются новые правила, бизнес-процессы;
- как должно функционировать предприятие завтра;
- какова новая культура в организации;
- как должно функционировать предприятие послезавтра.

Модель «3D-предприятие» строится для отражения взаимосвязей ключевых компонентов ИУС предприятия на выбранном историческом участке времени его развития в трех измерениях, предусмотренных 3D-схемой:

1. Ось уровня проектирования и использования ИУС содержит шесть горизонтальных разделов: потребности и планы, бизнес-модель, логическая модель, техническая конструкция, детальная реализация, практика использования.

2. Ось раздела обеспечения и аспекта работы ИУС содержит шесть вертикальных разделов: почему(цели), кто (деятели системы — люди и организационные единицы), что (информация), как (функции и процессы), когда (события и графики функционирования), где (размещение и коммуникации).

3. Ось времени, в котором развивается предприятие и его ИУС, содержит шесть возможных (но не единственных) стадий на верхней грани модели, соответствующих возможным стадиям жизненного цикла системы:

- стратегический анализ целей и потребностей, детальный анализ предприятия, (могут рассматриваться отдельно)
- конструирование (проектирование) решений,
- детальная реализация системы решений,
- внедрение (ввод в действие) решений,
- использование (эксплуатация) системы,
- совершенствование системы.

Модель описания ИТ-архитектуры Gartner. Одним из возможных достаточно простых форматов описания архитектуры Gartner является простое матричное представление, которое для каждой из основных областей архитектуры ИТ (данных, приложений, интеграции, общих сервисов и инфраструктуры) «последовательно накладывает» несколько спецификаций (бизнес-потребности, принципы, процессы и руководства, протоколы и стандарты, используемые продукты и технологии), отличающихся по уровню детализации и конкретизации. Данный подход позволяет обеспечить отслеживание логической связи между выбранными технологиями, их ценностью и потребностями для бизнеса.

В 2002 г. в компании Gartner была сформулирована новая концепция архитектуры предприятия, ставшая определенным обобщением ранней модели ИТ-архитектуры на основе матричного представления и косвенным отражением растущей важности вопросов взаимодействия предприятий. На формирование новой концепции также оказали влияние концепции сервис-ориентированной архитектуры и осознание факта существования различных стилей архитектуры информационных систем, соответствующих различным стилям бизнес-процессов. Модель Gartner 2002 года сформулирована в виде четырех уровней:

1. Среда бизнес-взаимодействия (Business Relationship Grid), описывающая новую модель «виртуального» бизнеса и процессы, связанные с кооперацией предприятий и бизнесом, B2B. Этот уровень идентичен понятию «отраслевой нервной системы» взаимодействующих предприятий. Он получил развитие в связи с распространением Интернета как среды взаимодействия и связан с понятиями доступа и межорганизационного взаимодействия.

2. Бизнес-процессы и стили бизнес-процессов описывающие, каким образом организация выполняет свои ключевые функции (бизнес-процессы предприятия, такие как обработка заказа, мониторинг производственных процессов, анализ использования критически важных ресурсов, совместная работа с информацией).

3. Шаблоны, описывающие модели и алгоритмы, которые могут широко использоваться для решения различных задач на предприятии. Отметим, что шаблоны охватывают не только область программного обеспечения, но и соответствующие сетевые и вычислительные ресурсы. Примерами шаблонов являются: трехуровневая архитектура прикладных систем (интерфейс — логика — данные), использование «толстого» клиента в архитектуре клиент/сервер, хранилища данных. Что касается приложений, то упор сделан на использовании шаблонов сервис-ориентированной архитектуры, т.е. реализации приложений в виде модульного набора различных типов сервисов. Это позволяет в перспективе интегрировать приложения как web-сервисы.

4. Технологические строительные блоки (кирпичики — bricks), соответствующие технологической архитектуре и включающие в себя операционные системы, серверы, базы данных, собственно данные и прочее.

В данной схеме первый и второй уровни ориентированы на совместную работу бизнес-руководителей и ИТ-специалистов (бизнес-архитектура), а третий и четвертый уровни входят во внутреннюю компетенцию ИТ-службы. При этом уровни ИТ-архитектуры соответствуют различным уровням выполнения операций реального бизнеса.

Такой подход позволяет продемонстрировать руководству предприятия механизм влияния решений в области ведения бизнеса на решения в области использования ИТ. Первый и второй уровни модели показывают важность и необходимость построения архитектуры предприятия по мере развития моделей ведения бизнеса в сторону «более виртуальных» структур («расширенных организаций»), успех которых будет в существенной степени зависеть от рациональной реализации архитектуры. Полная модель Gartner представляет собой «трехмерную» взаимосвязанную комбинацию (треугольную пирамиду) бизнес-архитектуры, технологической и информационной архитектур.

Подход компании Gartner представляет собой пример реализации методологии достаточно высокого уровня. Он задает только общую рамочную модель описания и не определяет ни форматов, ни какого-либо специализированного языка для описания. Что касается разработки архитектуры, то в данном подходе сформулированы важные и полезные рекомендации в виде последовательности шагов и задач участников, которые не детализируются до уровня моделей процесса разработки архитектуры.

Основу методики *META Group* составляет Технологическая архитектура масштаба предприятия (EWTA — Enterprise Wide Technical Architecture). Понимание наличия тесной связи между бизнесом и информационными технологиями привело к добавлению в представления архитектуры предприятия следующих доменов:

- бизнес-архитектура (EBA — Enterprise Business Architecture);

- архитектура информации (EAI — Enterprise Information Architecture);
- портфель прикладных систем предприятия (EAP — Enterprise Application Portfolio);
- управление программами и проектами в масштабах предприятия (EPM — Enterprise Program Management).

Первые два домена соответствуют эволюции понятия «архитектура предприятия», которая происходила на рынке в целом и продолжается в принятой сегодня практике выделения доменов архитектуры. Третий и четвертый домены — предложение META Group рассматривать архитектуру предприятия в интеграции с ключевыми процессами и проектами, такими как процесс управления корпоративными ИТ-программами, процесс выработки стратегии и планирования.

Кроме этого, в технологической архитектуре (EWTA) выделяют набор доменов, которые объединяют группы связанных между собой технологий и компонентов. Каждый домен технологической архитектуры включает описание принципов, технологий, стандартов, продуктов, конфигураций, лучших практик, которые являются многократно используемыми строительными блоками при построении ИТ-систем.

Объединяющим для всех доменов методики META Group является процесс формулировки бизнес-требований к ИТ-архитектуре, что оформляется в виде двух документов:

- «Видение общих требований» (CRV — Common requirements Vision);
- «Принципы концептуальной архитектуры» (CA — Conceptual Architecture).

Полное описание методики разработки архитектуры предприятия META Group содержится в документе Enterprise Architecture Desk Reference в котором приводятся следующие разделы:

- практическая реализация архитектуры через процесс управления корпоративными ИТ-программами и проектами;
- вопросы управления и контроля архитектурного процесса;
- оценка зрелости архитектуры;
- анализ технологических тенденций и планирование;
- управление портфелем ИТ-активов и проектов.

Методика TOGAF. Разработчиком методики описания архитектуры TOGAF (The Open Group Architecture Framework) является некоммерческое объединение The Open Group ряда ведущих производителей информационных технологий. TOGAF не эталонная модель, а как «средство для разработки архитектур информационных систем». Основное назначение TOGAF — ускорить и облегчить процесс разработки архитектуры конкретной организации, обеспечивая при этом возможность будущего развития.

В отличие от методики META Group ориентирована на разработку программной инфраструктуры ИС и в наилучшей мере подходит для описания интеграционных компонентов, использующихся для поддержки широкого спектра корпоративных приложений, прежде всего, критичных для бизнеса. В состав модели TOGAF входят два основных компонента: методика ADM

(Architecture Development Method), определяющая процесс разработки архитектуры, и базовая архитектура FA (Foundation Architecture). Базовая архитектура дополняется соответствующей базой данных ресурсов, включающей описания архитектурных принципов, примеров реализации, а также специализированный язык ADML.

Процесс разработки архитектуры по методике ADM разбит на фазы:

- фаза O подготовка — уточнение модели с учетом особенностей организации, определение принципов реализации проекта;
- фаза A — определение границ проекта, разработка общего представления архитектуры; утверждение плана работ и подхода руководством;
- фаза B — разработка бизнес-архитектуры предприятия;
- фаза C — разработка архитектуры данных и технологической архитектуры;
- фаза D — разработка архитектуры приложений;
- фаза E — проверка возможности реализации предложенных решений;
- фаза F — планирование перехода к новой системе;
- фаза G — формирование системы управления преобразованиями;
- фаза H — управление изменением архитектуры.

Каждая фаза разбивается на этапы (процессы), которые в свою очередь состоят из работы. Для каждой работы определяются решаемые входе ее выполнения задачи, входные и выходные документы.

Процесс разработки не заканчивается выбором оптимальной архитектуры и разработкой плана миграции. Необходимыми элементами являются задачи, выполняемые на фазах G и H. В частности, для обеспечения практического принятия архитектуры в организации и успеха проекта обязательным является формирование системы управления реализацией архитектуры (Implementation Governance). Важно отметить, что TOGAF распространяется свободно и может быть использована бесплатно любой организацией для разработки внутренних проектов. Лицензируется только коммерческое использование. В настоящее время с сайта The Open Group было загружено 90000 копий TOGAF. Количество проданных экземпляров серии TOGAF составило более 20000.

NASCIO Architecture Toolkit. NASCIO (National Association of State Information Resource Executives) — Национальная ассоциация руководителей государственных информационных ресурсов (США, 1969 г.) разработала набор шаблонов IT Architecture Toolkit — инструментарий для документирования ИТ-архитектуры организации.

Основное преимущество использования набора шаблонов IT Architecture Toolkit заключается в построении иерархической системы описаний элементов, удобной для поддержания жизненного цикла документа, т. е. в форме, предполагающей его возможные изменения в будущем по мере изменения требований бизнеса и совершенствования технологий. Структурная схема методики включает пять уровней, описанных в табл. 1.

Уровни описания архитектуры NASCIO

Уровни	Наименование	Назначение	Пример
1	Области (домены)	Области верхнего уровня, описывающие отдельные аспекты ИТ-систем	Информация
2	Дисциплины	Обеспечивают логическое деление доменов на разделы, которыми уже проще управлять и представляют собой связанные единицы в рамках соответствующей предметной области	Управление данными
3	Технологические дисциплины	дисциплины, поддерживающие функциональные технологические разделы архитектуры	Реляционные СУБД Плоские файловые системы Настольные БД Модели данных
4	Продуктовые компоненты	протоколы, продукты (семейства продуктов) и конфигурации, которые специфичны для каждой технологической дисциплины	MS SQL, Oracle, DB2 --- MS Access ER Win, MS Visio, Designer 2000
5	Документы соответствия	определяют руководства, стандарты и регулирующие документы, которые связаны с дисциплинами, технологическими дисциплинами и/или продуктовыми компонентами	стандарты предприятия по именованию хранимых процедур, по именованию таблиц и атрибутов, по защите БД; квоты на использование общего дискового пространства

Вся ИТ-архитектура подразделяется на набор областей (доменов) верхнего уровня, описывающих отдельные аспекты ИТ-систем, которые являются логическими блоками технологической архитектуры. Каждая область может включать одну и более дисциплин. Каждая дисциплина содержит одну и более технологических дисциплин. Каждой технологической дисциплине соответствует набор продуктовых компонент. С этого уровня начинается описание технических деталей технологической архитектуры. Документация для каждого компонента включает оценочные критерии, которые были использованы для включения продуктового компонента в общую технологическую архитектуру. Документы соответствия предписывают необходимость соблюдения тех или иных международных рекомендаций, стандартов, законодательных актов. Документы соответствия могут присутствовать на каждом из пяти уровней. Они обеспечивают основу для принятия важных решений о новых продуктах, протоколах, конфигурациях и т.д.

Важным преимуществом описанного подхода является возможность представления всего описания архитектуры в виде единой «гипертекстовой» базы

данных, что позволяет эффективно организовать процессы управления жизненным циклом отдельных документов, а также эффективно разграничить права доступа к отдельным разделам (например, документам, описывающим применяемые средства защиты информации) при сохранении целостности и единства описания.

Набор шаблонов NASCIO включает пять взаимосвязанных архитектур:

- архитектуру управления;
- бизнес-архитектуру;
- информационную архитектуру;
- технологическую архитектуру;
- архитектуру решений (Solution Architecture), обусловившую частичное изменение структуры модели.

Новым понятием в наборе шаблонов IT Architecture Toolkit является архитектура решений, облегчающая описание ее элементов в пределах предприятия посредством определенного перечня предлагаемых архитектору направлений при формулировании требований к решениям, спецификации проекта и логическим моделям проекта. Фактически данная архитектура представляет слой шаблонов из модели Gartner. В отличие от остальных архитектур, описания элементов архитектуры решений не содержат такого признака, как «существующий» или «целевой». Фактически все эти «решения» относятся как раз к стадии перехода от существующего состояния к целевому, так что по завершении соответствующего проекта они сохраняются со статусом «архивный». Для описания конкретного решения используются три типа шаблонов:

- «область» — тип шаблона, описывающий полное решение и связь решения с планом выполнения. Данный шаблон определяет концептуальную модель решения;

- «требования» — тип шаблона, содержащий формализованные требования к наборам решений, сгруппированных по типам. Организация шаблона схожа с ГОСТ 34.698-9 «Техническое задание на АС»;

- «дизайн» — тип шаблона, предоставляющий списки различных спецификаций проекта, основанные на определенных типах, представлениях и категориях.

Эти шаблоны содержат информацию, позволяющую оценить воздействия решений на состояние внешней среды в различных областях на определенный момент времени. Наряду с описанием элементов архитектуры в ходе процесса разработки определяется реализация стандартных процессов поддержки жизненного цикла архитектуры применительно к конкретным особенностям предприятия.

Модель представления архитектуры «4+1» (The 4+1 View Model of Architecture) Предложенная Ф. Кручтенем (Philippe Kruchten) из компании Rational в 1995 г. первоначально предназначалась для описания архитектуры систем, основанных на активном использовании программного обеспечения. Представляет собой простой и понятный способ описания архитектуры сложных систем, который состоит в использовании четырех основных представлений и объединяющего их сценария (рис. 17).

Логическое представление описывает назначение системы и функциональные требования в терминах конечных пользователей. Для этого представления используются различные абстрактные конструкции, такие как объекты и классы объектов, если используется объектно-ориентированная модель проектирования. Для их иллюстрирования могут применяться диаграммы классов в нотации языка UML либо, например, диаграммы «сущность-связь», если в разработке приложения доминируют данные.

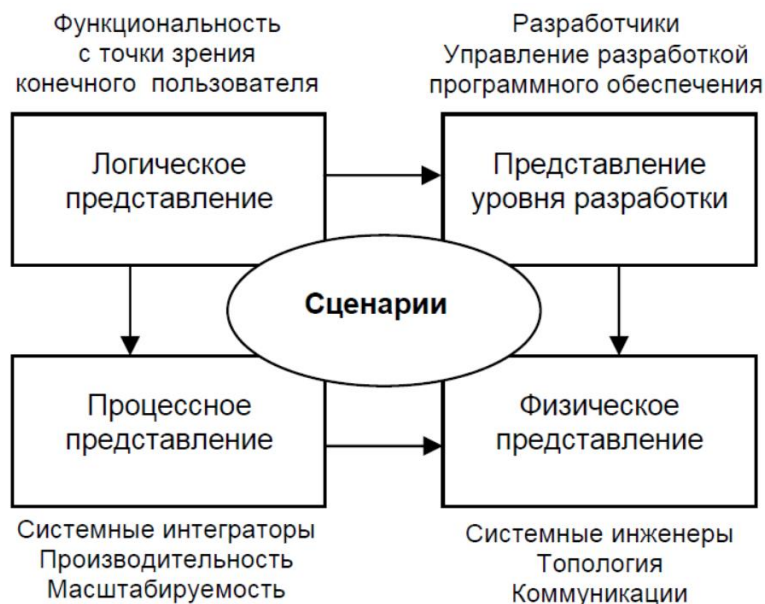


Рис. 17. Модель представления архитектуры «4 + 1»

Процессное представление описывает вопросы параллельного исполнения и синхронизации процессов. В нем учитываются некоторые нефункциональные требования к системе, включающие производительность и доступность. С помощью этого представления рассматриваются такие аспекты, как одновременное выполнение и распределение процессов, интеграция системы, устойчивость к сбоям, а также соответствие основных объектов абстракции, рассмотренных на уровне логического представления, архитектуре процессов.

Физическое представление описывает размещение программных компонентов системы на аппаратных платформах и аспекты, связанные с физическим расположением системы. В основном рассматриваются нефункциональные требования, такие как доступность, надежность, устойчивость, производительность, масштабируемость. Этот уровень описывает распределение элементов — сетей, процессов, задач и объектов — по различным узлам (элементам аппаратного обеспечения, объединенным в сеть).

Представление уровня разработки описывает статическую организацию программной системы в среде разработки, фактическую организацию модулей системы, разделение ее на подсистемы, которые могут разрабатываться независимо.

Каждое представление отражает специфические аспекты моделируемой системы. Сценарии, сформированные при объединении этих представлений, иллюстрируют их взаимодействие. Использование сценариев позволяет проверить корректность и согласованность описания представлений.

Сценарии описываются как последовательность взаимодействия объектов и процессов. Они отражают наиболее существенные требования, которым должна удовлетворять система. Использование сценариев позволяет:

- идентифицировать элементы архитектуры, которые требуются для эффективно работающей системы;
- выполнять проверку работоспособности и полноты архитектуры;
- предоставлять иллюстрацию процесса построения архитектуры;
- формировать основу для проведения тестирования архитектурного прототипа.

Стратегическая модель архитектуры SAM (Strategic Architecture Model) и методика ее создания разработана английской консалтинговой компанией Systems Advisers Ltd. SAM использует нотацию «сфер интересов» для представления целостного набора фактов о предприятии и «отношений», которые связывают эти факты в некоторые группы, что обеспечивает практический, целостный взгляд на структуру и операции, выполняемые предприятием.

SAM можно рассматривать как некоторую надстройку над моделью архитектуры предприятия Захмана. Она предоставляет общие структуры для определения архитектуры и механизмы, позволяющие осуществлять организацию и анализ информации об архитектуре. Методика создания SAM использует итеративный подход для создания архитектуры, сочетающий элементы разработки «сверху–вниз» и «снизу–вверх».

SAM выделяет три категории сфер (рис. 18):

1. Стабильные, описывающие достаточно стабильные элементы бизнеса и представляющие фундаментальные структуры: бизнес-функции, данные, бизнес-компоненты и инфраструктуру.

2. Подвижные, описывающие действия предприятия с точки зрения организации бизнеса в настоящем и будущем: организация, бизнес-процессы, прикладные системы и технологии. Эти сферы могут находиться в процессе постоянных изменений, обеспечивая адекватную реакцию на изменение экономических и рыночных условий.

3. Динамичные, задающие направления бизнеса, рабочие программы, управление изменениями. Они описывают основные области, в которых работает предприятие, и усилия, направленные на достижение целей бизнеса посредством реализации связанных между собой проектов.

Данная классификация позволяет:

- понимать, какая часть архитектуры предприятия носит достаточно стабильный характер, а какая требует постоянных изменений;
- идентифицировать относительно стабильные области, для которых полезна разработка архитектурных шаблонов.

Сферы интересов» SAM позволяют легко систематизировать всю информацию, имеющую отношение к определенной области. Внутри каждой сферы сохраняется информация об определенной предметной области, что обеспечивает простоту ее сопровождения и извлечения. Обычно применяется одна или несколько иерархических структур, которая заполняется либо «снизу вверх» путем сбора относящейся к предметной области информации, и обобщается на более высоких уровнях, либо «сверху вниз» с постепенной декомпозицией на более мелкие детали. После того как некоторая пара сфер определена с достаточной степенью детализации, элементы, составляющие эти сферы, могут быть связаны так, чтобы представить существующие в реальности связи между объектами анализа. Это обеспечивает возможность оптимизации и улучшений в различных областях деятельности предприятия.

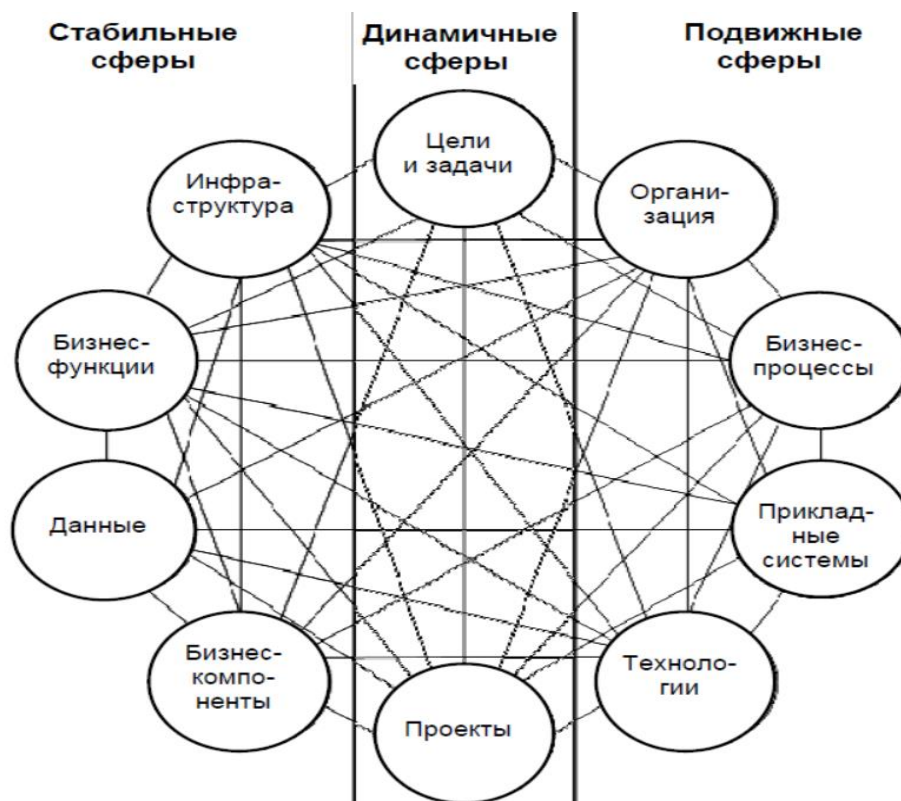


Рис. 18. Стратегическая модель архитектуры SAM

Минимальный объем информации, относящейся к какой-либо сфере, называется элементом (member). Например:

- конкретные подразделения («Отдел продаж», «ИТ-отдел» и др.) в сфере «Организация»;
- конкретные бизнес-процессы («Прием заказа», «Управление ИТ-активами» и др.) в сфере «Бизнес-процессы»;
- конкретные информационные объекты («Клиент», «Заказ», «Поставщик» и др.) в сфере «Данные»;
- география подразделений («Головной офис в Ярославле», «Офис продаж в Санкт-Петербурге», «Завод в Томске») в сфере «Местоположение».

Архитектурные концепции и методика Microsoft в большей степени сфокусированы на процессах разработки конкретных программных прикладных систем и создании технологической инфраструктуры, включая центры обработки данных различного масштаба и уровня надежности. Как и во многих других методиках, здесь выделяются четыре представления (домена): бизнес-архитектура; архитектура информации; прикладные системы; технологическая архитектура и рассматриваются на различных уровнях абстракции: концептуальном, логическом, физическом и уровне реализации. В идеале для каждой перспективы используется какой-то один тип моделей так, как это показано в табл. 2. Но в реальности могут использоваться и несколько различных моделей для описания каждого уровня абстракции.

Таблица 2

Архитектурные составляющие Microsoft

Уровень абстракции (перспектива)	Используемые модели	Домены архитектуры
Концептуальный уровень	Бизнес-модели	Бизнес-архитектура
Логический уровень	Модели приложений	Прикладные системы
	Модели данных	Архитектура информации
Физический уровень	Технологические модели	Технологическая архитектура
Уровень реализации	Модели реализации	

Выработаны достаточно подробные методики, отражающие различные аспекты архитектуры, среди которых явно выделяются следующие области:

- процессы разработки прикладных систем;
- процессы создания инфраструктуры;
- организация процессов эксплуатации систем и технологической инфраструктуры;
- создание соответствующих шаблонов, которые могут использоваться как при проектировании архитектуры систем, так и при ее построении.

В табл. 3 перечислены эти методики, они содержат рекомендации для специалистов, касающиеся следующих основных вопросов (назначение).

Таблица 3

Области охвата архитектурными представлениями Microsoft

Охват	Методика	Назначение	Фазы ЖЦ ИТ-решений
Процессы разработки прикладных систем и архитектуры	Microsoft Solutions Framework (MSF),	Как правильно создавать ИТ-системы	Проектирование Разработка
	Microsoft Systems Architecture (MSA)	Как правильно создавать технологическую инфраструктуру	Внедрение Адаптация

Охват	Методика	Назначение	Фазы ЖЦ ИТ-решений
Вопросы управления и эксплуатации т.е. архитектура системного управления	Microsoft Operations Framework (MOF)	Как правильно эксплуатировать технологическую инфраструктуру	Эксплуатация Техническая поддержка Изменения
	Microsoft Solutions for Management (MSM)	Как правильно строить процессы управления технологической инфраструктурой	Оптимизация

Методики Microsoft сосредоточены в основном на системном уровне — уровне архитектуры прикладных систем и обеспечивающей инфраструктуры. На рис. 19 отображено соответствие между методиками Microsoft и элементами архитектуры предприятия, а также используемыми шаблонами проектирования.

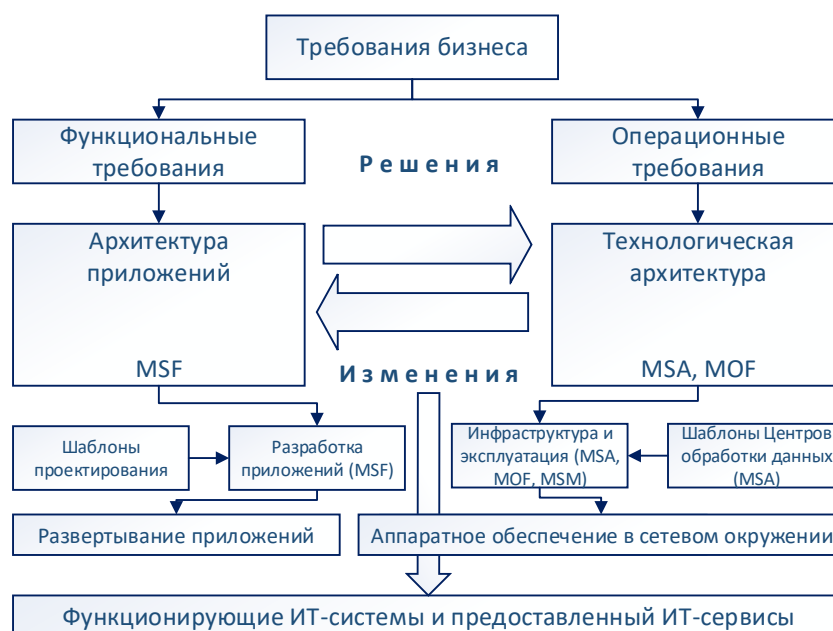


Рис. 19. Организация взаимосвязи компонентов архитектуры предприятия с использованием методик Microsoft

Microsoft выделяет два типа руководств и обеспечивающих методик, которые могут помочь системным архитекторам ускорить процессы разработки моделей при минимизации рисков.

Первый тип руководств — это архитектурные концепции, такие, например, как сервис-ориентированные подходы к проектированию архитектуры. Эти концепции обеспечивают:

- общее понимание и язык описания архитектуры;
- общие руководства, рекомендации по использованию специфических концепций;
- указания способов реализации концепции на практике в форме конкретных технологий и стандартов.

Второй тип руководств — это архитектурные шаблоны, основанные на практическом опыте большого количества успешно реализованных проектов создания распределенных прикладных систем и являющиеся следствием использования вышеописанных архитектурных концепций. Эти шаблоны содержат в себе лучшие практики проектирования распределенных приложений и средства по минимизации рисков неудач проектов, поскольку рекомендуют хорошо апробированные модели.

Использование данных концепций и шаблонов является важным условием успешного, быстрого и эффективного с точки зрения затрат создания систем и использования ИТ-технологий организациями. Поэтому, помимо методик MSF, MOF, MSA и MSM, компанией опубликованы подробные руководства по разработке архитектуры систем, а также шаблоны, которые могут применяться при проектировании корпоративных информационных систем.

3.3. Метод планирования архитектуры организации EAP

Один из признанных авторитетов в области архитектуры предприятия Стивен Спивак разработал модель планирования архитектуры предприятия, которая называется EAP (Enterprise Architecture Planning — планирование архитектуры предприятия), основу которого составляют:

- процесс планирования архитектуры предприятия, ориентированный на создание архитектуры, обеспечивающей поддержку бизнеса на основе учета конкретных данных, приложений и технологий, наиболее полно отвечающих потребностям предприятия;
- разработка плана реализации, определяющего процесс воплощения этой архитектуры. При этом предполагается, что созданию архитектуры предшествует разработка бизнес-стратегии, включающей миссию, бизнес-цели и способы их достижения.

EAP декларирует 10 этапов, определяющих состав и структуру слоев и элементов архитектуры, а также план ее проектирования, обеспечивающий реализацию как традиционных требований к архитектуре, так и специфических требований конкретной организации. На каждом из этапов выполняется от трех до девяти шагов, подробно описанных в табл. 4–13. Этапы организованы в виде четырехуровневой схемы метода EAP (рис. 20).

Первому уровню (исходная позиция) соответствует 1 этап «Инициация планирования» (табл. 4) — запуск планирования, выработка решений, которые необходимо принять для реализации соответствующей архитектуры предприятия, и определение состава необходимого для реализации инструментария, адаптация методологии, всего шесть шагов. Результатами этапа являются: цели, видение, методологии, инструментарий, команда, презентации, рабочий план.

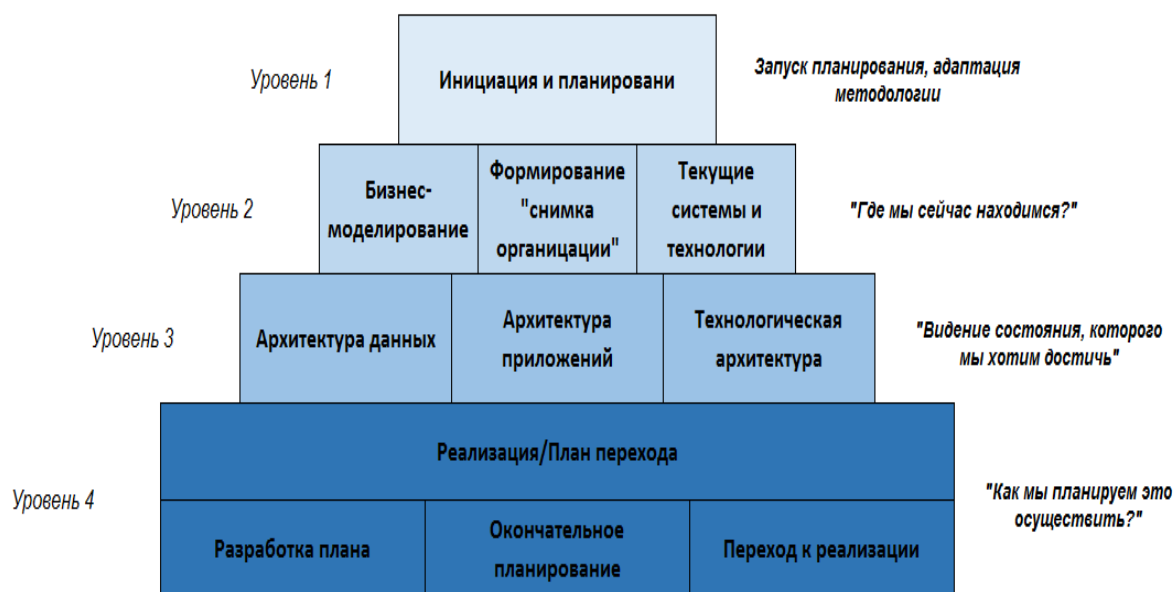


Рис. 20. EAP — Enterprise Architecture Planning (метод Спивака)

Таблица 4

Этап I — Инициация планирования

№ шага	Назначение	Задачи	Результат
1	Формальное определение области и целей планирования архитектуры для понимания участниками проекта конечных достигаемых результатов	<ul style="list-style-type: none"> • обзор организации и определение ее контекста (системных входов/выходов); • оценка характеристик деятельности предприятия по степени их влияния на реализацию проекта; • формирование перечня строго сформулированных целей и их достижимости; • формирование перечня подразделений, затрагиваемых грядущими изменениями ИТ-стратегии и корпоративной культуры 	<ul style="list-style-type: none"> • перечень согласованных и утвержденных целей, • список причастных к проекту подразделений организации
2	Исследование организации, системных входов/выходов и вариантов представлений менеджеров	<ul style="list-style-type: none"> • изучение всех исходных материалов по бизнесу (заказчики, продукты, сотрудники, цели и т.д.); • определение лиц, заинтересованных в создании архитектуры; • анализ опыта организаций, успешно выстроивших собственные архитектуры; • формирование видения организации в разрезе ИТ-среды, обеспечивающей достижение целей 	<ul style="list-style-type: none"> • согласованное и утвержденное видение организации, обеспечивающее политическую поддержку менеджмента
3	Адаптация методологии планирования	<ul style="list-style-type: none"> • формулирование принципов и требований к методологии; • оценка существующих на предприятии методов и стандартов; • изучение имеющихся на рынке подходов; 	<ul style="list-style-type: none"> • руководство по реализации адаптированной методологии планирования

№ шага	Назначение	Задачи	Результат
		<ul style="list-style-type: none"> • принятие решения об исполнителе (внутренние ресурсы или внешний консультант); • создание методологии, отвечающей потребностям данного предприятия; • разработка содержания отчетов, создаваемых на каждом из последующих этапов 	
4	Инвентаризация компьютерных ресурсов и оценка инструментария создания архитектуры предприятия.	<ul style="list-style-type: none"> • определение требований к инструментарию; • определение требований к аппаратуре; • оценка альтернатив для репозитория проекта; • выбор и приобретение подходящего программного инструментария; • разработка регламентов и процедур, обеспечивающих надлежащее использование продуктов; • разработка проектов отчетов, экранных форм и т. п.; • оценка трудозатрат на «канцелярскую» поддержку большого объема документации по архитектуре предприятия; • доведение решений по инструментарию до всех подразделений — потенциальных пользователей архитектуры 	<ul style="list-style-type: none"> • репозитарий инструментария создания архитектуры предприятия • альбом экранных форм, печатных форм, отчетов
5	Формирование проектной команды	<ul style="list-style-type: none"> • определение квалификационных требований по каждому этапу создания архитектуры; • оценка трудозатрат по каждому этапу создания архитектуры; • определение необходимого числа участников; • спецификация ролей и областей ответственности каждого члена команды; • подбор персонала; • обучение персонала (методологии и инструментарий); • выбор внешних консультантов и определение направлений их использования 	<ul style="list-style-type: none"> • приказ о создании проектной команды • должностные инструкции членов команды
6	Создание рабочего плана	<ul style="list-style-type: none"> • подготовка рабочего плана; • презентация и обсуждение плана; • утверждение плана 	<ul style="list-style-type: none"> • утвержденный рабочий план • утвержденный бюджет на реализацию проекта

Второй уровень (анализ текущего состояния) — определение точки отсчета для преобразования существующей архитектуры в целевую, а также формирование предварительного графика перехода, позволяет ответить на вопрос: где мы сейчас находимся? Состоит из трех этапов.

2 этап «Предварительное бизнес-моделирование», состоящий из трех шагов (табл. 5), результатом которого являются организационно-штатная структура и предварительная функциональная бизнес-модель предприятия.

Таблица 5

Этап II — Предварительное бизнес-моделирование

№ шага	Назначение	Задачи	Результат
1	Документирование организационной структуры	<ul style="list-style-type: none"> • формирование (редактирование) организационных схем; • и фиксация их в инструментарии; • идентификация видов деятельности в разрезе организационных единиц; • формирование отчетов по полученным результатам 	<ul style="list-style-type: none"> • обновленные организационные схемы; • перечень ролей и мест их выполнения; • оценку количества; • сотрудников по ролям
2	Определение структуры бизнес-модели (идентификация и определение бизнес-функций)	<ul style="list-style-type: none"> • определение основных видов деятельности; • выделение и классификация бизнес-процессов; • описание бизнес-процессов верхнего уровня; • построение функционального иерархического дерева; • сопоставление функций и исполняющих их организационных единиц, построение соответствующей матрицы 	<ul style="list-style-type: none"> • документация по бизнес-процессам верхнего уровня, включающая наименование, владельца, краткое описание, принадлежность к конкретной организационной единице; • матрица организационных проекций
3	Документирование бизнес-модели и ее распространение для верификации	<ul style="list-style-type: none"> • формирование отчетов по бизнес-модели; • распространение отчетов и проведение презентации; • сбор замечаний и предложений 	<ul style="list-style-type: none"> • утвержденная обобщенная бизнес-модель организации

3 этап «Формирование снимка организации», состоящий из трех шагов (табл. 6), результат — полная функциональная бизнес-модель.

4 этап «Описание текущих систем и технологий», состоящий из четырех шагов (табл. 7), результат — каталог информационных ресурсов и системные схемы. Целью этапа «Описание текущих систем и технологий» является документирование всех используемых в организации системных и технологических платформ, т. е. создание так называемого каталога информационных ресурсов

IRC (Information Resource Catalog), по-другому — системной энциклопедии, являющейся высокоуровневым объектом (а не детальным словарем данных).

Таблица 6

Этап III — Формирование снимка организации

№ шага	Назначение	Задачи	Результат
1	Планирование, подготовка и проведение интервью	<ul style="list-style-type: none"> • формирование списка интервьюируемых с указанием даты и времени проведения; • распределение интервьюирующих по видам деятельности и бизнес-процессам (функциональным направлениям); • подготовка инструкции для участников (цели и задачи, кто, когда, где, какие вопросы и др.); • анализ бизнес-модели 	<ul style="list-style-type: none"> • формы для интервьюирования и фиксации результатов; • форм для управления процессом интервьюирования; • результаты интервьюирования
2	Построение полной бизнес-модели	<ul style="list-style-type: none"> • функциональная декомпозиция процессов; • развитие функциональной декомпозиции до уровня бизнес-операций 	<p>Полная функциональная бизнес-модель дает ответы на следующие вопросы:</p> <ul style="list-style-type: none"> • какая информация используется при выполнении функций; • когда функция выполняется; • где и кем функция выполняется; • как часто функция выполняется; • какие улучшения возможны
3	Анализ бизнес-модели	<ul style="list-style-type: none"> • оценка качества декомпозиции и ее улучшение 	

Таблица 7

Этап IV — Описание текущих систем и технологий

№ шага	Назначение	Задачи	Результат
1	Определение видов данных для IRC и проектирование форм для сбора данных	<ul style="list-style-type: none"> • определение видов данных по приложениям; • определение видов данных по входам, выходам, файлам и БД приложений; • идентификацию технологических платформ и определение их декомпозиции по видам (например, принтеры — матричные, лазерные; языки — Си, Паскаль и т. п.); • проектирование форм для сбора данных; • подготовку детальных инструкций по заполнению форм 	<ul style="list-style-type: none"> • реестр видов данных; • реестр технологических платформ; • шаблоны форм для сбора данных; • инструкции по заполнению форм

№ шага	Назначение	Задачи	Результат
2	Сбор данных для IRC и их ввод (заполнение форм), а также сопоставление приложений и функций	<ul style="list-style-type: none"> • сбор системной документации; • ввод информации в инструментарий • сопоставление приложений и бизнес-функций и формирование соответствующей матрицы; • сопоставление приложений и технологических платформ, формирование соответствующей матрицы 	<ul style="list-style-type: none"> • заполненные формы; • матрица «Приложения — бизнес-функции»; • матрица «Приложения — платформы»
3	Интегрирование и верификация информации по текущим приложениям и технологическим платформам	<ul style="list-style-type: none"> • разработка потоковых диаграмм по каждому приложению; • проверка на соответствие требованиями собранной информации; • обсуждение результатов текущего состояния дел 	<ul style="list-style-type: none"> • верифицированный IRC; • пакет отчетов по IRC; • предложения по улучшению IRC на основе проведенных обсуждений
4	Подготовка к процессам администрирования и сопровождения IRC	<ul style="list-style-type: none"> • разработка политики, регламентов и процедур поддержания каталога информационных ресурсов в актуальном состоянии 	<ul style="list-style-type: none"> • утвержденные политики и регламенты; • инструкции для ввода информации в IRC

Третий уровень (планируемая перспектива) — определение технических деталей перспективной архитектуры (данные, приложения и технологии), формирует видение состояния, которого мы хотим достичь. Состоит из трех этапов.

5 этап «Формирование архитектуры данных», состоящий из четырех шагов (табл. 8), на нем определяются и идентифицируются основные разновидности данных, поддерживающих бизнес-функции; результаты: определения сущностей, ER-модель, матрица сущности-функции, отчет по архитектуре данных.

Таблица 8

Этап V — Формирование архитектуры данных

№ шага	Назначение	Задачи	Результат
1	Формирование списка кандидатов в сущности	<ul style="list-style-type: none"> • распределение компонентов бизнес-модели по членам команды в разрезе видов деятельности и бизнес-процессов; • подготовка каждым из участников списка кандидатов в сущности; • формирование общего списка кандидатов в сущности 	<ul style="list-style-type: none"> • идентификация всех потенциальных сущностей, необходимых для поддержки бизнеса
2	Определение сущностей, атрибутов и отношений	<ul style="list-style-type: none"> • определение и документирование сущностей; • построение ER-модели; • сопоставление файлов и БД из IRC с сущностями 	<ul style="list-style-type: none"> • стандартизованное описание сущностей; • графическая иллюстрация взаимодействия сущностей

№ шага	Назначение	Задачи	Результат
3	Сопоставление сущностей с бизнес-функциями и приложениями	<ul style="list-style-type: none"> каждой функции нижнего уровня детализации идентифицируется вид каждой из затрагиваемых ей сущностей (создается, изменяется, используется); приложения сопоставляются с сущностями по входам, выходам, файлам и БД 	<ul style="list-style-type: none"> матрица «Сущности — бизнес-функции»; матрица «Сущности — приложения»
4	Анализ результатов	<ul style="list-style-type: none"> подготовка отчета по архитектуре данных; анализ текущей архитектуры данных; распространение отчета и результатов анализа 	<ul style="list-style-type: none"> сформированная архитектура данных; рекомендации по развитию архитектуры данных

6 этап «Формирование архитектуры приложений», состоящий из пяти шагов (табл. 9), определяют основные виды приложений, необходимых для управления данными и поддержки бизнес-функций; результаты: определения приложений, матрицы приложений, анализ покрытия, отчет по архитектуре приложений. Архитектура приложений не является ни системным проектом, ни набором детальных требований к системам. Она только определяет, какие приложения будут управлять данными, и снабжает соответствующей информацией исполнителей бизнес-функций.

Таблица 9

Этап VI — Формирование архитектуры приложений

№ шага	Назначение	Задачи	Результат
1	Формирование списка потенциальных приложений	<ul style="list-style-type: none"> идентификация каждого из возможных приложений; формирование списка приложений с описанием назначения и целей использования 	<ul style="list-style-type: none"> реестр приложений, которые могут улучшить бизнес или обеспечить конкурентные преимущества
2	Определение приложений	<ul style="list-style-type: none"> распределение приложений между членами команды; определение состава каждого приложения (имя, номер, цель, общее описание и возможности, бизнес-преимущества); упрощение сложных приложений и ликвидация избыточности; выработка предварительных предложений по применимости имеющегося на рынке ПО и технологических платформ; построение схемы архитектуры приложений; оценка качества архитектуры приложений (понимаемость, полнота и состоятельность, прочность-устойчивость) 	<ul style="list-style-type: none"> стандартизованное описание приложений; графическая схема архитектуры приложений

№ шага	Назначение	Задачи	Результат
3	Сопоставление приложений и функций	<ul style="list-style-type: none"> • идентификация бизнес-функций, поддерживаемых приложениями; • идентификация бизнес-функций, не поддерживаемых приложениями 	<ul style="list-style-type: none"> • уточненная матрица «Приложения — бизнес-функции»; • матрица «приложения — организационные единицы»
4	Определение соответствия архитектуры приложений существующим в организации приложениям	<ul style="list-style-type: none"> • сопоставление каждого приложения из архитектуры приложений и имеющихся систем, определенных в IRC, • контроль полноты сопоставления (каждое существующее приложение из IRC должно быть соотнесено, по крайней мере, с одним из архитектурных приложений) 	<ul style="list-style-type: none"> • таблица соответствий архитектуры приложений и существующих приложений
5	Анализ результатов	<ul style="list-style-type: none"> • подготовка отчета по архитектуре приложений; • анализ текущей архитектуры приложений; • распространение отчета и результатов анализа 	<ul style="list-style-type: none"> • сформированная архитектура приложений; • рекомендации по развитию архитектуры приложений

7 этап «Формирование технологической архитектуры», состоящий из четырех шагов (табл. 10), определяет основные виды технологий, необходимых для обеспечения окружения приложений, управляющих данными; результаты: распределение данных/приложений, отчет по технологической архитектуре. Технологическая архитектура не является ни проектом сетевого оборудования и ПО, ни набором детальных требований к ним; она только определяет виды технических платформ, поддерживающих бизнес.

Таблица 10

Этап VII — Формирование технологической архитектуры

№ шага	Назначение	Задачи	Результат
1	Идентификация технических принципов и платформ	<ul style="list-style-type: none"> • формулирование общих принципов для технических платформ; • идентификация потенциальных кандидатов в платформы 	<ul style="list-style-type: none"> • реестр потенциальных кандидатов в платформы; • стратегия распределения приложений и данных, технических платформ
2	Определение платформ и их распределение	<ul style="list-style-type: none"> • определение мест размещения бизнес-функций; • распределение данных и приложений; • определение конфигурации технологических платформ (рабочих станций, сети, архитектуры бизнес-систем) • оценка концептуальной технологической архитектуры 	<ul style="list-style-type: none"> • описание распределения данных и приложений; • схема конфигурации технологических платформ; • предложения по развитию концептуальной архитектуры

№ шага	Назначение	Задачи	Результат
3	Сопоставление платформ с приложениями и бизнес-функциями	<ul style="list-style-type: none"> • обоснование технологических платформ путем их соотнесения с используемыми бизнес-функциями; 	<ul style="list-style-type: none"> • уточненная матрица «Платформы — бизнес-функции»; • матрица «Платформы — приложения»
4	анализ результатов	<ul style="list-style-type: none"> • подготовка отчета по технической архитектуре; • анализ текущей технической архитектуры; • распространение отчета и результатов анализа 	<ul style="list-style-type: none"> • сформированная технологическая архитектура; • рекомендации по развитию технологической архитектуры

Четвертый уровень (формирование плана реализации перспективной архитектуры) — позволяет ответить на вопрос: как мы планируем это осуществить? Состоит из трех этапов.

8 этап «Разработка плана реализации», состоящий из четырех шагов (табл. 11), результаты: последовательность, план перехода, цены и преимущества, факторы успеха и рекомендации.

Таблица 11

Этап VIII — Разработка плана реализации

№ шага	Назначение	Задачи	Результат
1	Формирование последовательности реализации приложений	<ul style="list-style-type: none"> • установка приоритетов и формирование последовательности реализации приложений; • сопоставление приложений и сущностей на основе бизнес-функций; • преобразование матрицы приложений, содержащих сущности данных, к виду, позволяющему установить последовательность реализации, определяемую данными с помощью соответствующей оптимизационной процедуры 	<ul style="list-style-type: none"> • матрица приложений, содержащих сущности данных; • список упорядоченных по приоритетам приложений
2	Оценка затрат и ресурсов, построение плана	<ul style="list-style-type: none"> • формирование критериев (количественных и качественных) последовательности реализации; • формирование последовательности модификации существующих систем и приобретения технологий 	<ul style="list-style-type: none"> • план модификации и/или замены существующих систем; • группировка приложений в проекты • последовательность реализации технологий

№ шага	Назначение	Задачи	Результат
3	Оценка стоимости и достоинств плана	<ul style="list-style-type: none"> • обоснование плана модификации и/или замены существующих систем/приложений 	<ul style="list-style-type: none"> • оценка стоимости проектов; • оценка преимуществ проектов; • технико-экономическое обоснование проектов
4	Определение факторов успеха и рекомендаций по их достижению	<ul style="list-style-type: none"> • выявление КФУ реализации проектов; • разработка рекомендаций по обеспечению КФУ 	<ul style="list-style-type: none"> • реестр КФУ; • рекомендации по достижению КФУ

9 этап «Заключительное планирование», состоящий из двух шагов (табл. 12), результаты: окончательный отчет, презентация.

Таблица 12

Этап IX — Заключительное планирование

№ шага	Назначение	Задачи	Результат
1	Формирование окончательного отчета по архитектуре предприятия	<ul style="list-style-type: none"> • обсуждение и утверждение архитектуры предприятия; • обсуждение и утверждение плана реализации архитектуры предприятия; • подготовка окончательного отчета по архитектуре предприятия и плана реализации 	<ul style="list-style-type: none"> • окончательный отчет по архитектуре предприятия
2	Формирование полного представления об архитектуре предприятия	<ul style="list-style-type: none"> • подготовка и проведение презентации по архитектуре предприятия; 	<ul style="list-style-type: none"> • презентации по архитектуре предприятия

10 этап «Переход к реализации», состоящий из семи шагов (табл. 13), результаты: совершенствование политик, стандартов, процедур, детализация проектных планов.

Таблица 13

Этап X — Переход к реализации

№ шага	Назначение	Задачи	Результат
1	Планирование перехода	<ul style="list-style-type: none"> • спецификация целей перехода; • формирование плана перехода; • назначение ответственности за переход; • определение руководителя-лидера 	<ul style="list-style-type: none"> • окончательный план перехода

№ шага	Назначение	Задачи	Результат
2	Адаптация подхода к реализации АП	<ul style="list-style-type: none"> • методологии перехода на новую архитектуру • инструменты реализации АП 	• адаптированные методологии совершенствования АП
3	Инвентаризация компьютерных ресурсов	<ul style="list-style-type: none"> • анализ существующего оборудования; • приобретение необходимого оборудования; • обеспечение надежности хранилища данных 	• полная обеспеченность оборудованием
4	Чистка архитектуры	<ul style="list-style-type: none"> • ревизия, добавление деталей и обновление 	• окончательная архитектура с учетом возможностей и ограничений
5	Организация и персонал	<ul style="list-style-type: none"> • изменение организационно-штатной структуры; • набор персонала; • проведение обучения 	• организованный и обученный персонал
6	Совершенствование политик	<ul style="list-style-type: none"> • введение стандартов на программирование; • введение процедурных стандартов 	<ul style="list-style-type: none"> • установленные стандарты; • регламентированные процедуры
7	Завершение реализации	<ul style="list-style-type: none"> • разработка детальных планов по приложениям; • определение и утверждение даты завершения перехода 	• реализованная АП

Если «наложить» метод ЕАР Спивака на модель Захмана, то можно сказать, что метод ЕАР является руководством по заполнению первых двух строк таблицы Захмана, которые описывают контекст архитектуры и концептуальную модель бизнеса предприятия. Проектирование систем, которое начинается с третьей строки таблицы Захмана, остается за рамками метода Спивака.

3.4. Построение архитектуры предприятия с использованием методологии ARIS

ARIS (англ. *Architecture of Integrated Information Systems*) — методология и программная платформа моделирования организации и ее бизнес-процессов, разработанная компанией IDS Scheer, первоначально предназначена для разработки и описания архитектуры интегрированных информационных систем. Автором идеи является Август-Вильгельм Шеер, профессор Института информационных систем при Университете Саарбрюкена, который в 1984 г. создал инновационную фирму по разработке программного обеспечения для моделирования бизнес-процессов. В результате появился инструментарий, существенно отличающийся от других подходов к автоматизированному моделированию процессов и объектов организации. Основные отличия ARIS:

- платформа ARIS основана на концепции интеграции различных стандартов описания деятельности организации;

- организация рассматривается в разных аспектах;
- использует множество типов моделей: оргструктуры, функции, данные, процессы и позволяет отражать их в единой интегрированной модели предприятия;
- имеется единый репозиторий (хранилище данных) всевозможных моделей организации;
- пользователи могут иметь разный уровень подготовки в области ИТ;
- служит базой для управления сложными проектами, поскольку благодаря структурным элементам содержит встроенные модели процедур для разработки интегрированных информационных систем.

Все это позволило использовать ARIS в качестве мощного инструмента разработки и описания архитектуры предприятия. ARIS используется в SAP R/3 — одной из самых распространенных в мире информационных систем управления ресурсами предприятия (ERP).

В методологии выделено четыре основных вида моделей (четыре представления), представляемых в виде «домика ARIS» (рис. 21), отражающих различные аспекты исследуемой системы:

- организационные модели (КТО?);
- функциональные модели (ЧТО?);
- информационные модели (НА ОСНОВЕ ЧЕГО?);
- модели процессов/управления (КАКИМ ОБРАЗОМ?).

Ряд авторов, изображая «домика ARIS», ставят его на фундамент модели, позволяющей ответить на вопрос ДЛЯ ЧЕГО?, которая включает в себя модель ресурсов и модель входов-выходов. Модель входов-выходов описывает что организация как система имеет на входе (сырье и материалы, пожелания клиентов, требования поставщиков и госорганов и т.п.) и что производит на выходе. Модель ресурсов описывает используемые в процессе преобразования входов в выходы технические ресурсы (оборудование, технологии и т.п.), материальные ресурсы, программно-алгоритмическое обеспечение, поддерживающее технологию и управление.

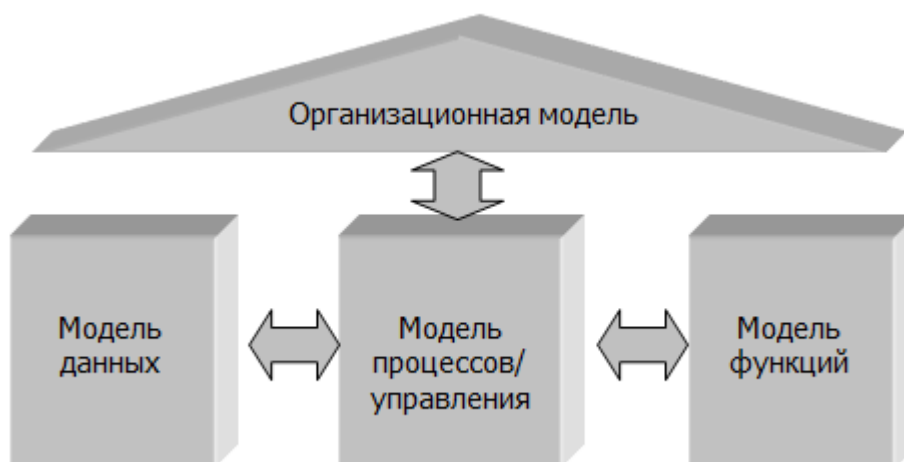


Рис. 21. Группы основных видов моделей ARIS

Для каждого из этих представлений можно построить несколько типов моделей, на основе которых с точки зрения каждого представления формулируются

требования к интегрированной информационной системе в целом, к ее компонентам, функционалу и т.д. Также, с точки зрения каждого представления могут быть разработаны спецификации проекта и даны описания реализации. Такое трехуровневое описание каждого из представлений обеспечивает целостность разрабатываемой системы. В ARIS Toolset используется трехфазовая модель жизненного цикла АИС, т.е. каждый из перечисленных аспектов имеет три уровня представления:

1. Уровень определения требований. На данном уровне разрабатываются модели, описывающие то, что должна делать система - как она организована, какие деловые процессы в ней присутствуют, какие данные при этом используются.

2. Уровень проектной спецификации. Этот уровень соответствует концепции информационной системы, определяющей основные пути реализации предъявленных на втором этапе требований.

3. Уровень описания реализации. На данном этапе жизненного цикла создания информационных систем происходит преобразование спецификации в физическое описание конкретных программных и технических средств. Это заключительный этап проектирования систем, за которым следует этап физической реализации (программирования). Уровень описания реализации порождает документы, на основе которых можно обеспечить процесс разработки программных модулей (или подбора готовых программных компонент, отвечающих поставленным требованиям), а также выбора и организации технических средств реализации системы.

Таким образом, ARIS представляет собой современный подход к структурированному описанию деятельности организации и представлению ее в виде взаимосвязанных и взаимодополняющих графических моделей, удобных для понимания и анализа. Методология основывается на концепции интеграции, предлагающей целостный взгляд на процессы, и представляет собой множество различных методик, объединенных в рамках единого системного подхода. Среди них такие известные как:

- функциональное моделирование;
- диаграммы eEPC (extended Event Driven Process Chain — событийная цепочка процесса);
- модели ERM (Entity Relationship Model — модель «сущность-связь»);
- UML (Unified Modeling Language — универсальный язык моделирования)
- OMT (Object Modeling Technique — методика объектно-ориентированного моделирования);
- ABC (Activity Based Costing — пооперационный расчет стоимости процессов);
- BSC (balanced scorecard — система сбалансированных показателей).

Несмотря на большее количество моделей в методологии ARIS в проектах по описанию и оптимизации деятельности наиболее часто используются модели, приведенные в таблице.

Наиболее часто используемые модели в методологии ARIS

Название модели		Вид и назначение модели
Английский вариант	Русский вариант	
OD-Objective diagram	Диаграмма целей	Иерархическая модель описывает стратегические цели компании и их взаимосвязь с другими элементами организации
PST-Product/Service tree	Дерево продуктов и услуг	Иерархическая модель описывает продукты и услуги, производимые компанией и их взаимосвязь с другими элементами организации
FT-Function tree	Дерево функций	Иерархическая модель описывает функции, выполняемые в компании и их иерархию
FAD-Function allocation diagram	Диаграмма окружения процесса	Процессная модель описывает окружение бизнес-процесса
VAD-Value added chain diagram	Диаграмма цепочки добавленной стоимости	Процессная модель, применяется для описания бизнес-процессов верхнего уровня
PSM — Process selection matrix	Матрица выбора процесса	Процессная модель, является альтернативой модели VAD и применяется для описания бизнес-процессов верхнего уровня
eEPC — Extended event driven Process Chain	Расширенная цепочка процессов, управляемая событиями	Процессная модель прототипом которого является классический стандарт WFD (Workflow Diagram). Применяется для описания бизнес-процессов нижнего уровня.
BPMN	Нотация моделирования бизнес-процессов	Процессная модель прототип классического стандарта IDF3. Применяется для описания бизнес-процессов нижнего уровня
ORG — Organizational chart	Организационная диаграмма	Иерархическая модель описывает организационную структуру компании
ASTD-Application system type diagram	Диаграмма типов информационных систем	Модель описывает структуру информационных систем, используемых в компании
IT infrastructure	ИТ-инфраструктура	Моделирует технические решения: какое ПО на каком аппаратном обеспечении работает, в какие сети объединено аппаратное обеспечение, какое сетевое оборудование используется
System landscape	Карта систем	Описывает, какие ИТ-системы принадлежат к какой логической единице (домену)

Рассмотрим наиболее подробно некоторые из этих моделей.

Организационная модель ARIS состоит из моделей, с помощью которых описывается организационная структура компании, а также другие элементы внутренней инфраструктуры организации. Основой этих моделей является организационная диаграмма. Объекты организационной диаграммы и их связи представляют структуру системы: иерархию организационных подразделений, должностей и конкретных лиц, многообразие связей между ними, а также территориальную привязку структурных подразделений. Основные типы объектов:



На рис. 22 продемонстрирован пример упрощенной организационной диаграммы. Модель строится иерархически — от верхнего уровня структуры к нижнему. Низшим уровнем является описание подразделений на уровне должностей — штатных единиц, занимаемых конкретными сотрудниками. На рис. 23 и 24 соответственно показано как с помощью организационной диаграммы смоделировать дивизиональную и матричную организационные структуры.

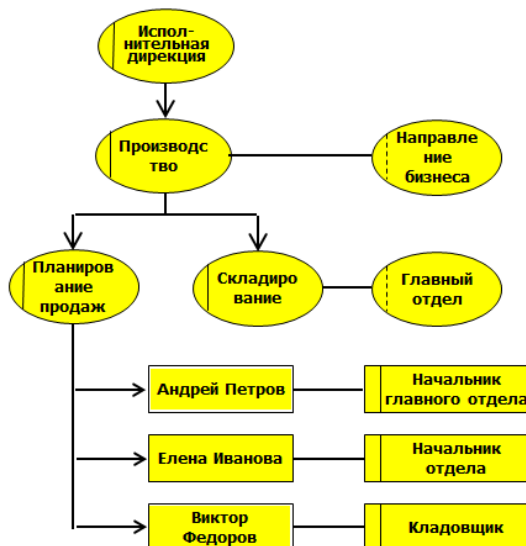


Рис. 22. Пример упрощенной организационной диаграммы

Моделирование и анализ организационной структуры должен проводиться с целью выявления:

- обоснованного количества уровней иерархии;
- наличие малого количества подчиненных у одного руководителя (норматив управляемости 5–7 объектов);
- наличие большого количества подчиненных подразделений у одного руководителя (несоблюдение норм управляемости);
- подчинения одних и тех же сотрудников различным руководителям.

В модели организационной структуры целесообразно показывать:

- подразделения предприятия и их взаимодействия;
- наименования должностей и фамилии ответственных лиц;
- физическое местоположение отделов на предприятии.

Всего методология ARIS содержит пять видов организационных моделей, сгруппированных по уровням представления. К первому уровню «определение требований» относятся организационная диаграмма (Organizational chart) и сменный график/календарь (Shift calendar). На втором уровне «спецификация проекта» моделируется топология сети (Network topology). Уровень «описание реализации» содержит диаграмму сети (Network diagram) и модель технических ресурсов (Technical resources).

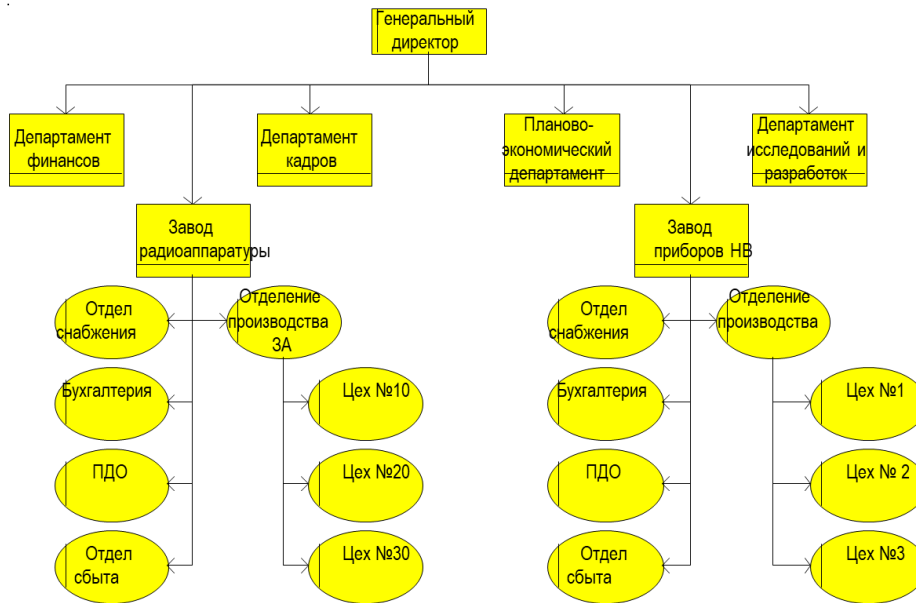


Рис. 23. Дивизиональная организационная структура

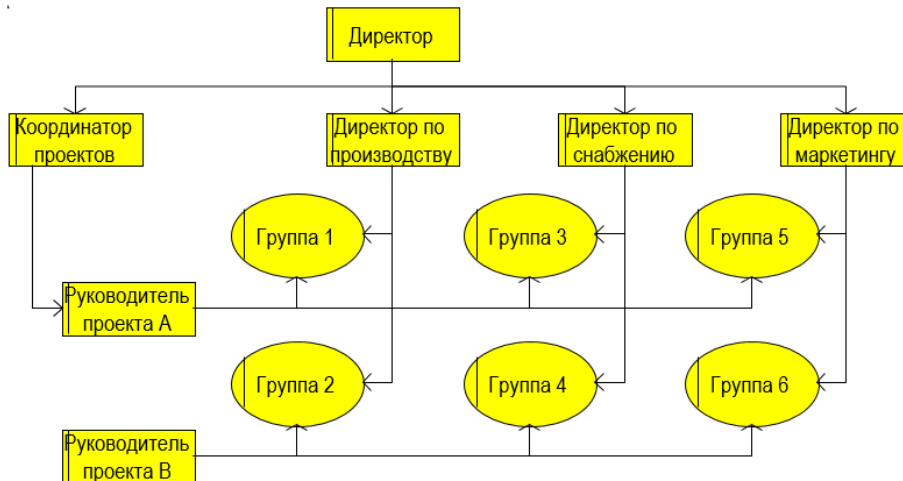


Рис. 24. Матричная организационная структура

Функциональная модель ARIS состоит из моделей, используемых для описания как стратегических, так и тактических целей организации, а также функций, выполняемых в организации и прочих элементов ее функциональной деятельности. В методологии представлено семь видов моделей:

I уровень — определение требований:

1. Диаграмма целей (Objective diagram).
2. Дерево функций (Function tree).
3. Модель функций SAP ALE (SAP ALE Function Model).
4. Диаграмма SAP приложений (SAP Applications Diagram).
5. Y-диаграмма (Y-Diagram).

II уровень — спецификации проекта:

6. Диаграмма типов прикладных систем (Applications System Type Diagram).

III уровень — описание реализации

7. Диаграмма прикладных систем (Applications System Diagram).

Основу функционального представления составляют первые две модели, содержащие иерархию целей, стоящих перед аппаратом управления, с совокупностью деревьев функций, необходимых для достижения поставленных целей.

Диаграмма целей строится в виде дерева целей, посредством построения которого описывается упорядоченная иерархия целей компании. Для этого осуществляется последовательная декомпозиция главной цели на подцели по следующим правилам:

- общая цель должна содержать описание конечного результата;
- при развертывании общей цели в иерархическую структуру исходят из того, что реализация подцелей каждого последующего уровня является необходимым и достаточным условием достижения целей предыдущего уровня;
- при формулировке целей разных уровней необходимо описывать желаемые результаты, а не способы их получения;
- подцели каждого уровня должны быть независимы друг от друга и невыводимы друг из друга;
- фундамент дерева целей должны составлять задачи, представляющие собой формулировку работ, которые могут быть выполнены определенным способом в установленные сроки.

В диаграмме используется только один тип объекта — цель. Группировка целей может проводиться по разным признакам. Методология ARIS поддерживает применение методики BSC в которой цели предприятия формулируются по четырем аспектам: финансы, клиенты и маркетинг, бизнес-процессы, персонал и системы. В соответствии с этим методология рекомендует на первом уровне иерархии выделять именно такие группы целей (рис. 25). В реальной ситуации дерево целей состоит из 5–8 уровней и имеет разветвленную структуру, в связи с этим, для его описания может использоваться совокупность диаграмм, каждая из которых содержит декомпозицию одной цели 2-го или 3-го уровней. На рис. 26 представлен пример такой декомпозиции для цели «Увеличение прибыли» в инструментальной среде ARIS.

Дерево функций (Function Tree) — второй основной элемент функциональной модели. Функция — описание элемента работы, образующего один логический этап в рамках процесса. В методологии ARIS используется диаграмма «Дерево функций», посредством которой функции могут быть описаны с различными уровнями детализации. При этом функции представляются не обязательно в хронологическом порядке. Используется только один тип объекта — функция (работа, действие, этап в рамках процесса).

На верхнем уровне дерева функций описываются наиболее сложные функции, представляющие собой отдельный бизнес-процесс или процедуру. Детализация функций образует иерархическую структуру их описаний.

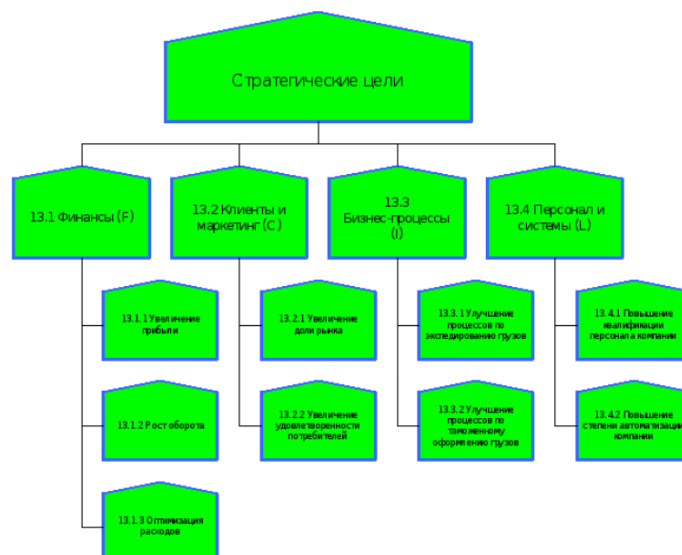


Рис. 25. Структура дерева целей

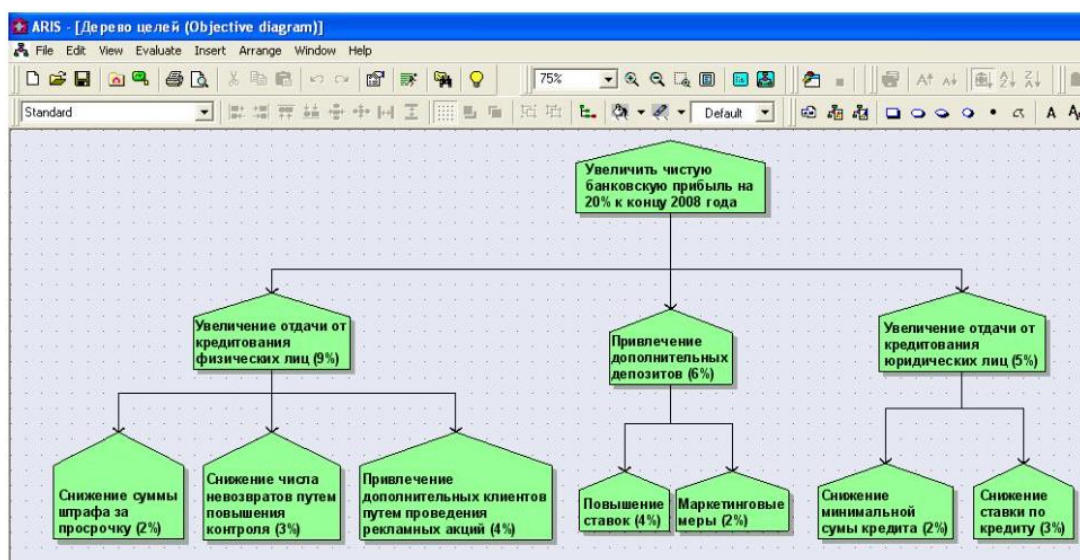


Рис. 26. Диаграмма целей, созданная в среде ARIS

Существуют разные принципы декомпозиции дерева функций:

- объектно-ориентированная подчиненность (Is object-oriented superior) — выделение функций, выполняемых над одним объектом. Эти функции описывают различные операции, выполняемые над одним и тем же объектом;
- процессно-ориентированная подчиненность (Is process-oriented superior) — выделение функций по принадлежности к одному процессу, то есть все выделенные функции являются этапами одного процесса. Критерием процессно-ориентированной детализации служат операции, которые выполняются над различными объектами в рамках бизнес-процесса;
- операционно-ориентированная подчиненность (Is execution-oriented superior) — при операционно-ориентированном подходе функция верхнего уровня декомпозируется на подфункции, каждая из которых выполняет ту же операцию,

но с различными объектами. Функции могут принадлежать различным процессам и привлекаться к обработке различных объектов. Однако выполняемый ими тип операции над отдельными объектами всегда один и тот же.

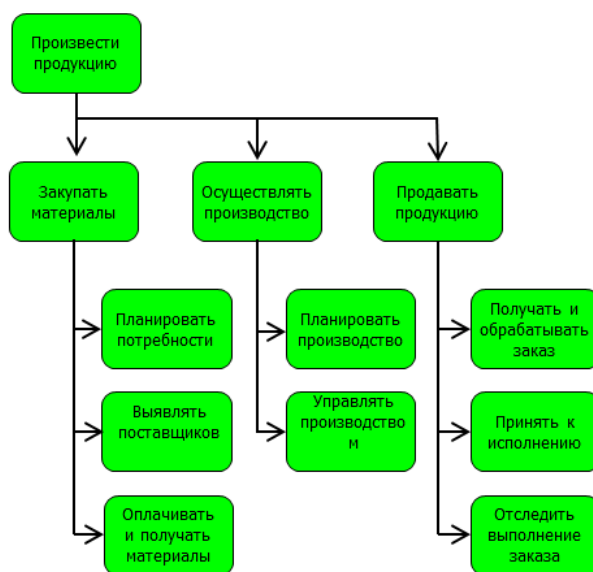


Рис. 27. Дерево функций

Самый нижний уровень дерева функций представляют базовые функции, которые уже не могут быть разделены на составные элементы.

Уровень спецификации проекта функциональной модели описывается с помощью диаграммы типов прикладных систем. Тип прикладной системы как вид объекта в рамках функциональной модели на этапе спецификации проекта и представляет собой обозначение группы прикладных систем, базирующихся на одной и той же технологии, в отличие от конкретной прикладной системы, которая рассматривается уже на уровне реализации функциональной модели. На диаграмме прикладных систем могут отображаться как отдельные прикладные системы, так и их модульная структура (т.е. отдельные модули), а также все копии одного типа прикладной системы, которые могут быть идентифицированы, например, уникальным номером лицензии. На рис. 28 представлены объекты этих диаграмм, а на рис. 29 пример диаграммы типов прикладных систем и прикладных систем.

Информационная модель ARIS описывает структуру информации, необходимой для реализации всей совокупности функций системы. Состоит из моделей, с помощью которых описывается информация, используемая в деятельности организации. В методологии представлено 20 видов моделей, в том числе 15 видов для определения требований, 4 — спецификации проекта и 1 — описание реализации. Формулировка требований в рамках модели данных включает описание семантической модели данных в рассматриваемой предметной области. Для определения требований создается репозиторий моделей, описывающих альбом форм документов с подробным описанием реквизитов, схемы документооборота, иерархия форм документов. В качестве базовой модели спецификации проекта используются схемы информационных потоков. Наиболее широко

распространенным методом описания реализации является модель сущность-связь (ERM) в нотации Чена является. Сущность (Entity) — это реальный или абстрактный объект, представляющий интерес для задач в конкретной области деятельности (в терминах баз данных — таблица). Атрибуты (Attribute) это свойства, описывающие типы сущностей (столбец в таблице). Связи (Relationship) демонстрируют отношения между сущностями. Для однозначной идентификации объекта используется первичный ключ (Primary key). Для установления связей между объектами используется внешний ключ (Foreign key), который указывает на наличие ссылки на первичный ключ другой сущности.

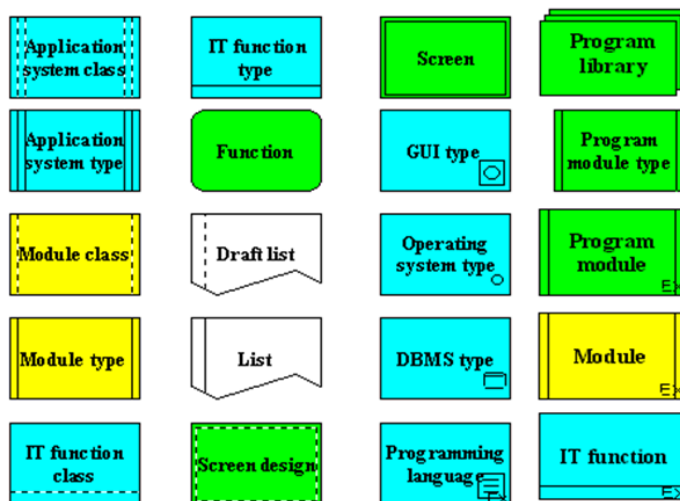


Рис. 28. Объекты диаграммы прикладной системы и ее типа

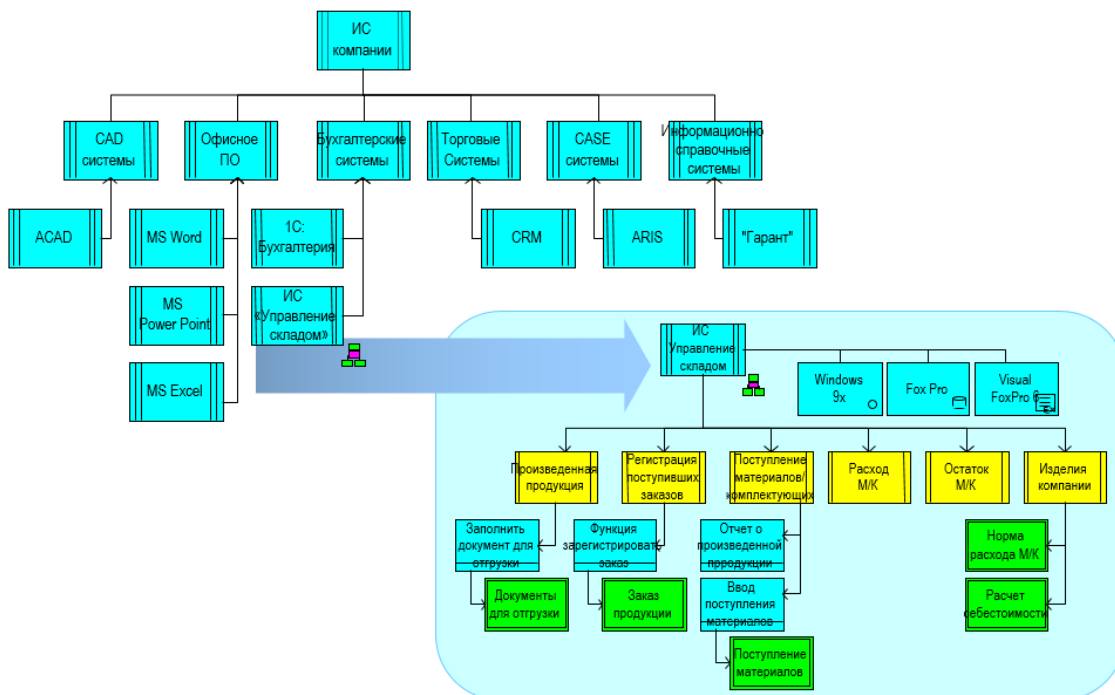


Рис. 29. Диаграммы прикладных системы и типов ПС

Процессная модель ARIS. Самая большая группа моделей (73 вида моделей). Состоит из моделей, используемых для описания бизнес-процессов компании, а также различных взаимосвязей между структурой, функциями и информацией. Совокупность управляющих моделей и моделей бизнес-процессов представляет собой комплексный взгляд на реализацию деловых процессов в рамках системы. К моделям процессов/управления относятся VAD-диаграммы и диаграммы eEPC (extended Event driven Process Chain). Также в ARIS поддерживаются нотации описания бизнес-процессов BPMN, IDFO и IDF3. Независимо от того какие типы моделей используются для представления процессной модели предприятия, она представляет собой иерархическую структуру, построение которой начинается с описания процессов верхнего уровня в виде VAD-диаграмм (рис. 30), в которой используется один вид объекта — процесс.

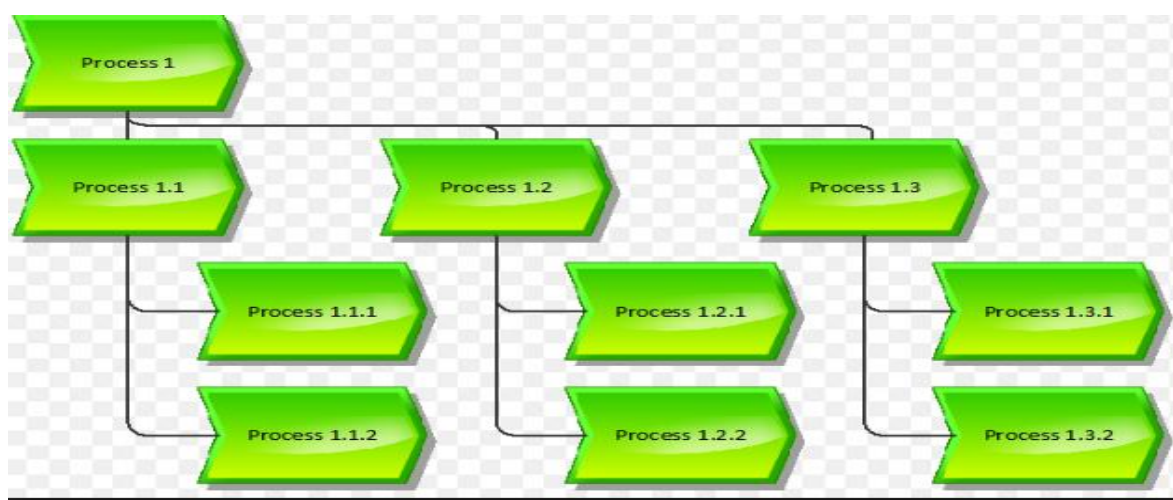


Рис. 30. Общий вид VAD-диаграммы

Диаграмма цепочки добавленной ценности (VAD) описывает функции организации, которые непосредственно влияют на реальный выход ее продукции или услуги. Эти функции создают последовательность действий, формирующих добавленные значения: стоимость, количество, качество и т.п. Построение диаграммы цепочки добавленной ценности целесообразно начать с обзорного представления взаимосвязанных частей процесса путем расположения элементов процесса согласно временной последовательности их выполнения.

На следующем этапе рекомендуется отразить взаимосвязь различных элементов процесса путем нанесения соответствующих связей. После отображения в модели структуры процесса каждый из элементов процесса рассматривается с точки зрения необходимости его детализации. Если это необходимо, то элемент декомпозируется на соответствующие блоки. На завершающем этапе в модель процесса добавляют элементы организационной структуры, отвечающие за выполнение процесса или участвующие в процессе, а также документы, используемые в процессе. На рис. 31 представлен пример VAD-диаграммы типового производственного предприятия.

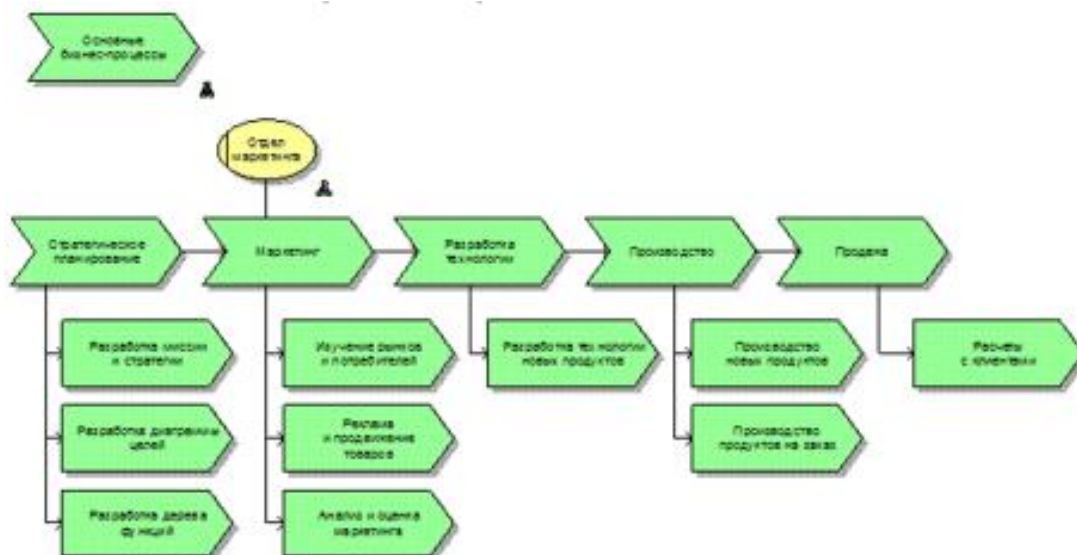


Рис. 31. Пример бизнес-процессов верхнего уровня

VAD-диаграммы используются чаще всего для описания процессов верхнего уровня, детализация процессов с указанием элементов организационной структуры и документов описывается с помощью eEPC-диаграмм.

Событийная цепочка процесса (eEPC) предназначена для детального описания процессов, выполняемых в рамках одного подразделения, несколькими подразделениями или конкретными сотрудниками. Она позволяет выявлять взаимосвязи между организационной и функциональными моделями. eEPC-диаграмма отражает последовательность действий в рамках одного бизнес-процесса, которые выполняются организационными единицами, а также ограничения по времени, налагаемые на отдельные функции. Для каждой функции должны быть определены начальное и конечное события, соответственно инициирующее и заканчивающее процесс. Кроме этого, рекомендуется отображать на диаграмме ответственных исполнителей, материальные и документальные потоки (входы, выходы, ресурсы), сопровождающие процесс, а также проведена декомпозиция на более мелкие функции (подфункции и т.д.).

Модель eEPC является наиболее информативной и удобной при описании деятельности подразделений организации. Основные типы объектов:



В отличие от «функций», выполняемых в течение определенного срока, «события» в диаграмме показывают возникшее в результате выполнения функций состояние, т. е. констатируют факт на текущий момент и в этом смысле не имеют временной протяженности.

Рассмотрим применение логических операторов. Оператор «И» используется если:

- функция может начать выполняться только после того, как произойдет несколько событий;

- завершение выполнения функции должно инициировать одновременно несколько событий;
- событие происходит только после обязательного завершения выполнения нескольких функций;
- одно событие инициирует одновременное выполнение нескольких функций.

Оператор «ИЛИ» используется если:

- функция может начать выполняться только после того, как произойдет одно или несколько событий;
- завершение выполнения функции может инициировать одно или несколько событий;
- событие происходит только после обязательного завершения выполнения одной или нескольких функций.

Оператор «Исключающее ИЛИ» используется если:

- функция может начать выполняться сразу после того, как произойдет одно из нескольких возможных событий;
- завершение выполнения функции может инициировать только одно из нескольких возможных событий в зависимости от условия;
- событие происходит сразу после завершения выполнения одной из нескольких возможных функций.

Преимуществом данной нотации является интуитивно понятный набор элементов и правил построения диаграмм. На рис. 32 приведен пример описания бизнес-процесса «Производство на заказ» в нотации eEPC.

Использование EPC помимо простоты и доступности имеет следующие преимущества:

- позволяет отразить все значимые организационные элементы на одной схеме (в отличие от простой блок-схемы);
- может использоваться на разных уровнях модели — описывать как глобальные процессы, так и делать детальные инструкции за счет того, что каждый функциональный блок может стать подпроцессом;
- легко делать сложные распараллеливания процесса, так как можно ввести любое количество событий в один ряд.

В то же время эта нотация не лишена недостатков, среди которых можно назвать следующие:

- необходимость придумывать события на каждые даже незначительные действия сильно усложняет схему;
- вероятны организационные разрывы из-за неудобного отслеживания назначений;
- качественное прописывание входов и выходов приводит к перегрузке схемы прямоугольниками, стрелками, которые начинают пересекаться и тем самым еще сильнее усложняют восприятие схемы;
- при распараллеливании работ очень сложно отразить исполнителей. Если один человек выполняет группу функций, картинка усложняется стрелками. Если присутствуют несколько исполнителей или мы не хотим рисовать длинные

стрелки, приходится дублировать «овальчики» с исполнителями. Все это очень скоро может привести к полной неразберихе на схеме.

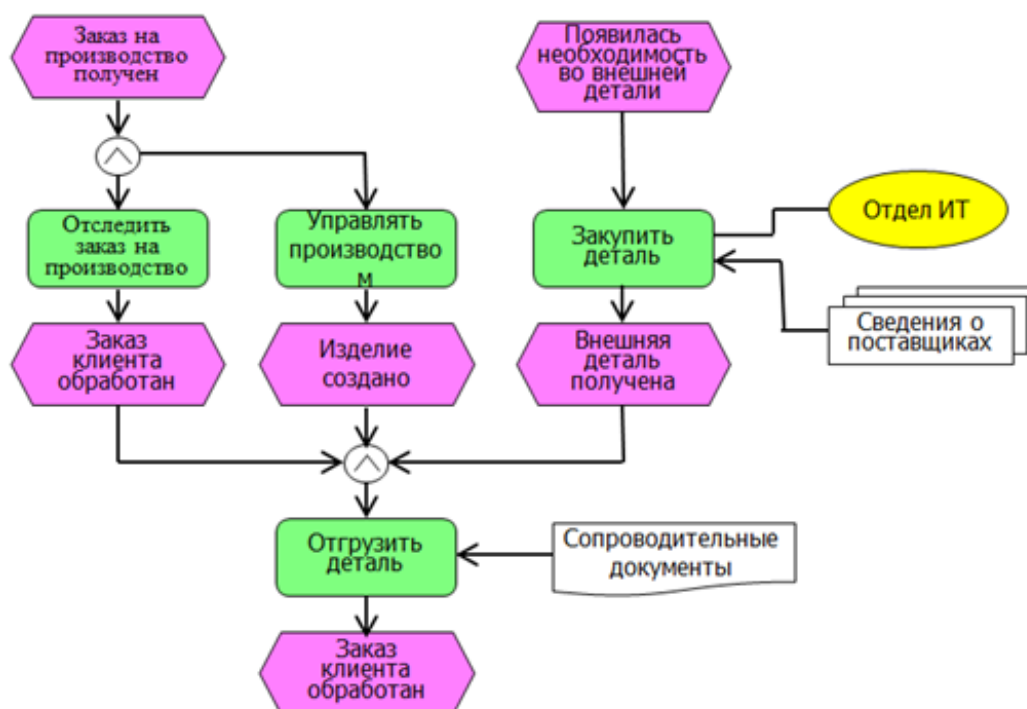


Рис. 32. Пример бизнес-процесса в нотации eEPC

Локальные рабочие процедуры, смоделированные в данной нотации, вполне удобны как для разработчика, так и для пользователя инструкции. Нотация подойдет и для проект-менеджеров, поскольку позволяет им наглядно планировать распределение работ в проекте на интуитивно понятном для разных участников проекта языке. Для разработки более сложной многоуровневой модели деятельности предприятия больше подходят другие нотации моделирования такие как IDFO и BPMN.

Серьезное преимущество ARIS перед другими инструментами заключается в том, что в нем хорошо развиты графические средства представления сформированных моделей. ARIS позволяет вывести в отчетный документ любую информацию, содержащуюся в базе данных проекта, а также получить аналитическую отчетность, структурированную по ряду признаков. ARIS является подходящим инструментом для детальной классификации, структурирования и наглядного представления операционных рисков. Методология ARIS подходит для крупных и (или) длительных проектов (например, внедрение системы непрерывного улучшения бизнес-процессов) на предприятиях с достаточным оборотом денежных средств. Это обусловлено стоимостью внедрения (например, разработка одного соглашения по моделированию может занять 1–3 месяца) и трудозатратами по сопровождению и поддержке. Методология ARIS в значительной большей степени предназначена для целей управленческого консалтинга и последующей поддержки решений, применения методологий эффективного для

анализа и оптимизации бизнес-процессов (реинжиниринга), внедрения ERP, систем управления качеством, проектирования информационных систем, а также для обеспечения процесса реорганизации.

Оптимизация деятельности в ARIS сводится к выделению, формализации и структурированию бизнес-процессов с целью формирования на их основе «сквозного» представления процессов организации. Внедрению ARIS обязательно должна предшествовать серьезная «ручная» проектно-аналитическая работа. Главным преимуществом методологии ARIS являются эргономичность и высокая степень визуализации бизнес-моделей, что делает данную методологию удобной и доступной в использовании всеми работниками организации (начиная от топ-менеджеров и заканчивая рядовыми работниками). Методология содержит более 80 моделей, и поэтому для осмысленного применения требуется время на изучение. В рейтинге Gartner Group система ARIS занимает лидирующее положение на рынке средств моделирования и анализа деловых процессов.

Список использованной и рекомендуемой литературы

1. Богомолова М.А. Архитектура предприятия : учеб. пособие / М.А. Богомолова. — Самара : Поволж. гос. ун-т телекоммуникаций и информатики, 2016. — 155 с. — URL: <http://www.iprbookshop.ru/71822.html>.
2. Васильев Р.Б. Управление развитием информационных систем / Р.Б. Васильев, Г.Н. Калянов, Г.А. Левочкина. — Москва : Горячая линия — Телеком, 2011. — 376 с. — ISBN 978-5-9912-0065-3.
3. Вдовин В.М. Теория систем и системный анализ : учеб. для бакалавров / В.М. Вдовин, Л.Е. Суркова, В.А. Валентинов. — 4-е изд. — Москва : Дашков и К°, 2019. — 644 с. — ISBN 978-5-394-03252-3.
4. Гриценко Ю.Б. Архитектура предприятия : учеб. пособие / Ю.Б. Гриценко. — Томск : Изд-во Том. гос. ун-та систем управления и радиоэлектроники, 2014. — 264 с. — ISBN 978-5-86889-512-8.
5. Данилин А. Архитектура и стратегия. «Инь» и «Янь» информационных технологий предприятия / А. Данилин, А. Слюсаренко. — Москва : Интернет-ун-т информ. технологий, 2005. — 502 с. — ISBN 5-9556-0045-0.
6. Лукьянов Б.В. Архитектура предприятия : учеб. пособие / Б.В. Лукьянов, П.Б. Лукьянов. — Саратов : Ай Пи Эр Медиа, 2019. — 153 с. — URL: <http://www.iprbookshop.ru/79895.html>.
7. Пешкова О.В. Стратегическое планирование информационных систем : учеб. пособие для магистрантов / О.В. Пешкова. — Иркутск : Изд-во БГУЭП, 2015. — 116 с.
8. Тельнов Ю.Ф. Инжиниринг предприятия и управление бизнес-процессами. Методология и технология : учеб. пособие для студентов магистратуры, обучающихся по направлению «Прикладная информатика» / Ю.Ф. Тельнов, И.Г. Федоров. — Москва : Юнити-Дана, 2017. — 207 с. — ISBN 978-5-238-02622-0.

Учебное издание

Пешкова Ольга Вячеславовна

Архитектура предприятия

Учебное пособие

Издается в авторской редакции

ИД № 06318 от 26.11.01.
Подписано в пользование 02.07.19.

Издательство Байкальского государственного университета.
664003, г. Иркутск, ул. Ленина, 11.
<http://bgu.ru>.