

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
АВТОМОБИЛЬНО-ДОРОЖНЫЙ ИНСТИТУТ

Н. В. Гуменюк

АРХИТЕКТУРА ПРЕДПРИЯТИЯ

Учебное пособие
для обучающихся образовательных учреждений
высшего профессионального образования

Донецк
2020

УДК 004.9:658(075.8)

ББК У212.5

Г945

Рекомендовано Ученым советом
ГОУВПО «Донецкий национальный технический университет»
в качестве учебного пособия для обучающихся образовательных учреждений
высшего профессионального образования
(Протокол № 1 от 28.02.2020)

Рецензенты:

Лепя Роман Николаевич – доктор экономических наук, профессор, заведующий отделом моделирования экономических систем ГУ «Институт экономических исследований»;

Моисеенко Игорь Алексеевич – доктор физико-математических наук, доцент, декан факультета математики и информационных технологий ГОУВПО «Донецкий национальный университет».

Автор:

Гуменюк Наталья Владимировна – доцент кафедры математического моделирования АДИ ГОУВПО «ДОННТУ»;

Гуменюк, Н. В.

Г945 Архитектура предприятия : учеб. пособие для обучающихся образоват. учреждений высш. проф. образования / Н. В. Гуменюк ; ГОУВПО «ДОННТУ». – Донецк : ДОННТУ, 2020. – 217 с. : ил., табл.

Учебное пособие содержит сведения по основным разделам дисциплины «Архитектура предприятия». В рамках теоретического курса изложены этапы становления архитектурного подхода, ключевые принципы построения доменов архитектуры предприятия, проанализированы вопросы взаимосвязи и интеграции бизнес-стратегии и стратегии использования современных информационных технологий, представлены наиболее востребованные модели и методы разработки корпоративной архитектуры.

Пособие содержит ряд практических рекомендаций по разработке организационно-функциональных моделей, что позволяет студентам выработать навыки построения бизнес-модели предприятия с использованием классической методологии структурного анализа.

Книга предназначена для обучающихся вузов, научных и педагогических работников, деятельность которых связана с применением информационных технологий в экономике.

УДК 004.9:658 (075.8)

ББК У212.5

© Гуменюк Н. В., 2020

© ГОУВПО «Донецкий национальный технический университет», 2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
ТЕМА 1 ОБЩИЕ ПОЛОЖЕНИЯ АРХИТЕКТУРЫ ПРЕДПРИЯТИЯ	7
1.1 Понятие и предмет изучения архитектуры предприятия	7
1.2 Исторические этапы становления архитектуры предприятия	11
1.3 Интегрированная концепция архитектуры предприятия.....	15
1.4 Домены архитектуры предприятия	18
1.5 Принципы, модели и стандарты в рамках архитектуры предприятия	21
1.6 Контекст и уровни абстракции архитектуры предприятия	23
1.7 Статический и динамический аспект архитектуры предприятия	30
Контрольные вопросы	33
ТЕМА 2 БИЗНЕС-АРХИТЕКТУРА ПРЕДПРИЯТИЯ	34
2.1 Контекст и основные элементы бизнес-архитектуры	34
2.2 Предприятие как объект исследования бизнес-архитектуры.....	37
2.3 Миссия и стратегия: роль в построении бизнес-модели.....	44
2.4 Организационная структура предприятия.....	50
2.5 Функциональная модель предприятия.....	61
2.6 Основные модели и инструменты описания бизнес-архитектуры ...	66
Контрольные вопросы	70
ТЕМА 3 СИСТЕМНАЯ АРХИТЕКТУРА	71
3.1 Архитектура информации	71
3.2 Модели и инструменты построения архитектуры приложений	75
3.3 Структура и принципы разработки технологической архитектуры..	83
Контрольные вопросы	98
ТЕМА 4 МОДЕЛИ И МЕТОДИКИ ПОСТРОЕНИЯ АРХИТЕКТУРЫ ПРЕДПРИЯТИЯ	99
4.1 Модель Захмана.....	102
4.2 Модель описания ИТ-архитектуры Gartner.....	108

4.3 Методика Meta Group	111
4.4 Методика TOGAF	115
4.5 Модель представления архитектуры «4+1»	117
4.6. Стратегическая модель архитектуры SAM	119
4.7 Архитектурные концепции и методики Microsoft.....	122
4.8 Методологические принципы использования стандарта COBIT 5 при построении архитектуры предприятия	127
4.9 Метод планирования архитектуры организации EAP	133
Контрольные вопросы	145
ПРАКТИЧЕСКИЕ РЕКОМЕНДАЦИИ ПОСТРОЕНИЯ БИЗНЕС- МОДЕЛЕЙ ПРЕДПРИЯТИЯ	147
1 Разработка бизнес-архитектуры предприятия на основании использования методологии структурного анализа IDEF0	147
Контрольные вопросы	169
2 Анализ и синтез моделей бизнес-процессов на основании использования методологии структурного анализа IDEF0	170
Контрольные вопросы	177
3 Моделирование взаимодействия бизнес-процессов предприятия на основании использования методологии IDEF3.....	177
Контрольные вопросы	184
4 Проведение стоимостного анализа, основанного на работах (ABC-анализ)	185
Контрольные вопросы	196
5 Моделирование функциональных требований к проектируемой системе при помощи диаграммы потоков данных DFD	197
Контрольные вопросы	203
6 Создание модели бизнес-архитектуры предприятия категории «To-Be» (реинжиниринг бизнес-процессов)	203
Контрольные вопросы	212
СПИСОК ЛИТЕРАТУРЫ.....	213

ВВЕДЕНИЕ

Информационные технологии стали неотъемлемой составляющей современного предприятия, определяя в значительной степени уровень и характер его производства, инновационность и конкурентоспособность. Анализ и выбор информационных технологий должны строиться на знаниях архитектуры бизнеса, что положило начало идее интеграции бизнеса и информационных технологий на основе архитектурного подхода, в котором эти два направления рассматриваются как единое целое и объединяются общим понятием – архитектура предприятия.

Именно такой подход и позволяет обеспечить единое видение бизнес-процессов организации и ИТ-инфраструктуры, обеспечивающей их реализацию. Эффективная архитектура предприятия характеризуется целостным и всеобъемлющим представлением как статического, так и динамического аспектов архитектуры, которые содержат следующие элементы:

- миссию предприятия и стратегическое планирование его деятельности;
- организационные структуры и сервисы, необходимые для реализации бизнес-архитектуры, позволяющей в полной мере удовлетворить требования миссии и стратегического планирования;
- системную архитектуру, которая требуется для эффективной реализации сервисов уровня бизнес-архитектуры.

Современные подходы к управлению качеством предполагают определение сети бизнес-процессов компании, поддерживаемых полностью либо частично автоматизированными информационными технологиями, и последующую работу по их улучшению. В связи с этим, адекватная бизнес-модель позволяет существенно облегчить решение стоящих перед каждым предприятием или учреждением насущных в современных условиях задач:

- реорганизация бизнеса, обусловленная переходом от функционально-индустриальной модели к процессной;

– применение информационных систем для управления бизнесом, обусловленное стремительным развитием современных информационных технологии и средств связи;

– сертификация бизнеса с применением международных стандартов ISO, обусловленная повышением требований к качеству товаров и услуг.

Учебное пособие состоит из двух частей. Материал первой части пособия раскрывает теоретико-методологические основы построения архитектуры современного предприятия, обобщая опыт разработок авторитетных ученых и ведущих корпораций мира. В рамках теоретического курса проанализированы ключевые этапы становления архитектурного подхода и развития научного направления архитектуры предприятия; подробно описываются статический и динамический аспекты построения архитектуры предприятия; рассматривается архитектура предприятия в разрезе различных слоев и контекст их использования в условиях изменяющейся внешней среды. При рассмотрении категории «архитектура предприятия» уделено достаточно много внимания вопросам миссии и стратегического планирования деятельности предприятия, взаимосвязи и интеграции бизнес-стратегии и стратегии ИТ.

Во второй части пособия представлены рекомендации по использованию классической методологии структурного анализа и проектирования для построения бизнес-модели предприятия, которые сопровождаются примерами реализуемыми с помощью CASE-средства Allfusion Process Modeler (BPWin). Задача практической части пособия состоит в обучении студентов технологии составления диаграмм по стандартам DFD, IDEFO, IDEF3, IDEF1X путем последовательного рассмотрения процесса моделирования бизнес-процессов, спецификации и анализа требований к программному обеспечению.

Таким образом, целью изучения дисциплины «Архитектура предприятия» является формирование у студента профессиональных знаний по теоретическим основам построения архитектур предприятия, а также закрепление умений и навыков использования современных методологий и средств проектирования бизнес-моделей.

ТЕМА 1 ОБЩИЕ ПОЛОЖЕНИЯ АРХИТЕКТУРЫ ПРЕДПРИЯТИЯ

1.1 Понятие и предмет изучения архитектуры предприятия

Современное предприятие представляет собой сложную социальную экономическую систему, обладающими следующими свойствами:

1) хозяйствующий субъект функционирует во времени, взаимодействует с внешней средой и в каждый момент экономической деятельности может находиться в одном из возможных состояний, определяемых кривой жизненного цикла;

2) на «вход» системы поступают ресурсы, а на «выходе» — результаты (продукция, работы, услуги) (рис.1);

3) внутри системы, на основе используемых бизнес-процессов, происходит преобразование поступающих ресурсов в результаты;

4) под воздействием внешней среды внутри системы возникают отклонения заданных показателей развития, которые являются фактором, определяющим переход системы из одного состояния в другое, и приводят к адаптации входных и выходных параметров системы.

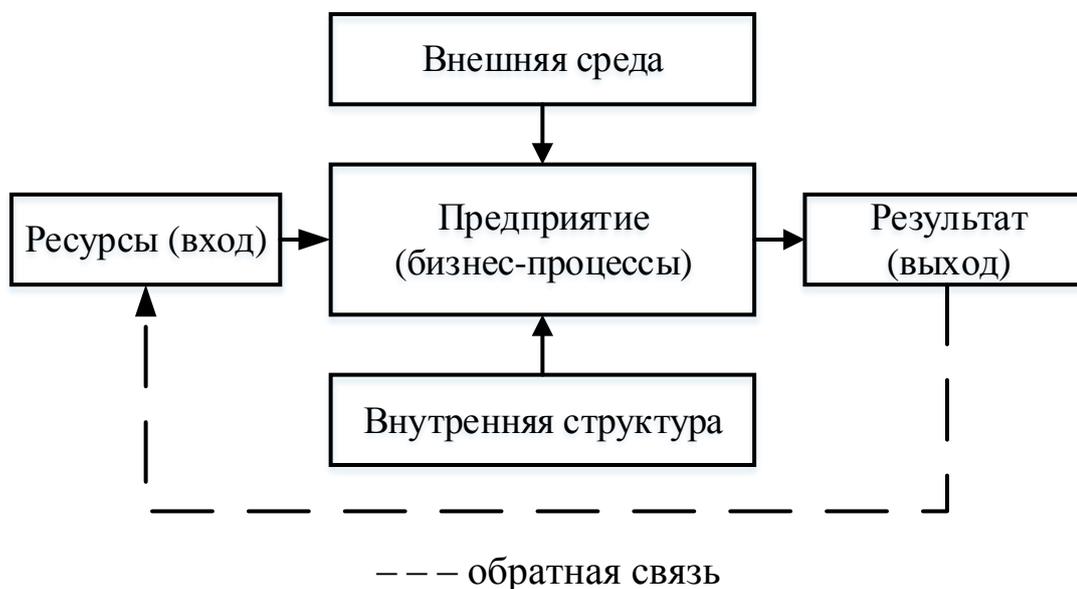


Рис. 1 – Схема функционирования предприятия

Чтобы потребляемые ресурсы (материальные, трудовые, финансовые, информационные) превратились в результат, нужно использовать соответствующие технологии, труд персонала предприятия и как необходимое условие управление персоналом. Во второй половине XX в. резко возросла сложность решаемых производственных задач, что потребовало широкого внедрения информационных технологий (ИТ).

Идея интегрированного взаимодействия бизнеса и ИТ-технологий стала основой *архитектурного подхода*, в котором эти две сущности рассматриваются как единое целое и объединены общим понятием — архитектура предприятия (АП).

При этом *управление предприятием* рассматривается как управление архитектурой в контексте достижения наибольшей эффективности функционирования предприятия [1, 2].



Рис. 2 – Сущность, предмет и цель архитектуры предприятия

Предмет архитектуры предприятия – существующие в организации бизнес-процессы и поддерживающие их организационно-управленческие и технические системы, использующие ИТ-технологии. Алгоритмы включают

модели, правила, приемы, обеспечивающие систематизацию и формализованное описание деятельности организации (предприятия) в соответствии с поставленными целью и задачами.

Цель архитектуры предприятия – подготовка управленческих решений, обеспечивающих повышение результативности бизнес-процессов и поддерживающих их организационно-управленческих и технических систем на основе взаимодействия бизнеса и IT-технологий.

В общем виде под архитектурой предприятия (Enterprise Architecture, EA) понимается всестороннее и исчерпывающее описание всех ключевых элементов предприятия и межэлементных отношений.

Согласно ГОСТ Р ИСО 15704–2008 «Промышленные автоматизированные системы. Требования к стандартным архитектурам и методологиям предприятия» архитектура предприятия должна включать роль людей, описание процессов (функции и поведение) и представление всех вспомогательных технологий на протяжении всего жизненного цикла предприятия [3].

В соответствии с документом Federal Enterprise Architecture Framework. Dev. by: The Chief Information Officers Council (USA) архитектура предприятия является стратегической информационной основой, которая определяет:

- 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.

Исходя из рассмотренных трактовок, на рис. 3 схематически представлена сущность архитектуры предприятия и роли ее составляющих элементов.

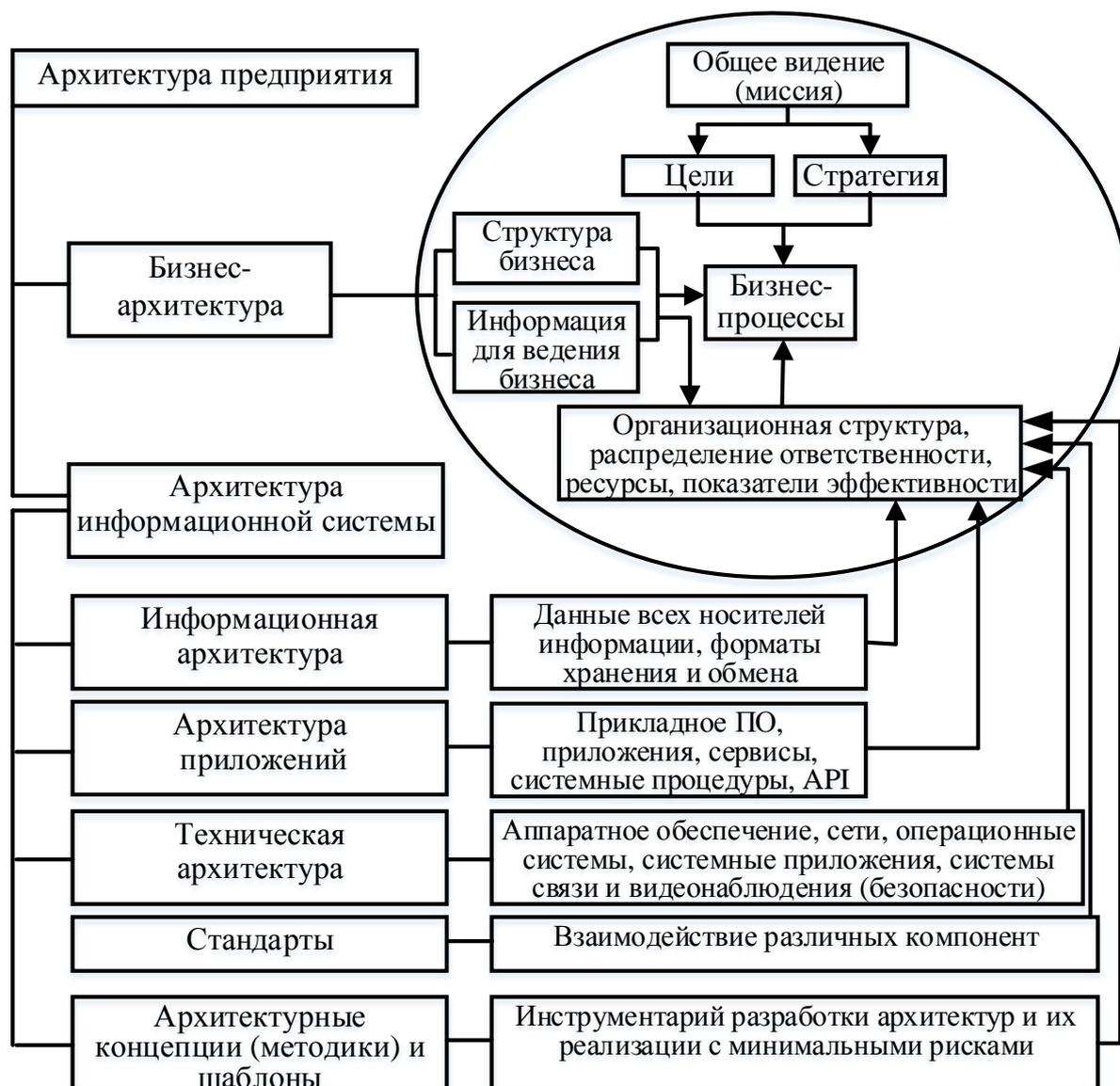


Рис. 3 – Содержание архитектуры предприятия

Так, миссия, зафиксированная как основной вектор и ключевое направление деятельности предприятия, определяет стратегию и цели его развития. В свою очередь, стратегия и цели работы предприятия воплощаются в жизнь на основе реализации основных и вспомогательных бизнес-процессов организации. Отметим, что на результативность выполнения бизнес-процессов оказывает влияние не только корректно сформулированные цели и задачи руководства, но и структура бизнеса, стиль его ведения, обеспеченность информацией для ведения бизнеса, а также действующая на предприятии организационная структура, система распределения функций и ответственности

между сотрудниками, наличие ресурсов и заложенные показатели эффективности. Таким образом, домен бизнес-архитектуры является определяющим в архитектуре предприятия и задает тенденции ее развития, при этом домены архитектуры информационных систем обеспечивают ее корректную работу. В состав архитектуры информационных систем входят следующие домены: информационная архитектура, архитектура приложений и технологическая архитектура, стандарты, архитектурные концепции и шаблоны, позволяющие реализовать потребности и запросы бизнес-архитектуры на всех уровнях абстракции, начиная от концептуального и заканчивая физическим уровнем и непосредственной реализацией.

1.2 Исторические этапы становления архитектуры предприятия

Сущность понятия «архитектура предприятия» с течением времени значительно трансформировалось. Во многом это отражало те управленческие задачи, которые решались с его помощью. Рассмотрим ключевые этапы становления архитектурного подхода и, соответственно, развития представлений о архитектуре предприятия (рис. 4).

Такой подход позволял добиться определенных частных выгод, связанных, прежде всего, с уменьшением стоимости закупок и эксплуатации информационных систем, сокращением издержек на разработку приложений и обучение персонала.

Однако он являлся ограниченным, так как не подразумевал ориентации на решение бизнес-задач, таких как, например, формирование единых в масштабе компании данных по клиентам.

Следующей ступенью явилось понятие *корпоративной информационно-технологической архитектуры масштаба предприятия* (EWITA – Enterprise-wide information technology architecture), которая подразумевала помимо технологического уровня включение описание архитектуры информации и архитектуры прикладных систем.

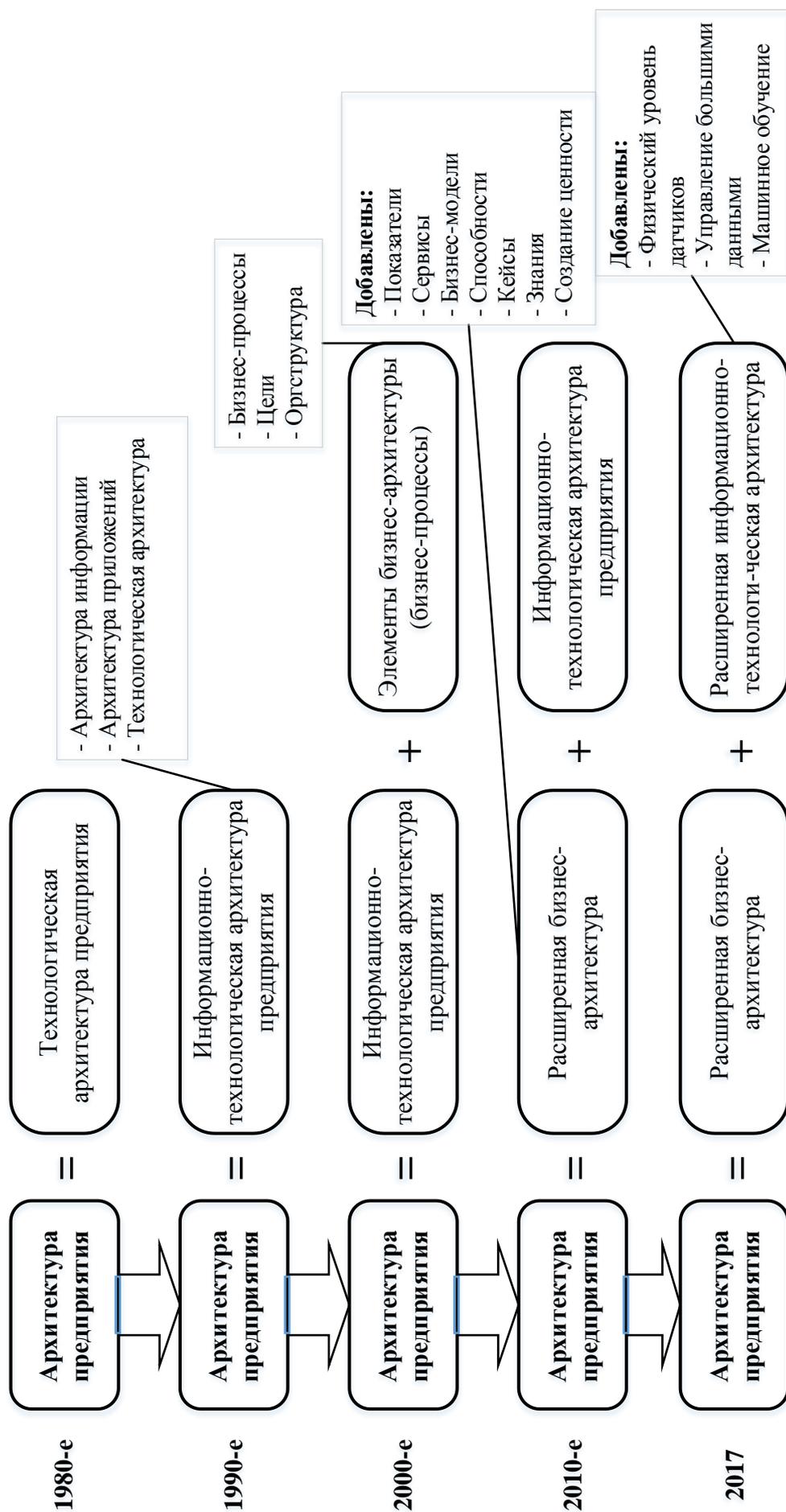


Рис. 4 – Эволюция концепции «Архитектура предприятия»

Основное направление работы заключается в совместном использовании общих данных, исключении дублирования бизнес-функций, координации управления пользователями, ресурсами, информационной безопасностью за счет улучшений в управлении портфелем прикладных систем компании.

Следующим этапом описания существующих в организации процессов и планируемых изменений явилось непосредственное введение понятия «*бизнес-архитектура*» (Business Architecture), которое стало дополнять ИТ-архитектуру и увязывать ее с достижением стратегических целей предприятия. Следует подчеркнуть, что в реальности это, фактически, две стороны одной медали. Должна существовать одна архитектура предприятия, которая показывает, как взаимосвязаны между собой все элементы ведения бизнеса, в том числе элементы, затрагивающие информационные технологии в компании. Преимуществами подобного включения бизнес-архитектуры в контекст целостной архитектуры предприятия выступают большая способность организации к изменениям, или динамичность (agility), а также синхронизация возможностей информационных технологий с бизнес-стратегией (business-IT alignment): обеспечение вариативности бизнес-стратегии за счет возможности изменений в обеспечивающих процессах и технологических решениях; использование лучших перспектив, с точки зрения возможностей ИТ по формированию самой бизнес-стратегии.

В 2010-х исследователи архитектуры предприятия пришли к заключению, что предложенное ИТ-специалистами наполнение термина «*бизнес-архитектура*» не позволяет отразить множество таких критически важных для бизнеса понятий, как «способности» (capabilities), «бизнес-модель» (способ создания ценности для покупателей и зарабатывания на этом денег), «непроцессная деятельность» (проекты, кейсы и пр.), «знания», «неформальная структура» и др.. Кроме того, практика использования архитектурного подхода в бизнес-инжиниринге показала, что важно не только обеспечить согласованность бизнеса и ИТ, но и добиться внутренней согласованности (coherence) разных элементов архитектуры предприятия (цели должны быть

согласованы с миссией предприятия и имеющимися способностями, задачи с целями, способностями и организационной структурой, показатели должны измерять достижение целей и задач, процессы должны улучшать выбранные показатели, а информационные системы усиливать именно критически значимые процессы и т.д.). Для того чтобы концепция архитектуры предприятия вышла за пределы департамента информационных технологий, топ-менеджеры должны увидеть в ней знакомые, актуальные для них элементы, а не только интерфейсы с ИТ. В целом, в последнее время существенно возросла роль бизнес-архитектуры, как все более значимой части АП [4].

Таким образом эволюция развития понятия архитектуры предприятия неразрывно связана с изменением роли ИТ в деятельности предприятий (рис. 5).

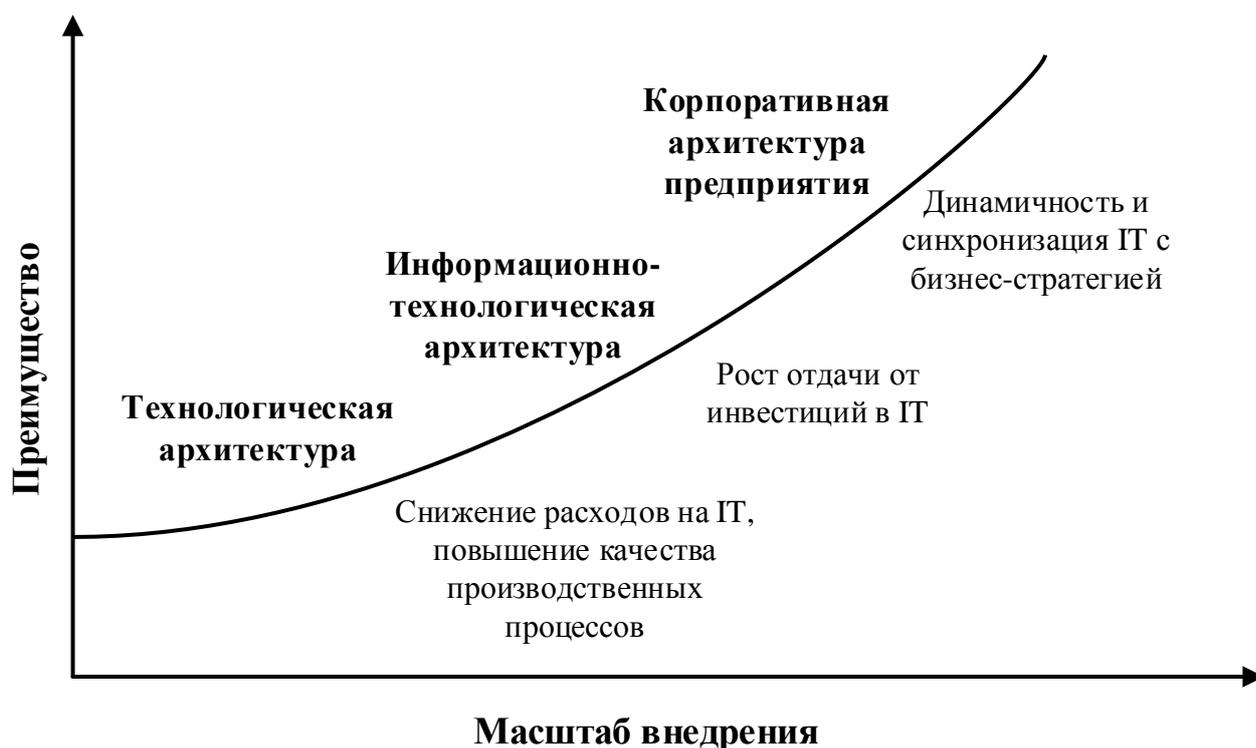


Рис. 5 – Динамика изменений представлений об архитектуре предприятия и роли информационных технологий [5]

Однако, такие изменения, происходят на фоне существования парадокса, который состоит в тенденции уменьшения возврата от инвестиций в ИТ.

Поэтому на данном этапе развития представлений о АП справедливо утверждение, что использование ИТ само по себе не приносит прямых преимуществ, а только создает условия для их получения.

1.3 Интегрированная концепция архитектуры предприятия

Являясь моделью всех ключевых элементов и межэлементных отношений исследуемой организации, архитектура описывает деятельность предприятия с двух позиций:

– с позиции логических терминов, таких как взаимодействующие бизнес-процессы и бизнес-правила, необходимая информация, структура и потоки информации, места расположения работы и пользователей;

– с позиции технических понятий, таких как аппаратные и компьютерные средства, программное обеспечение, коммуникация данных, защита и безопасность, а также используемые стандарты.

При этом обе группы понятий охватывают с одной стороны текущее состояние «как есть» и перспективное (или целевое) состояние «как должно быть». В соответствии с этим, архитектура предприятия рассматривается в статике и динамике. При этом статический аспект отражает характеристику предприятия в некоторый фиксированный момент времени со стороны трех основных компонентов: миссии, бизнес-архитектуры и системной архитектуры, а динамический аспект представляет собой описание процесса перехода предприятия от текущего состояния к некоторому желаемому состоянию в будущем.

Одна из главных задач архитектуры предприятия состоит в том, чтобы оптимизировать взаимные зависимости и существующие взаимоотношения и создать для этого соответствующую инфраструктуру и приложения, которые поддерживали бы все эти действия.

Эффективная архитектура предприятия должна обеспечивать целостный и всеобъемлющий взгляд на следующие аспекты:

- бизнес, включая движущие силы (ключевые факторы), видение и стратегию;
- организационные структуры и сервисы, которые требуются для реализации этого видения и стратегии;
- информация, системы и технологии, которые требуются для эффективной реализации этих сервисов.

Разработка архитектуры предприятия позволяет решить одну из существенных проблем взаимодействия бизнеса и ИТ – *синхронизация потребностей бизнеса и возможностей ИТ*, которая достигается за счет следующего:

- автоматизации процессов – там, где видится положительный возврат от инвестиций в технологии (ROI);
- уточнения понимания и формализации описания бизнес-процессов путем формального их моделирования;
- моделирование информации (разработка архитектуры информации), которая обеспечивает выполнение бизнес-процессов организации;
- формирование портфеля прикладных систем (определение архитектуры приложений), которые обрабатывают эту информацию в соответствии с некоторыми функциональными требованиями;
- построение инфраструктуры (формирование технологической архитектуры), которая обеспечивает работу прикладных систем на уровне, описанном в операционных требованиях (надежность, масштабируемость и т.д.).

Одной из концепций, ключевой для понимания роли архитектуры предприятия, является *концепция расширенной цепочки добавочной стоимости (value chain) ключевых бизнес-процессов*. Идея «цепочки добавочной стоимости» привлекла к себе внимание в середине 1980-х годов, когда Майкл Портер опубликовал книгу «Конкурентное преимущество» (Competitive Advantage, 1985). Большое количество специалистов в области корпоративной стратегии и управления активно используют на практике предложенные

Портером модели, такие как «модель пяти факторов влияния», для анализа конкурентной среды и связанных с этим и возможностей угроз.

При анализе деятельности организации Портер предложил сконцентрироваться на бизнес-процессах или цепочках создания добавочной стоимости, которые пересекают организационные границы, границы департаментов и функциональных образований (рис. 6). Цепочка создания добавочной стоимости включает все бизнес-процессы, которые должны быть выполнены от момента получения заказа от потребителя до поставки окончательного продукта. Портер рекомендовал прекратить мыслить в терминах организационного деления и вместо этого рассматривать все процессы и связанные с ними активности систематически и целостно. Он также высказал идею о том, что специфические активности в рамках процессов либо создают дополнительную стоимость в конечном продукте, либо нет, и предлагал компаниям фокусироваться на тех активностях, которые создают добавочную стоимость [6].



Рис. 6 – Бизнес-процессы и обеспечивающие информационные системы в рамках цепочек создания добавочной стоимости

Процессы и шаги процессов обеспечиваются потоком информации. Требуемая для этого информация представляет как бы «зеркальный образ процессов». Приложения, которые обеспечивают выполнение процессов, связаны между собой интерфейсами. После того как становится понятно, какие именно приложения и интерфейсы нужны для обеспечения выполнения процессов, можно рассматривать вопрос о том, какая ИТ инфраструктура нужна для их выполнения.

Таким образом, когда и бизнес-руководители, и руководители в области ИТ имеют общее понимание ключевых цепочек создания добавочной стоимости, то появляется гораздо больше возможностей для четкого объяснения потребностей каждой из вовлеченных в реализацию и автоматизацию процессов групп.

При этом исследования показывают, что огромное количество ценной информации может быть получено без выполнения анализа бизнес-процессов с «парализующей» степенью детализации. Иными словами, применимо **правило «80/20»**: *достаточно знать восемьдесят процентов необходимых деталей о процессах, что может быть получено двадцатью процентами возможных усилий на анализ этих процессов.* Это же правило применимо и к архитектуре предприятия в целом.

Характерной чертой современного бизнеса является исключительно высокая скорость происходящих изменений в окружающей среде, в связи с чем большой научный интерес сейчас представляет создание модели так называемой адаптивной организации, которая должна содержать внутренние механизмы для изменения в соответствии с требованиями внешнего окружения.

1.4 Домены архитектуры предприятия

Имеется множество методик описания архитектуры, и все они разбивают архитектуру предприятия на различное количество моделей и определений, которые относятся к таким областям, как бизнес, информация, прикладные

системы, технологическая инфраструктура. Обычно в составе архитектуры выделяют от четырех до семи основных представлений (предметных областей или доменов). Эти области последовательно покрывают архитектурные аспекты, отталкиваясь от потребностей функционирования организации (бизнеса) и обеспечивая весь набор технологий для реализации конкретного решения бизнес-проблемы.

Традиционно, на предприятии выделяют следующие *архитектурные слои или домены архитектуры* (рис. 7):

- **бизнес-архитектура** – описывает деятельность организации с точки зрения ее ключевых бизнес-процессов;
- **архитектура информации (данных)** – определяет, какие данные необходимы для поддержания бизнес-процессов (например, модель данных);
- **архитектура приложений** – определяет, какие приложения используются и должны использоваться для управления данными и поддержки бизнес-функций (например, модели приложений);
- **технологическая архитектура** (инфраструктура или системная архитектура) – определяет, какие обеспечивающие технологии (аппаратное и системное программное обеспечение, сети и коммуникации) необходимы для создания среды работы приложений, которые, в свою очередь, управляют данными и обеспечивают бизнес-функции.

Корпоративные миссия и стратегия		
Бизнес-архитектура		
Бизнес-процессы	Организационно-штатная структура	Система документооборота
Системная архитектура		
Приложения	Данные	Оборудование

Рис. 7 – Архитектурные слои на предприятии

В зависимости от конкретных потребностей организации и актуальности решения тех или иных проблем можно выделить и другие представления архитектуры, например, Архитектура интеграции, общих сервисов, Сетевая архитектура, Архитектура безопасности и т.д. Но большинство методик рассматривает эти предметные области как часть более обширных доменов, таких как архитектура приложений и технологическая архитектура, выделяя их в отдельные домены более низкого уровня на последующих этапах детального описания архитектуры предприятия.



Рис. 8 – Области, входящие в понятие Архитектуры предприятия

Как отдельную область, очень часто выделяют **архитектуру процессов управления информационными технологиями** (архитектуру операций), т.е. архитектура предприятия является неполной без структур управления и наборов процессов, которые поддерживают и обеспечивают как инфраструктуру и прикладные системы, так и непосредственно архитектурный процесс.

1.5 Принципы, модели и стандарты в рамках архитектуры предприятия

Основные составные элементы стратегии и архитектуры предприятия можно отобразить условно в виде следующей пирамиды (рис. 9).



Рис. 9 – Составные элементы доменов архитектуры предприятия

Из данного рисунка видно, что руководящие принципы, так же как и стандарты политики, руководства, процедуры, могут относиться абсолютно ко всем элементам архитектуры: данным, прикладным системам, технологической инфраструктуре.

Важно отметить, что для описания «верхней» части этой пирамиды используются, в основном, такие механизмы, как декларируемые принципы. «Средняя» часть пирамиды, т.е. непосредственно архитектура, описывается в форме набора соответствующих моделей. А «нижняя» часть пирамиды связана с выработкой соответствующих правил, процедур или выбором стандартов.

Ключевыми элементами, с точки зрения архитектуры, являются принципы, стандарты и модели.

Стандарты разрабатываются на основе принципов и описывают, как принципы будут реализованы на практике.

Модели являются графическим представлением принципов и стандартов и используются для описания архитектуры. При этом модели обеспечивают упрощенное представление о сложном реальном мире и создают абстрактные конструкции, в которых опущены несущественные детали и внимание сосредоточено на наиболее важных аспектах описываемого предмета. Кроме того, модели обеспечивают основу для обсуждения между различными заинтересованными сторонами одного и того же предмета.

В общем случае *описание архитектуры предприятия* включает следующие элементы:

- миссия и видение;
- руководящие принципы – утверждения, описывающие принципы и ключевые элементы философии использования информационных технологий;
- цели, задачи и стратегии;
- архитектура информационных технологий;
- политики (правила) – являются общими утверждениями, которые задают направления и цели, связанные с инициативами в области ИТ. Они носят, как правило, достаточно высокоуровневый и общий характер и обеспечивают скоординированный процесс планирования, закупку критически важных технологий, эффективную разработку систем и эффективное использование информационных технологий и ресурсов;
- ИТ-стандарты – это обязательные к использованию утверждения, касающиеся используемых технологий, продуктов и/или услуг. Они должны быть достаточно полными и в то же время определять разумный минимум требований, обязательных для использования. В случае, когда речь идет о стандартах, выбираемых государством, особенно важным является подход, когда стандарты описывают только наиболее общие и важные элементы технологий в соответствии с принципами честной конкуренции;

– процедуры – это инструкции, описывающие, как выполняются политики и стандарты. Процедуры устанавливают и описывают процессы, которые выполняются на регулярной основе;

– руководства или рекомендации (guidelines) – это описания лучших практик или приемлемых подходов к практической реализации политик и процедур. Руководства могут стать стандартами.

Реализация целей, задач и стратегий достигается через соответствующие ИТ-проекты, которые формулируются в планах на очередной период деятельности.

1.6 Контекст и уровни абстракции архитектуры предприятия

Движущей силой архитектуры предприятия является целостное видение, пронизывающее внутриорганизационные границы.

Если проанализировать цели ключевых подходов к описанию архитектуры предприятия, то в них проявляются общие черты:

– использование для анализа множества точек зрения на объект изучения. Важно понимать, что ни одна отдельно взятая точка зрения не является достаточной для понимания всего целого;

– для того чтобы обеспечить процесс синтеза, все модели, которые включены в архитектуру, связываются с другими моделями. Они являются либо более детальной декомпозицией, либо связанными между собой представлениями. Это богатство взаимосвязей между моделями напрямую определяет качество архитектуры.

Согласно определению Всемирной организации корпоративной архитектуры (GEAO – Global Enterprise Architecture Organization), **архитектура предприятия** *описывает те способы, с помощью которых общее видение деятельности организации отражено в структуре и динамике предприятия. На различных уровнях абстракции она дает единый набор моделей, принципов, руководств и политик, которые используются для создания, развития и*

обеспечения соответствия систем в масштабе и контексте деятельности всего предприятия в целом» [7].

В связи с этим, в зависимости от поставленной задачи и выбранного контекста ее решения, рассматривать архитектуру можно на различных уровнях абстракции (рис. 10).

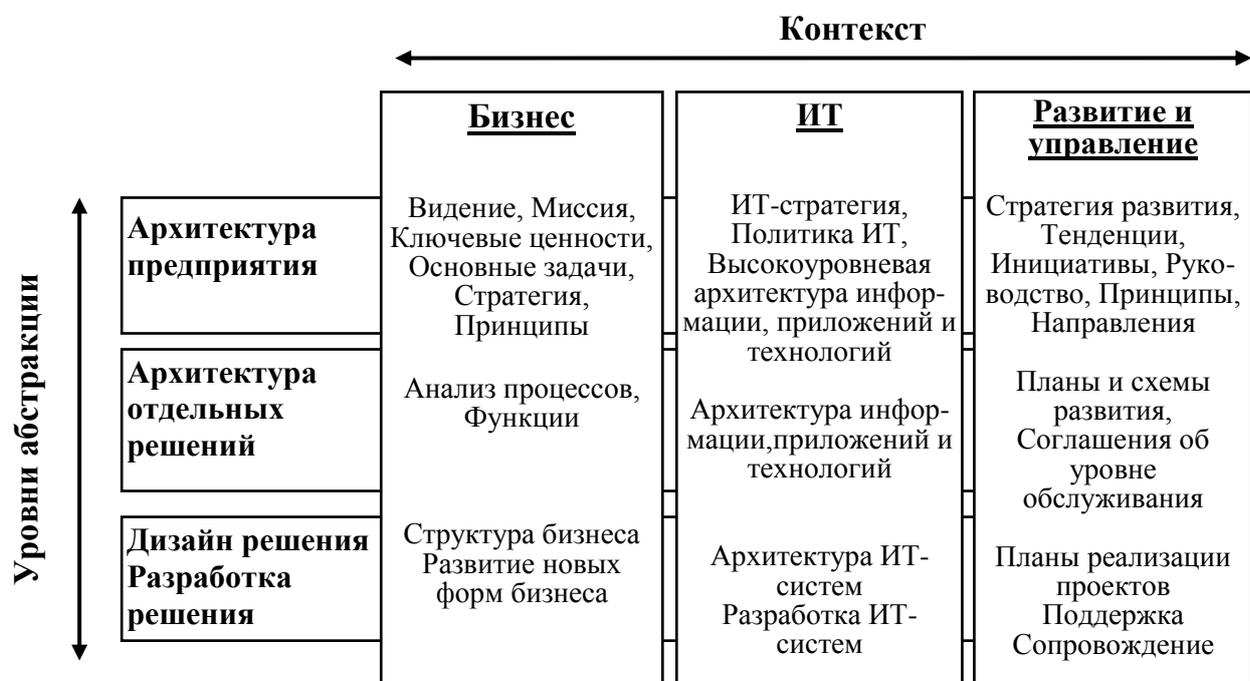


Рис. 10 – Концепции, соответствующие различным элементам и уровням абстракции архитектуры

В связи с этим, при описании архитектуры предприятия чрезвычайно важную роль имеют два понятия:

- перспектива (perspective) или уровень абстракции;
- представление (view) или предметная область, домен архитектуры.

Представление является упрощенным описанием или проекцией модели, которая рассматривается с некоторой определенной точки зрения и опускает все детали, которые не имеют отношения к этой точке зрения. Каждое из представлений (предметных областей) можно рассматривать и анализировать с различных перспектив или на нескольких уровнях абстракции.

Основная идея заключается в том, чтобы обеспечить возможность последовательного рассмотрения каждого отдельного аспекта системы в

координации со всеми остальными. Для любой достаточно сложной системы общее число связей, условий и правил обычно превосходит возможности одновременного рассмотрения. В то же время отдельное, в отрыве от других, рассмотрение каждого аспекта системы, чаще всего приводит к неоптимальным решениям в плане как производительности, так и стоимости реализации.

Эти разноплановые требования можно удовлетворить через прохождение этапов *концептуального, логического и физического проектирования*.

Понятие **перспективы** используется в искусстве для определения направления на изображаемый объект – например, вид здания с площади или из окна соседнего небоскреба. Помимо «геометрического» направления, понятие *перспективы может быть обобщено для отображения различного уровня восприятия объекта*, при чем оно достаточно широко применяется и при разработке программных систем.

Рассмотрим различные уровни абстракции, которые используются при описании различных областей архитектуры предприятия.

Уровень контекста описывает внешнюю среду, движущие силы и факторы, оказывающие действие на бизнес организации, видение, стратегию и то, как они влияют на деятельность организации и приоритеты. Этот достаточно полный набор утверждений затем используется в последовательной манере на различных этапах процесса принятия решений, что обеспечивает возможность отследить «в обратную сторону» то, какими внешними факторами, стратегией и видением определялись те или иные решения. В конечном итоге это создает возможность обеспечения соответствия информационных систем требованиям бизнеса.

На уровне контекста разработчики дают ответы на следующие вопросы:

- Каких целей хочет добиться организация?
- Почему организация занимается бизнесом: видение, миссия и цели?
- Каковы тенденции в индустрии, в которой работает организация?
- Как организация расположена и где она работает географически?
- Каковы факторы, определяющие достижение высоких результатов?

– Каковы на самом высоком уровне классы информации, которыми оперирует организация?

– Каковы функции этого бизнеса?

– В каких областях сосредоточена ключевая компетенция организации?

Концептуальный уровень является наиболее абстрактным и описывает те или иные элементы архитектуры в терминах бизнеса организации и в терминах конечных (непрофессиональных в смысле ИТ) пользователей системы. Эта перспектива отвечает на вопрос о том, как организовано и работает предприятие с целью успешной реализации своих задач в условиях, которые накладывает на организацию внешняя среда (контекст).

Концептуальный уровень описывает сервисы и взаимосвязи между сервисами, которые должны быть реализованы для обеспечения принципов, определенных на уровне контекста. Модели, которые создаются на этом уровне, основаны на понятии сервисов (они не содержат специфических деталей стандартов и продуктов, реализующих сервисы), поэтому они носят стабильный характер, если только сам бизнес не претерпевает фундаментальных изменений с точки зрения общего видения и целей. Этот уровень анализа обеспечивает фундамент, на котором можно строить логическую архитектуру. Ключевые **вопросы**, для рассмотрения, следующие:

– Какие области бизнеса должны быть поддержаны ИТ?

– Какая общая бизнес-архитектура (например, «фронт-офис», «мид-офис», «бэк-офис») будет использоваться?

– Как системы будут соотноситься с организационными структурами и бизнес-архитектурой, насколько ИС отдельных департаментов будут консолидированы в единый набор ключевых прикладных систем?

– Как выглядят бизнес-процессы, которые обеспечивают создание продуктов и оказание услуг?

– Какая информация требуется для каждого бизнес-процесса и как эта информация может повторно использоваться?

– Организован ли бизнес организации в централизованном или децентрализованном виде?

– Какой уровень делегирования полномочий должны обеспечивать системы?

– Какие существуют общие принципы по использованию технологий, характерные для индустрии, в которой работает организация, и типы оказываемых услуг?

– Какие вопросы по надзору и руководству использованием технологий должны быть рассмотрены на данном этапе?

Основная задача на этапе концептуального проектирования и создания бизнес-модели состоит в описании ключевых бизнес-процессов и данных, которые эти процессы используют таким образом, чтобы подчеркнуть цели и требования с точки зрения бизнеса в форме, свободной от описания применяемых технологий.

Логический уровень архитектуры показывает основные функциональные компоненты и их взаимосвязи между собой без технических деталей того, как на практике реализована функциональность этих компонент.

Он определяет классы прикладных систем, технологий и данных, которые должны быть поддержаны, но не в терминах конкретных продуктов и технологических решений. Логические модели отвечают на вопрос о том, как требования, идентифицированные в концептуальных моделях, будут реализованы. На этом уровне даются ответы на следующие вопросы:

– Какие приложения необходимы для поддержки бизнес-процессов?

– Кто является основными пользователями и заинтересованными сторонами в реализации данных прикладных систем?

– Как выглядят нормализованные модели данных для этих приложений?

– Какие прикладные системы нужны для управления данными: создания, чтения, внесения изменений и удаления данных?

– Какие нужны технологии для реализации этих прикладных систем?

– Как будет выглядеть распределенная архитектура прикладных систем?

– Какие стандарты должны быть приняты организацией?

Как правило, на логическом уровне существует несколько возможных подходов к построению решения (что отражает различные внешние факторы влияния, например, стоимость, гибкость, безопасность, управляемость).

Если мы говорим об архитектуре приложений, то логический уровень архитектуры приложения создается посредством создания модели приложений. *Модели приложений* описывают общую структуру прикладной системы, ее компоненты и взаимосвязи между ними, последовательности информационного обмена, общего описания данных и состояний, в которых может находиться система и ее компоненты.

Физический уровень описывает принципы проектирования, стандарты и правила, включая группирование критически важных компонент, а также модели развертывания. Это обеспечивает общую основу, в рамках которой на уровне реализации будет выполнена непосредственно разработка. Здесь же определяются критерии отбора технологических решений, которые должны быть либо разработаны, либо приобретены.

Если мы говорим о физическом уровне (архитектуре) прикладной системы, то каждый элемент модели приложения необходимо соотнести с реальными технологиями и технологическими стандартами, что делается через создание *технологических моделей приложения*. Как говорит само название, данный уровень абстракции описывает то, как логические структуры будут физически реализованы. Примеры вопросов, на которые отвечают на данном уровне абстракции, следующие:

- Каковы функциональные спецификации каждой прикладной системы?
- Будет ли организация разрабатывать специализированные приложения или покупать стандартные?
- Каковы критерии выбора и как будут оцениваться различные инициативы по реализации систем?
- Как данные будут представлены на физическом уровне?

При этом технологическая модель должна быть сформулирована в технологически нейтральной по отношению к конкретным поставщикам форме. Если придерживаться такого подхода, то на основе использования этой модели проектирования у организации остается свобода в использовании лучших в своем классе технологий для практической реализации системы в соответствии с разработанной архитектурой.

На следующем этапе проектирования технологическая модель приложения может быть материализована в форме **модели реализации**, что уже связано, например, с написанием детального кода программ, реализующих бизнес-логику. То есть модель реализации включает конкретные модели оборудования, топологию сети, производителя и версию СУБД, средства разработки и, собственно, готовый программный код.

Уровень реализации, самый нижний уровень или перспектива в описании архитектуры системы формулируется уже разработчиками системы в терминах использования тех или иных продуктов конкретных поставщиков.

Таблица 1 – Пример рассмотрения системы на различных уровнях абстракции

Перспектива (уровень абстракции)	Уровень детализации
Контекст	Компания представляется в виде «черного ящика» и является центральным «действующим лицом» (фактором) Бизнес моделируется с точки зрения внешних для бизнеса факторов Моделируются только бизнес-взаимодействия, средства игнорируются
Концептуальный	Моделируются потоки работ бизнес-процессов, идентифицированных на концептуальном уровне Система, реализующая процессы, является центральным актером в форме «черного ящика» Бизнес-процессы моделируются с точки зрения внешних для системы факторов. Рассматриваются средства коммуникаций, используемые для выполнения транзакций
Логический	Моделируется внутренняя архитектура системы Основные компоненты системы являются основными факторами Поведение системы моделируется с точки зрения внутренних для системы «черных ящиков»
Физический	Моделируется физическая структура реализации системы

1.7 Статический и динамический аспект архитектуры предприятия

Бизнес-архитектура на основе миссии, стратегии развития и долгосрочных бизнес-целей определяет необходимые организационную структуру и функциональную модель предприятия, описывающую направленные на реализацию текущих задач и перспективных целей бизнес-процессы. Бизнес-архитектура является областью деятельности высших руководителей, отвечающих за основные функции (бизнес) организации, и, как правило, содержит утверждения по поводу миссии и целей организации, критических факторов успеха, бизнес-стратегии, описания функций, а также структур и процессов, необходимых для их реализации.

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

В динамике архитектура предприятия представляет собой логически связанный цельный план действий и скоординированных проектов, необходимых для преобразования сложившейся архитектуры организации к состоянию, определенному как долгосрочная цель, базирующийся на текущих и планируемых бизнес-целях и бизнес-процессах организации [8, 9]. При этом ключевыми сущностями динамического аспекта АП являются понятия текущей и целевой архитектуры.

Текущая архитектура (Current architecture) описывает существующее состояние архитектуры предприятия. Называется также архитектурой «как есть» (As-Is) или базовым состоянием существующей архитектуры.

Текущая архитектура – это отображение объективной реальности, включающей в себя существующие компоненты (бизнес-процессы, информационные системы, технологические элементы) и их связи. Она представляет собой набор моделей с неизбежными упрощениями, ограничениями и субъективными искажениями.

Процесс разработки текущей архитектуры – это, в первую очередь, процесс документирования и поддержания информации о состоянии предприятия в актуальном виде, обеспечивающий регистрацию и контроль информации обо всех элементах архитектуры предприятия, включающий в себя ведение базы данных по архитектурным объектам; ведение управленческого учета и учета состояния.

Целевая архитектура (Target Architecture) описывает желаемое будущее состояние предприятия или «что должно быть сформировано» (To-Be). Другими словами, целевая архитектура является будущей моделью предприятия.

Целевую архитектуру можно назвать идеальной моделью предприятия, в основу которой заложены:

- стратегические требования к бизнес-процессам и информационным технологиям;
- информация о выявленных «узких местах» и путях их устранения;
- анализ технологических тенденций и среды бизнес деятельности.

Целевая архитектура (модель To-Be) и текущая архитектура (модель As-Is) позволяют описать начальное и конечное состояние предприятия – до и после внесения изменений в его структуру, оставляя без внимания сам процесс изменений.

Процесс перехода от текущей архитектуры предприятия к целевой переводит предприятие на новую спираль развития и, таким образом, мы можем говорить, что архитектура предприятия характеризуется определенным жизненным циклом, похожим на жизненный цикл информационных систем.

Таким образом, АП в общем случае описывается следующими последовательно зависимыми разделами:

- сформулированными миссией и стратегией предприятия, стратегическими целями и задачами;
- бизнес-архитектурой в текущем (As-Is – как есть) и планируемом (To-Be – как должно быть) состоянии;

– системной архитектурой в текущем (As-Is) и планируемом (To-Be) состоянии;

– планами 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 перспективному, определяемому стратегическими целями и задачами, а также преобразования как бизнес-архитектуры, так и системной архитектуры. При поэтапной миграции для целей формализации промежуточных результатов разрабатываются один или несколько промежуточных (миграционных) указанных элементов архитектуры. Планы миграции в соответствии с принятой на предприятии методологией управления проектами формализуются в виде отдельных проектов, включающих:

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

Выполнение плана миграции не означает замораживания развития бизнес- и системной архитектуры [9] (рис. 11).

Таким образом, планируемая системная архитектура является архитектурой To-Be только на определенном витке развития предприятия. Одновременно возврат к стратегическому уровню миссии и стратегических целей и задач не означает необходимость пересмотра миссии и стратегии.

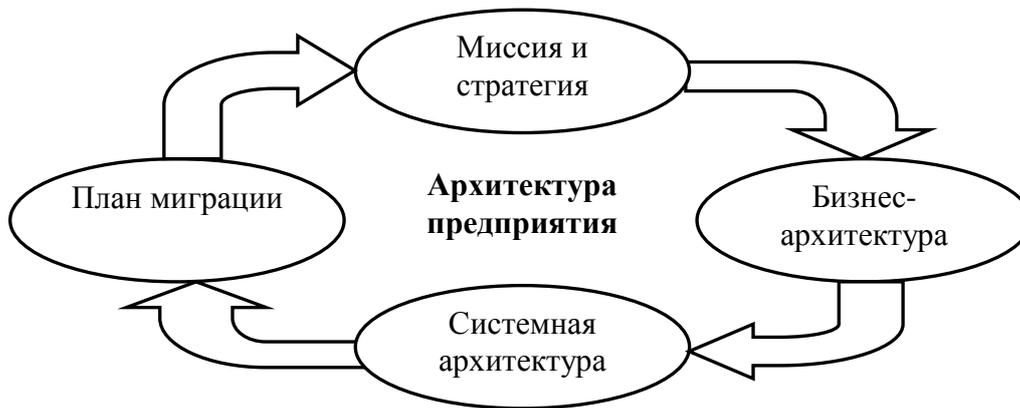


Рис. 11 – Циклическое развитие архитектуры предприятия

Но в конце каждого цикла обязательно проводится анализ эффективности разработанных и осуществленных мероприятий, при необходимости при второй итерации корректируются бизнес-архитектура, системная архитектура, реализуются новые планы миграции. В каждый момент времени может быть несколько циклов, каждый такой цикл не обязательно затрагивает все предприятие в целом, цикл может затрагивать отдельные направления, отдельные вопросы бизнеса и может быть зафиксирован в виде отдельного проекта.

Контрольные вопросы:

1. Что является предметом исследования АП?
2. Раскройте сущность АП и роли ее составляющих.
3. Перечислите ключевые этапы становления АП.
4. В чем заключается статический аспект модели АП?
5. Раскройте понятия текущей и будущей архитектуры.
6. Проанализируйте факторы, позволяющие достичь синхронизации потребностей бизнеса и возможностей ИТ.
7. Дайте характеристику основным доменам АП.
8. Раскройте сущность составных элементов доменов АП.
9. Дайте определение понятиям «перспектива» и «представление» в контексте моделирования АП.
10. Охарактеризуйте уровни абстракции, используемые при описании АП.

ТЕМА 2 БИЗНЕС-АРХИТЕКТУРА ПРЕДПРИЯТИЯ

2.1 Контекст и основные элементы бизнес-архитектуры

Ключом к построению хорошей бизнес-архитектуры является определение бизнес-процессов, их функций и характеристик. Это становится основой для построения архитектуры приложений, которые обеспечивают эти процессы.

Бизнес-архитектура предприятия – это целевое построение организационной структуры предприятия, увязанное с его миссией, стратегией, бизнес-целями. В ходе построения бизнес-архитектуры определяются необходимые бизнес-процессы, информационные и материальные потоки, а также организационно-штатная структура.

Под бизнес-архитектурой, как правило, понимается целостная организация бизнес-процессов, организационных, культурных и социальных областей деятельности предприятия. Она учитывает профиль предприятия, его цели, варианты реализации. Архитектура бизнес-процессов определяется основными функциями организации и может меняться под влиянием внешней среды. Бизнес-архитектура предприятия неразрывно связана с процессом его управления.

Под **управлением предприятием** обычно понимается деятельность компании с учетом изменений в окружающей экономической и социальной среде. Управленческий персонал распределяет финансовые, трудовые и материальные ресурсы для максимально эффективного достижения стратегических целей и задач предприятия.

В ходе разработки бизнес-архитектуры подробно рассматриваются различные модели построения предприятия, соответствующие стратегии его развития. *Модели бизнес-архитектуры* могут быть разделены на три класса:

– **классическая** или, другими словами, эталонная архитектура предприятия является идеальной моделью построения организации;

– **специализированная архитектура** – включает в себя модели, ориентированные на предприятия определенных отраслей или определенные фазы производства. В основе специализированных методологических моделей, как правило, лежат исторически сложившиеся алгоритмы управления в данных отраслях (например, банки, химическая промышленность, телекоммуникации);

– **специфическая архитектура** – так обычно называют исторически сложившуюся на данном предприятии модель бизнес-процессов.

Общее видение бизнес-архитектуры предприятия включает (рис. 12):

- анализ основных функций, цепочек создания добавленной стоимости;
- модели бизнес-сценариев;
- анализ информационных связей и процессов.

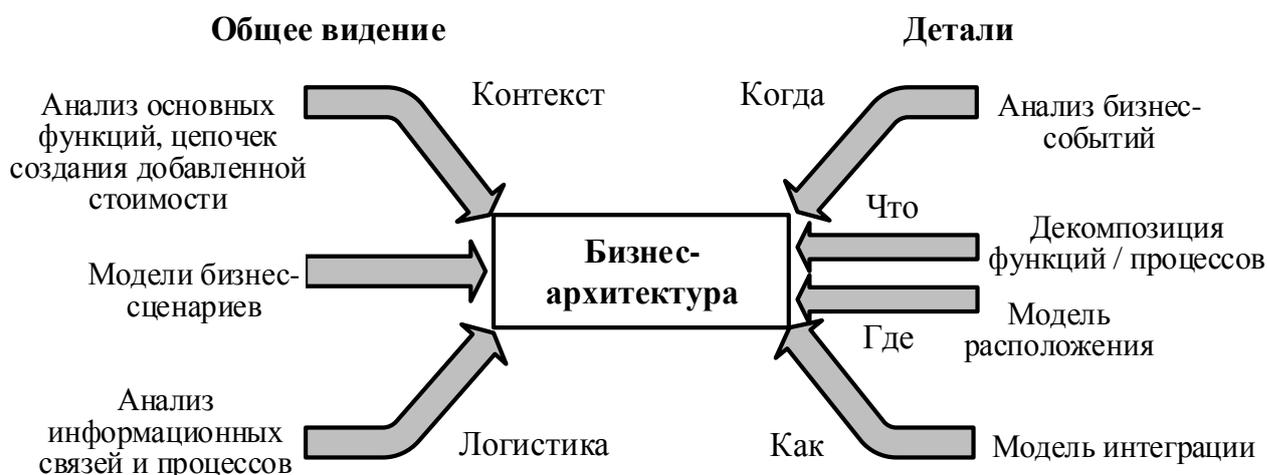


Рис. 12 – Общее видение бизнес-архитектуры

Построение бизнес-архитектуры начинается с общего обзора ситуации, который предполагает поиск ответов на следующие вопросы:

- Каков внешний контекст деятельности организации?
- В чем состоят основные функции и добавочная стоимость, которая является итогом деятельности организации?

– Какие сценарии развития бизнеса необходимо учитывать, и какова вероятность их реализации?

– Какие необходимы информационные взаимосвязи и процессы обработки информации?

Бизнес-архитектура как составной элемент архитектуры предприятия на основании миссии, стратегии развития и долгосрочных бизнес-целей определяет необходимые организационную структуру и бизнес-процессы, описываемые *функциональной моделью предприятия*, на основе которых в свою очередь реализуется процесс разработки и реализации продуктов.

Бизнес-архитектура включает [8]:

– 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;

– 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.

2.2 Предприятие как объект исследования бизнес-архитектуры

Предприятие [10] – это имущественный комплекс, используемый для осуществления предпринимательской деятельности. Предприятие как имущественный комплекс признается недвижимостью. Предприятие в целом или его часть могут быть объектом купли-продажи, залога, аренды и других сделок, связанных с установлением, изменением и прекращением вещных прав.

В современном экономическом словаре **предприятие** трактуется как обособленная специализированная единица, основанием которой является профессионально организованный трудовой коллектив, способный с помощью имеющихся в его распоряжении средств производства изготовить нужную потребителю продукцию (выполнить работы, оказать услуги) соответствующего значения, профиля и ассортимента [11].

Предприятия классифицируют по различным признакам:

по форме собственности:

- государственной;
- муниципальной;
- частной;
- собственности общественных организаций;
- иных форм собственности (смешанная собственность, собственность иностранных лиц, лиц без гражданства);

по масштабу:

- малые;
- средние;
- крупные;

по организационно-экономической форме:

- индивидуальные;

- партнёрства (хозяйственные товарищества и общества);
- корпорации (акционерные общества, госкорпорации);

по цели деятельности:

- коммерческие;
- некоммерческие;

по отраслевому и функциональному виду деятельности:

- промышленные;
- сельскохозяйственные;
- строительные;
- транспортные;
- торговые;
- производственно-торговые;
- торгово-посреднические;
- инновационно-внедренческие;
- лизинговые;
- банковские;
- страховые;
- туристические и др.;

по технологической целостности и степени подчиненности:

- головные;
- дочерние;
- ассоциированные;
- филиалы.

Как видно из приведенной классификации, деятельность предприятия не обязательно связана с коммерческой деятельностью. Это может быть и государственная организация, и общественное, в том числе неформальное, объединение участников, связанных общей целью.

Согласно более общему определению **предприятие** представляет собой комплексную систему культурных, технологических и процессных компонентов, организованных для достижения целей организации [10].

Предприятия (организации, фирмы) имеют право на добровольной основе объединять свою научно-техническую, производственную, коммерческую и другие виды деятельности, если это не противоречит действующему антимонопольному законодательству. Предприятия и другие первичные субъекты хозяйствования в рыночной экономике могут создавать разные по принципам и целям *добровольные объединения*:

– **ассоциации** – самая простая форма договорного объединения предприятий (фирм, компаний, организаций) с целью постоянной координации хозяйственной деятельности. Ассоциация не имеет права вмешиваться в производственную и коммерческую деятельность любого из ее участников (членов);

– **картели** – договорные объединения предприятий преимущественно одной отрасли для осуществления совместной коммерческой деятельности и регулирования сбыта изготовленной продукции;

– **синдикаты** — организационная форма существования разновидности картельного соглашения, предусматривающего реализацию продукции участников через создаваемый совместный сбытовой орган или сбытовую сеть одного из участников объединения. Таким же образом может быть организована закупка сырья для всех участников синдиката. Эта форма объединения предприятий присуща отраслям с массовым производством однородной продукции;

– **тресты** – монополистические объединения предприятий, ранее принадлежавших различным предпринимателям, в единый производственно-хозяйственный комплекс. При этом предприятия полностью теряют свою юридическую и хозяйственную самостоятельность, поскольку интегрируются все направления их деятельности;

– **корпорации** – договорные объединения субъектов хозяйствования на основе интеграции их научно-технических, производственных и коммерческих интересов с делегированием отдельных полномочий для централизованного регулирования деятельности каждого из участников;

– **концерны** – форма уставных объединений предприятий, характеризующаяся единством собственности и контроля. Объединение происходит чаще всего по принципу диверсификации, когда один концерн интегрирует предприятия разных отраслей экономики (промышленности, транспорта, торговли, науки, банковского или страхового дела). После создания концерна субъекты хозяйствования теряют свою самостоятельность, подчиняясь мощным финансовым структурам. В современных условиях значительно расширяется сеть международных концернов;

– **консорциумы** – временные уставные объединения промышленного и банковского капитала для достижения общей цели (например, осуществления совместного крупного хозяйственного проекта). Участниками консорциума могут быть государственные и частные фирмы, а также отдельные страны (например, Международный консорциум спутниковой связи);

– **холдинги (холдинговые компании)** – специфическая организационная форма объединения капиталов: интегрированное общество, которое непосредственно не занимается производственной деятельностью, а использует свои средства для приобретения контрольных пакетов акций предприятий, являющихся участниками концерна или другого добровольного объединения, с целью контролирования деятельности таких предприятий. Объединяемые в холдинги субъекты имеют юридическую и хозяйственную самостоятельность, однако право решения основных вопросов их деятельности принадлежит холдинговой компании;

– **финансовые группы** – объединения юридически и экономически самостоятельных предприятий разных отраслей экономики. В отличие от концерна, во главе финансовых групп становятся один или два банка, которые распоряжаются капиталом предприятий, входящих в состав финансовых групп, координируют все сферы их деятельности.

В настоящее время основными формами добровольных объединений предприятий (фирм, организаций) все больше становятся концерны, корпорации и финансовые (промышленно-финансовые) группы.

Предприятия помимо внешней интеграции могут иметь *сложную внутреннюю структуру*, с которой также могут происходить изменения. На рис. 1.1. приведен пример производственной структуры машиностроительного предприятия [8, 11].

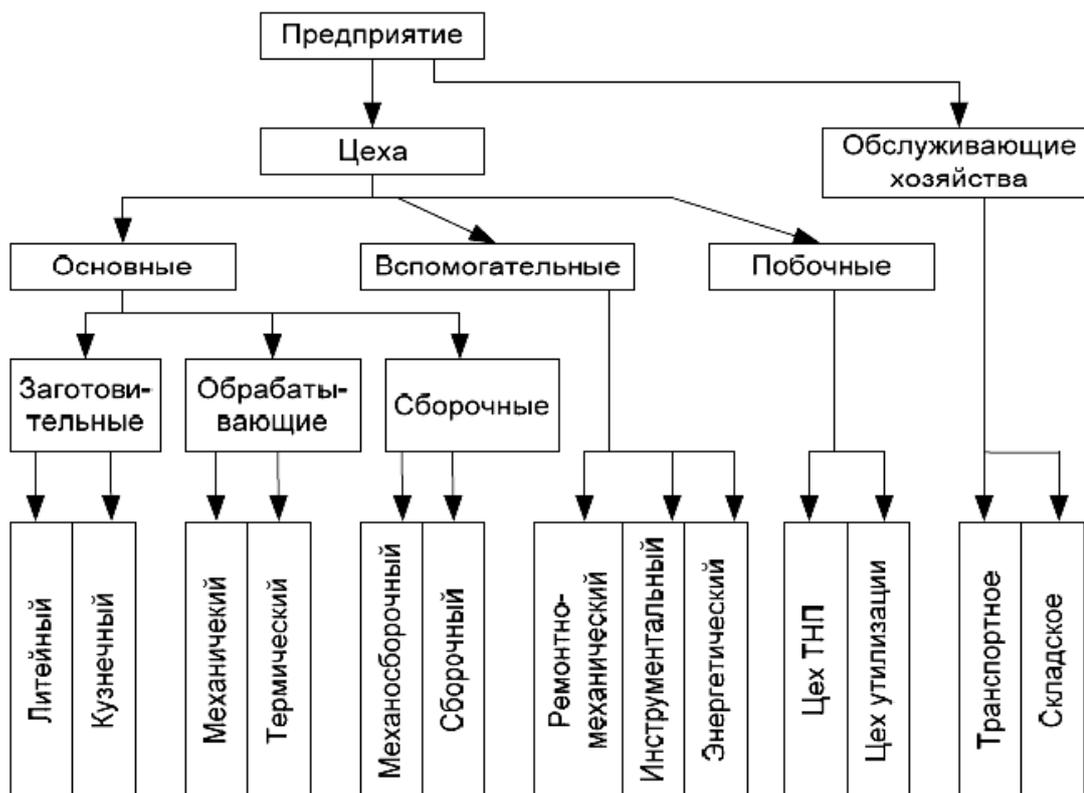


Рис. 13 – Производственная структура машиностроительного предприятия

В состав любого предприятия входят не только производственные подразделения, но и отделы аппарата управления, учреждения культурно-бытового назначения и др. Поэтому наряду с производственной структурой существует *общая структура предприятия*. Ее образует совокупность всех производственных, непроизводственных и управленческих подразделений предприятия. Типовая общая структура промышленного предприятия представлена на рис. 1.2.

Проанализируем роли ключевых элементов организационной структуры предприятия.

Экономическую службу на предприятии возглавляет *главный экономист* (заместитель директора по вопросам экономики), который отвечает за организацию плановой работы на предприятии. Подчиненные ему отделы осуществляют контроль над выполнением плановых задач, производят анализ деятельности предприятия. В его компетенции находятся также вопросы финансов, организации труда и заработной платы.

Главная задача *заместителя директора по производству* – обеспечение выполнения планов предприятия. С этой целью начальник производства и подчиненный ему производственный отдел, разрабатывающий оперативные планы выпуска продукции для каждого цеха, обеспечивают ритмичную работу по их выполнению, осуществляют контроль и регулирование производственного процесса.

Маркетинговые функции изучения спроса, рынков сбыта, рекламы, продвижения товаров, а также материально-технического обеспечения производства возложены на *заместителя директора по коммерческим вопросам*.

Заместитель директора по кадровым и социальным вопросам отвечает за реализацию кадровой политики предприятия, занимаясь вопросами отбора персонала, профессиональной ориентации и социальной адаптации, обучения, повышения в должности, увольнения и т. п. Кроме того, ему подчиняются службы, удовлетворяющие социальные потребности персонала.

Несколько подразделений аппарата управления предприятия подчинено непосредственно директору. Учет производства, контроль над использованием средств и соблюдением финансовой дисциплины, составлением баланса, расчеты с рабочими и служащими осуществляет *бухгалтерия*.

Функции контроля качества продукции, предупреждения брака, разработки и внедрения систем управления качеством возложены на независимый от любого заместителя директора *отдел технического контроля*.

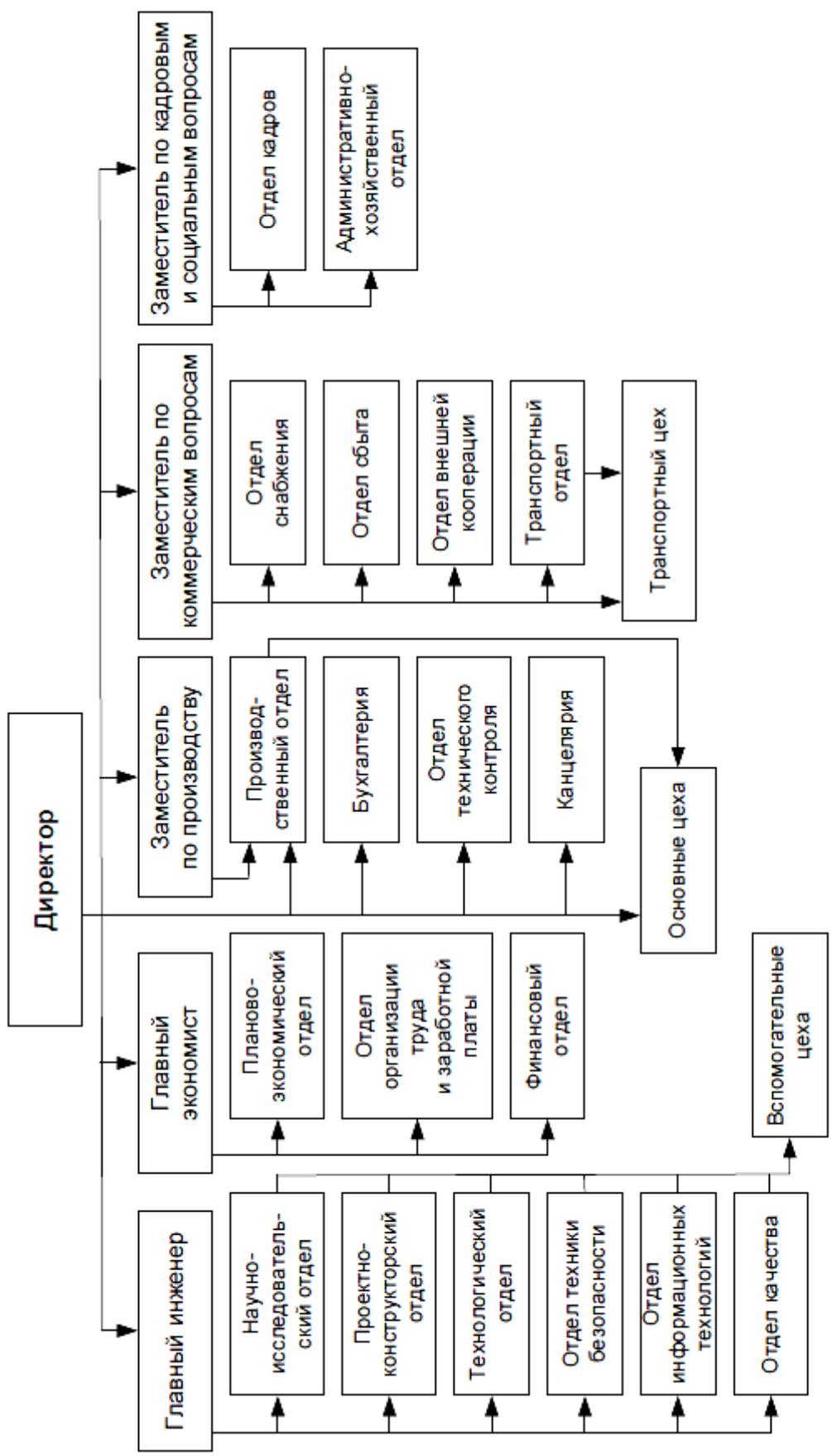


Рис. 14 – Типовая общая структура промышленного предприятия

Реализацию делопроизводства на предприятии, приемку входящей документации, ее регистрацию, учет, распределение, организацию внешнего документооборота, отправку и хранение документов обеспечивает *канцелярия*.

Директору предприятия также непосредственно подчинены *начальники цехов*, осуществляющие техническое и хозяйственное руководство соответствующими подразделениями. В состав аппарата управления цехом входит *заместитель начальника цеха по подготовке производства*. Он занимается вопросами разработки технологических процессов, обеспечивает участки необходимой документацией и оснасткой.

Помощник начальника цеха по производству осуществляет оперативное руководство производственными процессами.

Механик цеха организует ремонт оборудования и надзор за его эксплуатацией. *Начальник цеха* осуществляет управление производством с помощью *мастеров производственных участков*, которые уже непосредственно или через бригадиров организуют труд исполнителей.

Экономической работой в пределах цеха занимается *экономист*, а вопросами нормирования и оплаты труда – *нормировщик*.

2.3 Миссия и стратегия: роль в построении бизнес-модели

Миссия – одно из основополагающих понятий стратегического управления. Разные ученые давали различные формулировки миссии, но в целом можно выделить два подхода к пониманию миссии [12].

В широком смысле **миссия** – это философия и предназначение, смысл существования организации. *Философия организации* определяет ценности и принципы, в соответствии с которыми организация намеревается осуществлять свою деятельность. *Предназначение* определяет действия, которые организация намерена осуществлять, и цели, которых она намерена достичь. Философия организации определяется на этапе становления организации и редко меняется,

тогда как предназначение может меняться в процессе изменений как внутренней, так и внешней среды организации;

В узком понимании **миссия** – это определение целей и причин существования организации, т. е. миссия в таком понимании должна раскрывать смысл существования организации, в котором проявляется отличие данной организации от ей подобных.

Стратегия в широком толковании представляет собой искусство руководства общественными процессами, общий план руководства. Применительно к предприятию (организации), по определению Е. Кассельса, **стратегия** – это модель поведения, которой следует организация для достижения своих долгосрочных целей. В свою очередь, **стратегическое планирование** – это процесс осуществления систематизированных и взаимосогласованных работ с определением долгосрочных целей и направлений деятельности предприятия [8].

На основе общей миссии предприятия формулируются и устанавливаются общефирменные цели, которые должны отвечать определенным требованиям, суть которых состоит в следующем:

1) **конкретность и возможность измерения**. Формулирование целей в конкретных формах создает исходную базу для последующих правильных хозяйственных и социальных решений. Благодаря этому можно более точно определить, насколько эффективны действия предприятия по осуществлению поставленных целей;

2) **ориентированность во времени** (обозначение конкретных горизонтов прогнозирования). Цели устанавливаются на длительные или короткие промежутки времени. Долгосрочная цель имеет горизонт прогнозирования равный нескольким годам (пять, семь, десять лет); краткосрочная – в пределах одного года;

3) **достижимость и направленность на повышение эффективности деятельности предприятия**. Недостижимые или достижимые лишь частично цели приводят к негативным последствиям: блокированию стремления

работников эффективно хозяйствовать, снижению уровня мотивации, ухудшению показателей инновационной и производственной деятельности предприятия, снижению конкурентоспособности продукции на рынке;

4) **взаимосогласованность и поддерживаемость множественных целей предприятия**, т. е. действия и решения, необходимые для достижения одной цели, не могут препятствовать реализации других целей. Иное может привести к возникновению конфликтной ситуации между подразделениями предприятия (фирмы), ответственными за достижение различных целей.

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 [13]. На рис. 15 [8] отражены главные направления деятельности предприятия, которые могут декларироваться стратегией предприятия.

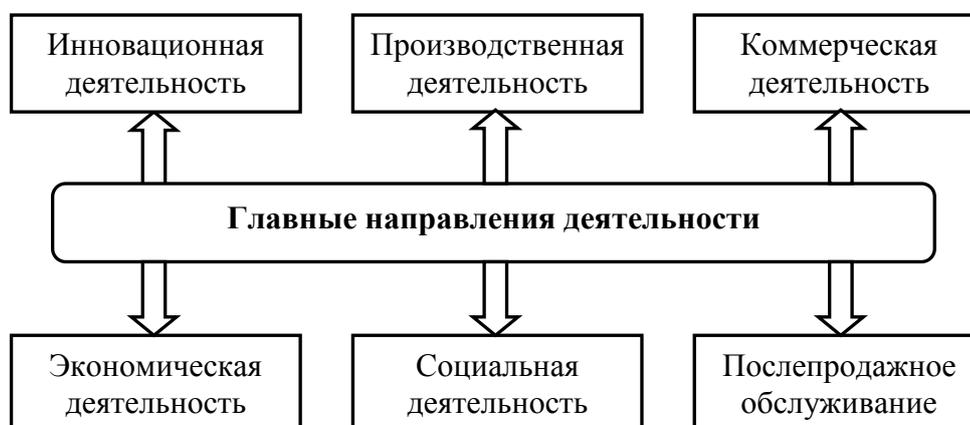


Рис. 15 – Главные взаимосогласованные направления деятельности предприятия

По своему содержанию стратегия является долгосрочным плановым документом, результатом стратегического планирования. *Основные этапы стратегического планирования* представлены на рис. 16.

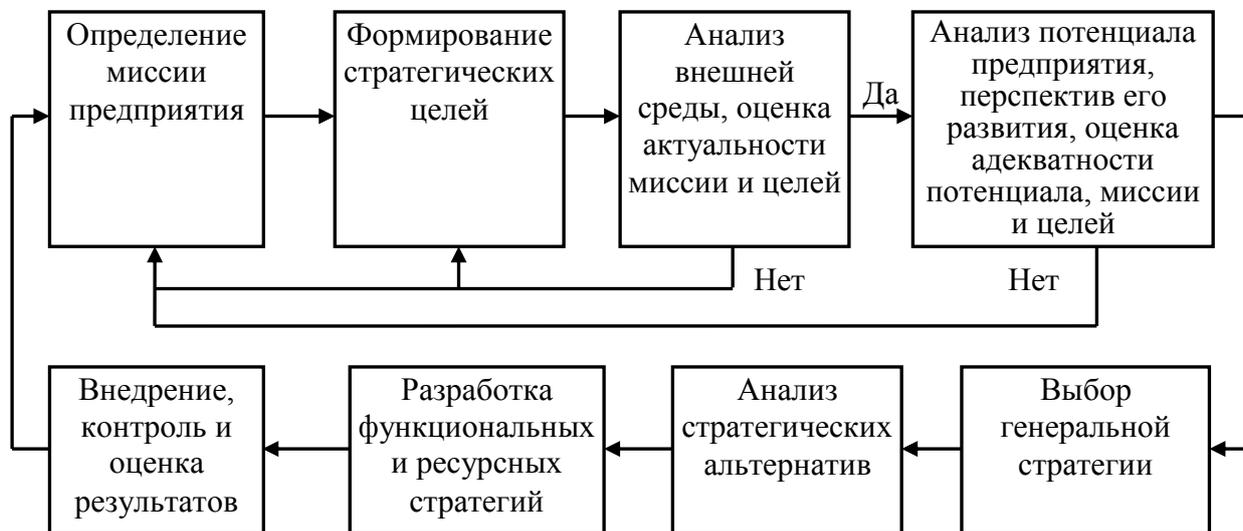


Рис. 16. Основные этапы стратегического планирования

Реальность и эффективность стратегии предприятия обеспечиваются при условии формулирования стратегических целей с учетом следующих требований:

- конкретность формулировки целей и их измеримость;
- четкая ориентированность целей во времени;
- достижимость, сбалансированность и ресурсная обеспеченность;
- однонаправленность и взаимосоответствие целей.

При этом рекомендуется устанавливать цели для каждого направления деятельности предприятия.

После определения миссии и целей начинается *диагностический этап* стратегического планирования. Первым важным шагом диагностического этапа является изучение внешней среды.

Анализ внешней среды представляет собой непрерывный процесс наблюдения, изучения и контроля внешних для предприятия факторов с целью

своевременного исчерпывающего определения возможных положительных явлений либо угроз субъекту хозяйствования.

Для разработки и осуществления стратегии большое значение имеет анализ рыночных факторов, проводимый в рамках анализа внешней среды. Рыночные факторы вследствие своей постоянной и высокой изменчивости оказывают непосредственное влияние на успешность предприятия. Речь идет в первую очередь о микроэкономическом анализе спроса, предложения и уровня конкуренции по определенной системе показателей.

Методы выбора генеральной стратегии можно разделить на две группы:

- 1) методы однопродуктового анализа;
- 2) методы «портфельного» анализа (матричные).

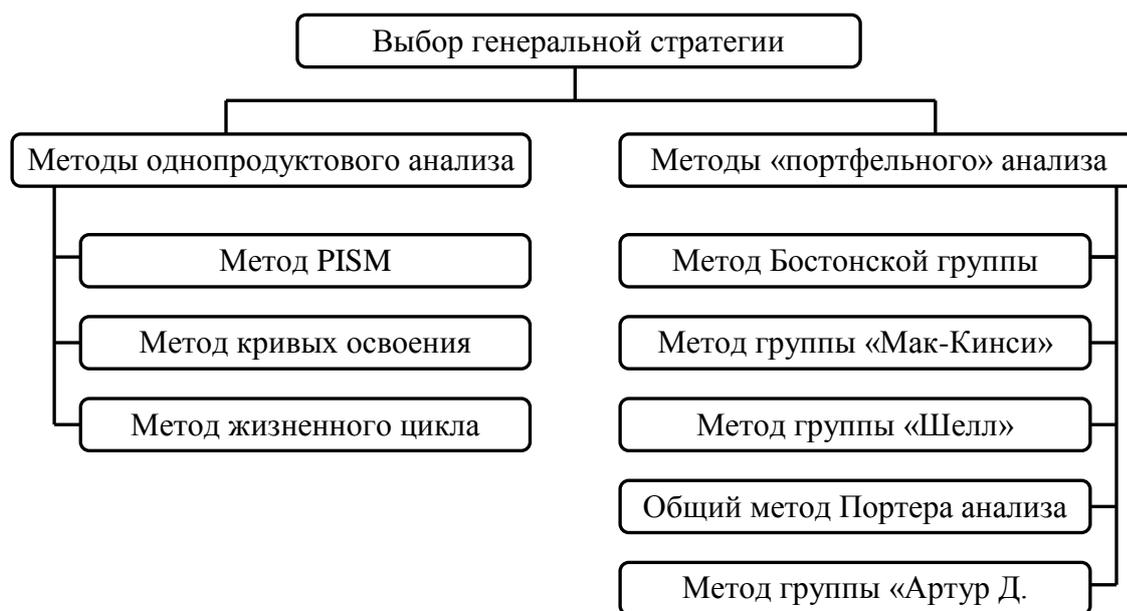


Рис. 18. Возможные методы выбора генеральной стратегии предприятия

Стержнем стратегического плана предприятия является *базовая стратегия*. В соответствии с циклом развития предприятия можно выбрать одну из базовых стратегий:

– **стратегию роста**, характеризующую намерение предприятия увеличивать объемы продажи, прибыли, капитальных вложений и т. п.;

– **стратегию стабилизации** при ощутимой нестабильности объемов продажи и прибыли;

– **стратегию выживания** – сугубо оборонную стратегию, применяемую при глубоком кризисе предприятия.

В рамках соответствующей базовой стратегии можно выбрать одно из нескольких возможных действий, которые принято называть стратегическими альтернативами (табл. 2).

Таблица 2 – Стратегические альтернативы на основе базовой стратегии

Варианты базовой стратегии	Критерии стратегии	Стратегические альтернативы
1. Стратегия роста	1. Объем продаж. 2. Доход. 3. Доля рынка. 4. Скорость рынка	1. Интенсификация рынка: захват новых рынков; географическая экспансия. 2. Диверсификация. 3. Межфирменное сотрудничество и кооперация. 4. Внешнеэкономическая деятельность
2. Стратегия стабилизации (наступательно-оборонительная)	1. Доход от объема продаж. 2. Доход от активов 3. Доход от акций. 4. Скорость оживления деятельности	1. Экономия: ревизия затрат, консультация, оживление деятельности. 2. Сдвиги: уменьшение затрат, восстановление дохода, активизация финансовой деятельности. 3. Обеспечение устойчивости: селективность, балансирование на рынках, финансовая экономия
3. Стратегия выживания (оборонительная)	1. Показатели анализа продуктов и рынков. 2. Показатели анализа финансового состояния. 3. Показатели анализа управления	1. Перестройка маркетинговой деятельности: изъятие товара из продажи

По функциональному признаку базовая стратегия может быть представлена маркетинговой стратегией, стратегией научно-исследовательских и экспериментально-внедренческих работ, производственной стратегией.

По ресурсному признаку выделяют стратегии кадров и социального развития, стратегии материально-технического обеспечения, организационные

стратегии, инвестиционные стратегии, стратегии технического развития и финансовые стратегии.

Каждая базовая стратегия, как правило, содержит следующие *компоненты*:

- цели, условия и основные направления деятельности в той или иной сфере, конечные результаты, обеспечивающие внедрение ресурсных стратегий;
- порядок и последовательность (в пространстве и во времени) решения качественных и количественных задач долгосрочных планов; ряд мероприятий, адекватных назначению базовой стратегии, что обеспечивает достижение поставленной цели.

Компоненты стратегического планирования отражаются в бизнес-плане предприятия.

Бизнес-план предприятия – это документ, в котором изложены сущность, направления и способы реализации предпринимательской идеи, охарактеризованы рыночные, производственные, организационные и финансовые аспекты будущего бизнеса, а также особенности управления. Этот документ является разрешительным основанием привлечения инвестиций для разработки и реализации предпринимательской идеи в виде инновационно-инвестиционных проектов.

Помимо стратегического планирования выделяют тактическое и оперативное планирование.

2.4 Организационная структура предприятия

В теории управления **организационная структура** определяется как абстрактная категория, характеризующаяся тремя параметрами:

- степенью сложности;
- степенью формализации;
- степенью централизации.

Сложность отражается множеством отличительных признаков организации. Чем глубже разделение труда, чем больше вертикальных уровней в иерархии управления и структурных подразделений, тем сложнее координировать деятельность людей в организации.

Объем разработанных правил и процедур, руководствуясь которыми организация направляет поведение своих сотрудников, и представляет собой *формализацию*. Чем больше правил и регуляторов, определяющих объем запрещенных и разрешенных действий работников организации, тем более формализованной является структура организации.

Централизация определяет место, где сосредоточено право принятия решений. Если все решения (или их большинство) принимаются высшими руководителями, тогда организация является централизованной. Децентрализация означает, что право принятия решений передается (делегировается) с высших уровней управления на низшие уровни.

В процессе реализации вышеназванных функций задача менеджмента заключается в разработке для всех компонентов организации такой формы и способов «состыковки» друг с другом, при которой предприятие не утрачивает целостности и функционирует максимально эффективно. Поэтому на любом предприятии имеет место соподчиненность его составных частей, уровней управления, четкое разделение власти, прав и ответственности.

Правила формирования организационной структуры предприятия [12]:

– стратегически важные для компании виды деятельности должны стать основными звеньями структуры, а статус их руководителей должен соответствовать значимости видов деятельности;

– при изменении стратегии обязательна корректировка организационной структуры;

– сторонним организациям можно передавать виды деятельности, которые выполняются ими дешевле, быстрее и более качественно. Как правило, эти виды деятельности не являются основными для компании, но иногда

передаются и основные, если последние не определяют конкурентное преимущество компании;

– все аспекты стратегически значимых видов деятельности должны быть подотчетны одному руководителю;

– взаимосвязанные виды деятельности должны выполняться скоординированно. Координация улучшается при встраивании в структуру специальных инструментов взаимодействия.

Для изображения структурных взаимосвязей основных уровней и подразделений предприятия, их соподчиненности на практике используются *схемы организационной структуры управления*. Эти схемы являются только «скелетом» системы управления, поскольку не раскрывают состава и содержания функций, прав и обязанностей подразделений и должностных лиц.

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 – это структура, между элементами которой существует только одноканальное взаимодействие. При такой структуре управления каждый подчиненный имеет лишь одного руководителя, который и выполняет все административные и специальные функции в соответствующем структурном подразделении.

Преимущества организационной структуры управления линейного типа: четкость взаимоотношений, 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 (звеньев).

Основой **функциональной организационной структуры** управления является разделение функций управления между отдельными подразделениями аппарата управления (рис. 19).



Рис. 19 – Пример функциональной структуры

Данное деление носит общий и неполный характер. Каждая функциональная структура должна учитывать специфику конкретной фирмы.

Преимуществами функциональной структуры является централизация, профессионализм и экономичность [14].

Централизация. В ситуации когда каждый функциональный руководитель отвечает только за одну производственную функцию, ответственность за реализацию производственного процесса в целом лежит на генеральном директоре компании.

Профессионализм. Функциональная структура стимулирует профессиональную специализацию работников, обеспечивает рост их квалификации, облегчает задачу координации деятельности специалистов внутри департамента, а в результате обеспечивает быстроту и четкость решения профессиональных задач.

Экономичность. Данная структура, в отличие от других, исключает дублирование функций в подразделениях организации.

Недостатки функциональной структуры управления: нарушение принципа единоначалия, противоречивость распоряжений, сложность координации деятельности управленческих служб, снижение оперативности работы органов управления. Работника в функциональной структуре управления отличает узкопрофессиональное видение проблем, ориентация на цели подразделения, в котором он работает, в ущерб общим целям организации. Данная структура не способствует развитию духа новаторства и предпринимательства.

Линейно-функциональная организационная структура управления опирается на распределение полномочий и ответственности по функциям управления и порядок принятия решений по вертикали. Она позволяет организовать управление по линейной схеме (директор → начальник цеха → мастер), а функциональные отделы аппарата управления предприятия лишь помогают линейным руководителям решать управленческие задачи. При этом линейные руководители не подчинены руководителям функциональных отделов аппарата управления.

Преимущество линейно-функциональной структуры состоит в быстрой реализации управленческих решений вследствие иерархичности ее построения, что способствует повышению уровня специализации и эффективности работы функциональных служб, делает возможным необходимый маневр ресурсами. Такая структура является наиболее целесообразной при массовом производстве с устоявшимся ассортиментом продукции и незначительными эволюционными изменениями технологии ее изготовления.

Недостаток состоит в том, что в условиях частых технологических изменений, обновления номенклатуры продукции использование этой структуры замедляет сроки подготовки и принятия управленческих решений, не обеспечивает надлежащей согласованности в работе функциональных отделов (подразделений).

Дивизиональная организационная структура управления базируется на углублении разделения управленческого труда. Применение данной структуры сопровождается процессом децентрализации 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. Следовательно, при дивизиональной структуре каждое производственное подразделение корпорации (концерна) имеет собственную достаточно разветвленную структуру управления, обеспечивающую автономное его функционирование. Лишь стратегические функции управления централизованы на корпоративном уровне. При дивизиональной структуре управления группирование видов деятельности субъекта хозяйствования осуществляется с применением принципа разделения труда по целям. Это означает, что вокруг определенного производства формируется автономная организационная общность (рис. 20).

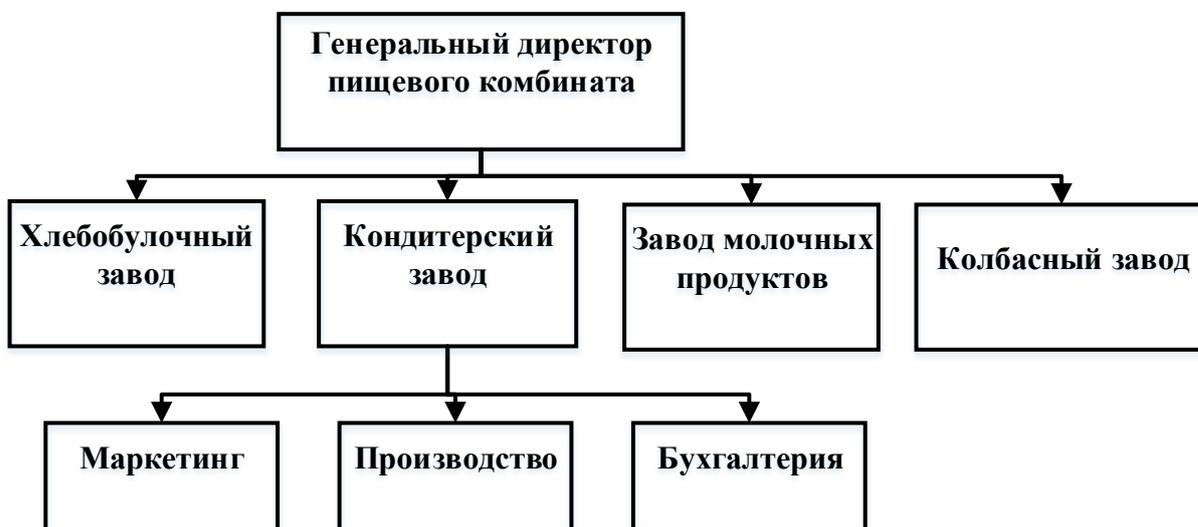


Рис. 20 – Пример дивизиональной структуры

При дивизиональной организационной структуре управления возможны три способа группирования производственных подразделений [13]:

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

В сильно диверсифицированных корпорациях дивизионы формируются на основе видов бизнеса или стратегических бизнес-групп, объединяющих до нескольких десятков предприятий родственных сфер деятельности. При этом вводится должность президента стратегической бизнес-группы, функциями которого являются общее руководство, координация и контроль деятельности предприятий группы. Президент группы, с одной стороны, является проводником политики высшего руководства компании в своей группе, а с другой – отстаивает интересы своих предприятий в головном офисе компании, что обеспечивает необходимый баланс интересов в фирме. Кроме того, сокращается число руководителей подразделений, подотчетных непосредственно высшему руководству, что повышает управляемость компании.

Преимуществами дивизиональной организационной структуры управления являются гибкое реагирование на изменения во внешней среде, быстрое принятие управленческих решений и улучшение их качества. Корпоративное руководство освобождается от оперативных функций и имеет возможность сконцентрироваться на стратегии. Недостатком является необходимость увеличения при такой структуре численности аппарата управления и затрат на его содержание. Чем более независимы дивизиональные лидеры от высшего руководства, тем опаснее разрыв стратегического и оперативно-тактического руководства. Утратив контроль над ситуацией в дивизионах, руководство компании не в состоянии сформировать реалистическую общую стратегию, а лидеры дивизионов, лишённые ориентиров развития, могут принимать решения, противоречащие интересам фирмы в целом.

Рассмотренные организационные структуры управления (линейная, функциональная, линейно-функциональная, дивизиональная) являются жестко

упорядоченными механическими структурами. С начала 60-х гг. XX в. получили распространение более гибкие типы организационных структур, способные модифицироваться в соответствии с изменениями внешней среды и потребностями самой организации. Такие структуры получили название *адаптивных, или органических*.

Основными *типами органических структур* являются проектные и матричные структуры.

Проектная организационная структура возглавляется руководителем, который обеспечивает реализацию проекта в определенный срок и в пределах выделенной сметы. Для осуществления проекта формируется временная команда специалистов из представителей необходимых функциональных служб. Руководителю проекта полностью подчинены выделенные под его реализацию ресурсы и члены проектной команды. По завершении проекта команда распускается, специалисты начинают работать над новым проектом либо возвращаются в свои подразделения.

Преимуществом структуры является возможность полной концентрации членов проектной группы и ее руководителя на выполнении одного проекта. Данная структура — одна из наименее бюрократизированных: решения принимаются всей командой. Должность руководителя позволяет координировать действия с другими подразделениями. Проектная структура является идеальной организационной базой для решения проблем инновационного характера [14].

С проектной структурой связан ряд специфических *трудностей*:

– продолжительный период «настраивания» команды на работу, включающий периоды формирования команды, приспособления ее членов друг к другу и нормализации отношений внутри коллектива;

– проблема трудоустройства высвободившихся специалистов по завершении проекта или отказе от проекта;

– дублирование существующих в организации функциональных служб – самый существенный недостаток структуры.

В матричной организационной структуре управления наряду с линейными руководителями предприятия и функциональным аппаратом управления формируются еще и временные специализированные звенья (проектные группы), которые создаются из специалистов постоянных функциональных отделов и лишь временно подчиняются руководителю проекта. Принципиальным моментом в матричной структуре является подчинение сотрудников двум руководителям одного иерархического уровня. Кроме того, матричная структура – это постоянная структура, в отличие от проектной (рис. 20).

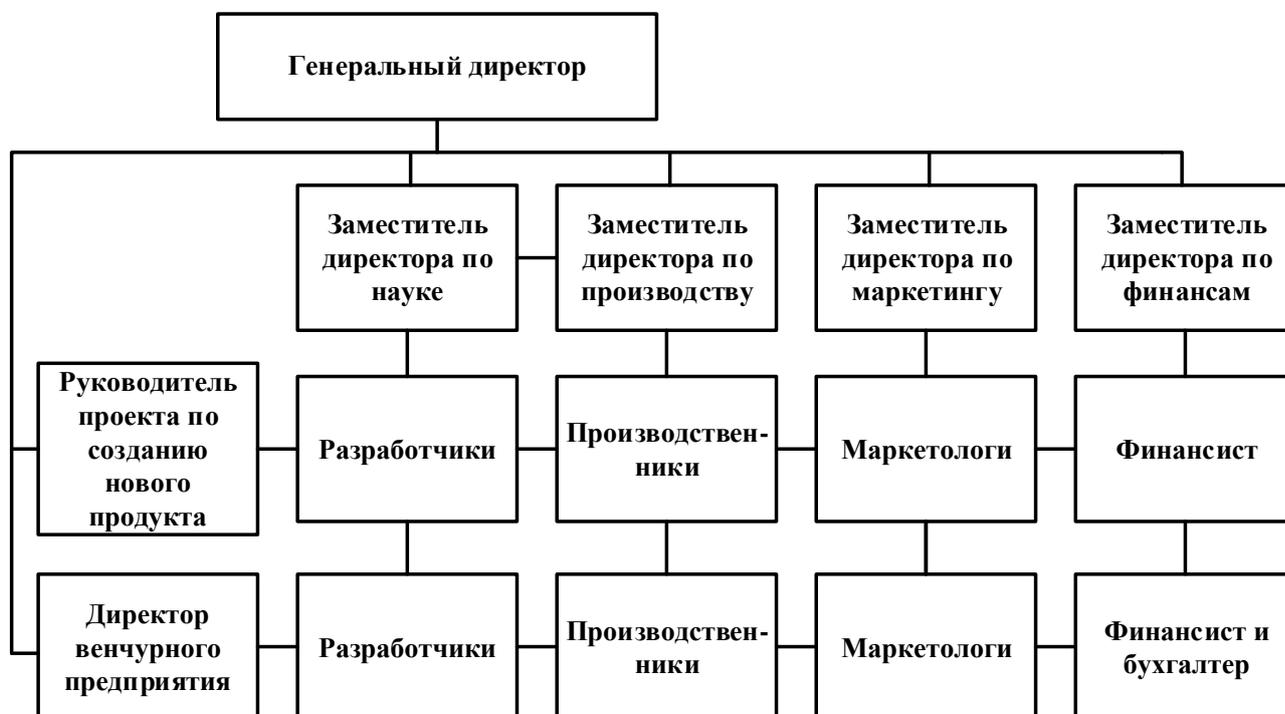


Рис. 20 – Пример матричной организационной структуры

Руководитель проекта ставит задачи перед членами проектной команды, составляет графики выполнения задач, осуществляет контроль затрат, сроков, количественных и качественных показателей. Руководители функциональных служб определяют способ решения поставленных перед соответствующими специалистами задач и контролируют ход их решения.

Матричная структура обладает всеми преимуществами проектной структуры, при том что лишена некоторых ее недостатков. Она позволяет перераспределять функциональных специалистов между проектами, избегать дублирования функциональных служб, в ней отсутствует проблема трудоустройства специалистов после завершения проекта.

Матричные структуры управления имеют и определенные *недостатки*: увеличение численности управленческого персонала и количества информационных связей между работниками подразделений, возможные конфликтные ситуации между ними; психологические проблемы персонала, связанные с неопределенностью и переменчивостью структуры.

Новые формы производственных структур.

Конец XX в. ознаменовался переходом от индустриального общества к информационному, для которого свойственны стремительные темпы изменения технологической, экономической и институциональной среды. На смену массовому производству стали приходить гибкие производственные системы, одновременно обеспечивающие большие объемы производства и приспособленные к работе под заказ потребителя.

Появились новые методы менеджмента, а, следовательно, и новые производственные структуры [15]:

- модель «подтянутого производства»,
- аутсорсинг,
- модель «горизонтальной корпорации»,
- сетевая организационная структура.

Модель «подтянутого производства» обеспечивает экономию затрат посредством автоматизации труда, компьютеризированного контроля над рабочими, «уплощения» производственной структуры, однако по существу производственную структуру она не меняет.

Широкое распространение получила вертикальная дезинтеграция корпораций, которая осуществлялась путем **аутсорсинга** – передачи субподрядов мелким и средним фирмам, отличающимся высокой

производительностью и гибкостью. Использование сети субподрядных фирм придавало корпорации определенную гибкость, недостаточную, однако, для существования в условиях непрерывно меняющейся среды.

Создатели модели **«горизонтальной корпорации»** полагали, что горизонтальные внутренние связи (координация деятельности подразделений) и горизонтальные внешние связи (кооперация) оказывают на эффективность работы больше влияния, чем традиционная вертикальная система управления, и в качестве организационных звеньев создали самоуправляемые комплексные рабочие группы.

Сетевая организационная структура является наиболее адекватной формой для новой экономики, как информационной, так и глобальной одновременно. Сетевая структура – это структура, где установлены устойчивые отношения координации и взаимодействия между самоуправляемыми фирмами [14]. Сети возникают в случаях, когда для координации деятельности фирм уже недостаточно чисто рыночных механизмов, а интеграция в рамках единой корпорации не обеспечивает необходимой гибкости системы. Сетевая же структура отличается особой гибкостью. Сеть может изменять не только выпуск продукции в связи с непредвиденными обстоятельствами, но и собственную структуру – добавлять новые или исключать ненужные фирмы.

Примерами сетевых структур являются совместные предприятия, соглашения об участии в совместных исследовательских проектах и производственных программах, консорциумы, франчайзинговые соглашения и другое.

Сетевая кооперация позволяет фирмам разделять затраты и комбинировать ресурсы, также разделять риск неудачного технологического решения. Большинство сетей формируется вокруг крупного мультинационального предприятия либо на базе стратегических альянсов между такими предприятиями.

2.5 Функциональная модель предприятия

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 по подразделениям. В функциональной модели должно быть отражено как внутреннее взаимодействие бизнес-процессов, так и взаимодействие предприятия с внешней средой (рис. 21). При построении бизнес-процессов важна скорость построения процессов, сокращение рисков совершения ошибок, логический анализ процессов и многие другие факторы.



Рис. 21 – Верхний уровень функциональной модели

Рассмотрим терминологию функционального моделирования бизнес-процессов.

Процесс (бизнес-процесс) – последовательность выполнения действий (функций), которые выполняются субъектами (ответственными за исполнение функций лицами) для достижения определенной задачи с определенным

результатом. Каждый процесс входит в соответствующую категорию процессов и «имеет ответственную должность», т. е. выполнение процесса происходит под руководством и наблюдением ответственного лица. Процессы состоят из функций и взаимосвязаны друг с другом.

Входящие процессы – процессы, ссылающиеся на данный процесс.

Исходящие процессы – процессы, на которые ссылается данный процесс.

Функции – составляющие процесса (шаг).

Функции бывают четырех видов:

– *исполнительная функция*, требующая выполнения какого-либо действия персоналом любого уровня;

– *функция анализа*, которая на основании анализа какого-либо действия может изменить дальнейшее выполнение процесса. Функция анализа имеет как минимум два выхода (решения). Эту функцию не может исполнять производственный персонал;

– *контрольная функция*, которая контролирует или проверяет какое-либо действие, не может выполняться производственным персоналом;

– *руководящая функция*, разрешающая или утверждающая выполнение определенных действий, таких как издание приказов, распоряжений и планов, подписание или утверждение документации, выбор стратегий, увольнение или прием персонала.

Руководящая функция может завершать или открывать какой-то процесс или действие. Эту функцию может выполнять только руководящий персонал. Правила и нормы выполнения функции определяются соответствующим документом (инструкцией, методикой), частота выполнения функции фиксируется записью в таких документах, как акт, журнал, отчет и т. д.

Должности, подразделяющиеся на три типа:

– *руководящая должность*, имеющая одного или нескольких подчиненных (например, директор или начальник отдела);

– *дополнительная должность*, не участвующая в процессах производства или оказания услуг (например, главный бухгалтер);

– *производственная должность*, принимающая участие в производстве продукции или оказании услуг и не имеющая подчинённых (например, монтажник или слесарь).

Категория процесса – группа процессов, объединённых общей задачей (управление персоналом, маркетинг, управление производством). В отличие от процесса, за категорию процессов «отвечает» не должность, а, как правило, целое подразделение. Категориями процессов могут являться, например, следующие: маркетинг, логистика, производство, услуги; менеджменты проектов, персонала, качества, безопасности; административный, финансовый, экологический и менеджмент; управление рисками.

Подкатегория процесса – дополнительный признак, различающий процессы, принадлежащие одной категории (например, в категории процесса «Финансовый менеджмент» выделяются такие подкатегории, как «Бюджетирование», «Бухгалтерский учёт», «Налоговый учёт» и т. д. Количество подкатегорий процессов в каждой категории не ограничивается.

Документ – внутренний или внешний документ, используемый в деятельности компании (приказ или нормативный акт).

Запись – документ, который ведётся в деятельности предприятия, например журнал о регистрации корреспонденции.

Дополнительная обязанность – внештатная единица (должность), которую из-за эпизодичности выполнения возложенных на нее функций нельзя ввести в штатное расписание, например аудитор системы менеджмента качества или глава аттестационной комиссии. Периодически эти обязанности могут быть переложены с одного сотрудника на другого.

Построение функциональной модели начинается с обязательного описания в виде классификаторов следующих управленческих регистров:

- бизнесов;
- бизнес-процессов;
- функций менеджмента;
- организационной структуры компании.

После этого следует закрепить с помощью матричных проекций элементы первых трех классификаторов за четвертым, т. е. определить ответственность персонала за бизнесы, бизнес-процессы и функции менеджмента. Организационно-функциональная модель закрепляется Положением об организационной структуре, которое содержит описание указанных классификаторов и матричных проекций, дополненное граф-схемой организационной структуры предприятия.

Исходя из миссии формируются базовый рынок и базовый продукт предприятия, которые после соответствующей детализации (дифференцирования продукта и сегментации рынка) позволяют определить бизнесы в виде товарных групп, ориентированных на соответствующие рыночные сегменты.

Проекция бизнесов на этапы производственного цикла, включающего закупки сырья и комплектующих, производство продуктов и их распределение, обеспечивает формирование основных бизнес-процессов. Проекция компонентов производственного обеспечения на тот же производственный цикл позволит сформировать перечень обеспечивающих бизнес-процессов, носящих вспомогательный характер. Для этого сначала на предприятии разрабатываются и утверждаются два базовых классификатора: компоненты менеджмента и этапы управленческого цикла. После этого элементы указанных классификаторов проектируются друг на друга, порождая на пересечении строк и столбцов матрицы перечень основных функций менеджмента.

Каждое предприятие может разработать для себя персональные классификаторы. Принципиальное значение имеет не столько наполнение классификаторов, сколько единство их восприятия всеми менеджерами предприятия, закрепленное внутрифирменными регламентами.

Закрепление функций за организационными звеньями производится согласно схеме, приведенной на рис. 22. Зона ответственности помечается крестиком на пересечении соответствующей строки (звена) и столбца (функции).

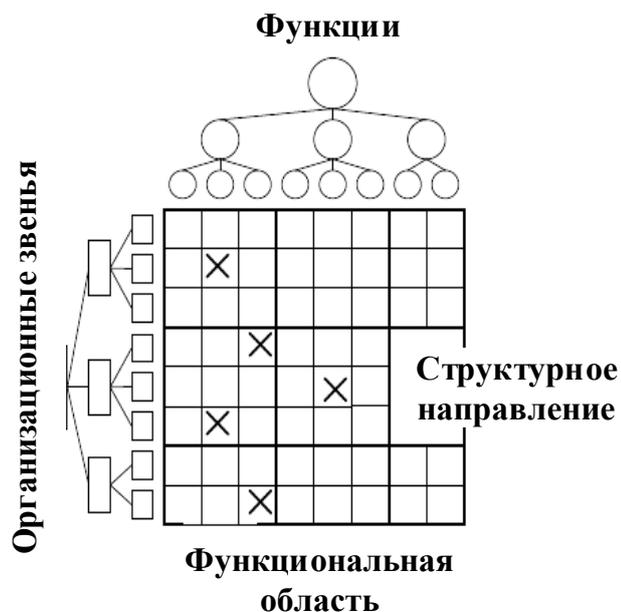


Рис. 22 – Матрица распределения функций по организационным звеньям

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.

Построение организационно-функциональной модели предприятия обеспечивает прозрачность бизнеса, что повышает его инвестиционную привлекательность, открывает путь к партнерству и сотрудничеству.

Преимущества процессного подхода к управлению предприятием:

- четкое распределение ответственности за функции;
- прозрачность деятельности;
- контроль не только результатов, но и процесса их получения;
- возможность проведения осознанного анализа деятельности персонала;
- снижение зависимости от человеческого фактора;
- накопление базы знаний по выходу из кризисных ситуаций;
- широкие возможности совершенствования деятельности.

Однако не рекомендуется включать в функциональную модель программы единичные бизнес-процессы, которые не типичны для деятельности компании и выполняются всего один раз.

2.6 Основные модели и инструменты описания бизнес-архитектуры

Обеспечение соответствия между ключевыми бизнес-процессами и архитектурой информационных технологий является самой важной составляющей всех усилий по созданию архитектуры.

Именно здесь возникает ряд коммуникационных проблем между представителями бизнеса и ИТ. Поэтому архитекторы нуждаются в простых, высокоуровневых средствах описания активностей и зависимостей в терминах, которые понятны бизнес-руководителям и пользователям и которые показывают соответствия с выполняемыми ими ролями. *Графические бизнес-модели* являются исключительно полезными в решении этой коммуникационной проблемы. Такие модели являются идеальным способом объяснить на достаточно высоком уровне критические активности и взаимосвязи в терминах бизнеса, и при этом они не требуют знаний в области ИТ. Модели обеспечивают важную связь между бизнес-целями и стратегиями деятельности организации и многими вариантами реализации ИТ (такими как готовые пакеты программ, унаследованные приложения, специально созданные заказные приложения, обслуживание по принципу аутсорсинга и подписки).

Модели бывают различных типов:

- модели процессов/потоков работ;
- функциональные модели;
- организационные модели;
- модели данных/ресурсов;
- временные модели типа диаграмм Ганта;
- модели причинно-следственных связей.

Факт заключается в том, что нет «одной, самой лучшей» модели для описания бизнес-процессов. Каждая из диаграмм описывает определенный ракурс или аспект контекста бизнес-архитектуры предприятия.

Для автоматизации моделирования процессов имеется специальный класс программных продуктов. Наиболее известными являются такие продукты, как

ARIS, Software Architect, BPWin (новое название – AllFusion Process Modeler), хотя зачастую стандартных графических пакетов типа Microsoft Visio, текстового редактора и электронной таблицы бывает достаточно.

Основное внимание при разработке бизнес-архитектуры должно уделяться картине в целом, поэтому рекомендуется начать с построения высокоуровневых моделей бизнес-процессов предприятия. Высокоуровневые модели, включенные в бизнес-архитектуру, должны давать необходимый минимум сведений о ключевых функциях, процессах, бизнес-событиях и потоках информации, достаточный для процесса принятия решений, поиска новых возможностей для инноваций. Из 10–20 основных процессов в первую очередь необходимо сосредоточиться на тех процессах, которые будут подвергнуты изменениям.

Основные шаги построения высокоуровневых моделей.

1. Идентификация критически важных для предприятия процессов (обычно не более восьми). Чаще всего это те ключевые процессы, которые:

– максимально влияют на способности организации реализовывать свою миссию, достигать цели, выполнять основные функции;

– открывают новые возможности;

– в настоящее время выполняются плохо и являются источниками неудовлетворенности клиентов, имеют возможности для оптимизации затрат.

2. Прослеживание связи между этими процессами и бизнес-стратегией, движущими силами и критически важными факторами успеха.

3. Построение модели высокого уровня для ключевых бизнес-процессов.

4. Определение для каждого шага процесса ответственных за выполнение шага. Это могут быть как внешние организации, так и подразделения компании.

5. Идентифицирование и документирование основных категорий информационных объектов.

Построение таких моделей и понимание их связей с ключевыми факторами позволяет понять в целом деятельность организации. Последующее

углубление в тщательно отобранные ключевые процессы и информационные потоки возможно с использованием основных инструментов, таких как:

- декомпозиция функций/процессов;
- анализ бизнес-событий.

Декомпозиция бизнес-процессов позволяет преодолеть сложность описываемой предметной области, представить объекты модели верхнего уровня в виде другой модели, раскрывающей то или иное внутреннее содержание данного объекта. Основные вопросы, на которые необходимо ответить при осуществлении декомпозиции бизнес-процессов:

- 1) каковы основные функции организации?
- 2) какие функции не несут в себе ценности?
- 3) какие функции пересекаются с другими бизнес-функциями?

Посредством декомпозиции и анализа бизнес-процессов должны быть получены следующие *результаты*:

- основные подпроцессы выбранных ключевых процессов (критически важных для бизнеса);
- границы основных организационных единиц;
- вклад каждой функции в цепочку создания добавочной стоимости;
- пересечения и излишние функции/процессы;
- возможные требования к прикладным системам и информации.

Анализ бизнес-событий позволяет понять, как инициируются бизнес-события и какие связанные с ними процессы происходят в цепочке создания добавочной стоимости предприятия. При этом берется конкретное событие, документируется текущий процесс его обработки и оцениваются возможности по его совершенствованию. Основные **вопросы**, на которые необходимо ответить при таком анализе:

- 1) кто является инициатором бизнес-события?
- 2) кто является основными участниками события?
- 3) как событие обрабатывается в рамках «расширенного» предприятия (партнеры и прочее)?

4) возможны ли инновации, связанные с событием и требуются бизнесом?

Посредством анализа бизнес-событий должны быть выявлены:

- основные инициаторы и участники бизнес-событий;
- партнеры;
- критически важные результаты, создающиеся и используемые позже в процессе обработки события;
- возможные новые формы ведения бизнеса.

Для построения в дальнейшем технологической архитектуры необходимо понимание того, где выполняются функции и процессы, а для построения архитектуры информации и архитектуры прикладных систем необходимо понимание ключевых внутренних и внешних точек интеграции, основных информационных потоков между участниками бизнес-процессов. Поэтому важно рассмотреть еще два аспекта:

- моделирование местоположений выполнения функций/процессов;
- модель интеграции функций/процессов.

Модель местоположений обеспечивает логистический взгляд на функции, выполняемые организацией, и идентифицирует в географическом плане то место, где они выполняются. Основные **вопросы**, на которые необходимо ответить при моделировании местоположений:

- 1) где выполняются основные функции?
- 2) какие функции связаны между собой?
- 3) существуют ли возможности по консолидации и рационализации?

В *результате* разработчик должен получить:

- распределение функций по местоположениям;
- связи между бизнес-функциями;
- требования к прикладным системам и технологической архитектуре;
- возможные требования к организационным изменениям.

Модель интеграции отражает высокоуровневые требования к интерфейсам между процессами и бизнес-событиями, требования к информации для новых бизнес-процессов, временные требования.

Результатами ее построения являются:

- ключевые внутренние и внешние точки интеграции;
- критичные информационные потоки между различными точками соединения моделей бизнес-событий;
- требования к технологической архитектуре, архитектуре приложений и информации;
- возможные требования к организационным изменениям.

Основопологающим требованием при построении модели бизнес-архитектуры является обеспечение ее адекватности, то есть достижение разумного баланса между детальностью и потребительскими качествами модели.

Контрольные вопросы:

1. Дайте определение понятию бизнес-архитектуры. В чем заключается ее контекст?
2. Перечислите составные элементы и подсистемы, входящие в состав бизнес-архитектуры.
3. Раскройте сущность понятия «предприятие». Приведите классификацию предприятий.
4. Проанализируйте основные виды добровольных объединений предприятий.
5. Дайте характеристику ключевым элементам организационной структуры предприятия.
6. Раскройте роль миссии и стратегии на предприятии.
7. Перечислите основные этапы стратегического планирования. Раскройте понятие базовой стратегии.
8. Проанализируйте типы организационных структур.
9. Раскройте сущность функциональной модели. Понятие и характеристики бизнес-процесса.
10. Проанализируйте значение и роль инструментов описания бизнес-архитектуры.

ТЕМА 3 СИСТЕМНАЯ АРХИТЕКТУРА

В системную архитектуру включены следующие элементы [16, 17]:

1) архитектура информации:

- базы данных и хранилища данных (БД);
- системы управления БД или хранилищами данных;
- правила и средства санкционирования доступа к данным;

2) архитектура приложений:

– собственно прикладные системы, поддерживающие исполнение бизнес-процессов;

– интерфейсы взаимодействия прикладных систем между собой, с внешними системами, источниками или потребителями данных;

– средства и методы разработки и сопровождения приложений;

3) технологическая архитектура:

– сетевая архитектура: 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: аппаратные средства вычислительной техники (серверы, рабочие станции, накопители и др.), 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.

3.1 Архитектура информации

Понятие **архитектуры информации** содержит описание создания возможностей быстрого принятия решений и распространения информации

внутри организации и за ее пределами посредством использования информационных технологий.

Можно сказать, что архитектура информации является «зеркальным отражением» бизнес-архитектуры. Бизнес-архитектура отвечает на вопрос «кто и что будет делать с учетом общего видения, целей и стратегий?», а архитектура информации – на вопрос «какая информация должна быть предоставлена для осуществления этих процессов исполнителями?». Архитектура информации включает в себя модели, описывающие процессы обработки информации (information value chain), основные информационные объекты, связанные с бизнес-событиями, информационные потоки, принципы управления информацией. Архитектура должна описывать операционные данные, требуемые для выполнения процессов, аналитические данные и «контент», публикуемый в веб-приложениях.

Разработка архитектуры информации как части архитектуры предприятия состоит не в создании структур БД или моделей всех данных, используемых предприятием. Суть процесса разработки архитектуры информации заключается в организации общего описания информации, требующейся для бизнеса, а также политики и правил работы с информацией. В связи с этим следует отметить, что в контексте архитектуры предприятия более правильно говорить об архитектуре и моделях информации, а не данных, хотя эти понятия и пересекаются. Модели архитектуры информации являются более абстрактными, они используют язык бизнеса и обеспечивают контекст, требуемый для моделирования данных. Модели данных предполагают четкие описания структуры объектов, атрибутов, отношений между сущностями.

Таким образом, понятие «архитектура информации» является расширением понятия «архитектура данных». В общем, архитектура информации рассматривается как процесс организации и представления значимой для бизнес-пользователей информации 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 данной области будет 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 [10]:

- 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.

Основные модели описания архитектуры информации.

Процессы разработки моделей информации и моделей данных предназначены для создания графических представлений информации, удовлетворяющей потребностям организации как в целом, так и в отдельных бизнес-процессах. 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 организационными 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-процесса соответствующей автоматизированной системы, приводится необходимый набор информационных объектов и их элементов данных.

Физическая модель данных предназначена для представления логической модели в функции хранения в системе управления базами данных.

3.2 Модели и инструменты построения архитектуры приложений

Архитектура приложений обеспечивает идентификацию прикладных систем, необходимых предприятию для выполнения бизнес-процессов, состоящих из этапов проектирования, разработки (или приобретения) и интеграции прикладных систем.

Архитектура приложений предприятия состоит из двух частей:

- портфеля прикладных систем предприятия;
- области разработки прикладных систем.

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.

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 [8]:

– **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. 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;

– **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 (разделение на функциональные составляющие, создание интерфейсов, настройку, 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), цикл разработки, контроль версий, настройка 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. Данная область выделяется, как правило, в организациях, в которых производится самостоятельная разработка приложений.

Модели и инструменты управления портфелем приложений. Сбор информации по имеющимся в организации прикладным системам является, на самом деле, нетривиальным занятием. Во-первых, к удивлению многих руководителей, выясняется, что количество этих систем достигает сотен. А во-вторых, чтобы такой каталог прикладных систем действительно был полезен, он должен включать в себя определенный набор информации:

- название системы;
- описание системы;
- список технологических компонентов. Это важно, поскольку они могут использоваться независимо для построения других решений. При этом основные компоненты должны быть отдельно описаны, включая их функции и техническое состояние;
- область применения с точки зрения бизнеса, т.е. функциональные возможности (например, CRM, финансы, управление кадрами, каналы продаж через Интернет и пр.);
- «владелец» системы со стороны бизнеса;
- оценка пользы прикладной системы для бизнеса;
- ответственный со стороны ИТ-подразделения;
- оценка технического состояния;
- оценка возможностей по обеспечению потребностей бизнеса;

– дата обновления этой информации.

Имеющийся каталог информационных систем предприятия является источником для анализа и управления портфелем приложений.

Существуют различные способы оценки портфеля и различные классификации прикладных систем предприятия. Одним из возможных способов оценки портфеля прикладных систем является оценка по двум критериям – *ценности с точки зрения бизнеса и техническому состоянию*.

Матрица оценки состояния прикладных информационных систем (Health Grid) [20], представлена на рис. 23.



Рис. 23 – Оценка портфеля прикладных систем по критериям «Бизнес-ценность» и «Техническое состояние»

Оценка портфеля служит отправной точкой в идентификации проблемных областей и возможностей для лучшего удовлетворения потребностей бизнеса и принятия решения об инвестициях в новые системы или обновление существующих.

В результате такой оценки прикладные системы относят к одной из четырех возможных категорий [10]:

1) *системы, находящиеся под угрозой вывода из эксплуатации (замены) или консолидации вследствие низкой ценности для бизнеса и плохого технического состояния.* Стоимость замены некоторых унаследованных систем может оказаться неоправданно высокой и будет иметь весьма ограниченную ценность с точки зрения бизнеса;

2) *системы, требующие переоценки или перепозиционирования* по причине низкой ценности для бизнеса, но в отличном техническом состоянии. Как правило, это прикладные системы, которые были недавно запущены в эксплуатацию в соответствии с рекомендациями, принятыми в рамках архитектуры предприятия. Однако объем и характер решаемых ими задач или ограниченность области применения в рамках каких-то узких организационных функций таковы, что их вклад в достижение ключевых бизнес-результатов незначителен. В этой ситуации рекомендуется провести идентификацию и анализ возможностей использования данных приложений или их компонентов в рамках остальных бизнес-процессов и организационных структур предприятия;

3) *требующие обновления системы,* представляющие высокую ценность для бизнеса, но в плохом техническом состоянии.

Эти прикладные системы исправно обслуживают ключевые бизнес-функции, но создают существенные проблемы, когда речь идет об эксплуатации и сопровождении этих систем, либо возникает необходимость использования информации из них, либо при необходимости интеграции данных систем с другими прикладными системами предприятия. Возможным выходом здесь является постепенный переход на использование более адаптивной архитектуры приложения (компонентного подхода, n-уровневой архитектуры, основанных на пересылке сообщений интерфейсов и т. д.);

4) *системы, требующие сопровождения и развития,* имеющие высокую ценность для бизнеса и в отличном техническом состоянии. Эти системы критически важны с точки зрения бизнеса и спроектированы в соответствии с современными представлениями об архитектуре прикладных систем.

Графическое расположение систем на матрице Health Grid, кроме двух параметров – ценность с точки зрения бизнеса и техническое состояние, – может нести в себе еще и дополнительную информацию.

Во-первых, каждой прикладной системе соответствует круг, диаметр которого пропорционален общей стоимости информационной системы, с учетом стоимости приобретения, эксплуатации и сопровождения (стоимости обновлений).

Во-вторых, круги окрашиваются в один из пяти цветов, характеризующих реальную важность системы, а не просто потенциальную ценность для бизнеса.

Такая оценка является только первым шагом в обеспечении соответствия между существующим и будущим портфелями прикладных систем и бизнес-стратегиями предприятия. В дополнение к этому необходимо выполнить следующее:

- оценить те потребности со стороны бизнеса, которые вообще никак не обслуживаются существующим портфелем прикладных систем. Бизнес-архитектура в этом плане должна обеспечивать список таких требований, в то время как архитектура информации должна уточнять эти требования с точки зрения информационного обеспечения;

- провести сравнение технологических и операционных требований (надежность, масштабируемость и т.д.) портфеля прикладных систем с имеющейся технологической архитектурой с целью идентификации тех возможностей инфраструктуры, которые в настоящее время отсутствуют, но могут потребоваться;

- согласовать проекты в области внедрения прикладных систем и развития инфраструктуры с учетом анализа на предыдущих двух шагах. Результатом должен быть синхронизированный план миграции как технологической архитектуры, так и архитектуры прикладных систем, с расстановкой приоритетов на основе определения ценностей проектов для бизнеса и технического состояния приложений и инфраструктуры.

Для оценки портфеля прикладных систем может быть также использована **модель, предложенная компанией Gartner**. Анализ портфеля инвестиций может быть существенно упрощен, если взять за основу 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 эффективность деятельности организации [8].

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 системами, должны проводиться четко и вовремя, но, например, сам факт своевременного 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.

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. *Преимущества* от их использования заключаются в следующем:

- 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.

Еще одна классификация связана с *ролью, выполняемой приложением в рамках портфеля информационных систем* организации:

- *критически важное для предприятия в целом (mission-critical)*. Приложение чрезвычайно важно для осуществления всей миссии компании, нарушения в работе приложения могут повлечь катастрофические последствия для бизнеса. Например, система биллинга оператора мобильной связи или система управления движением в аэропорту;
- *критически важное для бизнеса (business-critical)*. Приложение важно для поддержки отдельного направления бизнеса или обеспечивающего бизнес-процесса. Нарушения могут повлечь серьезные затруднения в бизнесе. Например, система приема заказов через Интернет;
- *вспомогательное (utility)*. Некритичное приложение, решающее частную, вспомогательную задачу. Например, система резервирования помещений для переговоров;

– *средства офисной автоматизации* (office productivity). Это приложения, используемые для автоматизации повседневной работы. Например, офисные пакеты и средства подготовки презентаций.

Заметим, что для разных компаний одни и те же «стандартные» приложения, такие, как электронная почта или система приема заказов, могут относиться к различным уровням в данной классификации.

3.3 Структура и принципы разработки технологической архитектуры

Основное назначение **технологической архитектуры** – обеспечение надежных ИТ-сервисов, предоставляемых в рамках всего предприятия в целом и координируемых централизованно, как правило, департаментами информационных технологий. Технологическая архитектура определяет набор принципов и стандартов, которые обеспечивают информационные руководства в отношении выбора и использования следующих технологий:

- аппаратных платформ;
- операционных систем;
- систем управления базами данных;
- средств разработки;
- языков программирования;
- сервисов электронной почты;
- систем безопасности;
- сетевой инфраструктуры и т. д.

Инфраструктурные сервисы в основном стандартизированы в рамках предприятия и используются сразу несколькими прикладными системами, расположенными над уровнем инфраструктурных сервисов и непосредственно обеспечивающими выполнение бизнес-процессов. При наличии необходимой инфраструктуры новые прикладные системы, которые потребуются предприятию для выполнения новых бизнес-процессов или реализации новых стратегий, могут быть созданы достаточно быстро и эффективно, что позволит

повысить динамичность и гибкость предприятия. Одной из частных задач, решаемых в рамках данной архитектуры, является формирование «списка закупаемых технологий» [18].

Существует два принципиально отличных подхода к формированию технологической архитектуры. *Первый подход* заключается в перечислении используемых на предприятии стандартов и теоретически позволяет уменьшить зависимость предприятия от конкретных поставщиков. Однако уменьшение этой зависимости имеет ограниченный успех, поскольку замена одного продукта другим, поддерживающим один и тот же набор стандартов, как правило, оказывается невозможной или затруднительной. Поэтому с середины 1990-х годов большинство предприятий стали использовать *второй подход*, который связан, в конечном итоге, с перечислением конкретных продуктов и технологий.

Упорядоченный в рамках технологической архитектуры список продуктов и технологий дает реальные преимущества [8]:

- технический персонал должен поддерживать уровень знаний, связанных с меньшим количеством продуктов, что уменьшает затраты на содержание персонала и его обучение;

- прикладные системы легче интегрировать между собой, когда они имеют много общих технических аспектов. Хотя заметим, что список технологий и поставщиков не является все-таки самым важным инструментом интеграции данных и систем. Вопросы семантики и согласования форматов, например, гораздо более сложны и не решаются выбором одной технологии;

- предприятие может получить экономию на масштабах, приобретая технологии ограниченного количества поставщиков (например, скидки на лицензии);

- много усилий может быть сэкономлено на процессах закупок, поскольку, после того как технология однажды выбрана, последующие закупки не требуют затрат времени на длительное изучение альтернатив.

Рисунок 24 иллюстрирует то, что технологическая инфраструктура предприятия располагается на нескольких «уровнях» и принятие решения о размещении той или иной части инфраструктуры в масштабе предприятия или отдельных бизнес-подразделений является стратегическим решением, которое должно основываться на принятых в организации принципах построения архитектуры.

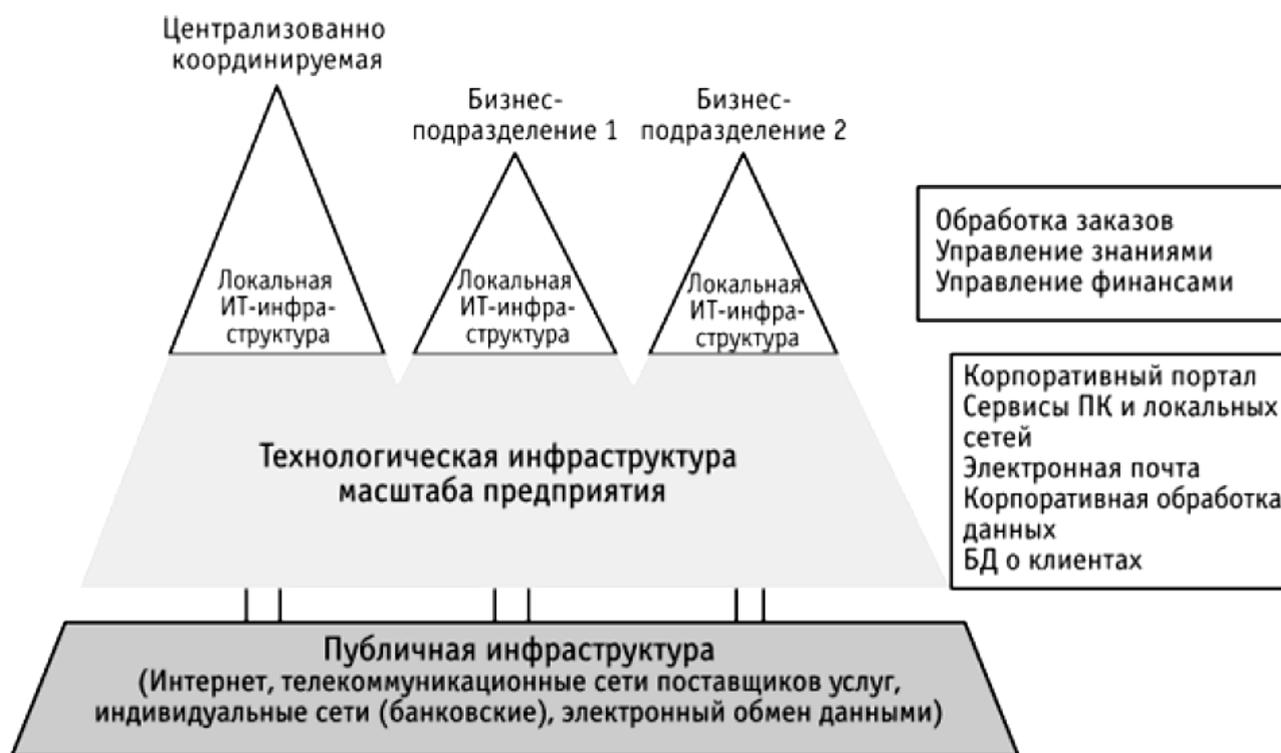


Рис. 24 – Различные уровни размещения инфраструктуры

На данном рисунке условно изображено предприятие с несколькими бизнес-подразделениями и разнообразным портфелем технологий: некоторые из этих технологий координируются и эксплуатируются централизованно, другие – на уровне отдельных подразделений. ИТ-сервисы, которые требуются для предприятия в целом, обеспечиваются комбинацией общекорпоративной и публичной инфраструктуры.

Инфраструктура уровня подразделения ориентирована на более специфические потребности соответствующих подразделений. *Например*, в

крупной компании обработка больших массивов производственных данных может производиться в едином корпоративном центре обработки данных. Все подразделения используют эту централизованную инфраструктуру, но имеют некоторые дополнительные локальные потребности, которые обеспечиваются локальной инфраструктурой. Одно из бизнес-подразделений, крупнейшее на предприятии, может и не иметь своей собственной локальной инфраструктуры, а использовать исключительно централизованные сервисы.

Gartner называет в технологической архитектуре **шесть архитектурных компонент (сервисов)**, в каждом из которых выделяется определенное количество технологических «строительных блоков»:

- *сервисы данных*: системы управления базами данных (технологии баз данных и методы доступа к базам), хранилища данных (хранилища и витрины данных), системы поддержки принятия решений;

- *прикладные сервисы*: языки программирования, средства разработки приложений, системы коллективной работы (средства групповой работы и электронной почты, средства управления документами), архитектура приложений (модель компонент, серверы приложений, серверы поддержки тонких клиентов), геоинформационные системы и средства;

- *программное обеспечение промежуточного слоя*;

- *вычислительная инфраструктура*: операционные системы и аппаратное обеспечение (приложения для настольных систем, операционные системы для настольных систем, мобильные устройства), среда для web-инфраструктуры (браузеры, web-порталы, web-серверы, средства управления и создания контента, серверы каталогов, форматы публикации информации), системы хранения, средства системного управления (средства сетевого управления, администрирование IP), топологии (топология распределенных приложений).

- *сетевые сервисы*: локальные сети (протоколы, кабельные системы, топология), глобальные сети, технологии доступа (пользователи с удаленным доступом, эмуляция терминалов и шлюзы, беспроводные технологии для

локальных и глобальных сетей, интегрированные средства передачи данных, обеспечение доступности, средства видеоконференций), голосовые технологии, сетевое аппаратное обеспечение (концентраторы, маршрутизаторы и пр.).

– *сервисы безопасности*: авторизация, внутренняя и внешняя аутентификация), сетевая безопасность, физическая безопасность центров обработки данных, прочие сервисы безопасности (обнаружение вторжений, защита от вирусов и т.д.).

Большое значение имеет *взаимосвязь между функциональными и операционными требованиями* к системам и различными областями архитектуры, такими как прикладные системы и технологическая архитектура. Функциональные требования к прикладной системе описывают ценность, которую представляет система с точки зрения реализации функций организации (бизнес-ценность). Архитектура приложений, по сути, является архитектурой всех автоматизированных сервисов, которые обеспечивают и реализуют функциональные требования, включая интерфейсы к бизнес-приложениям и другим прикладным системам. Архитектура приложений описывает структуру приложений и способы реализации данной структурой функциональных требований организации [10].

Хорошая технологическая архитектура может обеспечивать безопасность, доступность, надежность и целый список других операционных требований, но если приложение спроектировано без использования преимуществ технологической архитектуры, оно все равно будет функционировать плохо, и его будет сложно внедрять и сопровождать. Аналогично, хорошо спроектированная структура прикладной системы, которая точно соответствует требованиям бизнес-процессов и собрана из многократно используемых компонентов с применением новейших технологий, может не соответствовать реальной конфигурации используемого аппаратного и системного программного обеспечения.

Перспективным направлением развития является формирование **адаптивной технологической инфраструктуры**, которая способна в

определенных пределах, автоматически или полуавтоматически, «подстраиваться» под требования со стороны бизнес-приложений для обеспечения оптимальной работы.

Основные характеристики адаптивной системы:

– *самоконфигурирование* – организация системы в соответствии с требованиями;

– *самозащита* – предотвращение сбоев в системе в результате нарушения работы компонент системы и потери целостности данных;

– *самовосстановление* – диагностика неисправностей, локализация ошибок и устранение их последствий;

– *самооптимизация* – наиболее рациональное использование имеющихся ресурсов без вмешательства оператора.

Для решения этой задачи предложили свои решения практически все ведущие производители, включая HP (концепция Adaptive Enterprise, архитектура Darwin), IBM (On Demand), Sun (N1), Microsoft (Dynamic Systems Initiative) и другие. Важной частью этих решений является комплексность, использующая как возможности аппаратных платформ, включая разделяемые процессорные разделы, виртуальные дисковые массивы, серверы-"лезвия", так и специализированное программное обеспечение для "оркестровки" существующих ресурсов.

Основные идеи адаптивной инфраструктуры:

– все ИТ-ресурсы являются общими и разделяемыми;

– выделение ресурсов конкретным приложениям производится автоматически в соответствии с требованиями бизнеса;

– качество обслуживания является предсказуемым и стабильным, несмотря на непредсказуемый спрос на ресурсы.

На рис. 25 представлен пример реализации адаптивной инфраструктуры на практике.

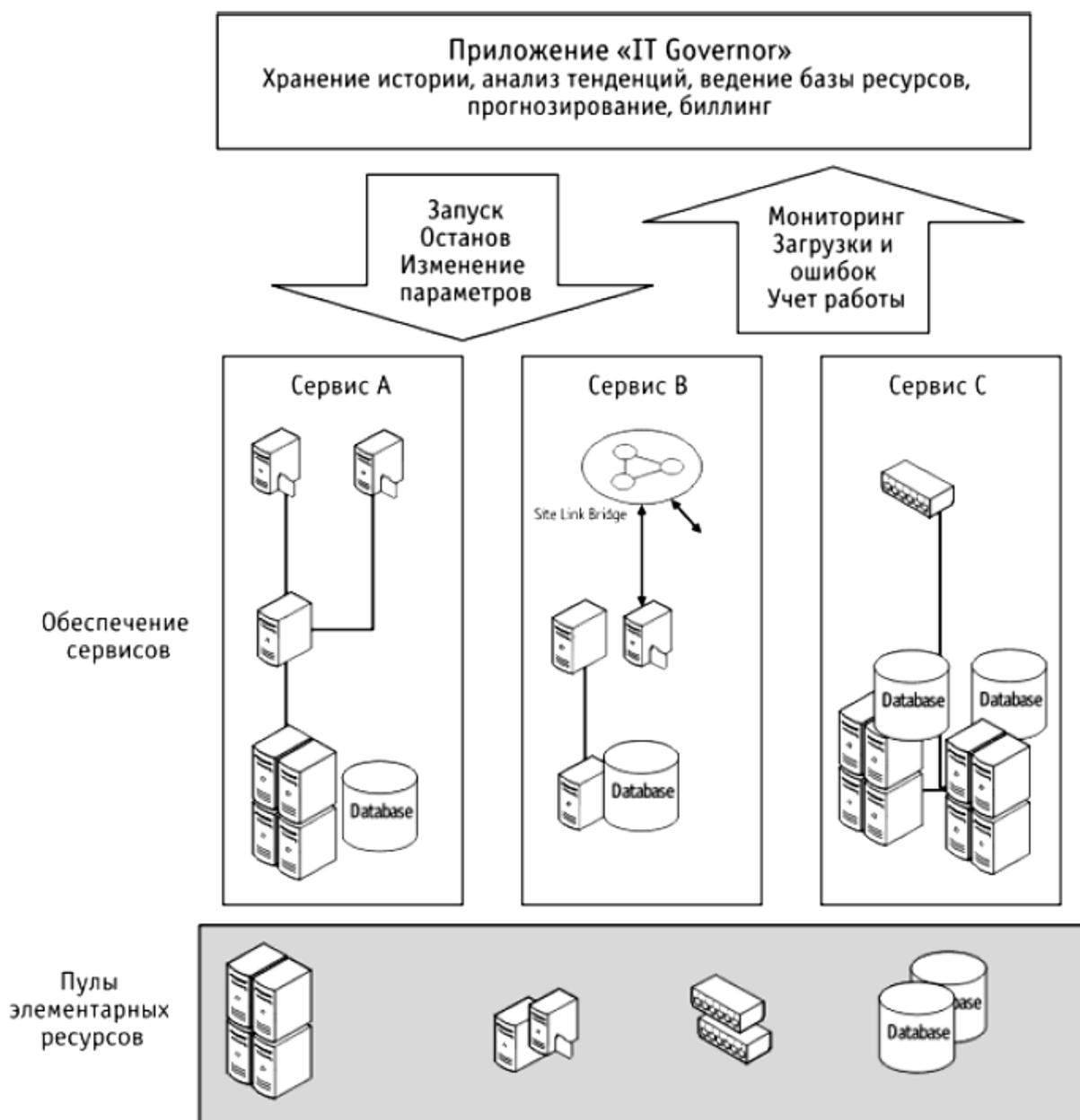


Рис. 25 – Инфраструктура реального времени

Проанализируем приведенную на рис. 25 схему:

- элементарными «строительными блоками» являются отдельные ресурсы – серверы и отдельные процессоры, виртуальные устройства хранения данных, сетевое оборудование, программные серверы приложений и СУБД;
- обеспечение требований бизнеса производится в рамках модели предоставления сервисов. Каждый сервис реализуется на основе нужной комбинации программных и аппаратных ресурсов, преимущественно в рамках

многозвенной архитектуры (уровень СУБД – уровень приложений – презентационный уровень);

– специальная «интеллектуальная» компонента системы – управляющий модуль (IT governor) на основе мониторинга времени реакции сервисов на запросы, прогнозных и исторических значений потребностей приложений, наличия ошибок/выхода из строя элементов системы и т.п. отвечает за оптимальную «оркестровку» – переконфигурирование виртуальных серверов, запуск серверов приложений, управление пулами однородных устройств, изменение приоритетов выполняемых приложений. Эта же компонента будет обеспечивать функции биллинга для учета реально использованных ресурсов со стороны приложений.

Использование архитектурных шаблонов.

Существенную роль при построении архитектуры предприятия играют стандартизованные элементы. Подобно тому, как проект здания может включать в себя элементы ранее созданных конструкций, так и реализация поддержки бизнес-процесса в информационной системе может использовать уже известные фрагменты программного кода и/или типовые конфигурации оборудования. Это позволяет, с одной стороны, значительно сократить сроки выполнения решения, с другой – уменьшить риски за счет использования фрагментов, проверенных на практике. Речь идет о выборе и использовании подходящих *шаблонов (patterns)*.

Шаблон – это общее решение некоторой повторяющейся проблемы в определенном контексте.

Характеристики шаблона:

– *общее решение* – шаблон не дает полного законченного решения. Он определяет класс проблемы и то, как эта проблема может быть решена с использованием определенного подхода, с демонстрацией аргументов в пользу этого подхода. Значимость шаблона состоит в том, что он сформулирован на достаточно высоком уровне абстракции, чтобы быть использованным в большом количестве ситуаций;

– *повторяющаяся проблема* – шаблоны используются в тех случаях, когда проблема не является уникальной, и они наиболее полезны для решения часто встречающихся проблем;

– *определенный контекст* – шаблон обеспечивает решение проблемы, границы которой в общих чертах определены. Понимая условия, в которых предлагаемое решение в форме шаблона является достаточно эффективным, далее строится собственное решение на основе этого шаблона.

Важность шаблонов для архитектуры предприятия в целом обусловлена следующими причинами:

– если используются корректные шаблоны, то вероятность получения адекватно работающей физической реализации архитектуры возрастает;

– разработка и использование шаблонов в рамках предприятия в целом обеспечивает преимущества, связанные с их многократным использованием для решения различных проблем. Это дает архитекторам возможности по использованию опыта и стандартизации решений при создании новых систем;

– использование шаблонов отделяет логический уровень от физического уровня архитектуры. Это позволяет создать долговременно работающие решения и придает гибкость, поскольку на последующем этапе эти достаточно постоянные конструкции могут быть связаны с конкретными технологическими решениями.

Далее концепция шаблонов была расширена и в область инфраструктуры, так что теперь можно вести речь о соответствующих комплексных программно-аппаратных решениях.

Инфраструктурные шаблоны – стандартизированный набор требований, компонент и сервисов, которые в совокупности формируют необходимую адекватную инфраструктуру для данной прикладной системы и реализации логики бизнес-процессов.

Организация инфраструктуры с помощью набора шаблонов позволяет единообразно определять компоненты и функциональные возможности, в результате чего эта часть ИТ-инфраструктуры может многократно

использоваться для различных типов прикладных систем, имеющих общие требования к инфраструктуре.

Большой интерес при создании бизнес-архитектуры предприятия представляют **бизнес-шаблоны**. Описание бизнес-шаблона включает:

- описание поддерживаемой бизнес-функции;
- данные, которые требуются для выполнения описанной бизнес-функции;
- бизнес-компоненты, которые являются представлением данных и функций бизнеса на языке информационных технологий;
- описание инфраструктуры, которая необходима для поддержки функций, данных и компонент.

Компания IBM предлагает стандартизированные решения, которые можно разделить в зависимости от уровня абстракции следующим образом:

- *бизнес-шаблоны* (Business pattern) предназначены для описания взаимодействия между участниками процесса;
- *шаблоны дизайна* (Design pattern) отражают внутреннюю компонентную структуру системы;
- *шаблоны уровня приложений* (Application pattern) определяют различные варианты взаимодействия между пользователями, приложениями и данными в системе, а также соответствующий прототип уровня выполнения;
- *шаблоны уровня выполнения* (Runtime pattern) описывают привязку компонент системы к физическим узлам и определяют конкретные возможные продукты и их комбинации.

Кроме этого, выделяют также два служебных шаблона: соответственно *интеграции доступа и интеграции приложений*.

Эти шаблоны предназначены для описания следующих областей:

- интерактивная – взаимодействие пользователя с предприятием (U2B);
- программное взаимодействие между приложениями различных предприятий (B2B);

- коллективная работа пользователей, включая электронную почту, обмен мгновенными сообщениями, общие форумы и т.п. (U2U);
- поиск информации в каталогах и базах данных, анализ данных, подписки (U2D);
- взаимодействие между приложениями «в рамках предприятия», в том числе и не обязательно с использованием web-интерфейсов;
- централизованный доступ к системе на уровне выбранного интерфейса (портал) или на более общем уровне (Web, речевая телефония, мобильные устройства и т.п.);
- обеспечение безопасности.

Шаблоны могут быть использованы по отдельности или в комбинации при реализации более сложных комплексных решений.

Сервис-ориентированная архитектура (SOA) и архитектура, управляемая моделями (MDA).

С помощью SOA организации имеют потенциальную возможность разрабатывать набор реализаций различных бизнес-процессов, которые могут быть многократно использованы предприятием как готовые сервисы.

Под **сервис-ориентированной архитектурой** понимается подход к проектированию прикладных информационных систем, который руководствуется следующими принципами:

- явное отделение бизнес-логики прикладной системы от логики презентации информации;
- реализация бизнес-логики прикладной системы в виде некоторого количества программных модулей (сервисов), которые доступны извне (пользователям и другим модулям), чаще всего в режиме «запрос-ответ», через четко определенные формальные интерфейсы доступа;
- при этом «потребитель услуги», который может быть прикладной системой или другим сервисом, имеет возможность вызвать сервис через интерфейсы, используя соответствующие коммуникационные механизмы.

В целом, SOA представляет собой модель взаимодействия компонент, которая связывает различные функциональные модули приложений (сервисы) между собой с помощью четко определяемых интерфейсов. Интерфейсы сами по себе не зависят от используемых аппаратных платформ, операционных систем или языков программирования, используемых для разработки этих приложений. Это позволяет отдельным сервисам взаимодействовать между собой одним и тем же стандартным, но в то же время универсальным способом. Такая особенность использования интерфейса, независимого от окружения и платформы, получила название модели «слабой связи». Ее очевидным преимуществом является повышенная гибкость и адаптируемость, поскольку замена или модернизация одной из компонент системы не сказывается на остальных.

Ориентация на сервисную архитектуру позволяет построить комплексную ссылочную модель архитектуры предприятия, которая в единой манере описывает как бизнес, так и ИТ:

Эта модель состоит из следующих основных компонент:

- *презентационный уровень* описывает интерфейсные сервисы для взаимодействия пользователей с информационной системой, включая корпоративные и публичные порталы, доступ с мобильных устройств, а также различные преобразования информации при взаимодействии с внешними системами и устройствами;

- *на уровне бизнес-сервисов* формируются модели и осуществляется управление выполнением бизнес-процессов предприятия с использованием специализированных средств, а также координация автоматизированных и «ручных» операций;

- *интеграционные сервисы* обеспечивают взаимодействие между приложениями;

- *сервисы уровня данных* реализуют средства извлечения и повторного использования данных из СУБД и приложений;

- *уровень инфраструктуры*, приложений и СУБД является основой для всей структуры, и именно здесь концентрируются основные инвестиции в ИТ.

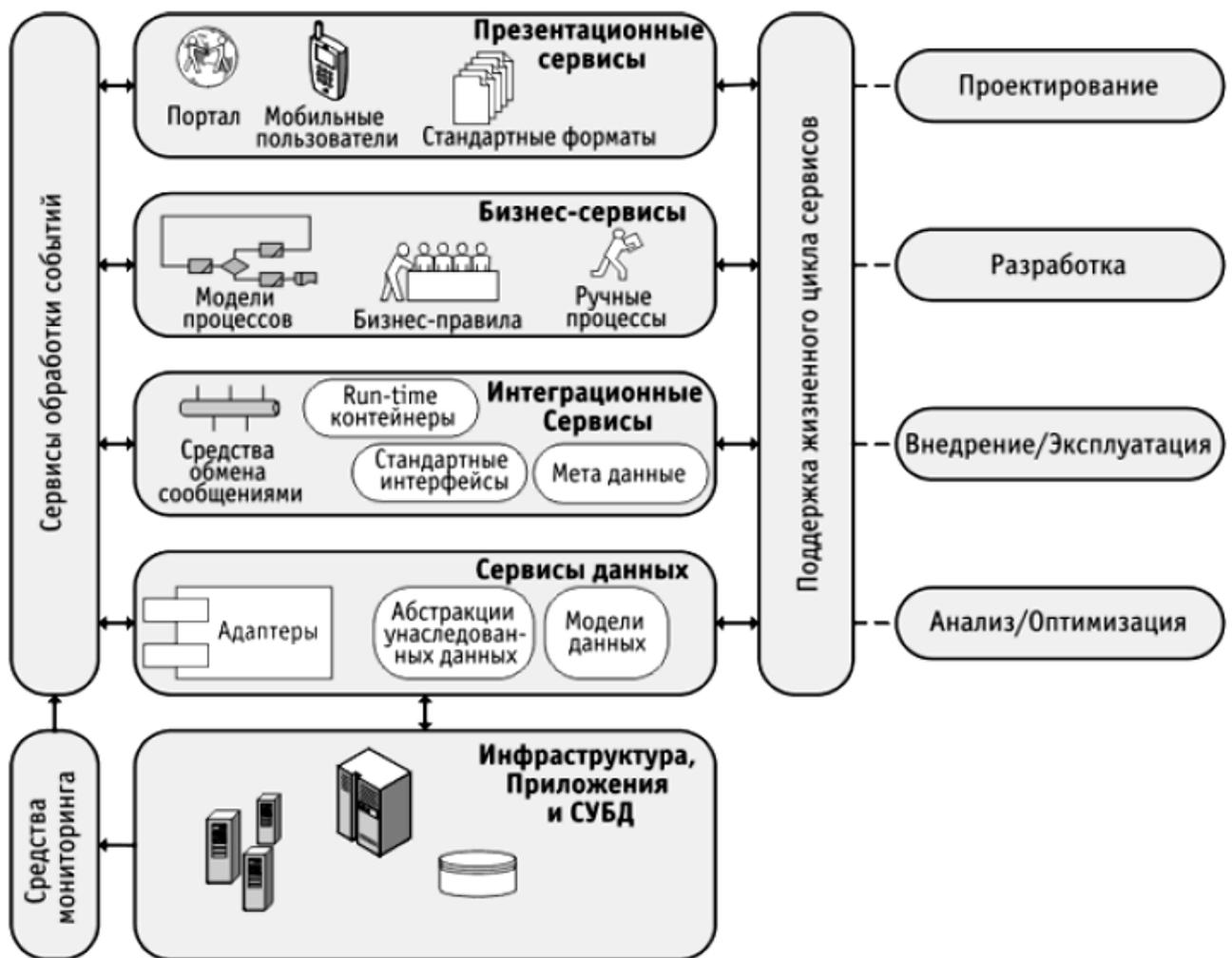


Рис. 26 – Ссылочная модель сервис-ориентированной архитектуры предприятия

На модели, представленной на рис. 26 взаимодействие между уровнями осуществляется через сервисы, выделенные на уровень обработки событий. Сервисы этой компоненты архитектуры обеспечивают сбор данных о событиях в масштабе всего предприятия, необходимое преобразование и маршрутизацию этих данных между разными уровнями, а также «обратную связь» между сервисами каждого отдельного уровня.

В предложенной модели, наряду с рассмотренными уровнями, отвечающими за взаимодействие различных групп сервисов в процессе деятельности предприятия, выделяется отдельная компонента архитектуры, которая описывает аспекты, связанные с жизненным циклом сервисов – то есть их созданием, эксплуатацией и оптимизацией.

Концепция MDA была предложена консорциумом OMG (Object Management Group) [21], в который сегодня входит более 800 известных производителей программного и аппаратного обеспечения. MDA является определенным обобщением идей SOA, с одной стороны, и повторно используемых программных компонент (шаблонов, паттернов) с другой, предназначенным прежде всего для повышения гибкости разрабатываемых приложений масштаба предприятия, чтобы обеспечить простоту обеспечения соответствия требованиям бизнеса в условиях изменения используемых инфраструктурных платформ.

MDA основана на следующих *принципах*:

- основой для разработки приложений масштаба предприятия являются детальные модели с общепринятой нотацией;
- построение систем может быть организовано в соответствии с рамочной системой моделей, которые позволяют отделить бизнес-логику приложений от конкретной реализации;
- существует формальное описание используемых моделей на основе системы метамodelей (в частности, для таких областей как распределенные вычисления и транзакции, операции в реальном времени и т.п.);
- принятие и широкое использование этого подхода основано на открытости промышленных стандартов и на поддержке со стороны производителей соответствующих средств разработки.

В рамках такого подхода сначала создается архитектура, которая описывает модель бизнес-функций и поведения прикладной системы независимо от технических деталей реализации. Эта разработка должна вестись в контексте всей корпоративной архитектуры организации. На основе этой модели, не зависящей от платформы реализации, может быть разработана одна или несколько специфических для конкретной платформы моделей, в зависимости от того, какая платформа используется и поддерживается организацией. Уже на основе этих специфических для конкретной платформы моделей разрабатывается код конкретной прикладной системы (рис. 27).

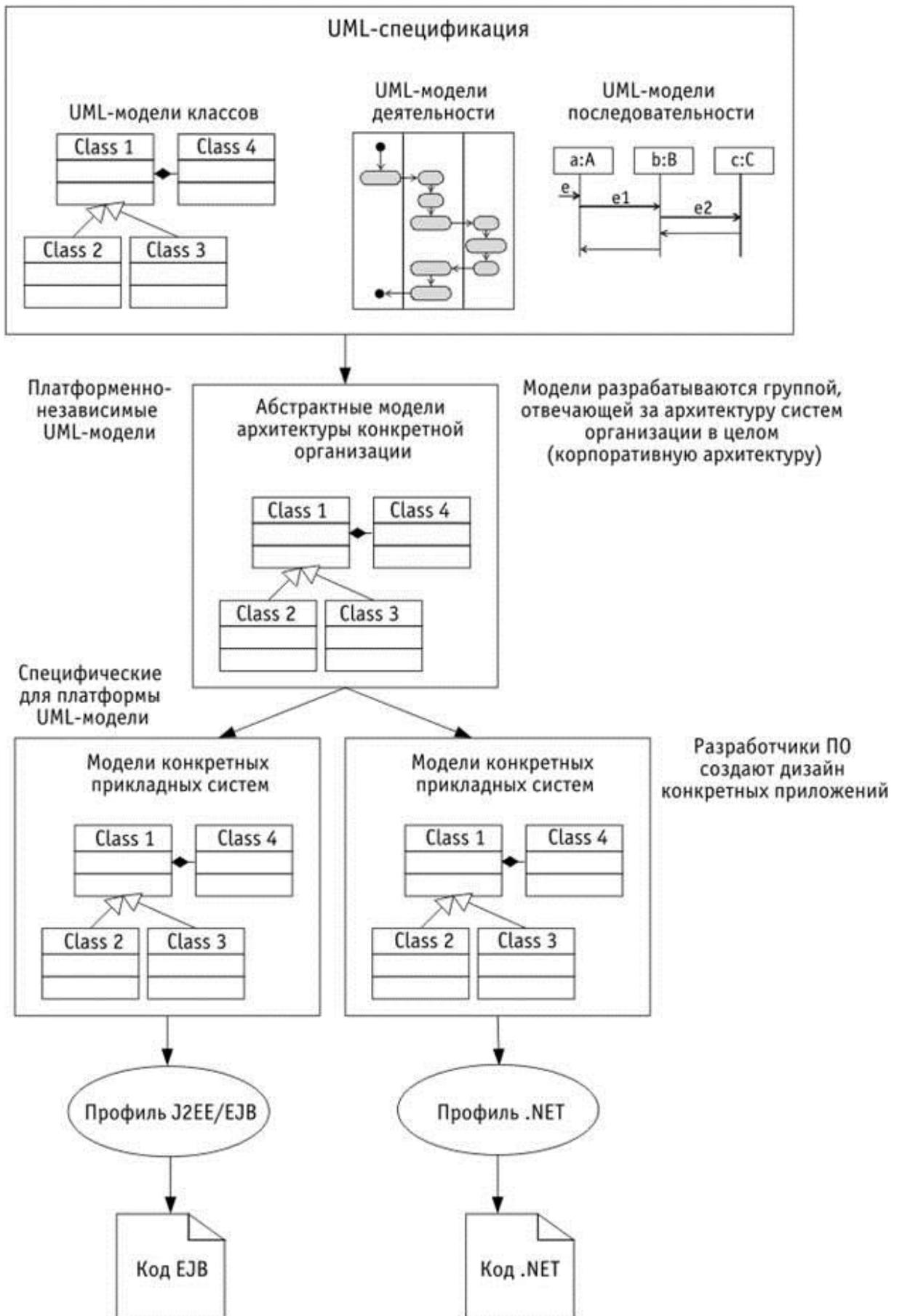


Рис. 27 – Создание прикладных систем в соответствии с подходом MDA

Этот подход не определяет, какие языки разработки, операционные системы или программное обеспечение промежуточного слоя будут использоваться на практике. Упор делается на описание того, как прикладные системы организованы с точки зрения процессов и как они интегрированы между собой. После того как эти высокоуровневые связи определены, могут использоваться соответствующие средства для разработки приложения с использованием конкретных языков и ПО промежуточного слоя. Таким образом, процесс позволяет сократить цикл разработки ИТ-систем и в то же время обеспечивает гибкость и возможность быстрого внесения изменений.

Контрольные вопросы:

1. Перечислите элементы системной архитектуры.
2. Раскройте сущность и задачи архитектуры информации.
3. Проанализируйте уровни представления архитектуры информации.
4. Дайте определение архитектуры приложений. Проанализируйте ее состав.
5. Какова роль каталога информационных систем предприятия и какую информацию он должен содержать?
6. Проанализируйте основные модели оценки портфеля прикладных систем.
7. Раскройте понятие технологической архитектуры, ее состав.
8. Дайте определение и развернутую характеристику адаптивной архитектуры.
9. Раскройте сущность архитектурного шаблона. Проанализируйте виды архитектурных шаблонов.
10. Дайте характеристику сервис-ориентированной архитектуры предприятия.

ТЕМА 4 МОДЕЛИ И МЕТОДИКИ ПОСТРОЕНИЯ АРХИТЕКТУРЫ ПРЕДПРИЯТИЯ

Эволюция от технологической до корпоративной архитектуры предприятия сопровождалась изменением смыслов использования архитектурного подхода от простого сокращения расходов на управление информационными технологиями и улучшения операционных процессов до идеи интеграции потребностей бизнеса и возможностей ИТ, поддержки динамичности и синхронизации ИТ и бизнес-стратегии. Соответственно появлялись новые методологии, которые адаптивно реагировали на изменения данной концепции (табл. 3).

Таблица 3 – Ключевые этапы разработки методологической базы архитектуры предприятия

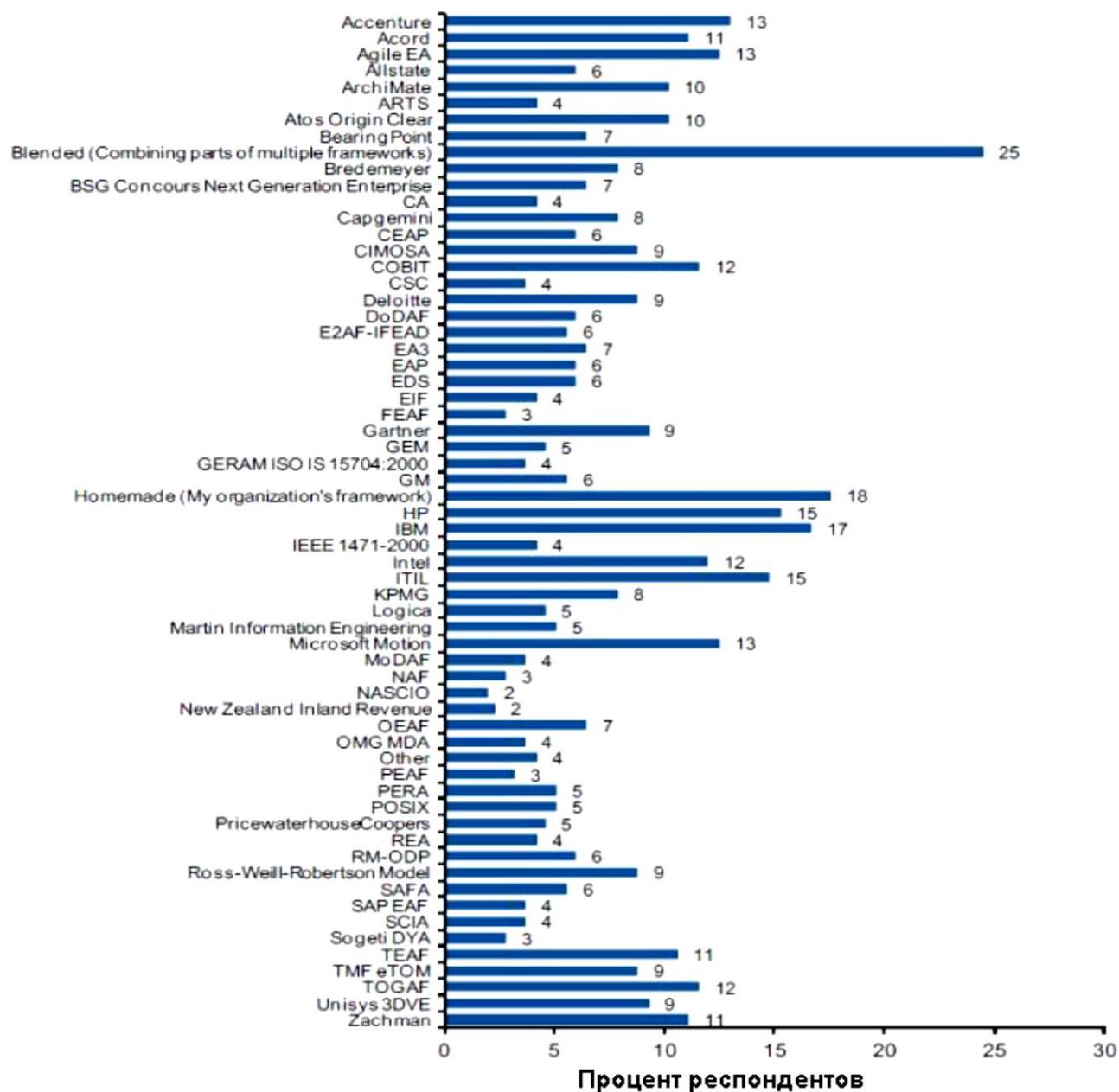
Событие	Год	Разработчик
1	2	3
Введение термина «архитектура информационных систем»	1987	Дж. Захман [22]
Разработка специализированной пяти уровневой эталонной модели предприятия	1989	Национальный институт стандартов и технологий (NIST) [23]
Разработка авторской методологии разработки архитектуры предприятия для эффективного управления предприятием	1992	Дж. Захман [22]
Представлена методология архитектурного описания EAP	1992	С. Спивак [24]
Разработан стандарт COBIT (Control Objectives for Information and Related Technology), регламентирующий методы управления информационными технологиями	1992	Ассоциация ISACA (Information Systems Audit and Control Association and IT Governance) [25]
Создание базовой архитектуры технического обеспечения для управления информацией TAFIM (Technical Architecture Framework for Information Management)	1994	Министерство обороны США (при участии Дж. Захмана) [26]
Разработана схема открытой системной архитектуры «Open System Architecture for CIM»	1994	Ассоциация CIMOSA [27]

Продолжение табл. 3

1	2	3
Разработка обобщенной стандартизированной методологии GERAM	1998	Рабочая группа IFIP—IFAC [28]
Создание методологии архитектуры федеральной организации (FEAF)	1998)	Федеральный совет директоров США по информационным технологиям [29, с. 63]
Разработка методологии архитектуры TOGAF (The Open Group Architecture Framework)	1998	Консорциум The Open Group [30]
Разработка концептуальной архитектурной схемы «3D-Предприятия»	2000	Е.З. Зиндер [31]
Модернизация и ренейминг методологии FEAF в методологию FEA	2002	Федеральный совет директоров США по информационным технологиям [29, с. 76]
Выпуск TOGAF 8.1 Enterprise Edition, DoDAF	2003	Консорциум The Open Group [30]
Разработка специализированной методологии Gartner	2005	Компания Gartner [32]
Выпуск TOGAF 9.1	2011	Консорциум The Open Group [30]
Разработка стандарта COBIT 4.1	2012	Ассоциация ISACA (Information Systems Audit and Control Association и IT Governance) [25]
Интеграция спектра инструментов для управления архитектурой предприятия	2012-настоящее время	Компании Software AG, IBM, Alfabet AG, Mega International, Trour Technologies и др. [1]

На сегодняшний день существует значительно большее множество методологий управления ИТ-инфраструктурой и предприятиями в целом на основе использования концепции архитектурного подхода. Выбор той или иной методологии связан с целями работы предприятий и огромным количеством критериев, которыми оперирует руководство при определении стратегических направлений его деятельности. Так, в исследовании компании Gartner [32] собран наиболее широкий перечень методологий управления архитектурой предприятия и проанализирована частота их использования (рис.28).

В рамках данного пособия невозможно охватить в полной мере все методики и модели построения архитектуры предприятия. Поэтому далее будут рассмотрены только наиболее востребованные и имеющие определяющее значение в становлении архитектурного подхода и развития концепции архитектуры предприятия.



N = 216

Рис. 28 – Доли компаний, использующих различные методологии архитектуры предприятия в качестве основных и сопутствующих

4.1 Модель Захмана

К наиболее значимым на сегодняшний день из ранних моделей, посвященных описанию и разработке архитектуры предприятия, относится схема Захмана [22]. Исторически схема Захмана, разработанная и развиваемая Джоном Захманом (John A. Zachman), была создана для проектирования и внедрения ИТ-систем, затем подход был обобщен для предприятия в целом. Первые публикации появились в 1980-х гг., а в 1987 г. – первый вариант обобщенной схемы. Долгое время являлась стандартом де-факто в области архитектуры предприятия.

Основная идея схемы Захмана заключается в том, чтобы обеспечить возможность последовательного описания каждого отдельного аспекта предприятия в координации с остальными. Метод преследует две основные цели: с одной стороны, логически разбить все описание архитектуры на отдельные разделы, с другой — обеспечить возможность рассмотрения целостной архитектуры на нескольких уровнях абстракции.

Захман предложил вместо традиционного подхода, связанного с рассмотрением отдельных аспектов работы системы в различные моменты времени, использовать рассмотрение системы с различных перспектив.

Для этого применяется матрица 6×6 , в которой каждая ячейка задает свой тип описания (моделей) свойств предприятия (рис. 29). Вся совокупность ячеек разделена на шесть столбцов матрицы — шесть аспектов деятельности предприятия:

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

	Данные ЧТО?	Функции КАК?	Сеть ГДЕ?	Организация КТО?	Расписание КОГДА?	Стратегии ПОЧЕМУ?	
Планировщик (1-й уровень)	Список важных понятий и объектов	Список основных бизнес- процессов	Список мест нахождения	Список организаций, важных для бизнеса	Список важных событий	Список бизнес- целей и стратегий	Сфера действия (контекст)
Владелец, менеджер (2-й уровень)	Концепту- альная модель данных	Модель бизнес- процессов	Схема логистики	Модель потока работ (workflow)	Календар- ный план реализации	Бизнес- план	Концептуальная модель предприятия
Конструктор, архитектор (3-й уровень)	Логические модели данных	Архитектура приложений	Модель распределенной архитектуры	Архитектура интерфейса пользователя	Структура процессов	Конкрети- зация ролей и бизнес- правил	Системная (логическая) модель
Проектиров- щик (4-й уровень)	Физическая модель данных	Системный проект	Технологи- ческая архитектура	Архитектура презентации	Структуры управления	Реализация ролей и бизнес- правил	Технологическая (физическая) модель
Разработчик (5-й уровень)	Описание структуры данных	Программ- ный код	Сетевая архитектура	Архитектура безопасности	Определение временных привязок	Реализация бизнес- логистики	Детали реализации
Пользователь (6-й уровень)	Фактические базы данных	Исполняемый код и инструкции к функциям	Описание взаимодействия в сети	Обученный персонал	Список фактических бизнес- событий	Работающие правила	Оценка функциони- рования

Рис. 29 – Модель Дж. Захмана

Эти аспекты предлагается описывать в шести разных, связанных между собой уровнях, сгруппированных в строки матрицы: от уровня высшего руководства («планировщика застройки») до технического специалиста.

Перспективы (строки в таблице) соответствуют различным уровням управления предприятием:

– *первый уровень* – уровень интересов высшего руководства и собрания акционеров. В применении к деятельности предприятия – это верхняя строка таблицы, представляющая, по сути, контекст модели. В данной строке демонстрируется планирование бизнеса в целом (бизнес-модель), вводятся общие понятия, определяющие бизнес (продукты, услуги, клиенты, расположение объектов бизнеса), формулируется бизнес-стратегия (колонка «Стратегия»). Данная строка определяет контекст всех последующих строк;

– *второй уровень* – отражает интересы бизнес-менеджеров и владельцев процессов, на нем определяется концептуальная модель, которая предназначена для описания ключевых и вспомогательных бизнес-процессов. Две верхние строки соответствуют наиболее общим представлениям и описывают существующее окружение, планы и цели;

– *третий уровень* – уровень организации «командной» работы бизнес-менеджеров, бизнес-аналитиков и менеджеров, отвечающих за разработку ИТ. Это уровень логической модели, здесь бизнес-процессы описываются в терминах информационных систем, включая различные типы данных, правила их преобразования и обработки для выполнения определенных на втором уровне бизнес-функций;

– *четвертый уровень* и последующие описывают детали, представляющие интерес для ИТ-менеджеров, проектировщиков, разработчиков. На нем определяется технологическая модель, включающая физическую модель и детали реализации, т. е. осуществляется привязка данных и операций над ними к выбранным технологиям реализации;

– *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 такие объекты, как инструкции для работы с системой, фактические базы данных.

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

Колонка «Данные» (ответ на вопрос «ЧТО?») 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» 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). Пятый уровень содержит описание модели на языке управления данными для формирования таблиц, готовые библиотеки классов, табличные пространства СУБД. Шестой уровень может описывать фактические наборы данных, в том числе такие характеристики, как журналы доступа, размеры реально занимаемого дискового пространства, статистика обращений и т. п.

Колонка «Функции» (ответ на вопрос «КАК?») предназначена для описания последовательной детализации способов реализации миссии предприятия на уровне отдельных операций. На первом уровне достаточным будет простое перечисление бизнес-процессов. Второй уровень будет содержать 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.

Колонка «Сеть» (ответ на вопрос «ГДЕ?»)) определяет пространственное распределение компонентов системы и сетевую организацию. На уровне планирования бизнеса достаточно определить расположение всех производственных объектов. На втором 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 компонентов информационной системы к узлам сети. Четвертый уровень служит для определения физической реализации в терминах аппаратных платформ и системного программного обеспечения, используемых для интеграции различных компонентов ИС между собой. На пятом уровне определяются используемые протоколы и спецификации каналов связи. На шестом уровне описывается функционирование реализованной сети.

Колонка «Организации» (ответ на вопрос «КТО?»)) определяет участников процесса. На уровне планирования бизнеса здесь представлен список подразделений предприятия и выполняемые ими функции. На втором 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), требования к интерфейсам пользователя и правила доступа к отдельным объектам (уровень 4), их физическая реализация на уровне кода или операторов определения доступа к таблицам в СУБД (уровень 5). Шестой уровень описывает обученных пользователей системы.

Колонка «Расписание» (ответ на вопрос «КОГДА?»)) определяет временные характеристики бизнес-процессов и работы системы. Детализация осуществляется сверху вниз, начиная от списка важных событий (уровень 1) и календарного плана (уровень 2), характеризующих выполнение 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 четвертом 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), шестой уровень представляет историю функционирования системы.

Колонка «Стратегии» (ответ на вопрос «ПОЧЕМУ?») служит для определения мотивации и задает порядок перехода от задач бизнеса к требованиям и элементам ИС. Исходной точкой является бизнес-стратегия (уровень 1), транслируемая последовательно в бизнес-план (уровень 2), правила и ограничения для реализации бизнес-процессов (уровень 3), соответствующие приложения, необходимые для включения в состав ИС и в дальнейшем в их физическую реализацию (четвертый уровень).

Таблица заполняется по следующим *правилам*:

- каждая клетка таблицы независима от других, вместе они образуют функционально полное пространство для описания системы («базис»);
- каждая клетка содержит соответствующее описание аспекта реализации системы в виде определенной модели или, возможно, простого описания (текстового документа);
- порядок следования колонок несущественен;
- базовые модели для каждой колонки уникальны;
- модели в клетках каждого ряда в совокупности образуют полное описание системы с выбранной перспективы;
- заполнение клеток проводится последовательно «сверху вниз».

Итак, модель Захмана позволяет концентрироваться на отдельных аспектах архитектуры и сохраняя общий взгляд на предприятие как единое целое, однако не поддерживает представление о динамике развития организации и ее ИС.

Обобщение подхода Захмана было предложено в работах Е.З. Зиндера [31]. 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 которой располагаются отрезки времени осуществления различных проектов и стадий развития информационных систем и всего предприятия.

Таким образом, была создана «объемная» схема архитектуры предприятия (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.

4.2 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 (рис. 30). Данный подход позволяет обеспечить отслеживание логической связи между выбранными технологиями, их ценностью и потребностями для бизнеса [8, 32].

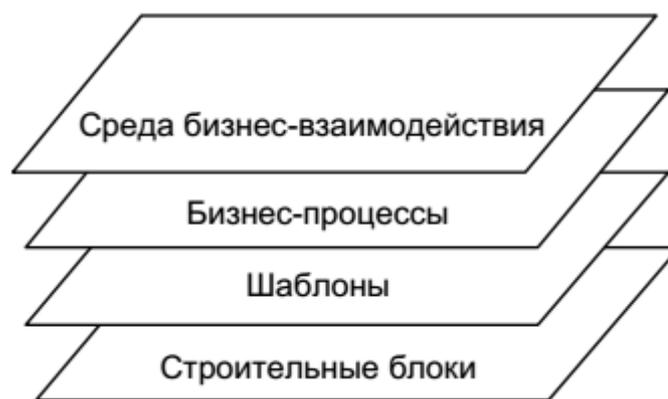


Рис. 30 – Уровни модели архитектуры Gartner

Модель Gartner 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 (системы электронной коммерции – программно-аппаратные комплексы, являющиеся инструментами для осуществления торгово-закупочной деятельности в сети Интернет). 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. Для приложений акцент сделан на использовании *шаблонов сервис-ориентированной архитектуры*, т. е.

реализации приложений в виде модульного набора различных типов сервисов, что позволяет в перспективе интегрировать приложения как web-сервисы;

4) *технологические строительные блоки (bricks)*, соответствующие технологической архитектуре и включающие в себя операционные системы, серверы, базы данных, собственно данные и пр.

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 (рис. 31).

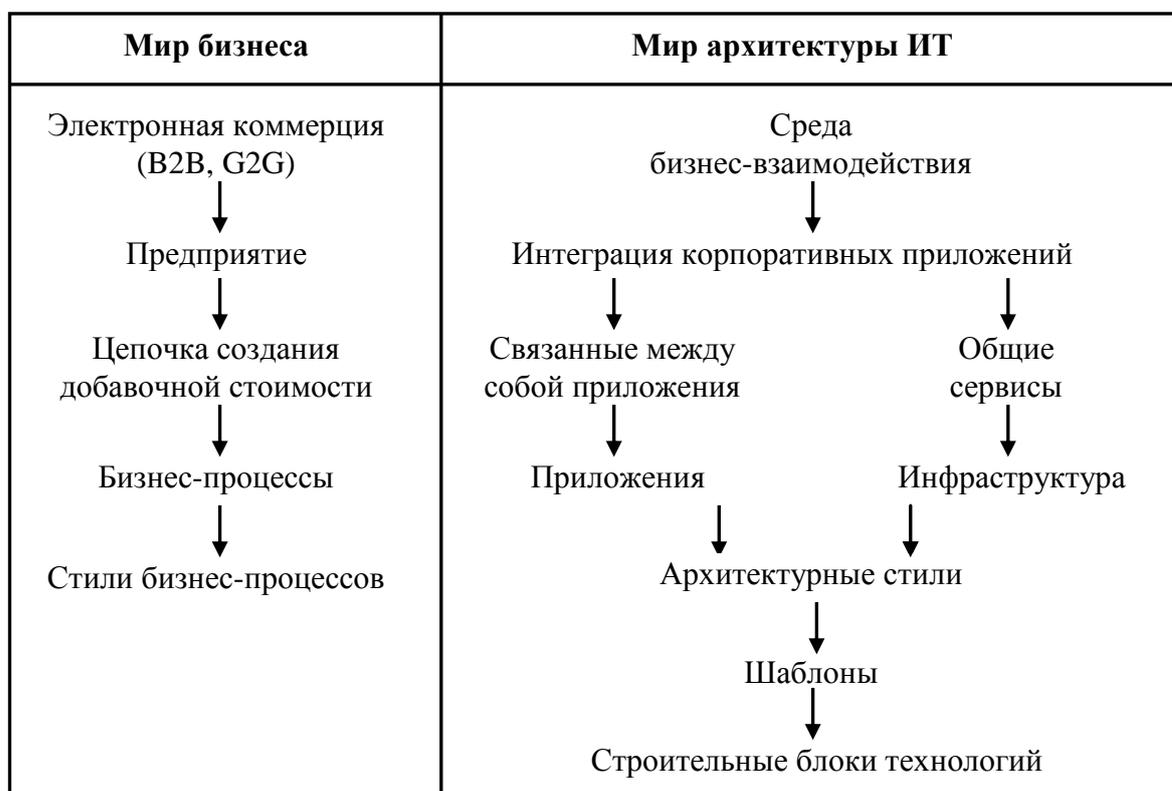


Рис. 31 – Архитектура ИТ в бизнес-контексте

4.3 Методика Meta Group

Методика META Group соответствует эволюции понятия «Архитектура предприятия», включая в свой состав общепринятые домены и рассматривает АП в интеграции с ключевыми процессами и проектами управления корпоративными ИТ-программами на основе процесса выработки стратегии и планирования.

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

- 1) «Видение общих требований» (CRV);
- 2) «Принципы концептуальной архитектуры» (CA).

Организация рабочего процесса разработки архитектуры и быстрое создание начальной версии архитектуры предприятия в соответствии с методикой META Group состоит в прохождении нескольких *этапов*:

- 1) разработка общих требований;
- 2) разработка концептуальной архитектуры;
- 3) разработка плана реализации.

Этап 1. Разработка общих требований включает:

- анализ развития внешней среды (технологические тенденции);
- выбор бизнес-стратегий и основных движущих сил;
- формирование требований к информационным системам со стороны бизнеса;
- формирование требований к технологической архитектуре, обеспечивающей адекватные возможности для информационных систем с точки зрения потребностей бизнеса.

Документ, представляющий видение общих требований, может быть оформлен в виде таблицы (табл. 4).

Таблица 4 – Пример оформления общих требований

Тенденция	Формулировка требования		
	Бизнес-стратегия предприятия	Требования к ИС	Требования к архитектуре
Наличие задержки в предоставлении услуги для 20 % клиентов	Увеличение доли рынка, достигаемое за счет рациональной организации процесса обслуживания, позволяющей сократить время ожидания для клиента	Оперативная (незамедлительная) передача в производство информации о заказах, независимо от канала и места их получения	Наличие ИТ-инфраструктуры, обеспечивающей управляемый доступ и своевременную передачу информации с целью достижения операционной эффективности

Важным аспектом является документирование явных связей между бизнес-стратегией (потребностями бизнеса) и требованиями к информационным системам и в конечном итоге установление логических связей с требованиями к технологической архитектуре.

Для установления логических связей рекомендуется использовать простые матрицы, представленные на рис. 32 [8, 10].

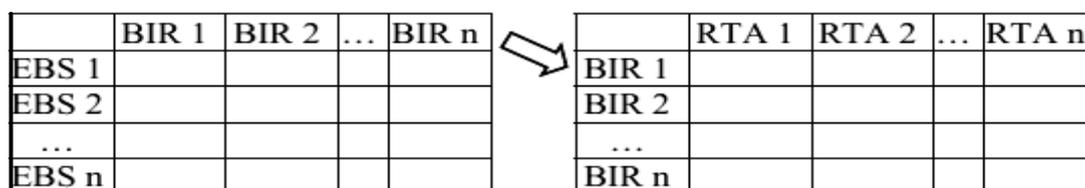


Рис. 32 – Матрица связей между элементами архитектуры предприятия: бизнес-стратегиями (EBS – Enterprise Business Strategy); требованиями к ИС (BIR – Business Information Requirements); требованиями к технологической архитектуре (RTA — Requirements for Technical Architecture)

Документированные связи послужат основой для принятия решений об инвестициях. Видение общих требований агрегирует все требования к технологической архитектуре, и это служит основой для формулировки принципов концептуальной архитектуры.

Этап 2. Разработка концептуальной архитектуры, определяющей логически связанный набор принципов, использование которых обеспечивает общее руководство для развития ИС предприятия и технологической инфраструктуры. На этом этапе параллельно ведется и разработка наиболее приоритетных доменов архитектуры. Здесь же выполняется анализ на несоответствие между текущим и желаемым состояниями архитектуры.

Концептуальная архитектура разрабатывается еще до создания других архитектурных доменов и основана на принципах, суть которых состоит в следующем:

- принципы представляют собой утверждения, которые касаются архитектурного процесса или содержания архитектуры;
- количество принципов ограничено, и они являются неизменным фундаментом для построения архитектуры;
- принципы должны быть утверждениями, чья справедливость для организации имеет «вечный» характер, поскольку они задают систему ценностей для архитектуры в целом.

В соответствии с методикой META Group результатом разработки принципов концептуальной архитектуры является выделение в технологической архитектуре (EWTA) набора доменов (предметных областей), которые объединяют группы связанных между собой технологий и компонентов. Каждый документ, описывающий домен технологической архитектуры, включает:

- формулировку миссии и стратегических целей домена;
- описание компонентов домена с точки зрения единого представления включенных в домен технологий;
- принципы проектирования, принятые в домене;
- обоснования и последствия принятия принципов;
- продукты и технические стандарты, обеспечивающие выполнение требований к технологической архитектуре. Выделяют стратегические (предпочтительные) стандарты, переходные (временно используемые),

устаревшие (подлежащие обновлению либо замене) и исследовательские или новые (находящиеся на этапе рассмотрения либо апробации);

– примеры лучших практических реализаций;

– конфигурации, формулируемые в случаях, когда необходимо облегчить принятие сложных решений или уменьшить общую стоимость владения за счет стандартных конфигураций;

– несоответствия между существующим состоянием домена технологической архитектуры и желаемым состоянием. Это служит основой для последующей работы группы, которая отвечает за данный домен архитектуры.

Взгляд на технологическую архитектуру с точки зрения предоставляемых ею инфраструктурных сервисов обусловлен распространением принципов сервис-ориентированной архитектуры. По мере повышения уровня абстракции выделяется *четыре группы сервисов*:

1) базовые инфраструктурные сервисы: общие, стандартные технологии, широко используемые в рамках всех ИТ-систем предприятия. Они ориентированы не на разработчиков прикладных систем, а на специалистов по инфраструктуре (примерами являются ПО пересылки сообщений промежуточного слоя, мониторы транзакций, сервисы каталогов);

2) общие (framework) инфраструктурные сервисы: общие, совместно используемые технологии, не содержащие готовой бизнес-логики, ориентированные на разработчиков и частично стандартизированные (примерами являются управление контентом, серверы приложений, серверы выполнения бизнес-правил);

3) общие (framework) бизнес-сервисы, которые могут быть использованы в рамках различных бизнес-процессов, поскольку содержат готовую, предопределенную бизнес-логику (примерами являются модули определения цены товара, персонализации информации, оценки кредитного рейтинга);

4) прикладные бизнес-сервисы, содержащие специфическую для отдельных бизнес-процессов высокоуровневую бизнес-логику (сервисы CRM-систем или систем управления поставками).

Этап 3. Разработка плана реализации, обеспечивающего миграцию в сторону желаемого состояния архитектуры. В полном описании методики META Group 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.

4.4 Методика TOGAF

Методика описания архитектуры TOGAF (The Open Group Architecture Framework) [30] была предложена некоммерческим объединением The Open Group, и позиционируется не как некоторая 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. Основным полем для применения TOGAF является программная инфраструктура ИС, и, соответственно, в наилучшей мере подходит для описания интеграционных компонентов, используемых для поддержки широкого спектра корпоративных приложений, прежде всего, критичных для бизнеса. Поскольку интеграционная архитектура сильно зависит от принимаемых решений в остальных областях, то в рамках методики TOGAF в необходимой степени рассматриваются и эти смежные области.

В состав модели TOGAF входят *два основных компонента*:

– методика ADM (Architecture Development Method), определяющая процесс разработки архитектуры;

– 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 включает следующие фазы:

– подготовка – уточнение модели с учетом особенностей организации, определение принципов реализации проекта;

– фаза А – определение границ проекта, разработка общего представления архитектуры; утверждение плана работ и подхода руководством;

– фаза В – разработка бизнес-архитектуры предприятия;

– фаза С – разработка архитектуры данных и архитектуры приложений;

– фаза D – разработка технологической архитектуры;

– 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, в свою очередь, разбивается на подпроцессы (этапы), отдельные работы и так далее.

Для каждого такого подпроцесса определяются решаемые в ходе его выполнения задачи, входные и выходные документы.

Важно отметить, что процесс предусматривает не обязательную, но возможную адаптацию метода к условиям конкретного предприятия, которая осуществляется на предварительной фазе. Это может быть вызвано как необходимостью учета других существующих стандартов предприятия, так и привлечением аутсорсинговых компаний к разработке архитектуры.

Процесс разработки не заканчивается выбором оптимальной архитектуры и разработкой плана миграции. Необходимыми элементами являются задачи, выполняемые на фазах G и H.

Важно отметить, что TOGAF распространяется свободно и может быть использована бесплатно любой организацией для разработки внутренних проектов.

4.5 Модель представления архитектуры «4+1»

Достаточно важную роль в развитии подходов к описанию архитектуры предприятия сыграла модель «4+1» (The 4+1 View Model of Architecture), предложенная Ф. Кручтенем (Philippe Kruchten) из компании Rational в 1995 г.

Модель предлагает простой и понятный способ описания архитектуры сложных систем, который состоит в использовании четырех основных представлений и объединяющего их сценария (рис. 33) [33].

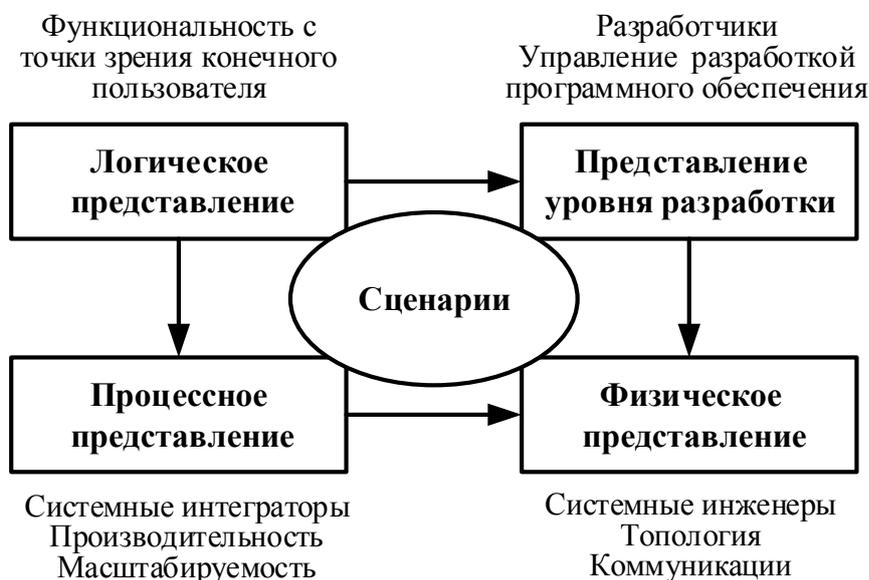


Рис.33 – Модель «4+1»

Представлениями в этой модели являются:

- 1) логическое представление;
- 2) процессное представление;

- 3) физическое представление;
- 4) представление уровня разработки.

Логическое представление является объектной моделью проектирования (в случае использования объектно-ориентированной модели проектирования). Основной целью логического представления в данной модели является описание функциональных требований (назначение системы в терминах конечных пользователей). Для этого представления используются различные абстрактные конструкции, такие как объекты и классы объектов. Для их иллюстрирования могут применяться диаграммы классов в нотации языка UML либо, например, диаграммы «сущность-связь», если в разработке приложения доминируют данные.

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. Архитектура процессов может быть представлена на различных уровнях абстракции. На высшем уровне система рассматривается как набор независимо выполняемых сетей взаимодействующих между собой программ. На более низких уровнях рассматриваются процессы и задачи.

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.

Описание архитектуры системы посредством четырех основных представлений иллюстрируется и проверяется путем использования *сценария*, сформированного при объединении этих представлений. Архитектура системы во многом определяется данными сценариями. Каждое представление отражает специфические аспекты моделируемой системы.

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.

4.6. Стратегическая модель архитектуры SAM

Стратегическая модель архитектуры SAM (Strategic Architecture Model) [35] использует нотацию «*сфер интересов*» для представления целостного набора фактов о предприятии и «отношений», которые связывают эти факты в полезные группы, что обеспечивает полезный 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 (рис. 34) .

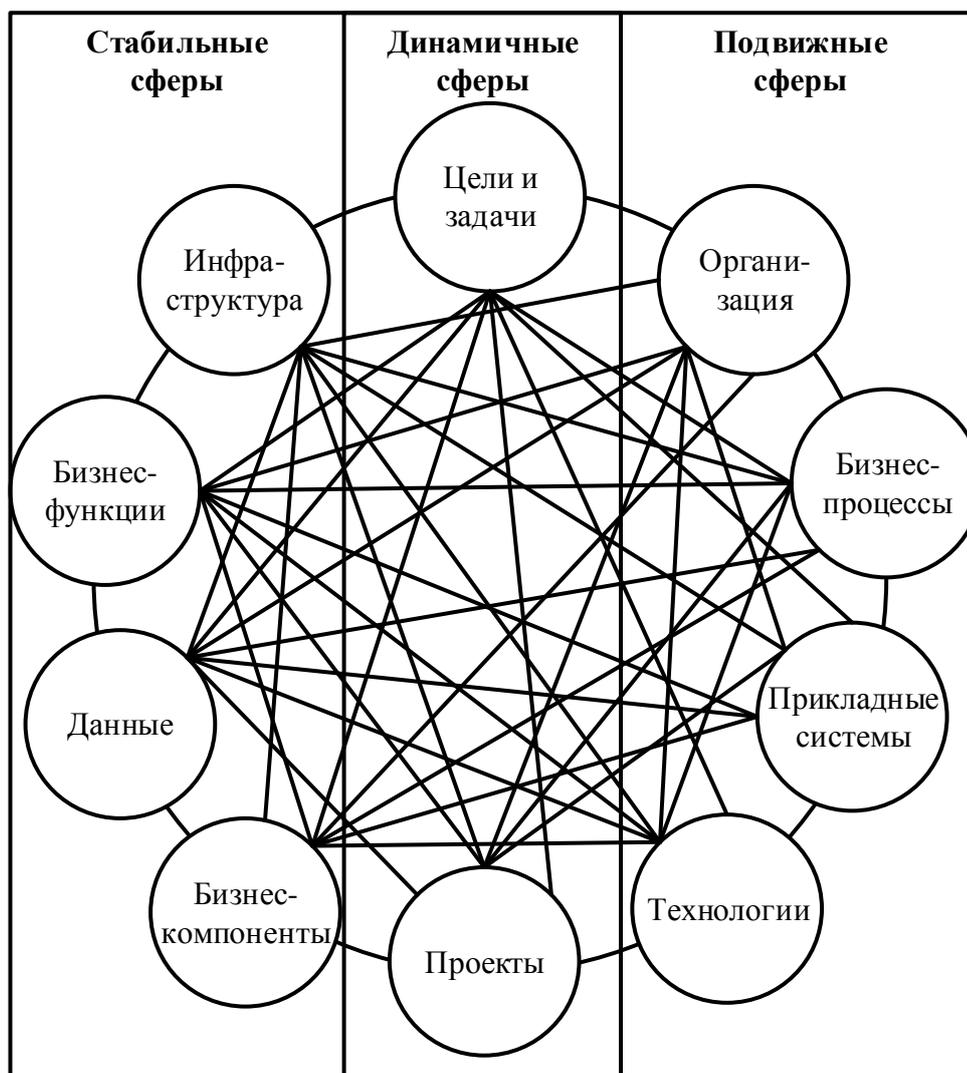


Рис. 34 – Типичные сферы интересов SAM

SAM использует итеративный подход для создания архитектуры, сочетающий элементы разработки «сверху-вниз» и «снизу-вверх». Сфера может заполняться в направлении «снизу-вверх» путем сбора относящейся к предметной области информации, которая обобщается на более высоких уровнях. Либо же заполнение может идти в направлении «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.

Наибольшую важность представляют следующие *сферы*: цели и задачи, организация, бизнес-процессы, прикладные системы, технологии, проекты, бизнес-компоненты, данные, бизнес-функции, инфраструктура.

Внутри каждой области интересов сохраняется информация об определенной предметной области, что обеспечивает простоту ее сопровождения и извлечения. Обычно применяется одна или несколько иерархических структур, которые напоминают ящик с файлами. Минимальный объем информации, относящейся к какой-либо сфере, называется *элементом* (member).

Группировка функций, создающих и обновляющих одни и те же элементы данных с помощью процесса «Коммутативная кластеризация», позволяет определить избыточное количество «строительных блоков» – *компонентов*, которые могут использоваться для построения систем и приложений, поддерживающих определенные бизнес-процессы.

Компоненты являются важными конструкциями в современных подходах к разработке систем. Они предлагают «сервисы», которые могут использоваться в совокупности с сервисами, предлагаемыми другими компонентами в рамках сервис-ориентированной архитектуры.

Можно выделить *три категории сфер* (рис. 34):

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 с целью создания отличий от конкурентов и обеспечения динамичности в деятельности. Сферы этой категории – организация, бизнес-процессы, прикладные системы и технологии – представляют собой области, которые организация может изменить достаточно быстро. 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. Такими областями являются бизнес-функции, данные, бизнес-компоненты и в определенной степени инфраструктура.

4.7 Архитектурные концепции и методики Microsoft

Крупные компании-поставщики инфраструктурных информационных технологий, такие как Microsoft, IBM, SAP и другие, располагают возможностями создания собственных методик разработки архитектуры предприятия с учетом области деятельности специализации.

Взгляды компании Microsoft [36] относительно АП в большей степени сфокусированы на процессах разработки конкретных программных прикладных систем и создании технологической инфраструктуры, включая центры обработки данных различного масштаба и уровня надежности. Аналогично многим методикам, Microsoft выделяет *четыре домена*:

- 1) бизнес-архитектура;
- 2) архитектура информации;
- 3) прикладные системы;
- 4) технологическая архитектура.

Данные представления рассматриваются на различных *уровнях абстракции*: концептуальном, логическом и физическом.

Кроме того, явно выделяются процессы разработки прикладных систем, организация процессов эксплуатации технологической инфраструктуры и создание соответствующих шаблонов, которые могут использоваться как при проектировании архитектуры систем, так и при ее построении. При этом компания Microsoft выработала достаточно подробные методики, отражающие различные аспекты архитектуры: Microsoft Solutions Framework (MSF), Microsoft Operations Framework (MOF), Microsoft Systems Architecture (MSA) и Microsoft Solutions for Management (MSM).

Методики MSF и MSA в большей степени относятся к процессу разработки архитектуры прикладных систем и инфраструктуры соответственно, а методики MOF и MSM – к архитектуре системного управления, т. е. к вопросам управления и эксплуатации. При этом MOF и MSF нацелены на различные, но связанные между собой фазы жизненного цикла ИТ-решений (рис. 35).

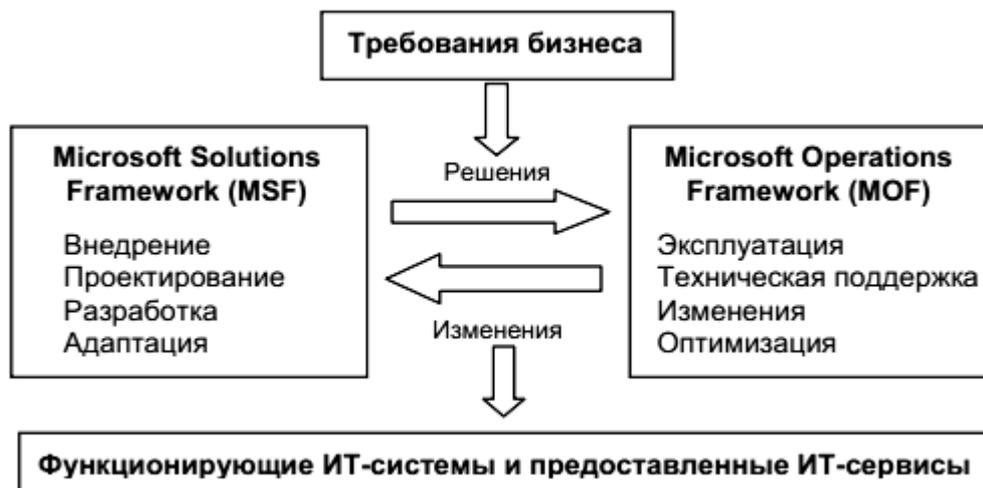


Рис. 35 – Организация взаимодействия MSF и MOF для удовлетворения запросов бизнеса

Методики Microsoft сосредоточены в основном на системном уровне – уровне архитектуры прикладных систем и обеспечивающей инфраструктуры. В этой достаточно «узкой» области полезными являются приведенные соотношения между различными перспективами описания системы и моделями, используемыми для описания на соответствующем уровне абстракции (рис. 36).

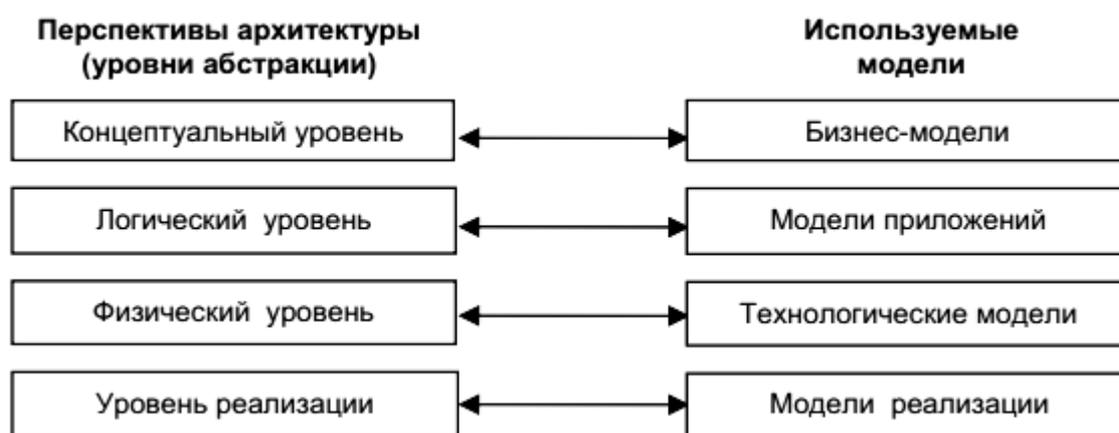


Рис. 36 – Различные перспективы архитектуры системы и используемые модели

На рис. 37 показаны взаимосвязи между различными перспективами в описании архитектуры и используемыми шаблонами проектирования, а также примерно отображено соответствие между методиками Microsoft и элементами архитектуры предприятия.

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. Данные 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.

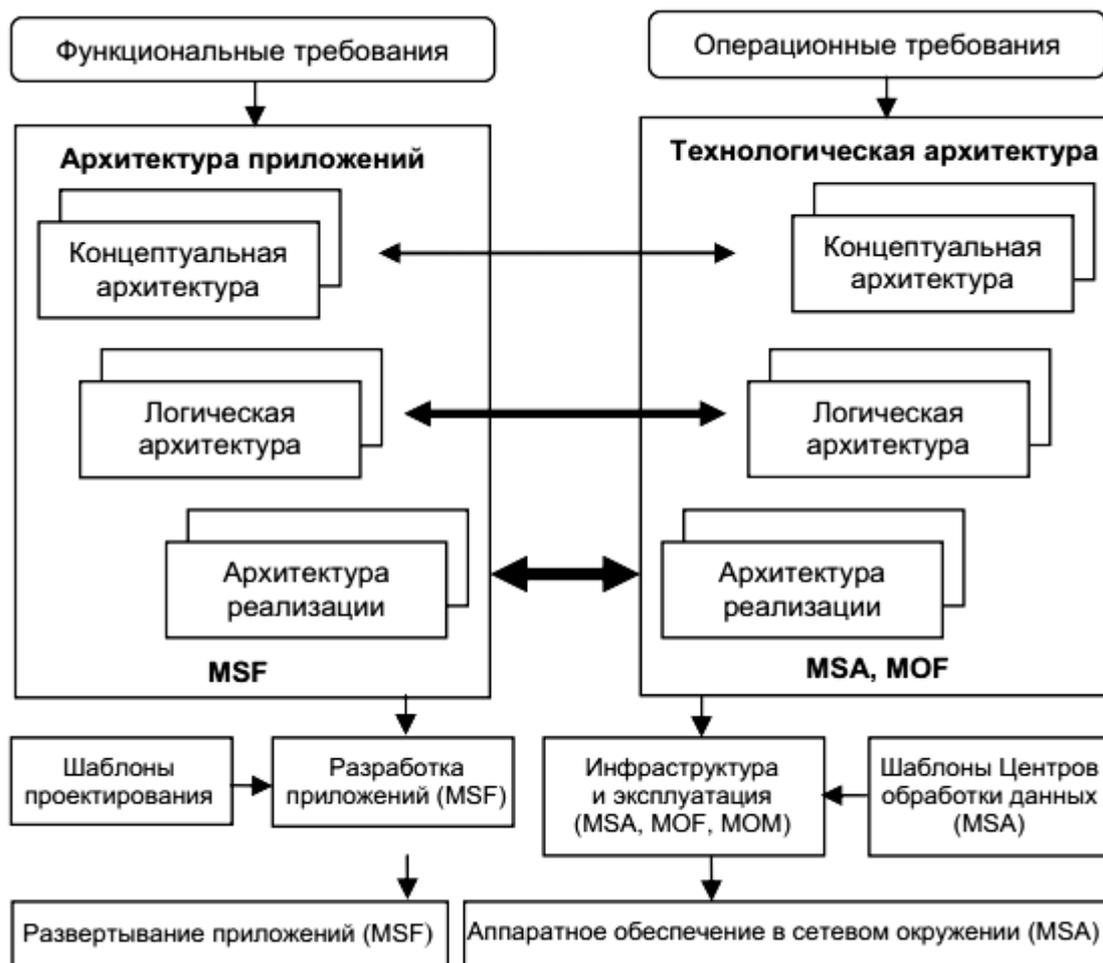


Рис. 37 – Организация взаимосвязи компонентов архитектуры предприятия с использованием методики Microsoft

Архитектурные концепции и шаблоны могут использоваться на различных уровнях проектирования архитектуры прикладной системы (рис. 38).

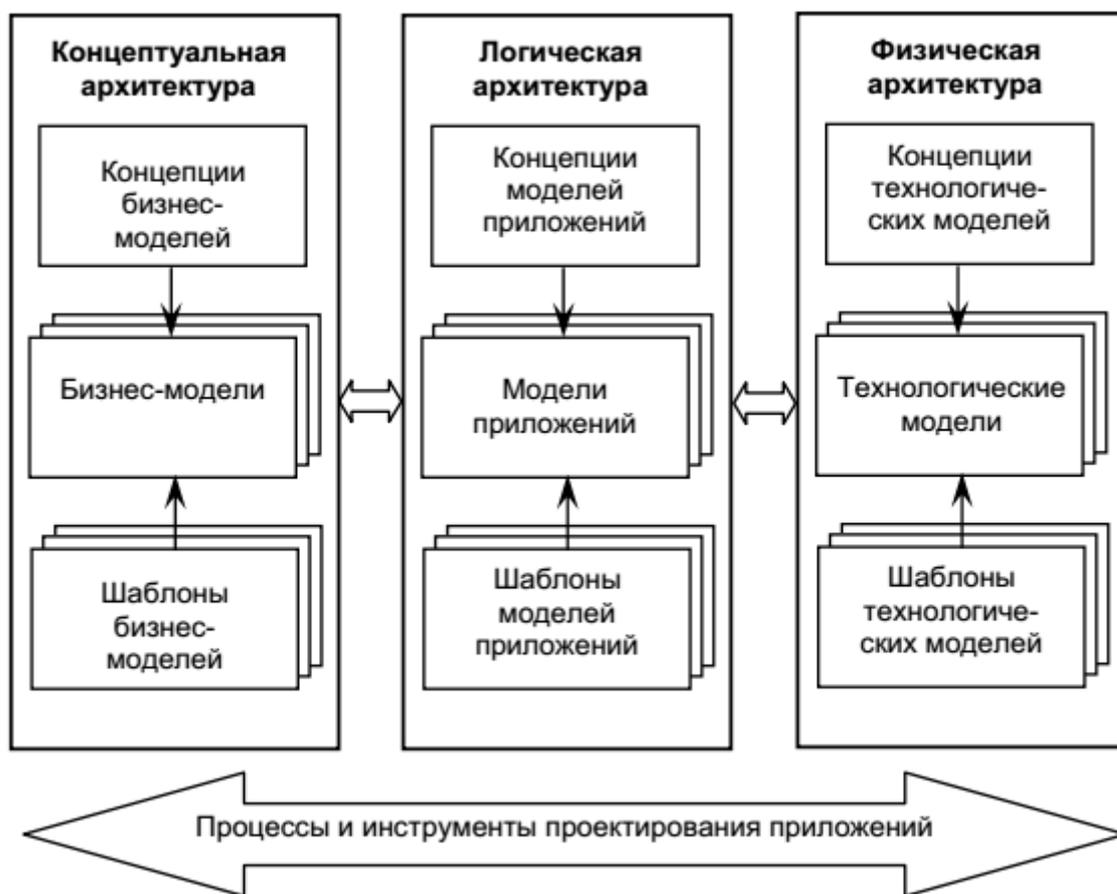


Рис. 38 – Концепции и шаблоны по построению архитектуры приложений

Уровнями проектирования являются:

- уровень концептуальной архитектуры в форме концепций построения бизнес-моделей и соответствующих шаблонов;
- уровень логической архитектуры в форме концепций построения моделей приложений и соответствующих шаблонов;
- уровень физической архитектуры в форме концепций построения технологических моделей и соответствующих шаблонов.

Использование данных концепций и шаблонов является важным условием успешного, быстрого и эффективного с точки зрения затрат создания систем и использования ИТ-технологий организациями.

4.8 Методологические принципы использования стандарта COBIT 5 при построении архитектуры предприятия

Весомое место среди методологий проектирования архитектуры предприятия занимает стандарт COBIT (Control Objectives for Information and Related Technologies). Развиваясь как система контроля и аудита информационных технологий на предприятиях различных сфер деятельности, на данный момент COBIT 5 интегрирует лучшие практики применения архитектурного подхода и представляет собой сбалансированную методологию решения задач управления ИТ-архитектурой в масштабах всего предприятия, как в областях функциональной ответственности информационных технологий, так и бизнеса, что в свою очередь позволяет оптимизировать выгоды предприятия от основных направлений его деятельности, минимизировать риски и эффективно использовать имеющиеся в его распоряжении ресурсы.

Основу реализации методологии COBIT 5 составляют пять фундаментальных принципов руководства и управления информационными технологиями на предприятии [37]:

- соответствие потребностям заинтересованных сторон;
- комплексный взгляд на предприятие;
- применение единой интегрированной методологии;
- обеспечение целостности подхода;
- разделение руководства и управления.

Анализ возможностей практической реализации указанных принципов позволил разработать концептуальную схему управления информационными технологиями на предприятии (рис. 39).

Проанализируем принципы, положенные в основу методологии COBIT 5 и составляющие базис представленной концептуальной схемы управления информационными технологиями.



Рис. 39 – Концептуальная схема управления информационными технологиями согласно методологии COBIT 5

Принцип 1 регламентирует процесс постановки стратегических целей и разработки на их основе тактических и оперативных задач деятельности с учетом потребностей заинтересованных сторон. Принципиальной задачей деятельности предприятия является получение выгоды путем создания некоторой ценности для потребителя. При этом эффективное управление предполагает сопоставление предполагаемых результатов с наличием имеющихся ресурсов и возникающих при реализации решений рисков различной природы. Методология COBIT 5 позволяет адаптировать исходную модель под запросы каждого предприятия путем сбора и анализа информации от достаточно широкого круга заинтересованных сторон относительно их

ожиданий от информационных и связанных технологий (выгоды, приемлемый уровень риска и цены) и приоритетов в обеспечении уверенности в получении планируемой ценности. Одни заинтересованные лица хотят получить отдачу в краткосрочном периоде, другим интересна долгосрочная стабильность. Одни готовы к высокой степени риска, а другие нет. Этими ожиданиями, различающимися и, возможно, конфликтующими, нужно эффективно управлять. Более того, заинтересованные стороны не только хотят участвовать в принятии решений, но и требуют прозрачности – как выполняемых работ, так и результатов. Кроме этого необходимо принимать во внимание необходимость управления возрастающей зависимостью успешности предприятий от внешних контрагентов бизнеса и ИТ, таких как аутсорсинговые компании, поставщики, консультанты, клиенты, поставщики облачных и других услуг, а также от разнообразия внутренних способов и механизмов формирования ожидаемой ценности.

Для решения этих задач COBIT 5 располагает инструментарием каскадирования целей от уровня стратегии предприятия до уровня управляемых и конкретных ИТ-целей задач, и связанных с ними процессов и практик реализации. Алгоритм его использования заключается в следующем:

- движущие силы заинтересованных сторон влияют на их потребности;
- потребности заинтересованных сторон связываются с целями предприятия;
- цели предприятия связываются с ИТ-целями;
- ИТ-цели связываются с целями факторов влияния.

Согласно *принципу 2* обеспечивается комплексный взгляд на предприятие, где руководство и управление информационными технологиями встраивается в руководство предприятием в целом и рассматривается как его неотъемлемая часть и ключевой актив деятельности.

Исходя из того, что информационные технологии работают по всей цепочке создания ценности, средства COBIT 5 позволяют описать все функции и процессы, необходимые для руководства и управления информационными и

связанными с информацией технологиями на предприятии, где бы ни проводилась обработка информации.

Принцип 3 обеспечивает целостный и системный взгляд на руководство и управление информационными и связанными технологиями на предприятии, основанный на наборе факторов влияния. Методология COBIT 5 описывает семь видов факторов влияния, которые определяют условия и среду реализации процессов руководства и управления как информационными технологиями, так и предприятием в целом:

– принципы, политики и подходы – обеспечивают трансляцию желаемого поведения в практические рекомендации по оперативному управлению предприятием;

– процессы – описывают структурированный набор практик и видов деятельности, необходимых для выполнения определенных задач и направленных на получение набора результатов, обеспечивающих достижение ИТ-целей;

– организационная структура – регламентирует состав и иерархию подразделений предприятия и является ключевой сущностью для принятия и реализации управленческих решений на предприятии;

– культура, этика и поведение;

– информация – включает в себя всю информацию, производимую и используемую на предприятии;

– услуги, инфраструктура и приложения – механизмы, предоставляющие предприятию инструменты обработки информации;

– люди, навыки и компетенции – необходимы для успешного выполнения всех видов деятельности, принятия эффективных управленческих решений и выполнения корректирующих действий.

При этом каждый из факторов влияния имеет следующие атрибуты: заинтересованные стороны, цели, жизненный цикл и общепринятые практики их использования, определяемые в процессе каскадирования целей и анализа выгод, возможностей оптимизации рисков и ресурсов.

Принцип 4 разделяет процессы руководства и управления. Руководство, согласно методологии COBIT 5, обеспечивает уверенность в достижении целей предприятия путём: сбалансированной оценки потребностей заинтересованных сторон, существующих условий и возможных вариантов; установления направления развития через приоритизацию и принятие решений; постоянного мониторинга соответствия фактической продуктивности и степени выполнения требований установленным направлению и целям предприятия. А управление заключается в планировании, построении, выполнении и отслеживании деятельности, в соответствии с направлением, заданным органом руководства, для достижения целей предприятия.

Эталонная модель процессов COBIT 5 представлена на рис. 40.

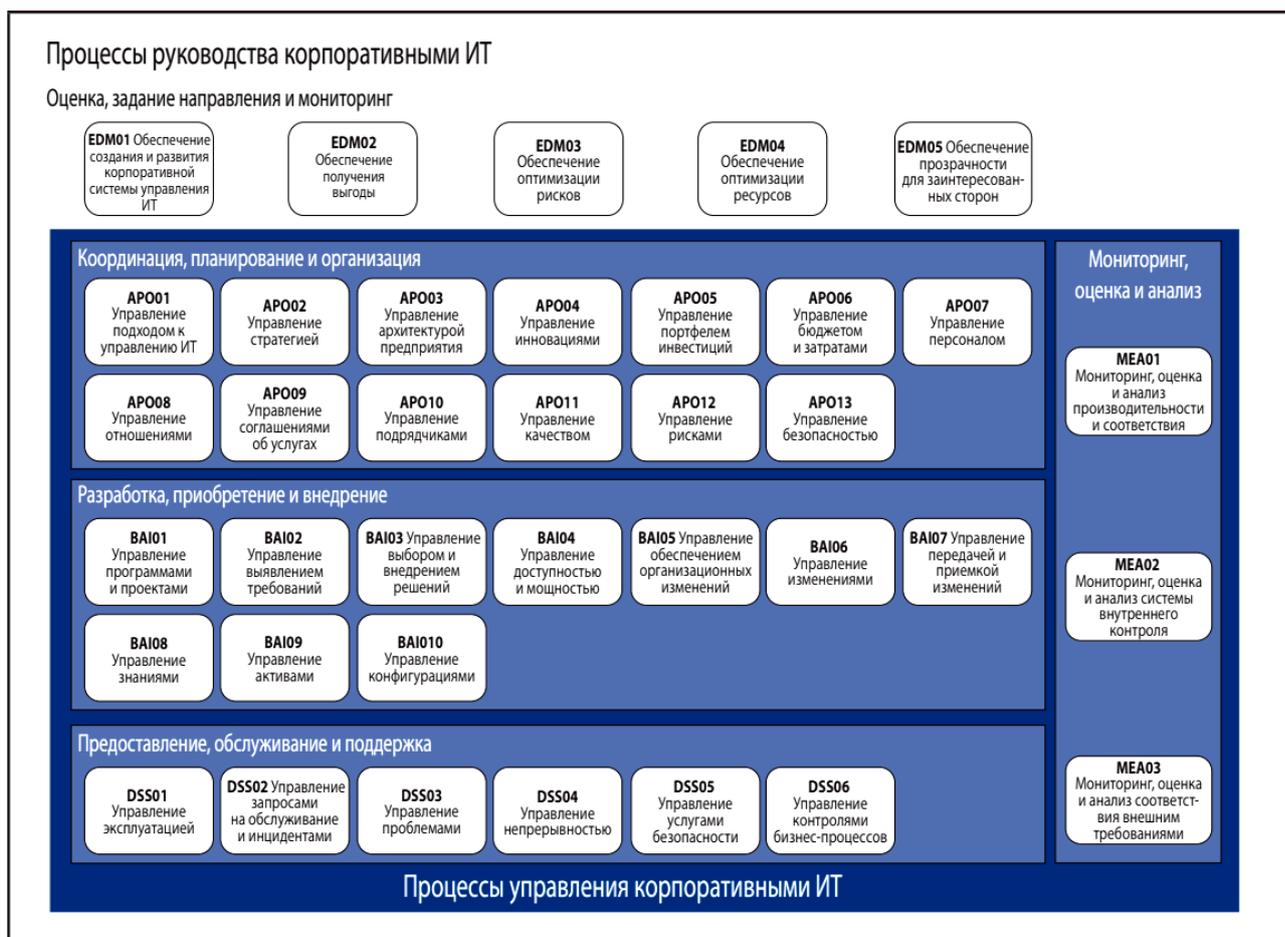


Рис.40 – Эталонная модель процессов COBIT 5

Данная модель разделяет все процессы руководства и управления ИТ на предприятии на два процессных домена:

– руководство – включает пять процессов руководства, а в каждом из них определены практики оценки, задания направления и мониторинга (Evaluate, Direct and Monitor (EDM)).

– управление – состоит из четырех суб-доменов, соответствующих областям ответственности:

1) координация, планирование и организация – Align, Plan and Organise (APO);
2) разработка, приобретение и внедрение – Build, Acquire and Implement (BAI);
3) предоставление, обслуживание и поддержка – Deliver, Service and Support (DSS);

4) мониторинг, оценка и анализ – Monitor, Evaluate and Assess (MEA).

Принцип 5 исключает противоречие, возникающее при отказе руководства предприятия от использования других методологий архитектурного подхода в пользу COBIT 5. Методология COBIT 5 разработана с крупнейших методологий и стандартов ИТ-индустрии, таких как Information Technology Infrastructure Library (ITIL®), The Open Group Architecture Framework (TOGAF®), Project Management Body of Knowledge (PMBOK®), PRojects IN Controlled Environments 2 (PRINCE2®), Committee of Sponsoring Organizations of the Treadway Commission (COSO), а также стандартов International Organization for Standardization (ISO). На рисунке 41 представлена схема пересечения методологических аспектов различных стандартов, которые были положены в разработку эталонной процессной модели COBIT 5.

Анализ соответствия методологии COBIT5 принципам ратифицированного в Российской Федерации международного стандарта ГОСТ Р ИСО/МЭК 38500-2017 «Информационные технологии (ИТ). Стратегическое управление ИТ в организации» [38], подтвердил ее целесообразность и эффективность использования как в контексте управления информационными технологиями в частности, так и архитектурой предприятия в целом.

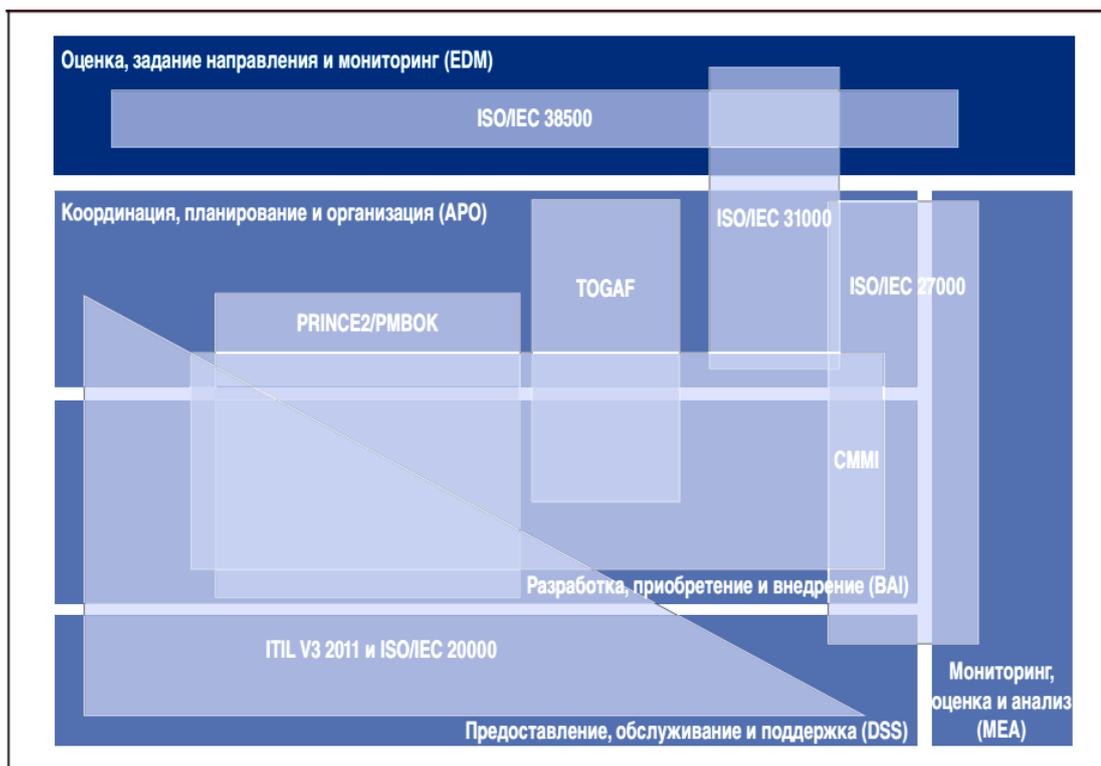


Рис. 41 – Связи COBIT 5 с прочими стандартами и подходами

Целесообразность применения методологии COBIT 5 в условиях работы современных предприятий подтверждена ее соответствием ряду международных стандартов и мировых практик использования архитектурного подхода.

Практическая значимость использования стандарта и эталонной модели COBIT 5 заключается в повышении наглядности и прозрачности управления информационными технологиями, повышении уровня безопасности и контроля их использования.

4.9 Метод планирования архитектуры организации EAP

Модель EAP соответствует рассмотренному ранее принципу сегментного подхода к разработке архитектуры и представляет собой метод планирования 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.

ЕАР декларирует *10 этапов* (табл. 5), 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.

Таблица 5 – Этапы планирования архитектуры

№ п/п	Название этапа	Результаты
1	Инициация планирования	Цели, видение, методологии, инструментарий, команда, презентации, рабочий план
2	Предварительное бизнес-моделирование	Организационно-штатная структура, предварительная функциональная бизнес-модель
3	Формирование снимка организации	Полная функциональная бизнес-модель
4	Описание текущих систем и технологий	Каталог информационных ресурсов, системные схемы
5	Формирование архитектуры данных	Определения сущностей, ER-модель, матрица сущности-функции, отчет по архитектуре данных
6	Формирование архитектуры приложений	Определения приложений, матрицы приложений, анализ покрытия, отчет по архитектуре приложений
7	Формирование технологической архитектуры	Распределение данных/приложений, отчет по технологической архитектуре
8	Разработка плана реализации	Последовательность, план перехода, цены и преимущества, факторы успеха и рекомендации
9	Заключительное планирование	Окончательный отчет, презентация
10	Переход к реализации	Совершенствование политик, стандартов, процедур, детализация проектных планов

Этапы организованы в виде четырехуровневой схемы метода ЕАР:

1) *уровень 1 (исходная позиция)* – выработка решений, которые необходимо принять для реализации соответствующей архитектуры

организации, и определение состава необходимого для реализации инструментария;

2) *уровень 2 (анализ текущего состояния)* – определение точки отсчета для преобразования существующей архитектуры в целевую, а также формирование временного графика перехода;

3) *уровень 3 (планируемая перспектива)* – определение технических деталей перспективной архитектуры (данные, приложения и технологии);

4) *уровень 4* – формирование плана реализации перспективной архитектуры.

Этап «Инициация планирования» включает в себя 7 шагов, цели, задачи и основные результаты которых описаны ниже.

Шаг первый – формальное определение области и целей планирования архитектуры для понимания участниками проекта конечных достигаемых результатов. Итог данного шага представлен перечнем согласованных и утвержденных целей, а также списком причастных к проекту подразделений организации.

Основные задачи первого шага состоят в следующем:

– обзор организации и определение ее контекста (системных входов/выходов);

– оценка характеристик деятельности предприятия по степени их влияния на реализацию проекта: способствующие реализации и затрудняющие реализацию (например: несоответствие существующих информационных систем требованиям и высокая стоимость их сопровождения; необходимость в интеграции и распределении данных; наличие неуспешных ИТ-проектов по причинам ограниченности менеджмента по времени и бюджету и т. п.);

– формирование перечня строго сформулированных целей и их достижимости;

– формирование перечня подразделений, затрагиваемых грядущими изменениями ИТ-стратегии и корпоративной культуры.

Шаг второй – исследование организации, системных входов/выходов и вариантов представлений менеджеров. Результатом является согласованное и утвержденное видение организации, обеспечивающее политическую поддержку менеджмента.

Основными задачами шага являются:

- 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.

Шаг третий – адаптация методологии планирования и создание руководства по ее реализации. Основными задачами шага являются:

- формулирование принципов и требований к методологии;
- оценка существующих на предприятии методов и стандартов;
- изучение имеющихся на рынке подходов;
- принятие решения об исполнителе (внутренние ресурсы или внешний консультант);
- создание методологии, отвечающей потребностям данного предприятия;
- разработка содержания отчетов, создаваемых на каждом из последующих этапов.

Шаг четвертый – инвентаризация компьютерных ресурсов и оценка инструментария создания архитектуры предприятия.

Основными задачами шага являются:

- 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.

Шаг пятый – создание проектной команды.

Основными задачами шага являются:

- определение квалификационных требований по каждому этапу создания архитектуры;
- оценка трудозатрат по каждому этапу создания архитектуры;
- определение необходимого числа участников;
- спецификация ролей и ответственности каждого члена команды;
- подбор персонала;
- обучение персонала (методологии и инструментарий);
- выбор внешних консультантов и определение направлений их использования.

Шаги шестой и седьмой – подготовка рабочего плана, его презентация и утверждение. В результате должен быть сформирован рабочий план и утвержден бюджет выполнения работ.

Задачей **этапа бизнес-моделирования** является обеспечение полной и исчерпывающей базой знаний всех участников проекта для ее использования при определении архитектуры и плана ее реализации. Бизнес-моделирование предполагает построение предварительной бизнес-модели, за которым следует построение полной бизнес-модели. Предварительная бизнес-модель идентифицирует функции и организационные единицы (исполнителей функций). По оценкам экспертов данный этап требует 25–30 % всех трудозатрат на моделирование, он осуществляется в 3 шага.

Шаг 1 – документирование организационной структуры. В качестве результатов имеет обновленные организационные схемы, перечень ролей и мест их выполнения, оценку количества сотрудников по ролям.

Основными задачами шага являются:

- формирование (редактирование) организационных схем и фиксация их в инструментарии;
- идентификация видов деятельности в разрезе организационных единиц;
- формирование отчетов по полученным результатам.

Шаг 2 – определение структуры бизнес-модели (идентификация и определение бизнес-функций). В качестве результатов имеет определенные функции, для которых определены следующие компоненты:

- имя;
- краткое описание либо декомпозиция на подфункции;
- принадлежность к конкретной организационной единице.

Основными задачами шага являются:

- определение основных видов деятельности и бизнес-процессов;
- функциональная декомпозиция процессов;
- развитие функциональной декомпозиции до уровня бизнес-операций;
- построение функционального иерархического дерева;
- оценка качества декомпозиции и ее улучшение;
- сопоставление функций и исполняющих их организационных единиц, построение соответствующей матрицы.

Шаг 3 – документирование бизнес-модели и ее распространение для верификации. Основными задачами шага являются:

- формирование отчетов по бизнес-модели;
- распространение отчетов и проведение презентации;
- сбор замечаний и предложений.

Полная функциональная бизнес-модель дает ответы на следующие вопросы:

- какая информация используется при выполнении функций;

- когда функция выполняется;
- где и кем функция выполняется;
- как часто функция выполняется;
- какие улучшения возможны.

Этап «Формирование снимка организации» включает в себя следующие 3 шага:

- 1) планирование, подготовку и проведение интервью;
- 2) построение бизнес-модели;
- 3) анализ бизнес-модели.

Планирование интервью включает: формирование списка интервьюируемых с указанием даты и времени проведения; распределение интервьюируемых по видам деятельности и бизнес-процессам (функциональным направлениям); подготовку инструкции для участников (цели и задачи, кто, когда, где, какие вопросы и др.); корректировку при необходимости плана создания архитектуры предприятия. Подготовка интервью включает разработку форм для управления процессом интервьюирования и фиксации результатов (прежде всего, для определения функций и информационных источников).

Главная цель интервьюирования – выявление необходимых данных по бизнес-модели. На следующих шагах осуществляется обработка результатов интервью, построение детальной модели, ее анализ, формирование пакета отчетов и проведение презентации.

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) – системной энциклопедии, являющейся высокоуровневым объектом (а не детальным словарем данных).

Построение каталога информационных ресурсов включает 4 шага.

Шаг 1 – определение видов данных для IRC и проектирование форм для сбора данных. Основные задачи шага включают:

- определение видов данных по приложениям;
- определение видов данных по входам, выходам, файлам и БД приложений;
- идентификацию технологических платформ и определение их декомпозиции по видам;
- проектирование форм для сбора данных;
- подготовку детальных инструкций по заполнению форм.

Шаг 2 – сбор данных для IRC и их ввод (заполнение форм), а также сопоставление приложений и функций. Основные задачи шага состоят в следующем:

- сбор системной документации;
- сопоставление приложений и бизнес-функций и формирование соответствующей матрицы;
- сопоставление приложений и технологических платформ и формирование соответствующей матрицы;
- ввод информации в инструментарий.

Шаг 3 – интегрирование и верификация информации по текущим приложениям и технологическим платформам, разработка потоковых диаграмм по каждой системе.

Шаг 4 – подготовка к процессам администрирования и сопровождения IRC, назначение которых состоит в поддержании каталога в актуальном состоянии. Здесь разрабатывается регламент поддержки, политики и процедуры сопровождения IRC, назначается ответственный по его сопровождению.

На этапе **«Формирование архитектуры данных»** определяются и идентифицируются основные разновидности данных, поддерживающих бизнес-функции. Архитектура данных представляется с помощью ER-модели. Этап содержит 4 шага:

- 1) формирование списка кандидатов в сущности;

- 2) определение сущностей, атрибутов и отношений;
- 3) сопоставление сущностей и бизнес-функций;
- 4) анализ результатов.

Целью *первого шага* является идентификация всех потенциальных сущностей, необходимых для поддержки бизнеса.

Здесь осуществляется распределение компонентов бизнес-модели по членам команды в разрезе видов деятельности и бизнес-процессов), подготовка каждым из участников списка кандидатов, формирование общего списка кандидатов в сущности.

Целью *второго шага* является создание стандартного определения и описания каждой сущности, обеспечение графической иллюстрации их взаимодействия. Здесь сущности определяются и документируются, осуществляется построение ER-модели, производится сопоставление файлов и БД из IRC с сущностями.

Целью *третьего шага* является сопоставление сущностей с бизнес-функциями и приложениями, результатами которого являются матрица сущности-функции и матрица сущности-приложения. При этом для каждой функции нижнего уровня детализации идентифицируется вид каждой из затрагиваемых ей сущностей (создается, изменяется, используется), а приложения сопоставляются с сущностями по входам, выходам, файлам и БД.

Целью *четвертого шага* является подготовка, распространение и анализ отчета по архитектуре данных.

На этапе **«Формирование архитектуры приложений»** определяются основные виды приложений, необходимых для управления данными и поддержки бизнес-функций. 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.

Основными шагами этапа являются:

- 1) формирование списка потенциальных приложений;

- 2) определение приложений;
- 3) сопоставление приложений и функций;
- 4) анализ применимости существующих приложений;
- 5) анализ результатов.

Цель *первого шага* – идентификация каждого из возможных приложений и формирование их списка, при этом особое внимание уделяется приложениям, которые могут улучшить бизнес или обеспечить конкурентные преимущества.

Цель *второго шага* – снабжение каждого приложения стандартным описанием (определением) и построение графической схемы архитектуры приложений. Основные задачи шага:

- 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 (полнота и состоятельность, прочность-устойчивость).

Целью *третьего шага* является идентификация бизнес-функций, поддерживаемых или выполняемых приложениями. Здесь для каждого приложения формируется матрица приложения-функции, а также перечень функций, не поддерживаемых ни одним приложением (с объяснением причин), а также матрица приложения, содержащего организационные единицы.

Цель *четвертого шага* состоит в определении соответствия архитектуры приложений существующим в организации приложениям. Кроме того, строится таблица соответствий архитектуры приложений и существующих приложений.

На *пятом шаге* производится подготовка, распространение и анализ отчета по архитектуре приложений.

Этап «Формирование технологической архитектуры» определяет основные виды технологий, необходимых для обеспечения окружения приложений, управляющих данными. Технологическая архитектура не является ни проектом сетевого оборудования и ПО, ни набором детальных требований к ним. Она только определяет виды технических платформ, поддерживающих бизнес. Основными шагами этапа являются:

- 1) идентификация технических принципов и платформ;
- 2) определение платформ и их распределение;
- 3) сопоставление платформ с приложениями и бизнес-функциями;
- 4) анализ результатов.

Целью *первого шага* является формулирование общих принципов для технических платформ и идентификация потенциальных кандидатов в платформы.

Цель *второго шага* – определение стратегии распределения приложений и данных, технологических платформ на основании сформулированных принципов.

Основными *результатами* шага являются распределение данных и приложений, конфигурация технологических платформ, оценка концептуальной архитектуры. Основными задачами шага являются:

- определение мест размещения бизнес-функций;
- распределение данных и приложений;
- определение конфигурации технологических платформ (рабочих станций, сети, архитектуры бизнес-систем);
- оценка концептуальной технологической архитектуры.

Цель *третьего шага* – обоснование технологических платформ путем их соотнесения с используемыми бизнес-функциями, формирование таблиц платформ приложений бизнес-функций.

На *четвертом шаге* производится подготовка, распространение и анализ отчета по технологической архитектуре.

Этап «Разработка плана реализации» включает:

- 1) формирование последовательности реализации приложений;
- 2) оценку трудозатрат и ресурсов, построение плана;
- 3) оценку стоимости и достоинств плана;
- 4) определение факторов успеха и рекомендаций по их достижению.

Целью *первого шага* является установка приоритетов и формирование последовательности реализации приложений (например, приложения, порождающие данные, должны быть реализованы перед реализацией приложений, использующих эти данные).

Основные результаты шага: матрица приложений, содержащих сущности данных, список упорядоченных по приоритетам приложений, план модификации и/или замены существующих систем, группировка приложений в проекты, последовательность реализации технологии.

Основными *задачами первого шага* являются:

- 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;
- формирование критериев (количественных и качественных) последовательности реализации;
- формирование последовательности модификации существующих систем и приобретения технологий.

Остальные шаги этапа традиционны для задачи планирования.

На этапе **«Заключительное планирование»** осуществляется подготовка окончательного отчета по архитектуре предприятия, подготовка и проведение презентации.

Этап «Переход к реализации» включает следующие шаги:

- 1) планирование перехода (спецификацию целей перехода, формирование плана перехода, назначение ответственности за переход, определение руководителя-лидера);

- 2) адаптацию подхода (методологии, инструменты);
- 3) инвентаризацию компьютерных ресурсов (приобретение необходимого оборудования, обеспечение надежности хранилища);
- 4) чистку архитектуры (ревизию, добавление деталей и обновление);
- 5) изменение организационно-штатной структуры;
- 6) набор персонала;
- 7) проведение обучения;
- 8) введение стандартов на программирование;
- 9) введение процедурных стандартов;
- 10) разработку детальных планов по приложениям;
- 11) определение и утверждение даты завершения перехода.

Если «наложить» метод EAP Спивака на модель Захмана, то можно сказать, что метод EAP 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 [24].

Данный подход помог многим компаниям и государственным ведомствам в организации процесса моделирования, стратегического бизнес-планирования, реорганизации деловых процессов, проектирования различных систем, выработки стандартов на данные, управления проектами. В частности, этой методикой пользовались такие организации, как Federal Express, Министерство энергетики США, Штаб Военно-воздушных сил США.

Контрольные вопросы:

1. Дайте развернутую характеристику модели Захмана.
2. Каково назначение уровней архитектуры согласно модели Gartner.
3. Перечислите и дайте характеристику ключевым этапам реализации методики Meta Group.

4. Проанализируйте процесс разработки архитектуры согласно методике TOGAF.
5. Выделите особенности применения модели архитектуры «4+1».
6. Дайте характеристику стратегической модели архитектуры SAM.
7. Раскройте сущность архитектурной концепции методике Microsoft.
8. Проанализируйте методологические принципы использования стандарта COBIT 5 при построении архитектуры.
9. Перечислите ключевые этапы планирования архитектуры.
10. Раскройте сущность метода EAP (планирования архитектуры организации).

ПРАКТИЧЕСКИЕ РЕКОМЕНДАЦИИ ПОСТРОЕНИЯ БИЗНЕС-МОДЕЛЕЙ ПРЕДПРИЯТИЯ

1 Разработка бизнес-архитектуры предприятия на основании использования методологии структурного анализа IDEF0

Методология IDEF0 предписывает построение иерархической системы диаграмм – единичных описаний фрагментов системы. Сначала проводится описание системы в целом и ее взаимодействия с окружающим миром (*контекстная диаграмма*), после чего проводится функциональная декомпозиция – система разбивается на подсистемы и каждая подсистема описывается отдельно (*диаграммы декомпозиции*). Затем каждая подсистема разбивается на более мелкие и так далее до достижения нужной степени подробности однозначного моделирования деятельности системы.

В основе методологии лежат четыре основных понятия: функциональный блок, интерфейсная дуга, декомпозиция, глоссарий.

Функциональный блок (Activity Box) представляет собой некоторую конкретную функцию в рамках рассматриваемой системы. По требованиям стандарта название каждого функционального блока должно быть сформулировано в глагольном наклонении (например, «производить услуги»). На диаграмме функциональный блок изображается прямоугольником (рисунок 42). Каждая из четырех сторон функционального блока имеет свое определенное значение (роль), при этом:

- верхняя сторона имеет значение «Управление» (Control);
- левая сторона имеет значение «Вход» (Input);
- правая сторона имеет значение «Выход» (Output);
- нижняя сторона имеет значение «Механизм» (Mechanism).

IDEF0 требует, чтобы в диаграмме было не менее трех и не более шести блоков. Эти ограничения поддерживают сложность диаграмм и модели на уровне, доступном для чтения, понимания и использования.

Блоки в IDEF0 размещаются по степени важности, как ее понимает автор диаграммы. Этот относительный порядок называется *доминированием*. Доминирование понимается как влияние, которое один блок оказывает на другие блоки диаграммы. Наиболее доминирующий блок обычно размещается в верхнем левом углу диаграммы, а наименее доминирующий – в правом углу.



Рис. 42 – Функциональный блок

Взаимодействие работ с внешним миром и между собой описывается в виде стрелок, изображаемых одинарными линиями со стрелками на концах. Стрелки представляют собой некую информацию и именуются существительными.

Интерфейсная дуга (Arrow) отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию, представленную данным функциональным блоком. Интерфейсные дуги часто называют потоками или стрелками.

С помощью интерфейсных дуг отображают различные объекты, в той или иной степени определяющие процессы, происходящие в системе. Такими объектами могут быть элементы реального мира (детали, вагоны, сотрудники и т. д.) или потоки данных и информации (документы, данные, инструкции и т. д.).

Стрелки делятся на *пять видов*:

- *входа* (входят в левую грань работы) – изображают данные или объекты, изменяемые в ходе выполнения работы;
- *управления* (входят в верхнюю грань работы) – изображают правила и ограничения, согласно которым выполняется работа;
- *выхода* (выходят из правой грани работы) – изображают данные или объекты, появляющиеся в результате выполнения работы;
- *механизма* (входят в нижнюю грань работы) – изображают ресурсы, необходимые для выполнения работы, но не изменяющиеся в процессе работы (например, оборудование, человеческие ресурсы и т. д.);
- *вызова* (выходят из нижней грани работы) – изображают связи между разными диаграммами или моделями, указывая на некоторую диаграмму, где данная работа рассмотрена более подробно.

Входные дуги изображают объекты, используемые и преобразуемые функциями. *Управленческие* дуги представляют информацию, управляющую действиями функций. *Выходные* дуги изображают объекты, в которые преобразуются входы. Дуги *механизмов* IDEF0 изображают физические аспекты функции (склады, людей, организации, приборы). Таким образом, стороны блока графически сортируют объекты, изображаемые касающимися блока дугами.

Любой функциональный блок по требованиям стандарта должен иметь, по крайней мере, одну управляющую интерфейсную дугу и одну исходящую, иначе его рассмотрение не имеет никакого смысла.

Декомпозиция (Decomposition) является основным понятием стандарта IDEF0. Принцип декомпозиции применяется при разбиении сложного процесса на составляющие его функции. При этом уровень детализации процесса определяется разработчиком модели.

Декомпозиция позволяет постепенно и структурировано представлять модель системы в виде иерархической структуры отдельных диаграмм, что делает ее менее перегруженной и легко усваиваемой.

Глоссарий. Последним из понятий IDEF0 является глоссарий (Glossary). Для каждого из элементов IDEF0 – диаграмм, функциональных блоков, интерфейсных дуг – существующий стандарт подразумевает создание и поддержание набора соответствующих определений, ключевых слов, повествовательных изложений и т. д., которые характеризуют объект, отображенный данным элементом. Этот набор называется *глоссарием* и является описанием сущности данного элемента. Глоссарий гармонично дополняет наглядный графический язык, снабжая диаграммы необходимой дополнительной информацией.

Модель IDEF0 всегда начинается с представления системы как единого целого – одного функционального блока с интерфейсными дугами, простирающимися за пределы рассматриваемой области. Такая диаграмма с одним функциональным блоком называется *контекстной диаграммой*.

В пояснительном тексте к контекстной диаграмме должна быть указана *цель (Purpose)* построения диаграммы в виде краткого описания и зафиксирована *точка зрения (Viewpoint)*.

Цель. Определение и формализация цели разработки IDEF0-модели является крайне важным моментом. Фактически цель определяет соответствующие области в исследуемой системе, на которых необходимо фокусироваться в первую очередь.

Точка зрения определяет основное направление развития модели и уровень необходимой детализации. Четкое фиксирование точки зрения позволяет разгрузить модель, отказавшись от детализации и исследования отдельных элементов, не являющихся необходимыми, исходя из выбранной точки зрения на систему. Правильный выбор точки зрения существенно сокращает временные затраты на построение конечной модели.

Выделение подпроцессов. В процессе декомпозиции функциональный блок, который в контекстной диаграмме отображает систему как единое целое, подвергается детализации на другой диаграмме (рис. 43).

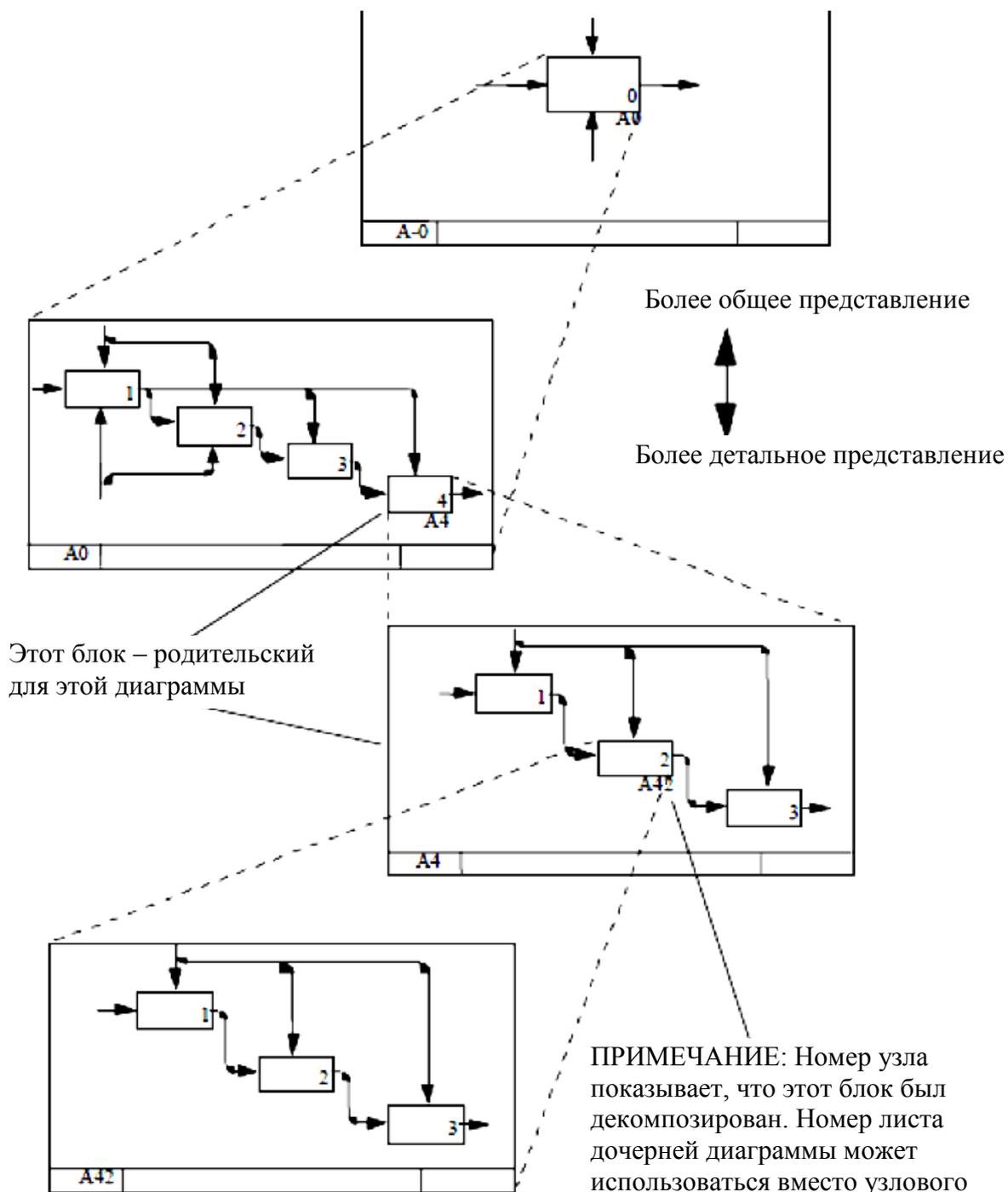


Рис.43 – Декомпозиция диаграмм в IDEF0

Получившаяся диаграмма второго уровня содержит функциональные блоки, отображающие главные подфункции функционального блока контекстной диаграммы, и называется *дочерней (Child Diagram)* по отношению к нему (каждый из функциональных блоков, принадлежащих дочерней диаграмме, соответственно называется *дочерним блоком – Child Box*). В свою

очередь, функциональный блок – предок называется родительским блоком по отношению к дочерней диаграмме (Parent Box), а диаграмма, к которой он принадлежит – родительской диаграммой (Parent Diagram). Каждая из подфункций дочерней диаграммы может быть далее детализирована путем аналогичной декомпозиции соответствующего ей функционального блока. В каждом случае декомпозиции функционального блока все интерфейсные дуги, входящие в данный блок или исходящие из него, фиксируются на дочерней диаграмме. Этим достигается структурная целостность IDEF0-модели.

Туннелирование. Иногда отдельные интерфейсные дуги высшего уровня не имеет смысла продолжать рассматривать на диаграммах нижнего уровня, или наоборот – отдельные дуги нижнего отражать на диаграммах более высоких уровней – это будет только перегружать диаграммы и делать их сложными для восприятия. Для решения подобных задач в стандарте IDEF0 предусмотрено понятие туннелирования. Обозначение «*туннеля*» (Arrow Tunnel) в виде двух круглых скобок вокруг начала интерфейсной дуги обозначает, что эта дуга не была унаследована от функционального родительского блока и появилась (из «туннеля») только на этой диаграмме. В свою очередь, такое же обозначение вокруг конца (стрелки) интерфейсной дуги в непосредственной близости от блока-приемника означает тот факт, что в дочерней по отношению к этому блоку диаграмме эта дуга отображаться и рассматриваться не будет. Чаще всего бывает, что отдельные объекты и соответствующие им интерфейсные дуги не рассматриваются на некоторых промежуточных уровнях иерархии, – в таком случае они сначала «погружаются в туннель», а затем при необходимости «возвращаются из туннеля».

Обычно IDEF0-модели несут в себе сложную и концентрированную информацию, и для того, чтобы ограничить их перегруженность и сделать удобочитаемыми, в стандарте приняты соответствующие **ограничения сложности**. Рекомендуется представлять на диаграмме от **трех до шести функциональных блоков**, при этом количество подходящих к одному

функциональному блоку (выходящих из одного функционального блока) *интерфейсных дуг предполагается не более четырех.*

Моделирование деловых процессов, как правило, выполняется с помощью *case-средств*. К таким средствам относятся **Allfusion Process Modeler (BPWin)**, Silverrun, Oracle Designer, Rational Rose и др. Функциональные возможности инструментальных средств структурного моделирования деловых процессов будут рассмотрены на примере case-средства BPWin.

При запуске BPWin по умолчанию появляется основная панель инструментов, палитра инструментов (вид которой зависит от выбранной нотации) и, в левой части, навигатор модели – Model Explorer (рис. 44).

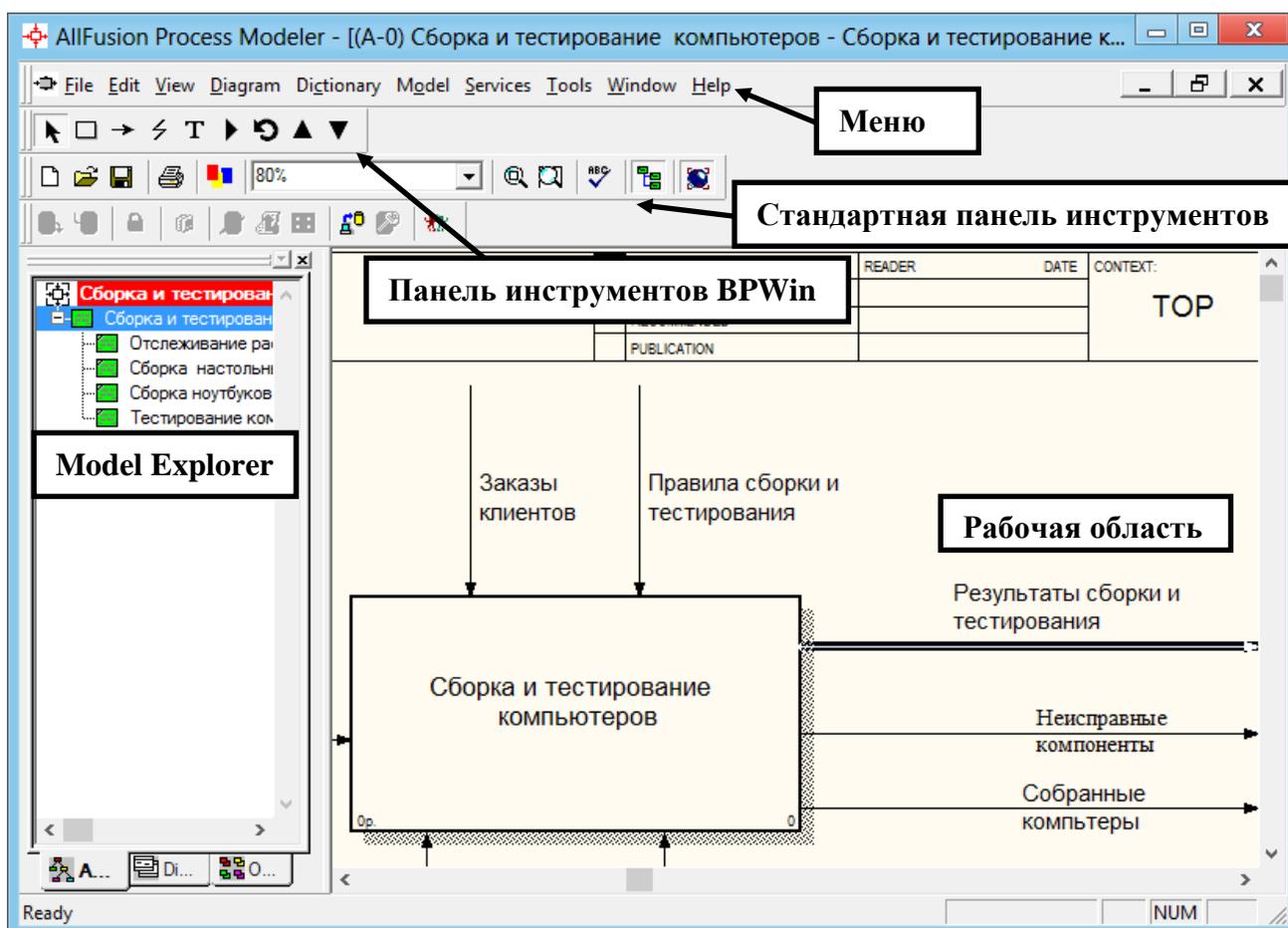


Рис. 44 – Интегрированная среда разработки модели BPWin

При создании новой модели возникает диалог, в котором следует указать, будет ли создана модель заново, или она будет открыта из файла либо из репозитория ModelMart, затем внести имя модели и выбрать методологию, в которой будет построена модель (рис. 45).

Кроме того, VPwin позволяет установить шрифт по умолчанию для объектов определенного типа на диаграммах и в отчетах.

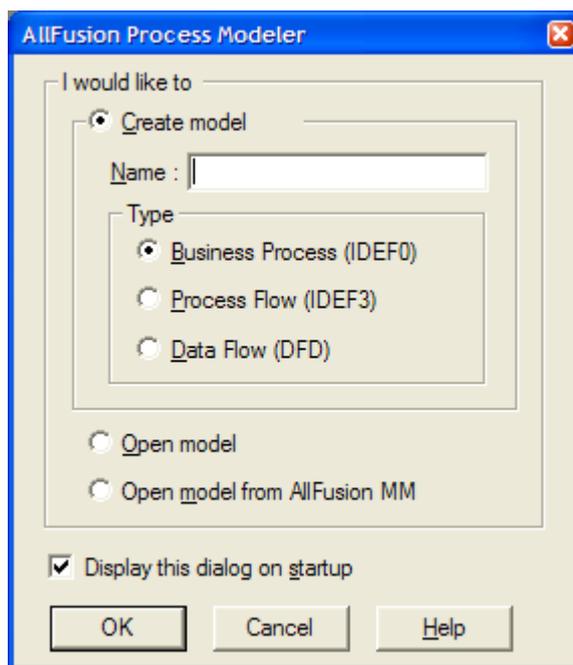


Рис. 45 – Диалоговое окно создания модели

Для этого следует выбрать меню Model/Default Fonts, после чего появляется следующее каскадное меню, каждый пункт которого служит для установки шрифтов для определенного типа объектов:

- *Context Activity* – работа на контекстной диаграмме;
- *Context Arrow* – стрелки на контекстной диаграмме;
- *Decomposition Activity* – работы на диаграмме декомпозиции;
- *Decomposition Arrow* – стрелки на диаграмме декомпозиции;
- *Node Tree Text* – текст на диаграмме дерева узлов;
- *Frame User Text* – текст, вносимый пользователем в каркасе диаграмм;
- *Frame System Text* – системный текст в каркасе диаграмм;

- *Text Blocks* – текстовые блоки;
- *Parent Diagram Text* – текст родительской диаграммы;
- *Parent Diagram Title Text* – текст заголовка родительской диаграммы;
- *Report Text* – текст отчетов.

В BPWin возможно построение смешанных моделей, т. е. модель может содержать одновременно диаграммы как IDEF0, так и IDEF3 и DFD. Состав палитры инструментов изменяется автоматически, когда происходит переключение с одной нотации на другую. Функциональность панели инструментов доступна из основного меню BPWin (табл. 6).

Таблица 6 – Описание элементов управления основной панели инструментов BPWin 4.0

Элемент управления	Описание	Соответствующий пункт меню
1	2	3
	Создать новую модель	File/New
	Открыть модель	File/Open
	Сохранить модель	File/Save
	Напечатать модель	File/Print
	Вызвать генератор отчетов Report Builder	Tools/Report Builder
	Выбор масштаба	View/Zoom
	Масштабирование	View/Zoom
	Проверка правописания	Tools/Spelling
	Включение и выключение навигатора модели Model Explorer	View/Model Explorer
	Включение и выключение дополнительной панели инструментов работы с ModelMart	ModelMart

Модель в BPWin рассматривается как совокупность работ, каждая из которых оперирует с некоторым набором данных. Работа изображается в виде прямоугольников, данные – в виде стрелок. Если щелкнуть по любому объекту

модели левой кнопкой мыши, появляется контекстное меню, каждый пункт которого соответствует редактору какого-либо свойства объекта.

В качестве рабочего примера рассматривается деятельность вымышленной компании «Computer Word». Компания занимается в основном сборкой и продажей настольных компьютеров и ноутбуков. Компания не производит компоненты самостоятельно, а только собирает и тестирует компьютеры.

Основные виды работ в компании таковы:

- 1) продавцы принимают заказы клиентов;
- 2) операторы группируют заказы по типам компьютеров;
- 3) операторы собирают и тестируют компьютеры;
- 4) операторы упаковывают компьютеры согласно заказам;
- 5) кладовщик отгружает клиентам заказы.

Компания использует лицензионную бухгалтерскую информационную систему, которая позволяет оформить заказ, счет и отследить платежи по счетам.

Моделирования бизнес-архитектуры в среде Allfusion Process Modeler (BPWin). Создание функциональной модели

Запустите BPWin. (Кнопка *Start*  /BPWin  .

Если появляется диалог *ModelMart Connection Manager*, нажмите на кнопку *Cancel* (Отмена).

Щелкните по кнопке . Появляется диалоговое окно *I would like to* (рис. 46). Внесите в текстовое поле *Name* имя модели «Деятельность компании» и выберите *Type – Business Process (IDEF0)*. Нажмите кнопку ОК.

Откроется диалоговое окно *Properties for New Models* (Свойства новой модели).

Введите в текстовое поле *Author* (Автор) имя автора модели и в текстовое поле *Author initials* его инициалы. Нажмите последовательно кнопки *Apply* и *OK*. Автоматически создается незаполненная контекстная диаграмма (рис. 47).

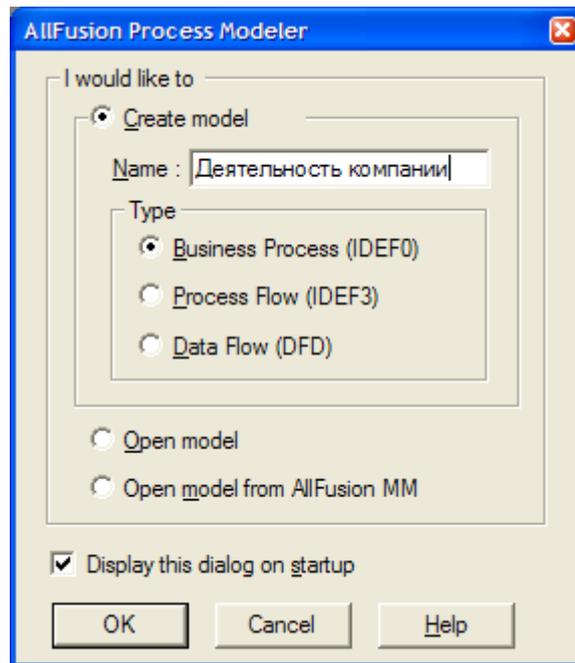


Рис. 46 – Диалоговое окно I would like to

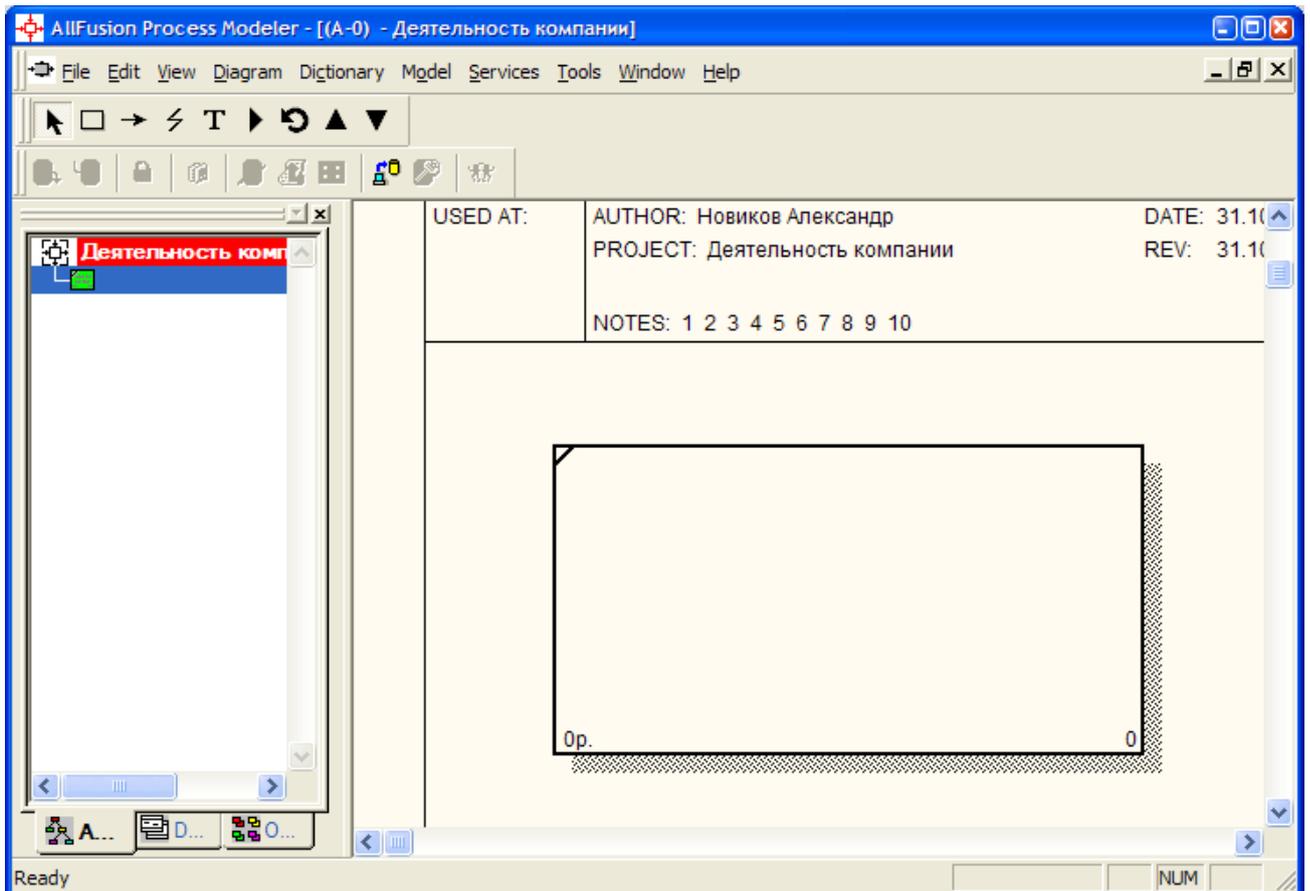


Рис. 47 – Незаполненная контекстная диаграмма

Обратите внимание на кнопку  на панели инструментов. Эта кнопка включает и выключает инструмент просмотра и навигации – *Model Explorer* (Браузер модели). *Model Explorer* имеет три вкладки – *Activities* (, *Diagrams* () и *Objects* (). Во вкладке *Activities* щелчок правой кнопкой по объекту в браузере модели позволяет выбрать опции редактирования его свойств.

Если непонятно, как выполнить то или иное действие, можно вызвать контекстную помощь – клавиша F1 или воспользоваться меню Help.

Перейдите в меню *Model/Model Properties*. Во вкладке *General* диалогового окна *Model Properties* в текстовое поле *Model name* следует внести имя модели «Деятельность компании», а в текстовое поле *Project* имя проекта «Модель деятельности компании», в текстовое поле *Time Frame* (Временной охват) – *As-Is* (Как есть) (рис. 48).

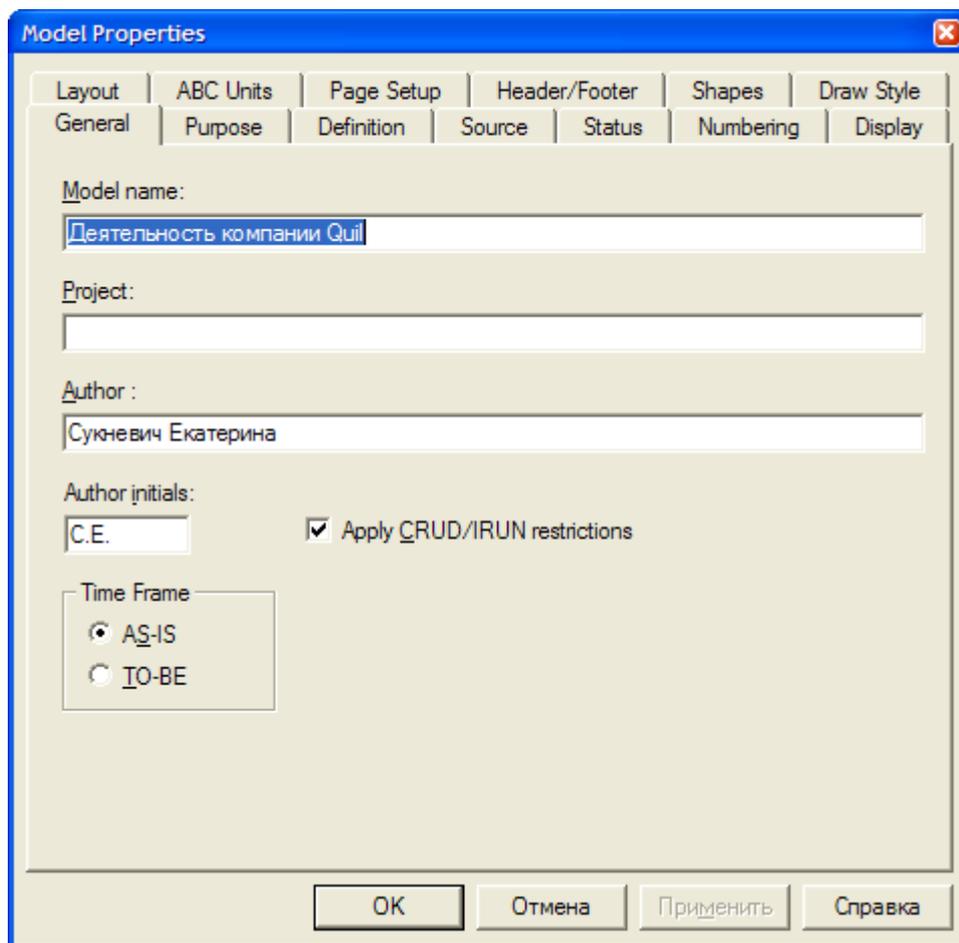


Рис. 48 – Вкладка *General* диалогового окна *Model Properties*

Во вкладке *Purpose* диалогового окна *Model Properties* в текстовое поле *Purpose* (цель) внесите данные о цели разработки модели – «Моделировать текущие (As-Is) бизнес-процессы компании», а в текстовое поле *Viewpoint* (точка зрения) – «Директор».

Во вкладке *Definition* диалогового окна *Model Properties* в текстовое поле *Definition* (Определение) внесите «Это учебная модель, описывающая деятельность компании» и в текстовое поле *Scope* (Охват) – «Общее управление бизнесом компании: исследование рынка, закупка компонентов, сборка, тестирование, и продажа продуктов», во вкладке *Source* внесите «Материалы учебного курса». Во вкладке *Status* установите *WORKING*, нажмите ОК (рис. 49).

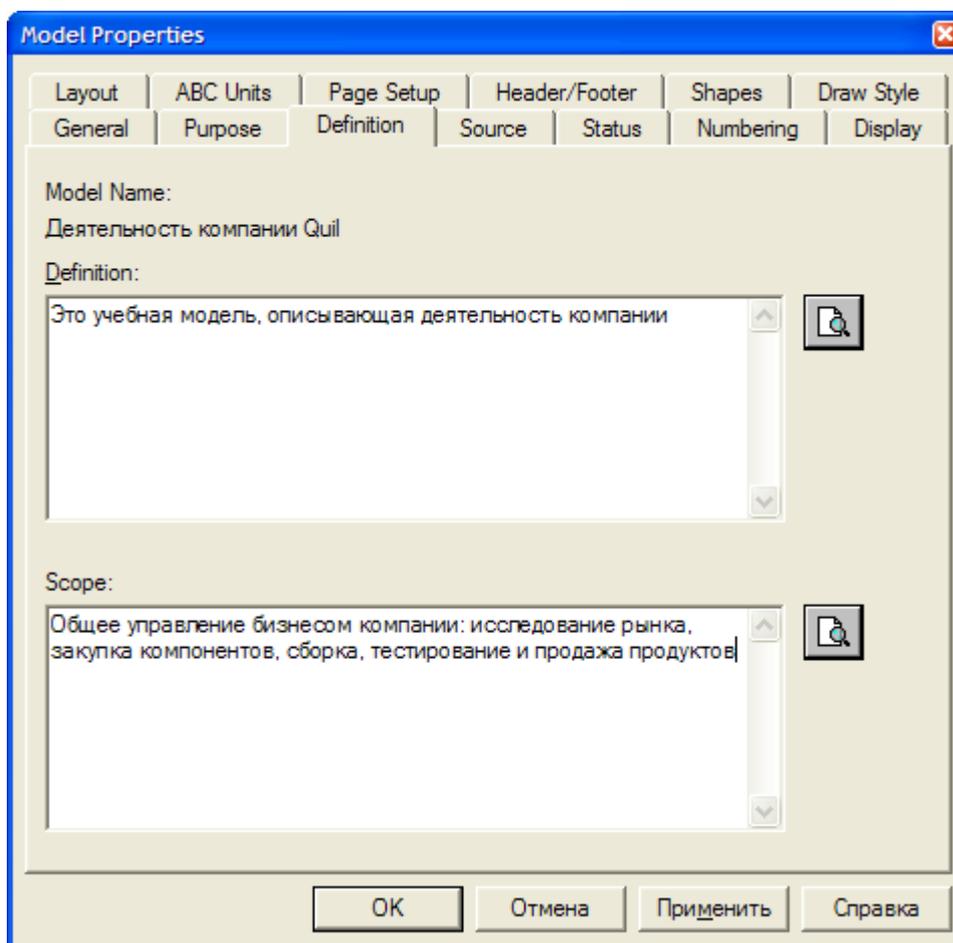


Рис. 49 – Вкладка *Definition* диалогового окна *Model Properties*

Перейдите на контекстную диаграмму и правой кнопкой мыши щелкните по прямоугольнику представляющему, в нотации IDEF0, условное графическое обозначение работы. В контекстном меню выберите опцию *Name*. Во вкладке *Name* внесите имя «Деятельность компании».

Во вкладке *Definition* диалогового окна *Activity Properties* в текстовое поле *Definition* (Определение) внесите «Текущие бизнес-процессы компании» (рис. 50). Текстовое поле *Note* (Примечания) оставьте незаполненным.

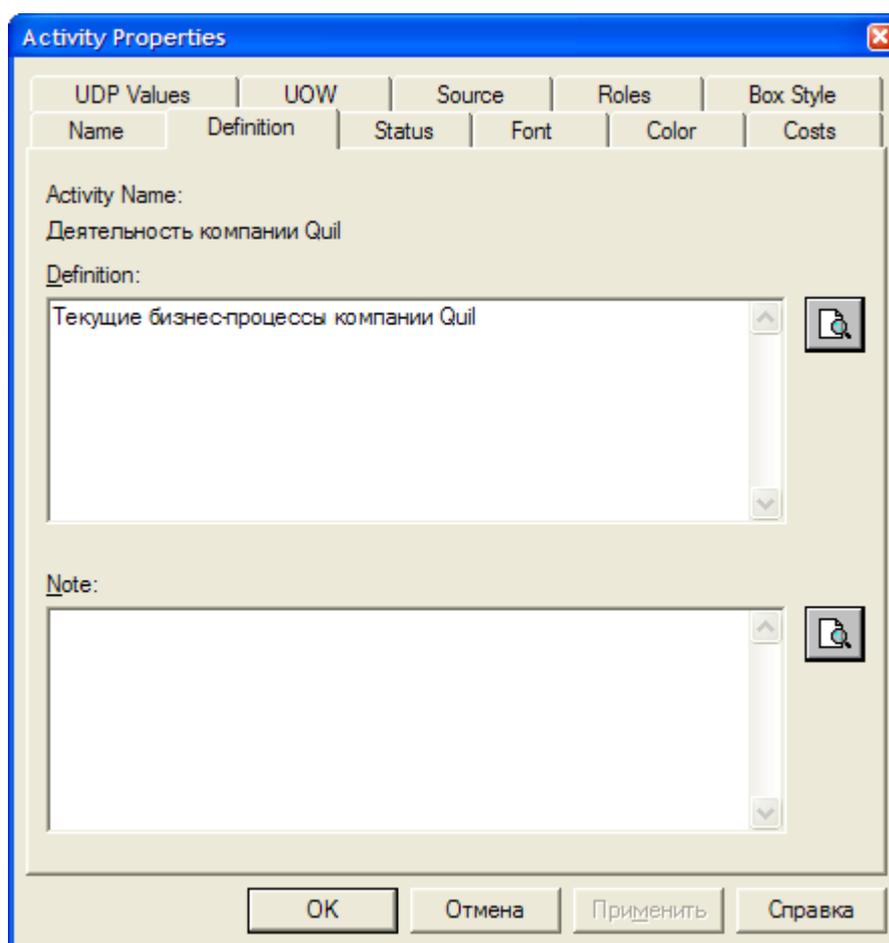


Рис. 50 – Вкладка *Definition* диалогового окна *Activity Properties*

Создайте ICOM-стрелки на контекстной диаграмме (таблица 7). Задайте *Arrow Name* (Название стрелки), *Arrow Definition* (Определение стрелки) и *Arrow Type* (Тип стрелки).

Таблица 7 – Стрелки контекстной диаграммы

Название стрелки (Arrow Name)	Определение стрелки (Arrow Definition)	Тип стрелки (Arrow Type)
Звонки клиентов	Запросы информации, заказы, техподдержка и т.д.	Input
Правила и процедуры	Правила продаж, инструкции по сборке, процедуры тестирования, критерии производительности и т.д.	Control
Проданные продукты	Настольные и портативные компьютеры	Output
Бухгалтерская система	Оформление счетов, оплата счетов, работа с заказами	Mechanism

С помощью кнопки **T** внесите текст в поле диаграммы – точку зрения и цель. Результат построения контекстной диаграммы представлен на рисунке 51.

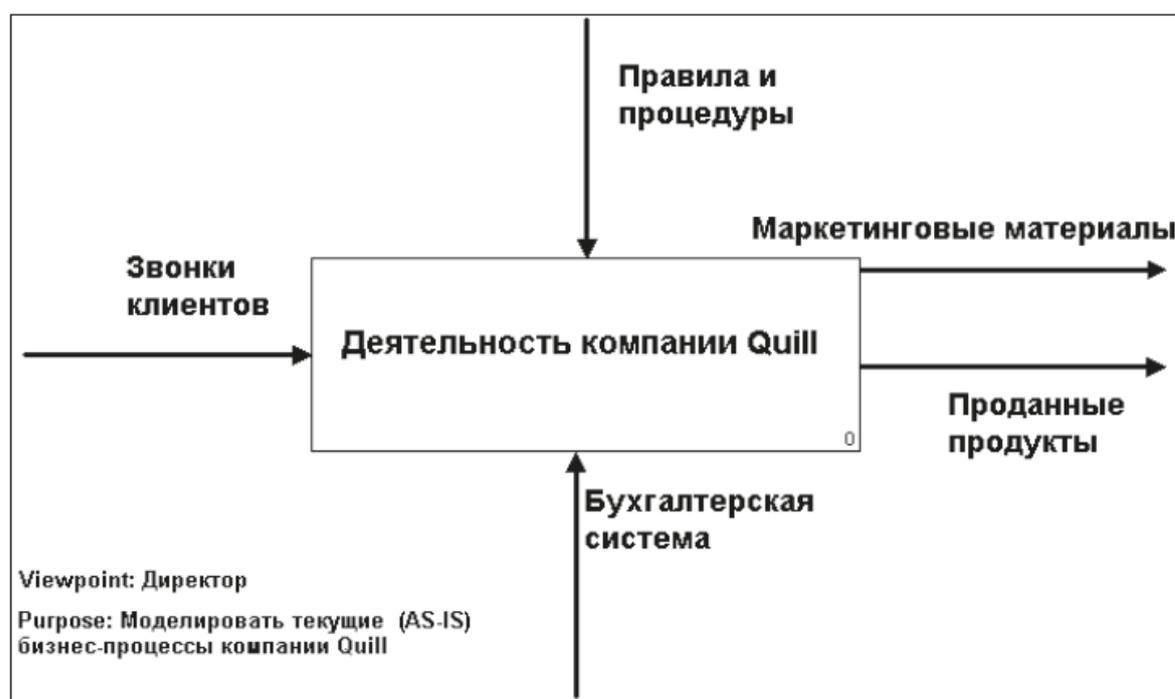


Рис. 51 – Контекстная диаграмма

Создание диаграммы декомпозиции

Выберите кнопку **▼** перехода на нижний уровень в палитре инструментов и в диалоговом окне *Activity Box Count* (рис. 52) установите число работ на диаграмме нижнего уровня – 3 – и нажмите кнопку ОК.

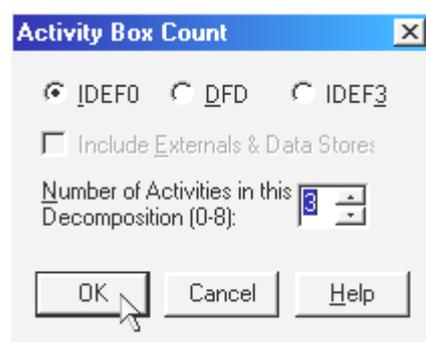


Рис. 52 – Диалоговое окно Activity Box Count

Автоматически будет создана диаграмма декомпозиции. Правой кнопкой мыши щелкните по работе, расположенной в левом верхнем углу области редактирования модели, выберите в контекстном меню опцию *Name* и внесите имя работы. Повторите операцию для оставшихся двух работ. Затем внесите определение, статус и источник для каждой работы согласно данным табл. 8.

Таблица 8 – Описание работ диаграммы декомпозиции

Название работы (Activity Name)	Определение работы (Activity Definition)
Продажи и маркетинг	Телемаркетинг и презентации, выставки
Сборка и тестирование компьютеров	Сборка и тестирование настольных и портативных компьютеров
Отгрузка и получение	Отгрузка заказов клиентам и получение компонентов от поставщиков

Для изменения свойств работ после их внесения в диаграмму можно воспользоваться словарем работ.

Вызов словаря производится при помощи пункта главного меню *Model*. Если описать имя и свойства работы в словаре, ее можно будет внести в диаграмму позже с помощью кнопки  в палитре инструментов. Невозможно удалить работу из словаря, если она используется на какой-либо диаграмме. Если работа удаляется из диаграммы, из словаря она не удаляется. Имя и описание такой работы может быть использовано в дальнейшем. Для добавления работы в словарь необходимо перейти в конец списка и щелкнуть правой кнопкой по последней строке. Возникает новая строка, в которой нужно внести имя и свойства работы. /*Diagram Object Editor* (рис. 53).

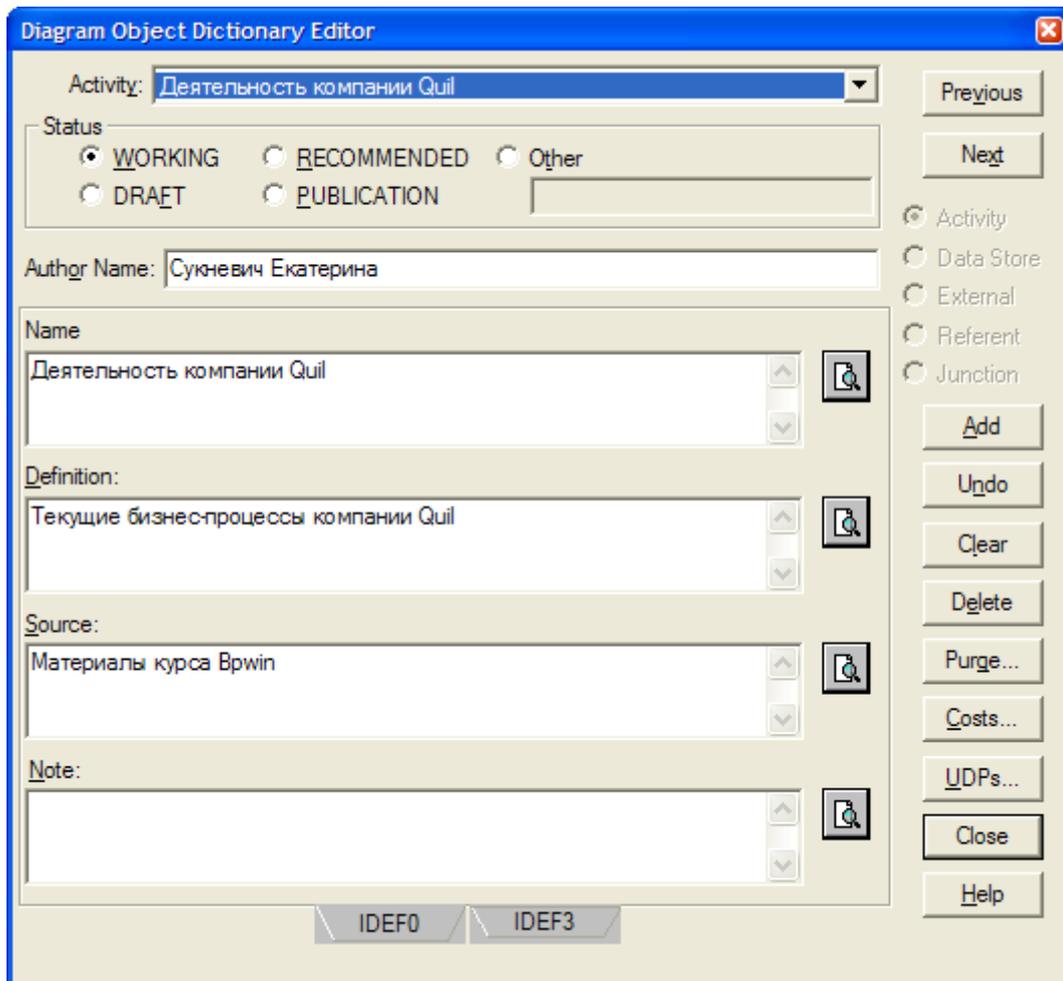


Рис. 53 – Диалоговое окно словаря

Для удаления всех имен работ, не используемых в модели, щелкните по кнопке  *Purge* (Очистить).

Перейдите в режим рисования стрелок и свяжите граничные стрелки, воспользовавшись кнопкой  на палитре инструментов так, как это показано на рисунке 54.

Правой кнопкой мыши щелкните по ветви стрелки управления работы «Сборка и тестирование компьютеров» и переименуйте ее в «Правила сборки и тестирования» (рис. 55).

Внесите определение для новой ветви: «Инструкции по сборке, процедуры тестирования, критерии производительности и т. д.» Правой кнопкой мыши щелкните по ветви стрелки механизма работы «Продажи и маркетинг» и переименуйте ее как «Система оформления заказов».

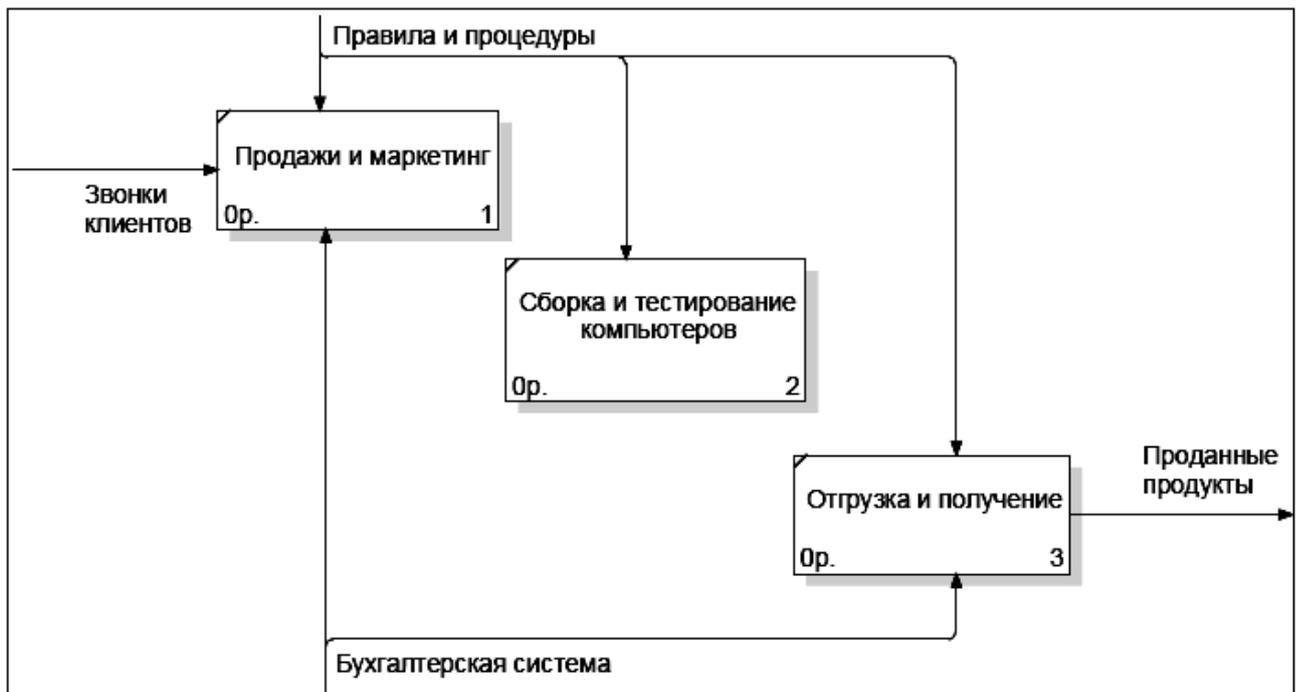


Рис. 54 – Связанные граничные стрелки на диаграмме А0

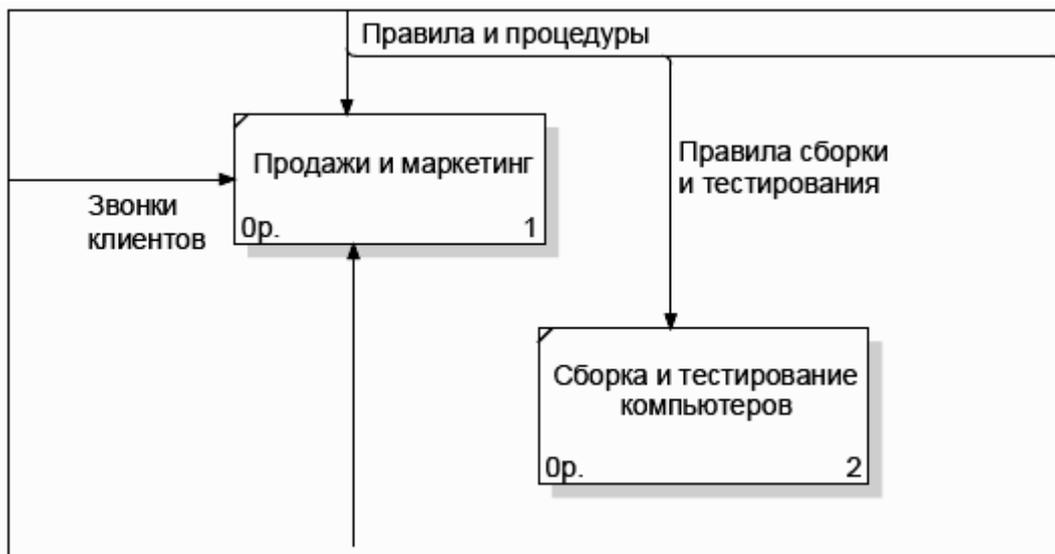


Рис. 55 – Стрелка «Правила сборки и тестирования»

Альтернативный метод внесения имен и свойств стрелок – использование словаря стрелок (вызов словаря – меню *Model/Arrow Editor...*).

Если описать имя и свойства стрелки в словаре, ее можно будет внести в диаграмму позже. Вы не можете удалить стрелку из словаря, если она используется в какой-либо диаграмме. Если Вы удалите стрелку из диаграммы,

из словаря она не удаляется. Имя и описание такой стрелки может быть использовано в дальнейшем.

Для добавления стрелки в словарь щелкните по кнопке *Clear*, внесите имя и свойства стрелки, затем щелкните по кнопке *Add*. Для удаления всех имен стрелок, не используемых в модели, щелкните по кнопке *Purge Unused*.

Создайте новые внутренние стрелки, как показано на рисунке 56.

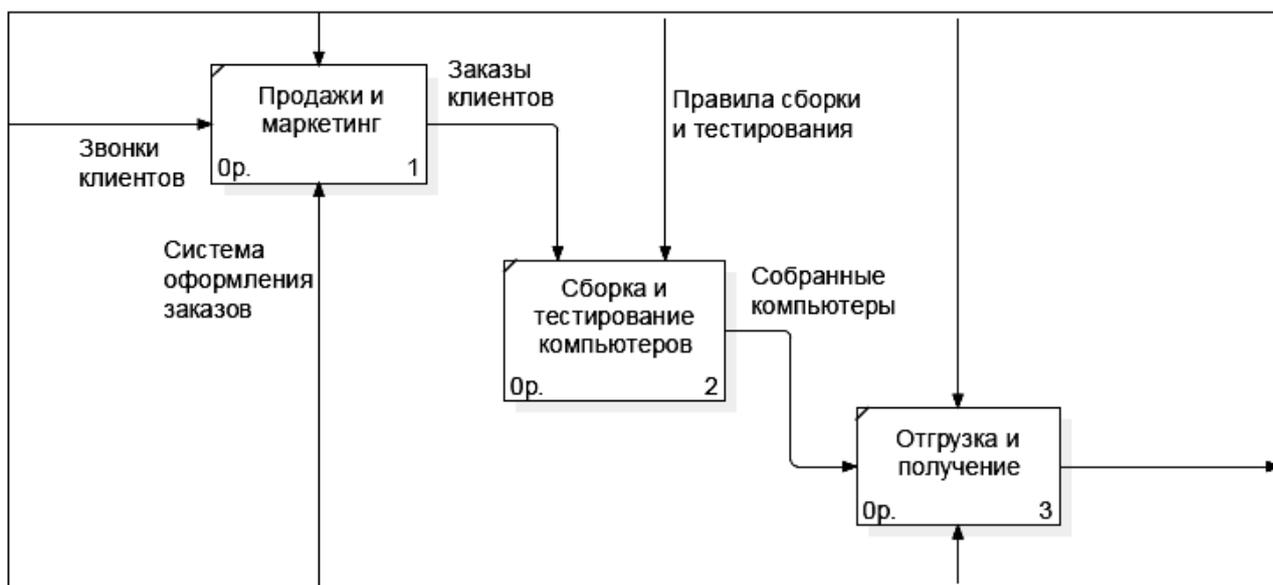


Рисунок 56 – Добавление стрелок

Создайте **стрелку обратной связи (по управлению)** «Результаты сборки и тестирования», идущую от работы «Сборка и тестирование компьютеров» к работе «Продажи и маркетинг». Измените, при необходимости, стиль стрелки (толщина линий) и установите опцию *Extra Arrowhead* (Дополнительный Наконечник стрелы) (из контекстного меню). Методом drag&drop перенесите имена стрелок так, чтобы их было удобнее читать. Если необходимо, установите из контекстного меню *Squiggle*. Результат изменений показан на рис. 57.

Создайте новую граничную стрелку выхода «Маркетинговые материалы», выходящую из работы «Продажи и маркетинг». Эта стрелка автоматически не попадает на диаграмму верхнего уровня и имеет квадратные скобки на наконечнике . Щелкните правой кнопкой мыши по квадратным скобкам и выберите пункт меню *Arrow Tunnel*.

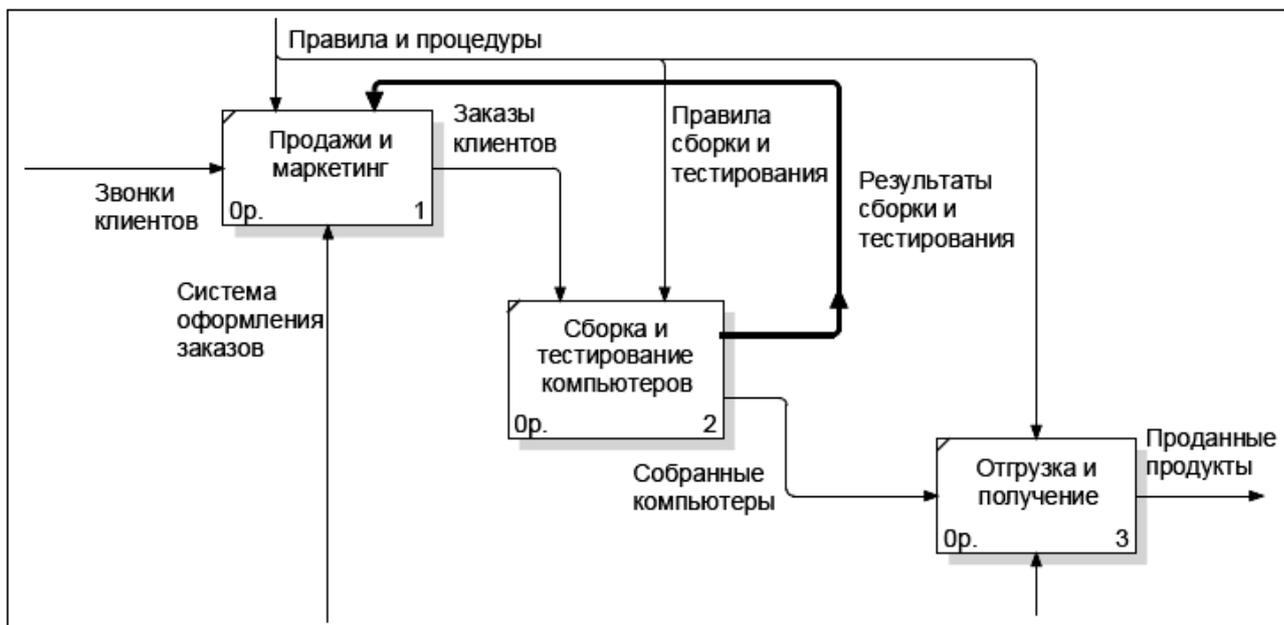


Рис. 57 – Добавление стрелки по управлению

В диалоговом окне *Border Arrow Editor* (Редактор Граничных Стрелок) выберите опцию *Resolve it to Border Arrow* (Разрешить как Граничную Стрелку).

Для стрелки «Маркетинговые материалы» выберите опцию *Trim* (Упорядочить) из контекстного меню. Результат показан на рисунке 58.

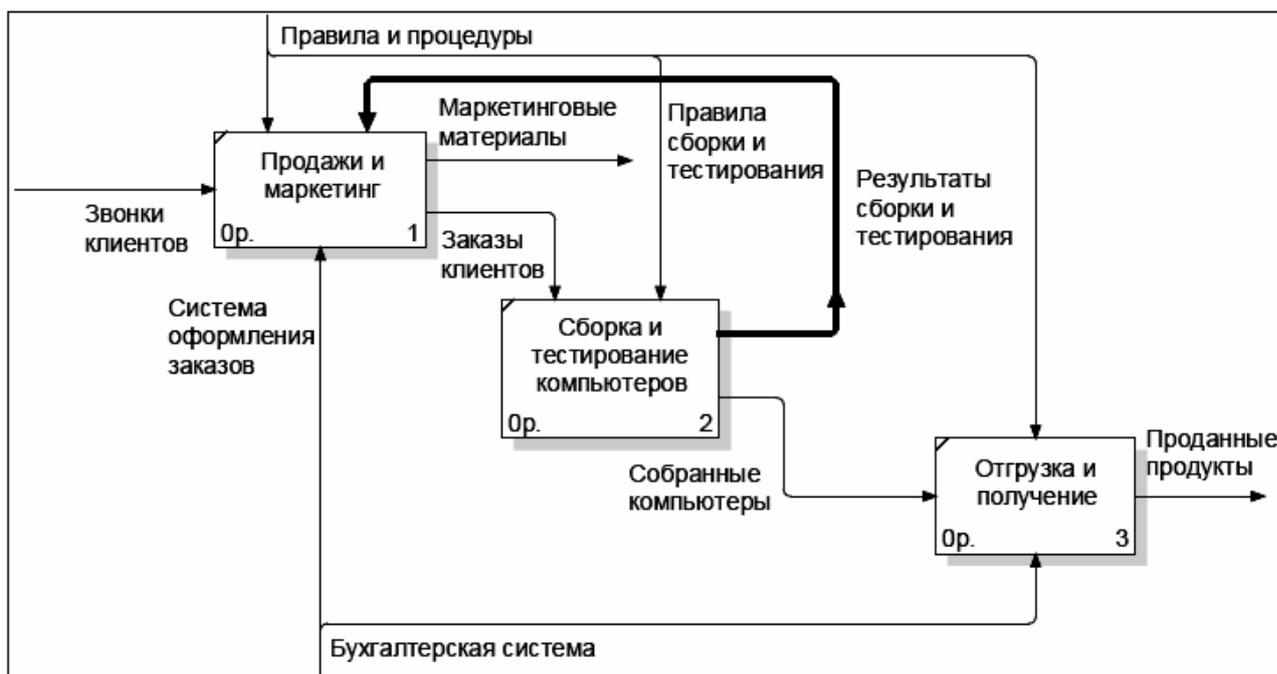


Рис. 58 – Результат декомпозиции – диаграмма A0

Создание диаграммы декомпозиции A2

Декомпозируем работу «Сборка и тестирование компьютеров». В результате проведения экспертизы получена следующая информация.

Производственный отдел получает заказы клиентов от отдела продаж по мере их поступления.

Диспетчер координирует работу сборщиков, сортирует заказы, группирует их и дает указание на отгрузку компьютеров, когда они готовы.

Каждые 2 часа диспетчер группирует заказы – отдельно для настольных компьютеров и ноутбуков, и направляет на участок сборки.

Сотрудники участка сборки собирают компьютеры согласно спецификациям заказа и инструкциям по сборке. Когда группа компьютеров, соответствующая группе заказов, собрана, она направляется на тестирование. Тестировщики тестируют каждый компьютер и в случае необходимости заменяют неисправные компоненты.

Тестировщики направляют результаты тестирования диспетчеру, который на основании этой информации принимает решение о передаче компьютеров, соответствующих группе заказов, на отгрузку.

На основе этой информации внесите новые работы и стрелки (таб. 9 и 10).

Таблица 9 – Определение работ для декомпозиции

Название работы (Activity Name)	Определение работы (Activity Definition)
Отслеживание расписания и управление сборкой и тестированием	Просмотр заказов, установка расписания выполнения заказов, просмотр результатов тестирования, формирование групп заказов на сборку и отгрузку
Сборка настольных компьютеров	Сборка настольных компьютеров в соответствии с инструкциями и указаниями диспетчера
Сборка ноутбуков	Сборка ноутбуков в соответствии с инструкциями и указаниями диспетчера
Тестирование компьютеров	Тестирование компьютеров и компонентов. Замена неработающих компонентов

Таблица 10 – Описание стрелок для декомпозиции

Наименование стрелки (Arrow Name)	Источник стрелки (Arrow Source)	Тип стрелки источника (Arrow Source Type)	Приемник стрелки (Arrow Dest.)	Тип стрелки приемника (Arrow Dest. Type)
Диспетчер	Персонал производственного отдела	Mechanism	Отслеживание расписания и управление сборкой и тестированием	Mechanism
Заказы клиентов	Граница диаграммы	Control	Отслеживание расписания и управление сборкой и тестированием	Control
Заказы на настольные компьютеры	Отслеживание расписания и управление сборкой и тестированием	Output	Сборка настольных компьютеров	Control
Заказы на ноутбуки	Отслеживание расписания и управление сборкой и тестированием	Output	Сборка ноутбуков	Control
Компоненты	«Tunnel»	Input	Сборка настольных компьютеров	Input
			Сборка ноутбуков	Input
			Тестирование компьютеров	Input
Настольные компьютеры	Сборка настольных компьютеров	Output	Тестирование компьютеров	Input
Персонал производственного отдела	«Tunnel»	Mechanism	Сборка настольных компьютеров	Mechanism
			Сборка ноутбуков	Mechanism
Правила сборки и тестирования	Граница диаграммы	Control	Сборка настольных компьютеров	Control
			Сборка ноутбуков	Control
			Тестирование комп-в	Control
Результаты сборки и тестирования	Сборка настольных компьютеров	Output	Граница диаграммы	Output
	Сборка ноутбуков	Output		
	Тестирование компьютеров	Output		
Результаты тестирования	Тестирование компьютеров	Output	Отслеживание расписания и управление сборкой и тестированием	Input
Собранные компьютеры	Тестирование компьютеров	Output	Граница диаграммы	Output
Тестировщик	Персонал производственного отдела	Output	Тестирование компьютеров	Mechanism
Указание передать компьютеры на отгрузку	Отслеживание расписания и управление сборкой и тестированием	Output	Тестирование компьютеров	Control

Туннелируйте и свяжите на верхнем уровне граничные стрелки, если это необходимо. Результат показан на рисунке 59.



Рис. 59 – Результат декомпозиции – диаграмма A2

Контрольные вопросы:

1. Что представляет собой модель в нотации IDEF0?
2. Что обозначают работы в IDEF0?
3. Назовите порядок наименования работ.
4. Какое количество работ должно присутствовать на одной диаграмме?
5. Что называется порядком доминирования?
6. Как располагаются работы по принципу доминирования?
7. Каково назначение сторон прямоугольников работ на диаграммах?
8. Перечислите типы стрелок.
9. Назовите виды взаимосвязей.
10. Что называется граничными стрелками?
11. Объясните принцип именования разветвляющихся и сливающихся стрелок.

2 Анализ и синтез моделей бизнес-процессов на основании использования методологии структурного анализа IDEF0

Диаграмма дерева узлов показывает иерархию работ в модели и позволяет рассмотреть всю модель целиком, но не показывает взаимосвязи между работами. Процесс создания модели работ является итерационным, следовательно, работы могут менять свое расположение в дереве узлов многократно. Чтобы не запутаться и проверить способ декомпозиции, следует после каждого изменения создавать диаграмму дерева узлов.

Диаграммы «только для экспозиции» (FEO) часто используются в модели для иллюстрации других точек зрения, для отображения отдельных деталей, которые не поддерживаются явно синтаксисом IDEF0. Диаграммы FEO позволяют нарушить любое синтаксическое правило, поскольку по сути являются просто картинками — копиями стандартных диаграмм и не включаются в анализ синтаксиса.

Слияние и расщепление моделей. Возможность слияния и расщепления моделей обеспечивает коллективную работу над проектом. Так, руководитель проекта может создать декомпозицию верхнего уровня и дать задание аналитикам продолжить декомпозицию каждой ветви дерева в виде отдельных моделей. После окончания работы над отдельными ветвями все подмодели могут быть слиты в единую модель. С другой стороны, отдельная ветвь модели может быть отщеплена для использования в качестве независимой модели, для доработки или архивирования.

BPWin использует для слияния и разветвления моделей **стрелки вызова**. Для слияния необходимо выполнить следующие **условия**:

- обе сливаемые модели должны быть открыты в BPWin;
- имя модели-источника, которое присоединяют к модели-цели, должно совпадать с именем стрелки вызова работы в модели-цели;
- стрелка вызова должна исходить из недекомпозируемой работы (работа должна иметь диагональную черту в левом верхнем углу);

– имена контекстной работы подсоединяемой модели-источника и работы на модели-цели, к которой мы подсоединяем модель-источник, должны совпадать;

– модель-источник должна иметь, по крайней мере, одну диаграмму декомпозиции.

При слиянии моделей объединяются и словари стрелок и работ. В случае одинаковых определений возможна перезапись определений или принятие определений из модели-источника. То же относится к именам стрелок, хранилищам данных и внешним ссылкам.

После подтверждения слияния (кнопка ОК) модель-источник подсоединяется к модели-цели, стрелка вызова исчезает, а работа, от которой отходила стрелка вызова, становится декомпозируемой – к ней подсоединяется диаграмма декомпозиции первого уровня модели-источника. Стрелки, касающиеся работы на диаграмме модели-цели, автоматически не мигрируют в декомпозицию, а отображаются как неразрешенные. Их следует туннелировать вручную. В процессе слияния модель-источник остается неизменной, и к модели-цели подключается фактически ее копия. Не нужно путать слияние моделей с синхронизацией. Если в дальнейшем модель-источник будет редактироваться, эти изменения автоматически не попадут в соответствующую ветвь модели-цели.

Разделение моделей производится аналогично. Для отщепления ветви от модели следует щелкнуть правой кнопкой мыши по декомпозированной работе (работа не должна иметь диагональной черты в левом верхнем углу) и выбрать во всплывающем меню пункт *Split Model*. В появившемся диалоге *Split Options* следует указать имя создаваемой модели. После подтверждения расщепления в старой модели работа станет недекомпозированной (признак – диагональная черта в левом верхнем углу), будет создана стрелка вызова, ее имя будет совпадать с именем новой модели, и, наконец, будет создана новая модель, причем имя контекстной работы будет совпадать с именем работы, от которой была «оторвана» декомпозиция.

Создание отчетов в BPWin. Отчеты по модели вызываются из пункта меню *Report*. Всего имеется *семь типов отчетов*:

– *Model Report* – включает информацию о контексте модели – имя модели, точку зрения, область, цель, имя автора, дату создания и др.;

– *Diagram Report* – отчет по конкретной диаграмме. Включает список объектов (работ, стрелок, хранилищ данных, внешних ссылок);

– *Diagram Object Report* – наиболее полный отчет по модели. Может включать полный список объектов модели (работ, стрелок с указанием их типа и др.) и свойства, определяемые пользователем;

– *Activity Cost Report* – отчет о результатах стоимостного анализа;

– *Arrow Report* – отчет по стрелкам. Может содержать информацию из словаря стрелок, информацию о работе-источнике, работе-назначении стрелки и информацию о разветвлении и слиянии стрелок;

– *Data Usage Report* – отчет о результатах связывания модели процессов и модели данных;

– *Model Consistency Report* – отчет, содержащий список синтаксических ошибок модели.

Реализация этапов моделирования в среде Allfusion Process Modeler (BPWin). Создание диаграммы узлов

Выберите меню *Diagram/Add Node Tree*. В первом диалоге гида *Node Tree Wizard* внесите имя диаграммы, укажите диаграмму корня дерева и количество уровней (рис. 60).

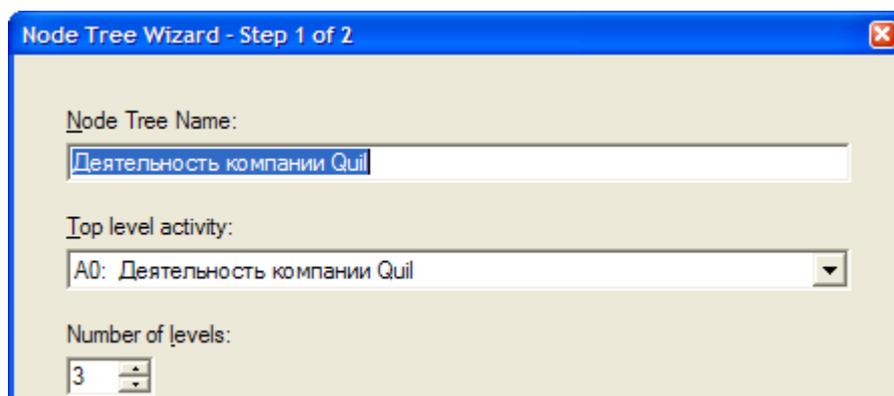


Рис. 60 – Первый диалог Node Tree Wizard

Во втором диалоге установите опции, как на рисунке 61.

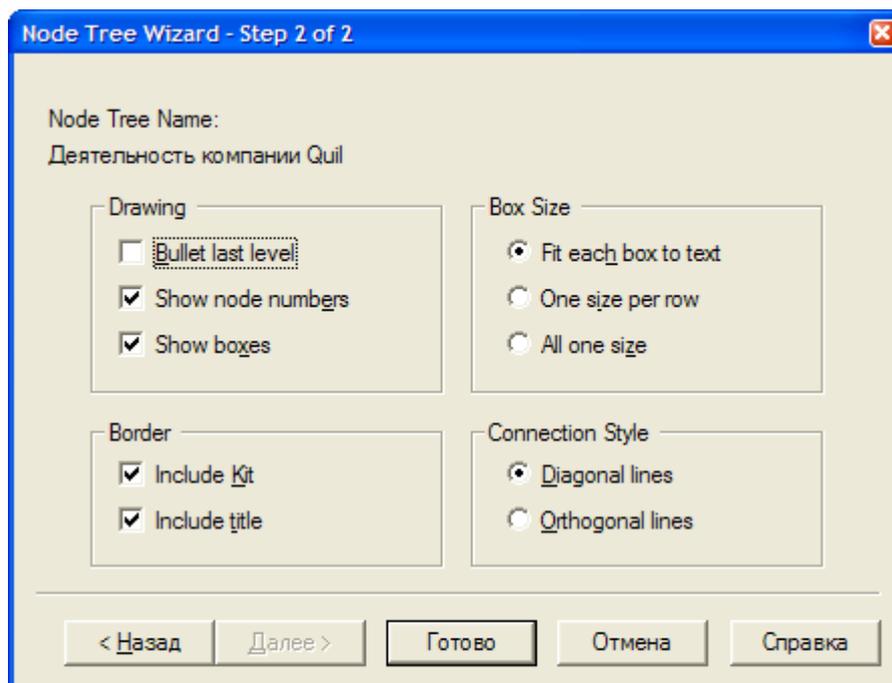


Рис. 61 – Второй диалог гида Node Tree Wizard

Щелкните по *Finish* (Готово). Создается диаграмма дерева узлов. Результат можно увидеть на рисунке 62.



Рис. 62 – Диаграмма дерева узлов

Диаграмму дерева узлов можно модифицировать. Нижний уровень может быть отображен не в виде списка, а в виде прямоугольников, так же как и верхние уровни.

Для модификации диаграммы правой кнопкой мыши щелкните по свободному месту, не занятому объектами, выберите меню *Node tree Diagram Properties* и во вкладке *Style* диалога *Node Tree Properties* отключите опцию *Bullet Last Level*. Щелкните по ОК. Результат показан на рисунке 63.

Создание FEO диаграммы. Предположим, что при обсуждении бизнес-процессов возникла необходимость детально рассмотреть взаимодействие работы «Сборка и тестирование компьютеров» с другими работами. Чтобы не портить диаграмму декомпозиции, создайте *FEO-диаграмму*, на которой будут только стрелки работы «Сборка и тестирование компьютеров».

Выберите пункт меню *Diagram/Add FEO Diagram*.

В диалоге *Add New FEO Diagram* выберите тип и внесите имя диаграммы FEO. Щелкните по ОК.

Для определения диаграммы перейдите в *Diagram/Diagram Properties* и во вкладке *Diagram Text* внесите определение.



Рис. 63 – Результат выполнения задания

Удалите лишние стрелки на диаграмме FEO. Результат показан на рис. 64.

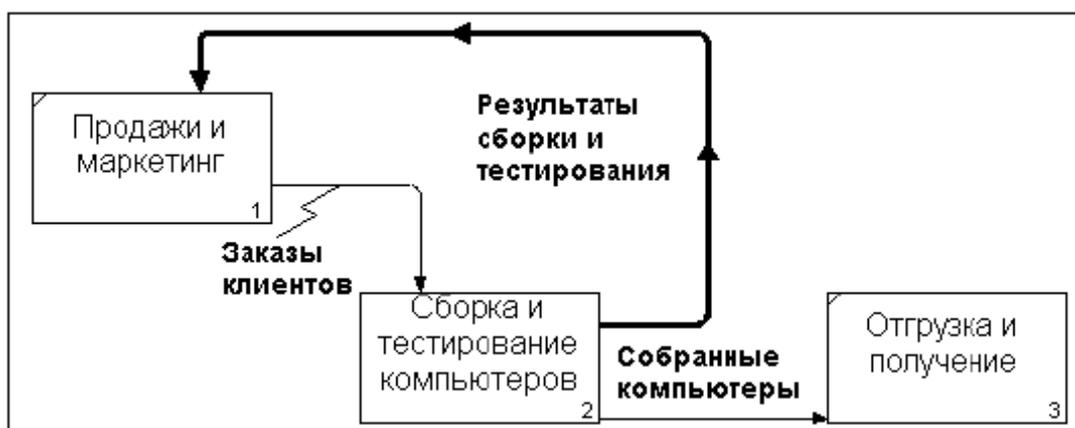


Рис. 64 – Диаграмма FEO

Для перехода между стандартной диаграммой, деревом узлов и FEO используйте *Кнопку*  на палитре инструментов.

Расщепление модели. Перейдите на диаграмму A-0. Правой кнопкой мыши щелкните по работе «Сборка и тестирование компьютеров» и выберите *Split model*. В диалоге *Split Option* внесите имя новой модели «Сборка и тестирование компьютеров», установите опции, как на рисунке, и щелкните по ОК (рис. 65).

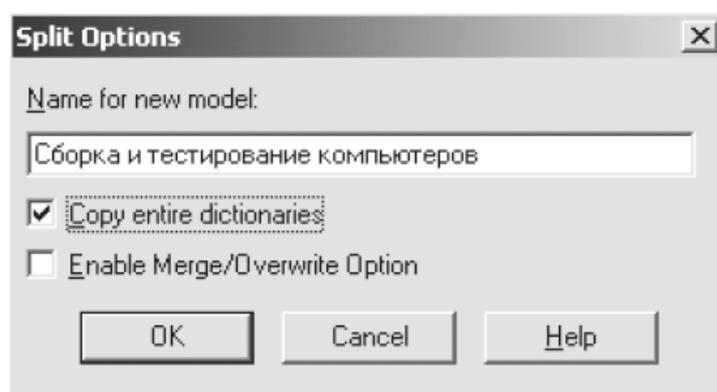


Рис. 65 – Диалог Split Option

Посмотрите на результат: в *Model Explorer* появилась новая модель, а на диаграмме A0 модели «Деятельность компании» появилась стрелка вызова «Сборка и тестирование компьютеров».

Создайте в модели «Сборка и тестирование компьютеров» новую стрелку «Неисправные компоненты». На диаграмме A0 это будет граничная стрелка выхода, на диаграмме A-0 – граничная стрелка выхода от работ «Сборка настольных компьютеров», «Тестирование компьютеров» и «Сборка ноутбуков».

Слияние модели. Перейдите на диаграмму A0 модели «Деятельность компании». Правой кнопкой мыши щелкните по работе «Сборка и тестирование компьютеров» и выберите *Merge model*.

В диалоге *Merge Model* включите опцию *Cut/Paste entire dictionaries* и щелкните по ОК.

Посмотрите на результат. В *Model Explorer* видно, что две модели слились. Модель «Сборка и тестирование компьютеров» осталась и может быть сохранена в отдельном файле. На диаграмме A0 модели «Деятельность компании» исчезла стрелка вызова «Сборка и тестирование компьютеров». Появилась неразрешенная граничная стрелка «Неисправные компоненты». Направьте эту стрелку ко входу работы «Отгрузка и получение».

Создание отчета по модели. В меню *Tools/Reports/Model Report* задайте опции генерирования отчета (установите галочки) и нажмите кнопку *Preview* (Предварительный просмотр) (рис. 66).

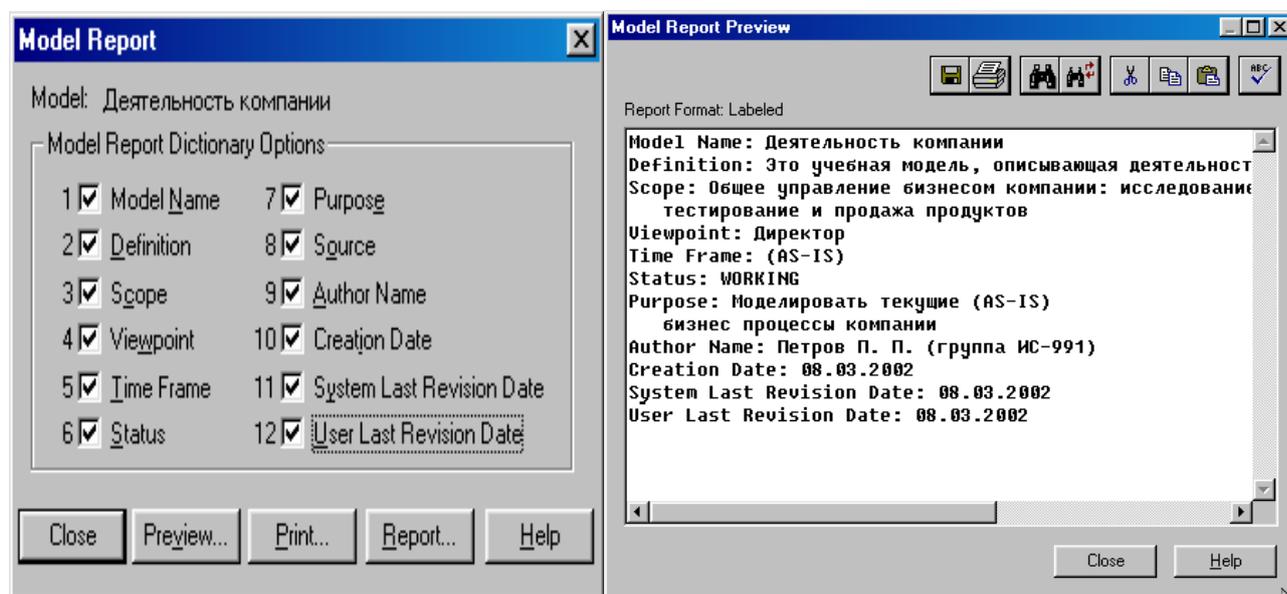


Рис. 66 – Формирование отчета

Контрольные вопросы:

1. В чем заключается суть построения диаграммы дерева узлов?
2. Каким образом можно преобразовать диаграмму дерева узлов?
3. Определите сущность и назначение FEO-диаграммы.
4. Перечислите этапы создания FEO-диаграммы.
5. Обоснуйте необходимость слияния и расщепления моделей.
6. Перечислите этапы слияния моделей в Allfusion Process Modeler (BPWin).
7. Перечислите этапы расщепления моделей в Allfusion Process Modeler (BPWin).
8. Какие типы отчетов возможно выполнить в BPWin?
9. Перечислите этапы создания отчета в среде BPWin.

3 Моделирование взаимодействия бизнес-процессов предприятия на основании использования методологии IDEF3

Метод моделирования IDEF3, являющийся частью семейства стандартов IDEF, предназначен для таких моделей процессов, в которых важно понять последовательность выполнения действий и взаимозависимости между ними. IDEF3 приобрел широкое распространение среди системных 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.

Диаграммы IDEF3 также называют **WorkFlow diagramming** – методологией моделирования, использующей графическое описание информационных потоков, взаимоотношений между процессами обработки

информации и объектов, являющихся частью этих процессов. Диаграммы WorkFlow используются для анализа процедур обработки информации.

Цель IDEF3 – дать аналитикам описание последовательности выполнения процессов, а также объектов, участвующих совместно в одном процессе.

Диаграмма является основной единицей описания в IDEF3-модели. Организация диаграмм в IDEF3 является наиболее важной, если модель редактируется несколькими людьми. В этом случае разработчик должен определять, какая информация будет входить в ту или иную модель.

Единицы работы – Unit of Work (UOW), также называемые работами, являются центральными компонентами модели. В IDEF3 работы изображаются прямоугольниками и имеют имя, обозначающее процесс действия и номер (идентификатор). В имя обычно включается основной результат работы (например, приготовление обеда).

Связи показывают взаимоотношения работ. Все связи в IDEF3 являются однонаправленными и делятся на **3 типа**:

– **старшая (Precedence) линия** – сплошная линия (—→), связывающая единицы работ. Рисуеться слева направо или сверху вниз. Показывает, что работа-источник должна закончиться прежде, чем работа-цель начнется;

– **линия отношения (Relation Link)** – пунктирная линия (- - - →), используемая для изображения связей между единицами работ, а также между единицами работ и объектами ссылок;

– **потоки объектов (Object Flow)** – стрелка с двумя наконечниками (—→→), применяется для описания использования объекта в двух или более единицах работы, например когда объект порождается в одной работе и используется в другой.

Старшая связь показывает, что работа-источник заканчивается ранее, чем начинается работа-цель. Часто результатом работы-источника становится объект, необходимый для запуска работы-цели. В этом случае стрелку, обозначающую объект, изображают с двойным наконечником. Имя стрелки

должно ясно идентифицировать отображаемый объект. Поток объектов имеет ту же семантику, что и старшая стрелка.

Отношение показывает, что стрелка является альтернативой старшей стрелке или потоку объектов в смысле задания последовательности выполнения работ – работа-источник не обязательно должна закончиться, прежде чем работа-цель начнется. Более того, работа-цель может закончиться прежде, чем закончится работа-источник.

Логика взаимодействия стрелок – перекрестки

Перекрестки (Junction) – используются для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий, которые могут или должны быть завершены перед началом следующей работы.

Различают *перекрестки для слияния (Fan-in Junction)* и *разветвления (Fan-out Junction) стрелок*. Перекресток не может использоваться одновременно для слияния и для разветвления. Для внесения перекрестка служит кнопка  (табл. 11).

Таблица 11 – Типы перекрестков

Обозначение	Наименование	Смысл в случае слияния стрелок (значение Fan-in)	Смысл в случае разветвления стрелок (значение Fan-out)
	Asynchronous (асинхронное) AND	Все предшествующие процессы должны быть завершены	Все следующие процессы должны быть завершены
	Synchronous (синхронное) AND	Все предшествующие процессы завершены одновременно	Все следующие процессы запускаются одновременно
	Asynchronous (асинхронное) OR	Один или несколько предшествующих процессов должны быть завершены	Один или несколько следующих процессов должны быть запущены
	Synchronous (синхронное) OR	Один или несколько предшествующих процессов завершены одновременно	Один или несколько следующих процессов запускаются одновременно
	XOR (Exclusive OR)	Только один процесс завершен	Только один следующий процесс запускается

Объекты-ссылки – являются специальными символами, которые ссылаются на внешние части описания процесса. Они добавляются на диаграмму для того, чтобы обратить внимание редактора на что-либо важное, что невозможно связать со стрелкой, работой или перекрестком.

Для внесения объекта-ссылки служит кнопка . Объект-ссылка отображается в виде прямоугольника. Объекты-ссылка должны быть связаны с единицами работ или перекрестками пунктирными линиями. При внесении объектов-ссылок необходимо указать их тип (табл. 12).

Таблица 12 – Типы объектов-ссылок

Тип объекта-ссылки	Цель описания
ОБЪЕКТ	Описывает участие важного объекта в работе
GOTO	Инструмент циклического перехода (в повторяющейся последовательности работ). Если все работы цикла присутствуют на текущей диаграмме, цикл может также изображаться стрелкой, возвращающейся на стартовую работу. GOTO может ссылаться на перекресток
JOB (Unit of behavior)	Применяется, когда необходимо подчеркнуть множественное использование какой-либо работы, но без цикла. Обычно этот тип ссылки не используется для моделирования автоматически запускающихся работ
NOTE	Используется для документирования важной информации, относящейся к графическим объектам на диаграмме. NOTE является альтернативой внесению текстового объекта в диаграмму
ELAB (Elaboration)	Используется для усовершенствования графиков или их более детального описания. Обычно употребляется для детального описания разветвления и слияния стрелок на перекрестках

Создание диаграммы IDEF3. Перейдите на диаграмму A2 и декомпозируйте работу «Сборка настольных компьютеров». В диалоге *Activity Box Count* установите число работ 4 и нотацию IDEF3 (рис. 67).

После этого на экране возникает диаграмма IDEF3, содержащая работы (UOW). Правой кнопкой мыши щелкните по работе, выберите в контекстном меню *Name* и внесите имя работы «Подготовка компонентов». Затем во вкладке *Definition* внесите определение «Подготавливаются все компоненты компьютера согласно спецификации заказа». Во вкладке *UOW* внесите свойства работы (табл. 13).

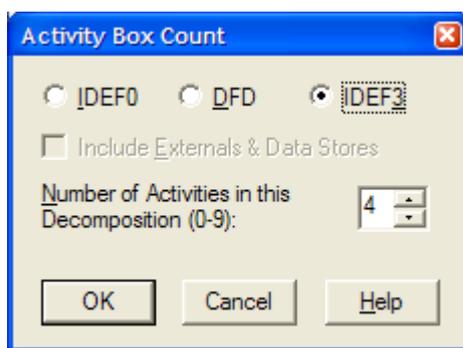


Рис. 67 – Создание диаграммы IDEF3

Таблица 13 – Свойства UOW

Свойство	Описание
Objects	Компоненты: винчестеры, корпуса, материнские платы, звуковые карты, дисководы, SD-ROM и флоппи, модемы, программное обеспечение
Facts	Доступные операционные системы: Windows NT, Windows XP, Windows Vista, Windows 7
Constraints	Установка модема требует установки дополнительного программного обеспечения

Внесите в диаграмму еще 3 работы с помощью кнопки .

Задайте имена работ:

- установка материнской платы и винчестера;
- установка модема;
- установка дисковода CD-ROM;
- установка флоппи-дисковода;
- инсталляция операционной системы;
- инсталляция дополнительного программного обеспечения.

С помощью кнопки  панели инструментов создайте объект ссылки. Внесите имя объекта внешней ссылки «Компоненты». Свяжите стрелкой объект ссылки и работу «Подготовка компонент».

Свяжите стрелкой работы «Подготовка компонент» (выход) и «Установка материнской платы и винчестера». Измените стиль стрелки на *Object Flow*. В IDEF3 имя стрелки может отсутствовать, хотя VPWin показывает отсутствие имени как ошибку. Результат представлен на рис. 68.

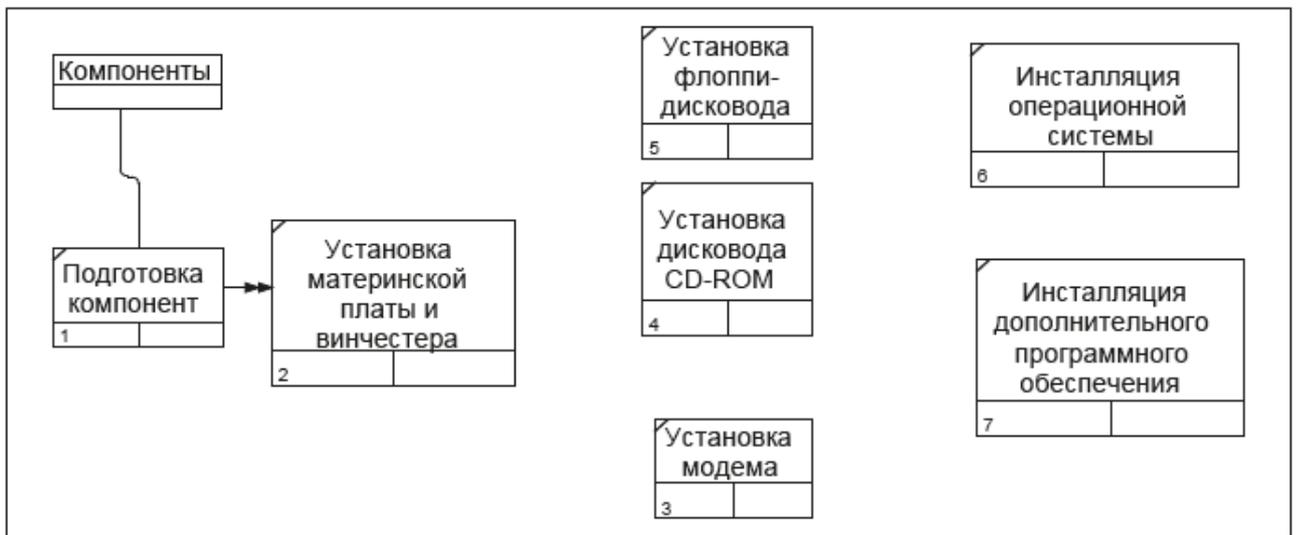


Рис. 68 – Результат создания UOW и объекта ссылки

С помощью кнопки  на палитре инструментов внесите два перекрестка типа «асинхронное или» и свяжите работы с перекрестками, как показано на рисунке 69.

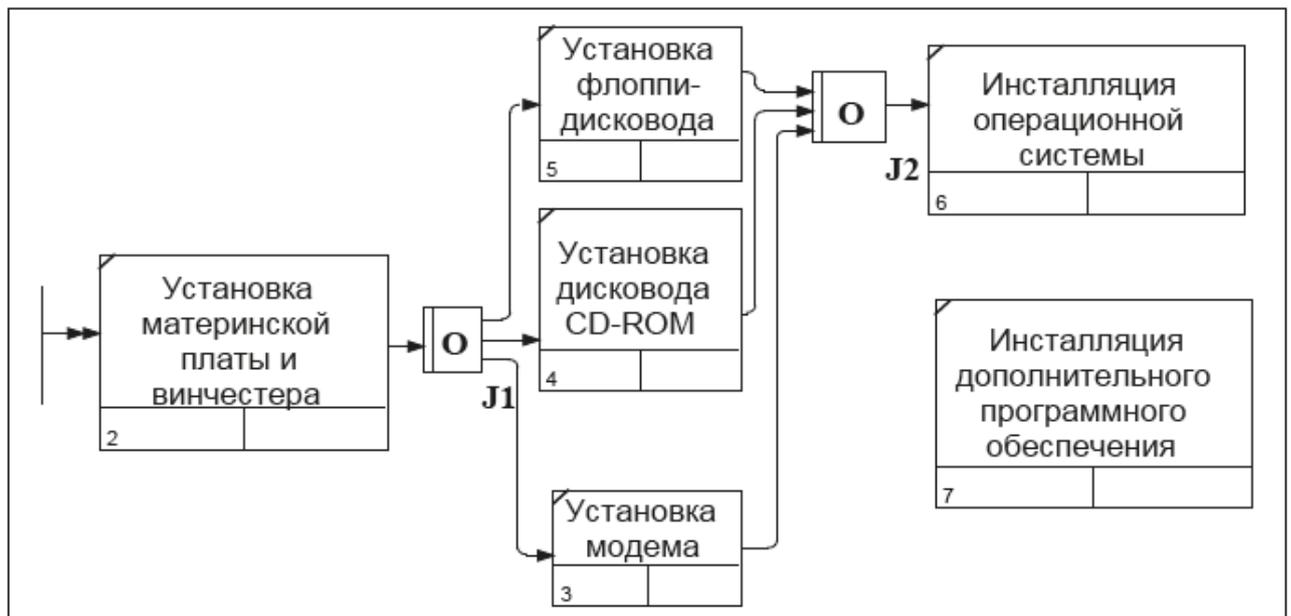


Рис. 69 – Диаграмма IDEF3 после создания перекрестков

Правой кнопкой щелкните по перекрестку для разветвления (*fan-out*), выберите *Name* и внесите имя «Компоненты, требуемые в спецификации заказа».

Создайте два перекрестка типа «исключающего или». С помощью кнопки  палитры инструментов создайте объект ссылки. Внесите имя объекта внешней ссылки «Программное обеспечение», и свяжите работы, как представлено на рисунке 70.

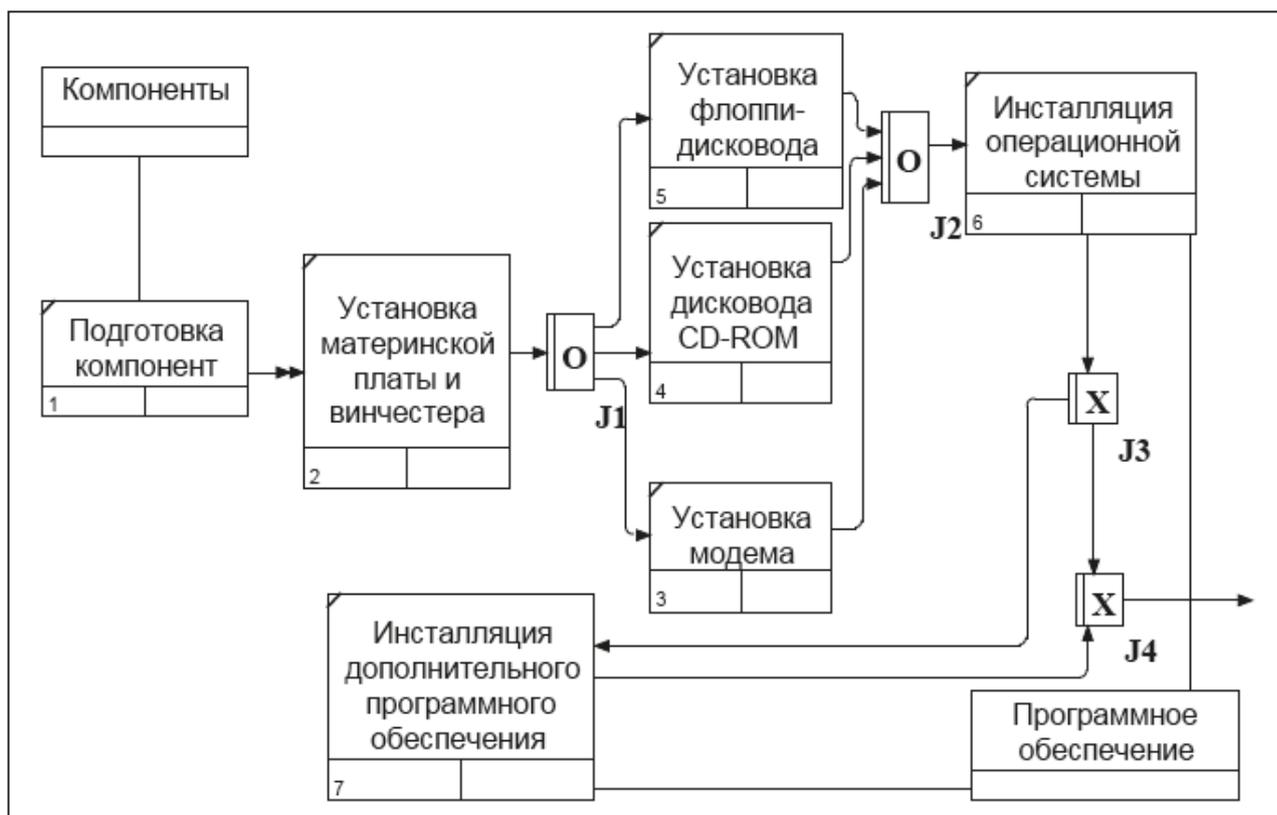


Рис. 70 – Результат моделирования

Создание сценария. Выберите пункт меню *Diagram/Add IDEF3 Scenario*. Создайте диаграмму сценария на основе диаграммы IDEF3 «Сборка настольных компьютеров» (A22.1).

Удалите элементы, не входящие в сценарий (рис. 71).



Рис. 71 – Результат выполнения задания

Контрольные вопросы:

1. Что описывает диаграмма IDEF3?
2. Что называется перекрестком?
3. Назовите типы перекрестков.
4. Что называется объектом-ссылкой?
5. Какие бывают типы объектов-ссылок?
6. Как добавить объект-ссылку?
7. Объясните механизм дополнения диаграммы IDEFO диаграммой DFD.
8. Перечислите составные элементы диаграмм IDEF3.
9. Что показывают связи в диаграммах IDEF3?
10. Перечислите типы стрелок в диаграммах IDEF3.

4 Проведение стоимостного анализа, основанного на работах (ABC-анализ)

В структурном проектировании аналитику предоставляется **два инструмента для оценки модели:**

- стоимостный анализ, основанный на работах, (*Activity Based Costing, ABC*);
- анализ на основе свойств, определяемых пользователем (*User Defined Properties, UDP*).

Функциональное оценивание – ABC – это технология выявления и исследования стоимости выполнения той или иной функции (действия). Исходными данными для функционального оценивания являются затраты на ресурсы (материалы, персонал и т. д.). В сравнении с традиционными способами оценки затрат, при применении которых часто недооценивается продукция, производимая в незначительном объеме, и переоценивается массовый выпуск, ABC обеспечивает более точный метод расчета стоимости производства продукции, основанный на стоимости выполнения всех технологических операций, выполняемых при ее выпуске. Стоимостный анализ представляет собой соглашение об учете, используемое для сбора затрат, связанных с работами, с целью *определения общей стоимости процесса*.

Стоимостный анализ основан на модели работ, потому что количественная оценка невозможна без детального понимания функциональности предприятия. Обычно ABC применяется для того, чтобы понять происхождение выходных затрат и облегчить выбор нужной модели работ при реорганизации деятельности предприятия (Business Process Reengineering, BPR). С помощью стоимостного анализа можно решить такие задачи, как определение действительной стоимости производства продукта, определение действительной стоимости поддержки клиента, идентификация наиболее дорогостоящих работ (тех, которые должны быть улучшены в первую очередь), обеспечение менеджеров финансовой мерой предлагаемых изменений и т. д.

ABC-анализ может проводиться только тогда, когда создание модели работы закончено, а именно:

– модель работы последовательна (следует синтаксическим правилам IDEF0);

– корректна (отражает бизнес), полная (охватывает всю рассматриваемую область);

– стабильна (проходит цикл экспертизы без изменений).

ABC включает следующие основные понятия:

– **объект затрат** – причина, по которой работа выполняется, обычно основной выход работы. Стоимость работ является суммарной стоимостью объектов затрат;

– **двигатель затрат** – характеристики входов и управлений работы («Заказы клиентов», «Правила сборки и тестирования», «Персонал производственного отдела»), которые влияют на то, как выполняется и как долго длится работа;

– **центры затрат**, которые можно трактовать как статьи расхода.

Список центров затрат упорядочен. Общие затраты по работе рассчитываются как сумма по всем центрам затрат. При вычислении затрат вышестоящей (родительской) работы сначала вычисляется произведение затрат дочерней работы на частоту работы (число раз, которое работа выполняется в рамках проведения родительской работы), затем результаты складываются. Этот достаточно упрощенный принцип подсчета справедлив, если работы выполняются последовательно.

Результаты стоимостного анализа могут существенно повлиять на очередность выполнения работ.

Пример. Предположим, что для оценки качества изделия необходимо провести три работы:

– внешний осмотр – стоимость 50 у. е.;

– пробное включение – стоимость 150 у. е.;

– испытание на стенде – стоимость 300 у. е.

Предположим также, что с точки зрения технологии очередность проведения работ не существенна, а вероятность выявления брака одинакова (50 %). Пусть необходимо проверить восемь изделий. Если проводить работы в убывающем по стоимости порядке, то затраты на получение готового изделия составят:

$$300 \text{ у. е. (испытание на стенде)} \cdot 8 + 150 \text{ у. е. (пробное включение)} \cdot 4 + \\ + 50 \text{ у. е. (внешний осмотр)} \cdot 2 = 3100 \text{ у. е.}$$

Если проводить работы в возрастающем по стоимости порядке, то на получение готового изделия будет затрачено:

$$50 \text{ у. е. (внешний осмотр)} \cdot 8 + 150 \text{ у. е. (пробное включение)} \cdot 4 + \\ + 300 \text{ у. е. (испытание на стенде)} \cdot 2 = 1600 \text{ у. е.}$$

Следовательно, с целью минимизации затрат первой должна быть выполнена наиболее дешевая работа, затем – средняя по стоимости и в конце – наиболее дорогая.

Результаты стоимостного анализа наглядно представляются на специальном отчете.

Свойства, определяемые пользователем (UDP)

ABC позволяет оценить стоимостные и временные характеристики системы. Если стоимостных показателей недостаточно, существует *возможность внесения собственных метрик* – свойств, определенных пользователем (User Defined Properties, UDP). UDP позволяют провести дополнительный анализ, хотя и без суммирующих подсчетов. Каждой работе можно поставить в соответствие набор UDP. Результат задания можно проанализировать в отчете.

Реализация стоимостного анализа (Activity Based Costing)

Условие. На производственном участке работают 5 сборщиков и 1 тестировщик. В среднем в день собирается 12 настольных компьютеров и 20 ноутбуков. Двое из сборщиков – стажеры.

Оплата:

- диспетчера – 15000 руб. в месяц;
- сборщика и тестировщика – 300 руб. в час;
- стажера – 90 руб. в час.

Средняя стоимость компонентов:

- для настольного компьютера – 24000 руб.;
- для ноутбука – 42000 руб.

Практическая реализация задания.

Перейдите на диаграмму A2. В диалоге *Model Properties* (вызывается из меню *Mode/Model Properties*) во вкладке *ABC Units* (рис. 72) установите единицы измерения денег и времени – рубли и дни.

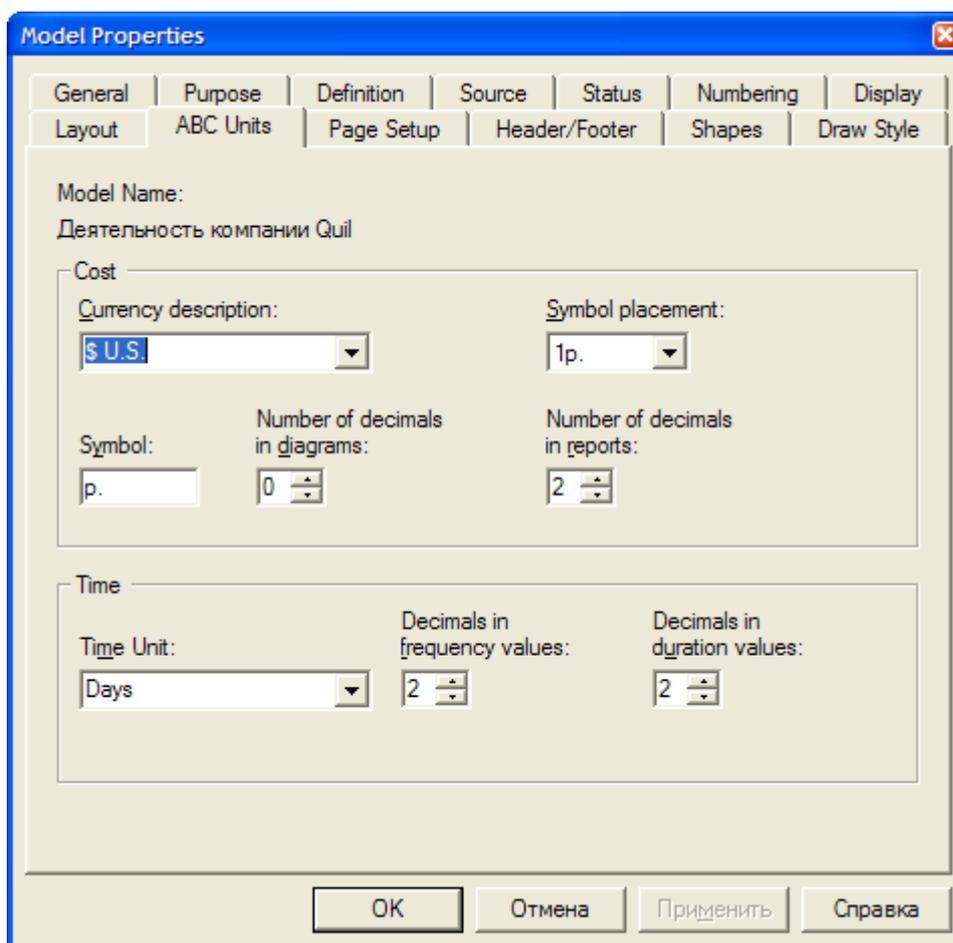


Рис. 72 – Вкладка ABC Units диалога Model Properties

Перейдите в *Dictionary/Cost Center* и в диалоге *Cost Center Dictionary* внесите название и определение центров затрат согласно таблице 14.

Таблица 14 – Центры затрат ABC

Центр затрат	Определение
Управление	Затраты на управление, связанные с составлением графика работ, формированием партий компьютеров, контролем над сборкой и тестированием
Рабочая сила	Затраты на оплату рабочих, занятых сборкой и тестированием компьютеров
Компоненты	Затраты на закупку компонентов

Для отображения стоимости каждой работы в нижнем левом углу прямоугольника с названием работы диаграммы A2 перейдите в меню *Model/Model Properties* и во вкладке *Display* диалога *Model Properties* включите опцию *ABC Data* (рис. 73).

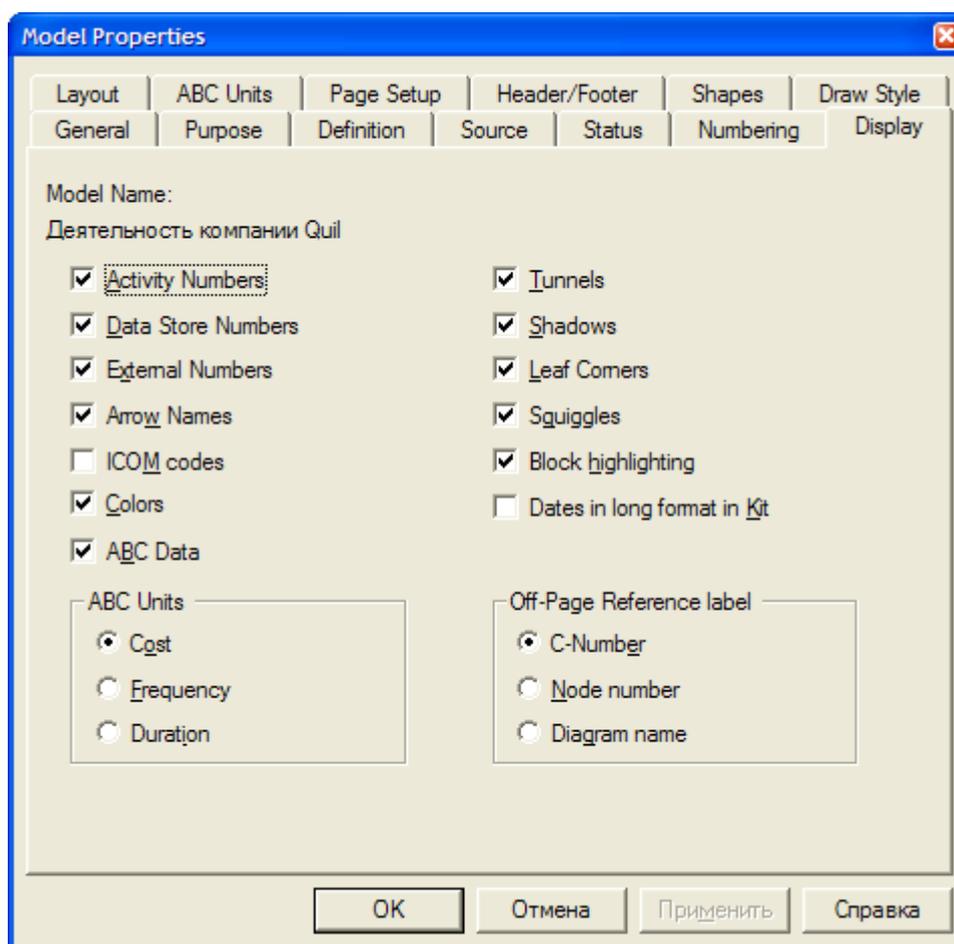


Рис. 73 – Вкладка Display диалога Model Properties

Для отображения частоты или продолжительности работы переключите кнопки в группе *ABC Units*. Для назначения стоимости работы следует щелкнуть по ней на диаграмме A2 правой кнопкой мыши и выбрать в контекстном меню *Cost* (рис. 74). Для задания стоимости выполнения работы, на диаграмме A2 внесите параметры ABC согласно таблице 15.

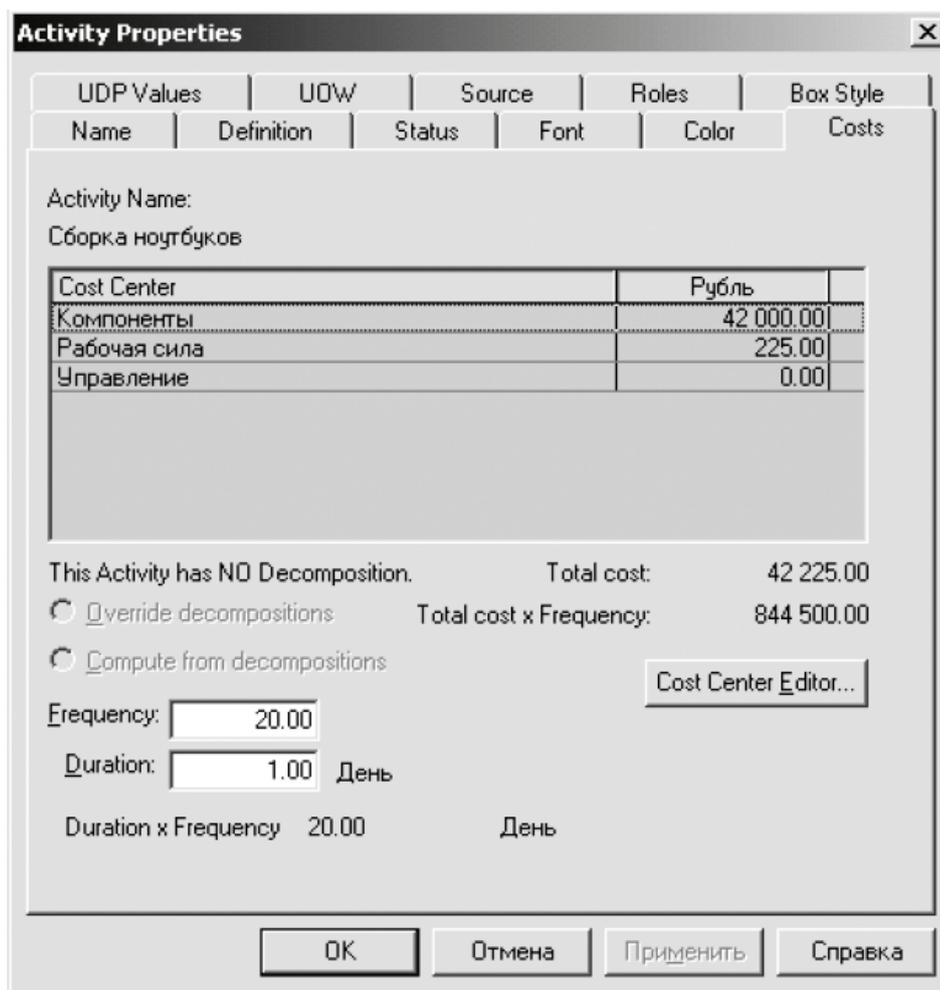


Рис. 74 – Вкладка Cost диалога Activity Properties

Таблица 15 – Стоимости работ к диаграмме A2

Activity Name	Cost Center	Cost Center Cost, руб.	Duration, день	Частота (Frequency)
Отслеживание расписания и управление сборкой и тестированием	Управление	750,00	1,00	1,00
Сборка настольных компьютеров	Рабочая сила	150,00	1,00	12,00
	Компоненты	24000,00		
Сборка ноутбуков	Рабочая сила	225,00	1,00	20,00
	Компоненты	42000,00		
Тестирование компьютеров	Рабочая сила	60,00	1,00	32,00

Результат стоимостного анализа отобразится в левом нижнем углу блока работы, для которой он был проведен (рис. 75).

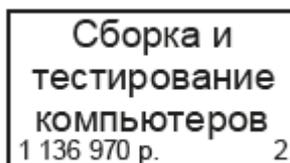


Рис. 75 – Отображение стоимости работы

Сгенерируйте отчет работы *Activity Cost Report* и внесите его в отчет о проделанной работе (рисунок 76).

Activity Name	Activity Cost (Рубль)	Cost Center	Cost Center Cost (Рубль)
Деятельность компании Quill	1 136 970.00	Компоненты	1 128 000.00
		Рабочая сила	8 220.00
		Управление	750.00
Продажи и маркетинг	0.00		
Сборка и тестирование компьютеров	1 136 970.00	Компоненты	1 128 000.00
		Рабочая сила	8 220.00
		Управление	750.00
Отслеживание расписания и управление сборкой и тестированием	750.00	Управление	750.00
Сборка настольных компьютеров	24 150.00	Компоненты	24 000.00
		Рабочая сила	150.00

Рис. 76 – Отчет Activity Cost Report

Использование категорий UDP (критерий пользователя)

Перейдите в меню *Dictionary/UDP Keywords* и в диалоге *UDP Keywords List* внесите ключевые слова критериев пользователя (UDP) (рис. 77):

- расход ресурсов;
- документация;
- информационная система.

Создайте критерий пользователя (UDP). Для этого перейдите в *Dictionary/UDP* и в словарь внесите имя *UDP*, например «Приложение».

Для UDP типа *List* необходимо в поле *Value* задать список значений. Для UDP «Приложение» внесите значение «Модуль оформления заказов» (рис. 78).

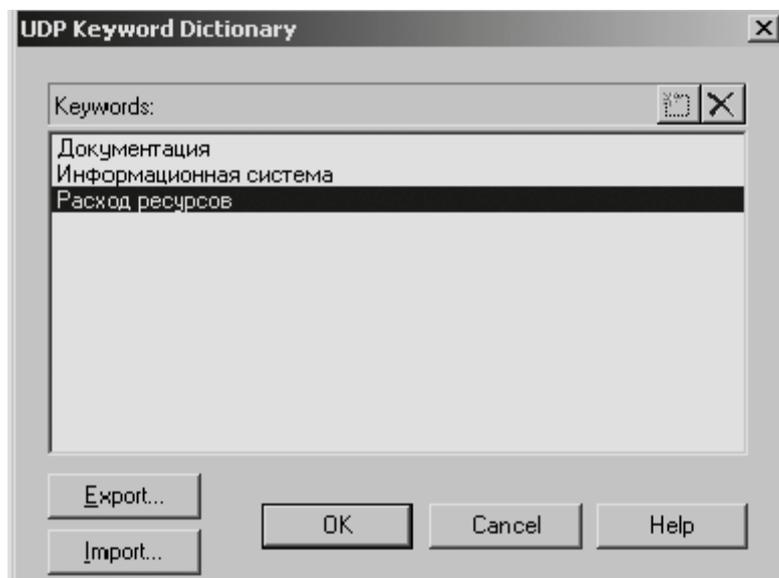


Рис. 77 – Словарь ключевых слов UDP

Name	D	UDP Datatype	S	Value	Keyword
Дополнительная документация		Command List		Edit List	Документация
Загрязнение окружающей среды		Text List (Single selection)		Edit List	
История изменения		Paragraph Text			Документация
Приложение		Text List (Multiple selections)		Edit List	Информационная с
Расход электроэнергии		Real Number	2		Расход ресурсов

Рис. 78 – Словарь UDP

Затем внесите другие значения в соответствии с данными таблицы 16.

Для подключения к UDP ключевого слова перейдите к полю *Keyword* и щелкните по полю выбора.

Для назначения пользовательского критерия оценки (UDP) какой-либо работы следует выделить ее правой кнопкой мыши и выбрать в контекстном меню опцию UDP. Появляется вкладка *UDP Values* диалогового окна *Activity Properties* (рис. 79).

Таблица 16 – Наименование и свойства UDP

Наименование UDP	Тип	Значение	Ключевое слово
Приложения	Text List (Multiple Selection)	Модуль оформления заказов Модуль создания и контроля расписания выполнения работ Модуль учета комплектующих и оборудования Модуль процедур сборки и поиска неисправностей	Информационная система
Дополнительная документация	Command List	Winword.exe sample1.doc Winword.exe sample2.doc PowerPNT.exe sample3.ppt	Документация
История изменения	Paragraph Text		Документация
Загрязнение окружающей среды	Text List (Single Selection)	Очень высокое Высокое Среднее Низкое	
Расход электроэнергии	Real Number		Расход ресурсов

Далее необходимо внести значения UDP для работ согласно данным таблицы 17. После внесения UDP типа *Command* или *Command List* щелчок по кнопке  приведет к запуску приложения.

Таблица 17 – Значения UDP

Имя работы (Activity Name)	Дополнит. Документация	Приложения	История изменения	Расход электроэнергии	Загряз-е окруж. среды
Сборка настольных компьютеров		Модуль учета комплектующих и оборудования Модуль процедур сборки и поиска неисправностей		20,00	Среднее
Сборка ноутбуков		Модуль учета комплектующих и оборудования Модуль процедур сборки и поиска неисправностей		25,00	Среднее
Тестирование компьютеров		Модуль учета комплектующих и оборудования Модуль процедур сборки и поиска неисправностей		40,00	Среднее
Отслеживание расписания и управление сборкой и тестированием	Winword.exe Sample2.doc	Модуль создания и контроля расписания выполнения работ	История изменения спецификаций	10,00	Низкое



Рис. 79 – Вкладка UDP Values диалогового окна Activity Properties

В диалоге *Activity Properties* щелкните по кнопке *Filter*. В появившемся диалоге *Diagram Object UDP filter* (рисунок 80) отключите ключевые слова «Информационная система». Щелкните по ОК. В результате в диалоге *Activity Properties* не будут отображаться UDP с ключевыми словами «Информационная система».

Отметим, что пользовательские критерии оценки (UDP) можно закрепить не только за работами, но и за стрелками.

Просмотрите отчет по UDP, используя меню *Tools/Report/Diagram Object Report*.

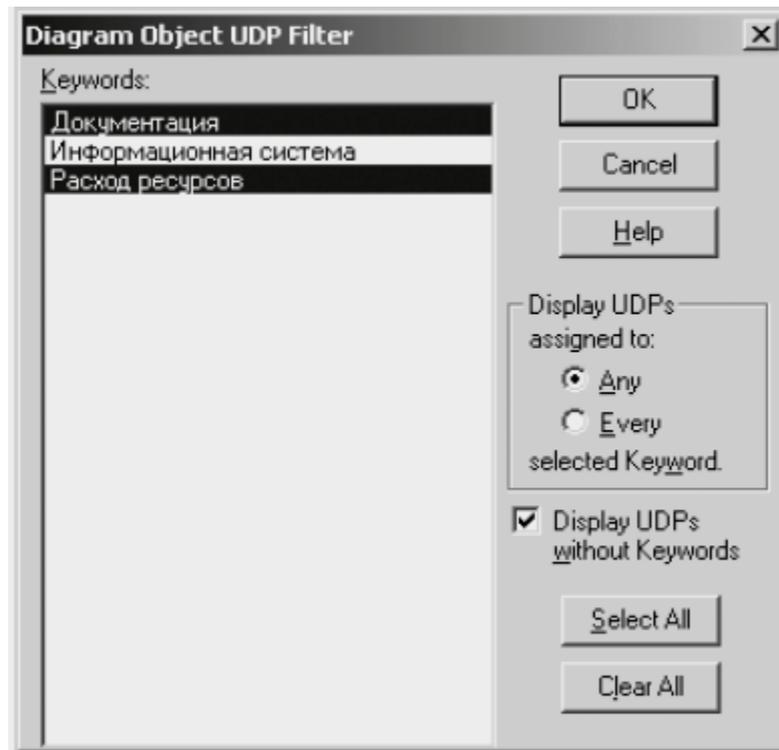


Рис. 80 – Диалог Diagram Object UDP filter

Выберите опции отчета:

- *Start from Activity* – A2. Сборка и тестирование компьютеров;
- *Number of levels* – 2;
- *User Defined Properties* – Расход электроэнергии;
- *Report Format* – DDE Table.

Щелкните по кнопке *Report*. В появившемся диалоговом окне выберите *MS Word – New* и нажмите кнопку ОК (рис. 81).

В редакторе Word создан файл «Документ x», в котором средствами Word можно добавить строку в таблице для расчета итогового значения расхода электроэнергии (вместо «x» редактор Word подставит какую-то цифру). Итоговый документ представлен на рисунке 82.

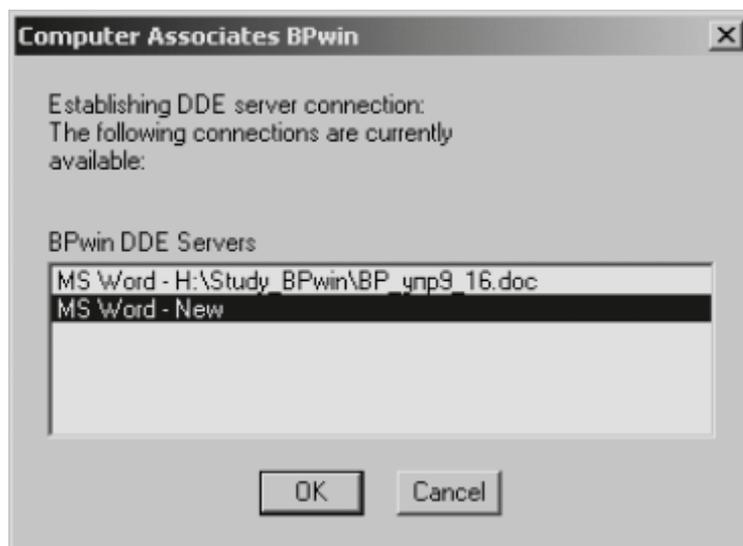


Рис. 81 – Окно формирования отчета

Activity Name	Расход электроэнергии
Сборка и тестирование компьютеров	
Отслеживание расписания и управление сборкой и тестированием	10,00
Сборка настольных компьютеров	20,00
Сборка ноутбуков	25,00
Тестирование компьютеров	40,00
ИТОГО	95,00

Рис. 82 – Отчет о расходе электроэнергии

Контрольные вопросы:

1. Опишите инструменты для оценки модели.
2. Опишите технологию выявления и исследования стоимости выполнения той или иной функции.
3. На какой модели основан Стоимостной анализ?
4. Для чего применяется Функциональное оценивание (ABC)?
5. Какие основные понятия включает методология ABC?
6. Какие характеристики позволяет оценить ABC?
7. Для чего применяется User Defined Properties (UDP)?

5 Моделирование функциональных требований к проектируемой системе при помощи диаграммы потоков данных DFD

Целью методики **DFD (Data Flow Diagrams)** является построение модели рассматриваемой системы в виде **диаграммы потоков данных**, обеспечивающей правильное описание выходов (отклика системы в виде данных) при заданном воздействии на вход системы (подаче сигналов через внешние интерфейсы). Диаграммы потоков данных являются основным средством моделирования функциональных требований к проектируемой системе.

При создании диаграммы потоков данных используются четыре основных понятия: потоки данных, процессы (работы) преобразования входных потоков данных в выходные, внешние сущности, накопители данных (хранилища).

Потоки данных являются абстракциями, используемыми для моделирования передачи информации (или физических компонентов) из одной части системы в другую. Потоки на диаграммах изображаются именованными стрелками, ориентация которых указывает направление движения информации.

Назначение процесса (работы) состоит в продуцировании выходных потоков из входных в соответствии с действием, задаваемым именем процесса. Имя процесса должно содержать глагол в неопределенной форме с последующим дополнением (например, «получить документы по отгрузке продукции»). Каждый процесс имеет уникальный номер для ссылок на него внутри диаграммы, который может использоваться совместно с номером диаграммы для получения уникального индекса процесса во всей модели.

Хранилище (накопитель) данных позволяет на указанных участках определять данные, которые будут сохраняться в памяти между процессами. Фактически хранилище представляет «срезы» потоков данных во времени. Информация, которую оно содержит, может использоваться в любое время

после ее получения, при этом данные могут выбираться в любом порядке. Имя хранилища должно определять его содержимое и быть существительным.

Внешняя сущность представляет собой материальный объект вне контекста системы, являющейся источником или приемником системных данных. Ее имя должно содержать существительное, например, «Склад товаров». Предполагается, что объекты, представленные как внешние сущности, не должны участвовать ни в какой обработке.

Кроме основных элементов, в состав DFD входят словари данных и миниспецификации.

Словари данных являются каталогами всех элементов данных, присутствующих в DFD, включая групповые и индивидуальные потоки данных, хранилища и процессы, а также все их атрибуты.

Миниспецификации обработки – описывают DFD-процессы нижнего уровня. Фактически миниспецификации представляют собой алгоритмы описания задач, выполняемых процессами: множество всех миниспецификаций является полной спецификацией системы.

Цель представления системы в виде DFD-диаграммы – продемонстрировать, как каждый процесс преобразует свои входные данные в выходные, а также выявить отношения между этими процессами.

Согласно методике построения DFD, модель системы определяется как иерархия диаграмм потоков данных, описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи потребителю.

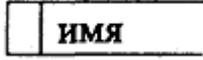
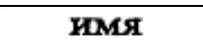
Эти диаграммы представляют сеть связанных между собой работ. Их удобно использовать для описания документооборота и обработки информации. DFD показывают, как объекты (включая данные) двигаются от одной работы к другой. Это представление потоков совместно с хранилищами данных и внешними сущностями делает модели DFD более похожими на физические характеристики системы – движение объектов (data flow), хранение объектов (data stores), поставка и распространение объектов (external entities).

DFD описывает:

- функции обработки информации (работы);
- документы (стрелки, arrow), объекты, сотрудников или отделы, которые участвуют в обработке информации;
- внешние ссылки (external reference), которые обеспечивают интерфейс с внешними объектами, находящимися за границами моделируемой системы;
- таблицы для хранения документов (хранилища данных, data store).

Для построения диаграмм DFD в BPWin используется нотация Гейна-Сарсона (табл. 18).

Таблица 18 – Нотация Гейна-Сарсона

Компонент	Обозначение
Поток данных	
Процесс	
Хранилище	
Внешняя сущность	

Для дополнения модели IDEF0 диаграммой DFD нужно в процессе декомпозиции в диалоге *Activity Box Count* указать тип диаграммы DFD.

К преимуществам методики DFD относятся:

- возможность однозначно определить внешние сущности, анализируя потоки информации внутри и вне системы;
- возможность проектирования сверху вниз, что облегчает построение модели «как должно быть»;
- наличие спецификаций процессов нижнего уровня, что позволяет преодолеть логическую незавершенность функциональной модели и построить полную функциональную спецификацию разрабатываемой системы.

К **недостаткам модели** отнесем: необходимость искусственного ввода управляющих процессов, поскольку управляющие воздействия (потоки) и управляющие процессы с точки зрения DFD ничем не отличаются от обычных;

отсутствие понятия времени, т. е. отсутствие анализа временных промежутков при преобразовании данных (все ограничения по времени должны быть введены в спецификациях процессов).

Условие. Работа по продажам и маркетингу заключается в ответах на телефонные звонки клиентов, предоставлении клиентам информации о ценах, оформлении заказов, внесении заказов в информационную систему и исследовании рынка.

При оформлении заказа важно проверить, существует ли такой клиент в базе данных и, если не существует, внести его в базу данных и затем оформить заказ. Оформление заказа начинается со звонка клиента. В процессе оформления заказа база данных клиентов может просматриваться и редактироваться. Заказ должен включать как информацию о клиенте, так и информацию о заказанных продуктах. Оформление заказа подразумевает чтение и запись информации о прочих заказах.

На основе построенных ранее контекстных и декомпозиционных диаграмм, необходимо разработать диаграмму потока данных для процесса «Оформление заказа».

Практическая реализация

На первом шаге выполнения задания декомпозируйте работу «Продажи и маркетинг» (IDEF0). Создайте следующие работы:

- предоставление информации о ценах;
- оформление заказов;
- исследование рынка.

Результат декомпозиции представлен на рисунке 83.

В процессе оформления заказа база данных клиентов может просматриваться и редактироваться. Заказ должен включать как информацию о клиенте, так и информацию о заказанных продуктах. Оформление заказа подразумевает чтение и запись информации о прочих заказах.

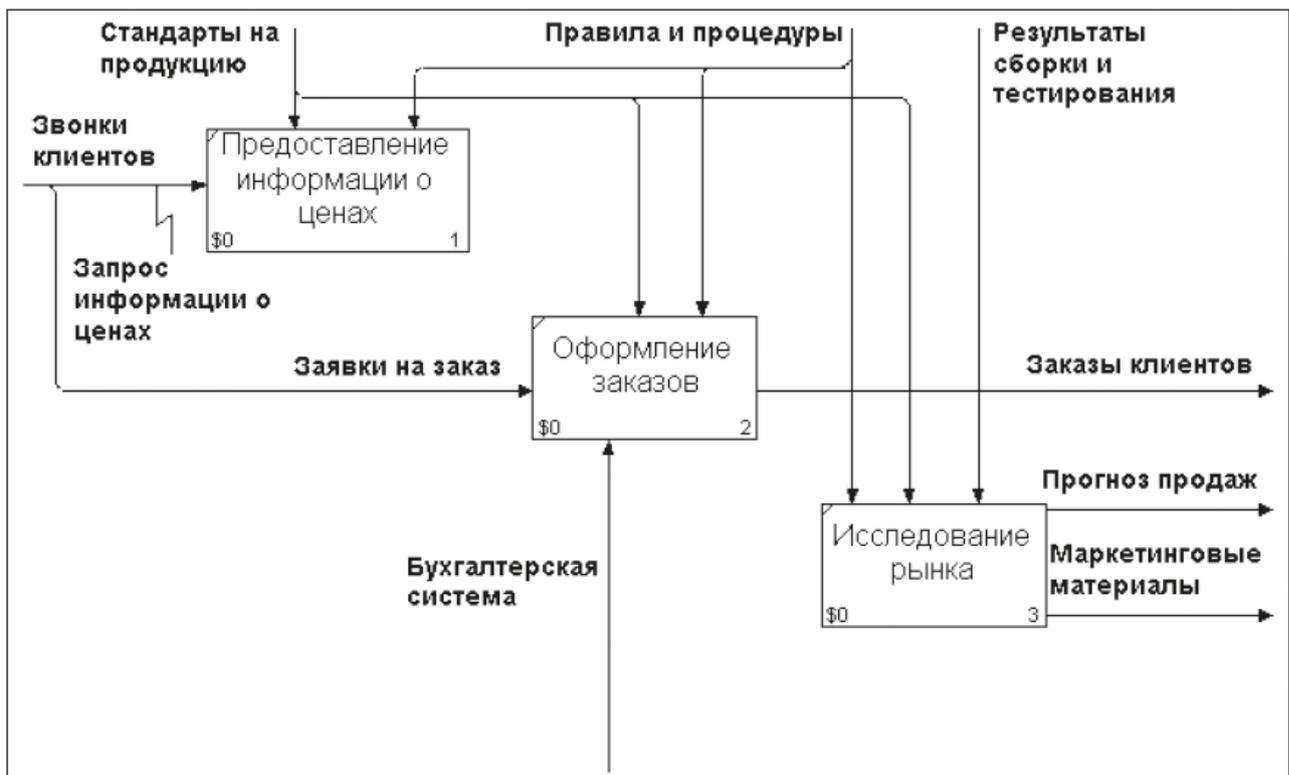


Рис. 83 – Результат выполнения

В процессе декомпозиции согласно правилам DFD необходимо преобразовать граничные стрелки во внутренние, начинающиеся и заканчивающиеся на внешних ссылках.

Декомпозируйте работу «Оформление заказов» на диаграмме A2.

В диалоге *Activity Box Count* выберите количество работ 2 и нотацию DFD. Щелкните по ОК и внесите в новую диаграмму DFD A22 имена работ:

- проверка и внесение клиента;
- внесение заказа.

Используя кнопку , внесите хранилища данных:

- список клиентов;
- список продуктов;
- список заказов.

Удалите граничные стрелки с диаграммы DFD A22. Используя кнопку  на палитре инструментов, внесите внешнюю ссылку «Звонки клиентов».

Создайте внутренние ссылки согласно рисунку 84. При именовании стрелок используйте словарь.

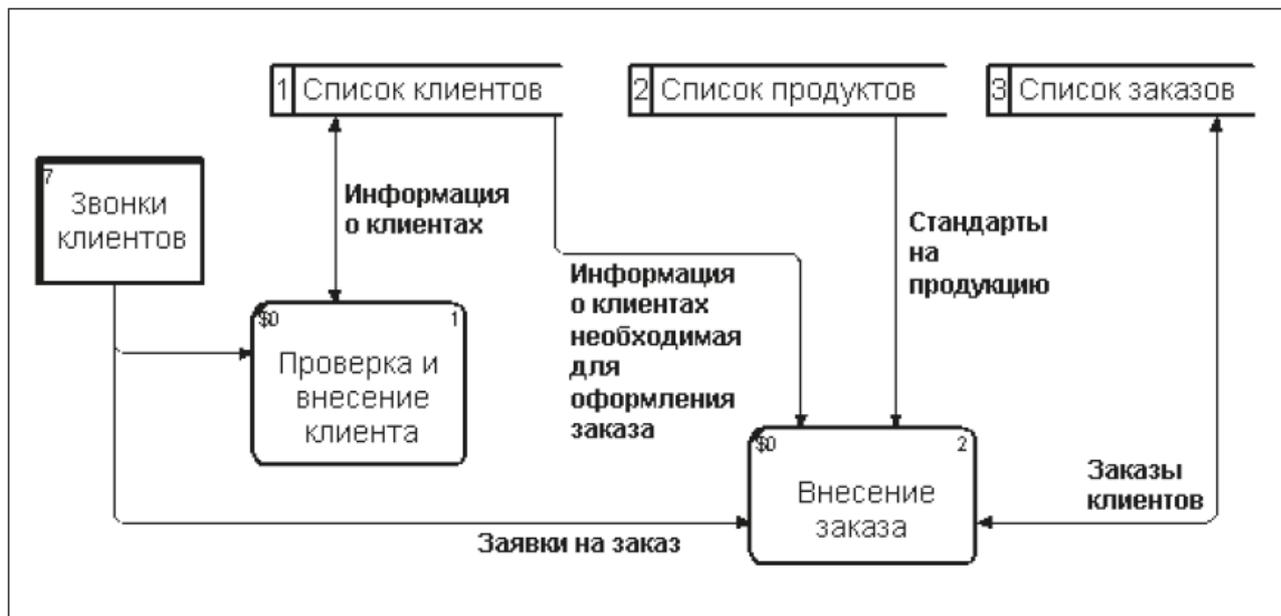


Рис. 84 – Диаграмма A22

Обратите внимание, что стрелки «Информация о клиентах» и «Заказы клиентов» двунаправленные. Для того чтобы сделать стрелку двунаправленной, щелкните правой кнопкой по стрелке, выберите в контекстном меню пункт *Style* и во вкладке *Style* выберите опцию *Bidirectional*.

На родительской диаграмме A2 туннелируйте (*Change to Tunnel*) стрелки, подходящие и исходящие из работы «Оформление заказов» (рис. 85).

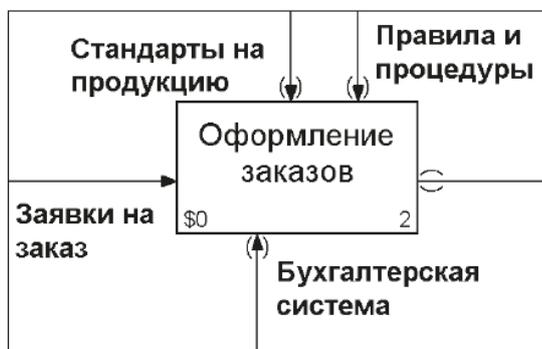


Рис. 85 – Работа «Оформление заказов» на диаграмме A2

Контрольные вопросы:

1. Что описывает диаграмма DFD?
2. Какая нотация используется в BPWin для построения диаграмм DFD?
3. Перечислите составные части диаграммы DFD.
4. Объясните механизм дополнения диаграммы IDEF0 диаграммой DFD.
5. Назовите преимущества методики DFD.
6. Что представляет собой «внешняя сущность»?
7. Чем являются словари данных?
8. Перечислите недостатки модели DFD.
9. Каким должно быть имя хранилища?

6 Создание модели бизнес-архитектуры предприятия категории «To-Be» (реинжиниринг бизнес-процессов)

Бизнес-процесс состоит из бизнес-операций и характеризуется определенным во времени началом и концом, интерфейсом для взаимодействия с другими процессами.

Современные достижения в области информационных систем и технологий дают возможность проведения *инжиниринга* и *реинжиниринга* бизнес-процессов.

Под **бизнес-инжинирингом** понимается выполнение комплекса проектировочных работ по разработке методов и процедур управления бизнесом, когда без изменения принятой организационной структуры управления в организации достигается улучшение его финансового положения.

Реализация инжиниринга строится на компьютеризации рабочих мест специалистов и применении ими для проектирования бизнеса ряда методик:

- выделение пошаговых процедур проектируемого процесса;
- внедрение описывающей процедуры системы обозначений;
- использование прагматических решений, позволяющих описывать степень соответствия спроектированного варианта бизнеса заданным целям.

Целью **реинжиниринга** является системная реорганизация материальных, финансовых и информационных потоков, направленная на упрощение организационной структуры, перераспределение и минимизацию использования различных ресурсов, сокращения сроков реализации потребностей клиентов, повышение качества их обслуживания.

Реинжиниринг бизнес-процессов – это создание новых, более эффективных бизнес-процессов без учета предшествующего развития.

Технология реинжиниринга основана на том, что в процессе управления пользователь активно использует современные информационные технологии для обучения, стратегического и тактического планирования, анализа возможных путей перестройки и улучшения бизнес-процессов, управления изменениями, реализацию проектов и др.

Для реализации принципов реинжиниринга бизнес-процессов в программной среде Allfusion Process Modeler (BPWin) предусмотрена возможность построения моделей «То-Ве», которые позволяют осуществить переход от действующей архитектуры предприятия (модели «As-Is») к ее модернизации и проектированию на новом этапе развития информационных технологий в соответствии с современными потребностями бизнеса.

Пример. В результате анализа принимается решение реорганизовать функции производства и тестирования компьютеров, функциональности «Продажи и маркетинг» и «Отгрузка и получение» оставить пока без изменений. Принято решение сформировать *отдел дизайна*, который должен:

- формировать конфигурации компьютеров;
- разрабатывать корпоративные стандарты;
- подбирать приемлемых поставщиков;
- разрабатывать инструкции по сборке, процедуры тестирования и устранения неполадок для всего производственного отдела.

Работа «Сборка и тестирование компьютеров» должна быть реорганизована и названа «Производство продукта». Будут созданы работы

«Разработать конфигурацию», «Планировать производство» и «Собрать продукт».

Предлагаемые изменения обусловили изменения в ролях и выполняемых функциях персонала рассматриваемого предприятия:

– дизайнер должен разрабатывать систему, стандарты на продукцию, документировать и передавать спецификации в отдел маркетинга и продаж. Он должен определять, какие компоненты (аппаратные и программные) должны закупаться для сборки компьютеров, обеспечивать документацией и управлять процедурами сборки, тестирования и устранения неполадок;

– функции диспетчера в работе «Сборка и тестирование компьютеров» должны быть заменены функциями планировщика;

– планировщик должен обрабатывать заказы клиентов и генерировать заказы на сборку, получать коммерческий прогноз из отдела маркетинга и формировать требования на закупку компонентов, и собирать информацию от поставщиков;

– диспетчер должен составлять расписание производства на основании заказов на сборку, полученных в результате работы «Планировать производство», получать копии заказов клиентов и отвечать за упаковку и комплектацию заказанных компьютеров, передаваемых в работу «Отгрузка и получение».

Практическая реализация. На первом этапе построения модели необходимо провести расщепление модели, после чего становится возможна ее модификация.

Измените свойства модели «Деятельность компании»:

– *Model Name* – Предлагаемая модель компании;

– *Time Frame* – To-Be;

– *Purpose* – документировать предлагаемые изменения бизнес-процессов компании.

Переименуйте работу «Сборка и тестирование компьютеров» в «Производство продукта». Расщепите эту работу в модель с тем же названием.

Модифицируйте отщепленную модель. Переместите работу «Тестирование компьютеров» с диаграммы А0 «Производство продукта» на диаграмму А2.1 «Сборка настольных компьютеров».

Переименуйте работу «Сборка настольных компьютеров» на диаграмме А0 в «Сборку продукта».

Удалите работу «Сборка ноутбуков».

Переименуйте стрелку «Заказы на настольные компьютеры» в «Заказы на изготовление».

Переименуйте работу «Отслеживание расписания и управление сборкой и тестированием» в «Планирование производства».

Создайте работу «Разработать конфигурацию».

Создайте ветвь стрелки «Персонал производственного отдела», назовите ее «Дизайнер» и направьте как механизм к работе «Разработать конфигурацию».

Создайте стрелку «Стандарты на продукцию» и направьте ее от выхода работы «Разработать конфигурацию» к границе диаграммы. Туннелируйте эту стрелку (*Resolve Border Arrow*). Создайте ветвь этой стрелки, идущую к управлению работы «Планирование производства» и назовите ее «Список необходимых компонентов».

Удалите стрелку «Правила сборки и тестирования». Создайте ветвь стрелки «Стандарты на продукцию», идущую к управлению работы «Сборка продукта» и назовите ее «Правила сборки и тестирования».

Переименуйте стрелку «Диспетчер» в «Планировщика производства». Добавьте стрелку «Прогноз продаж» как граничную управляющую к работе «Планирование производства».

Добавьте стрелку «Информация от поставщика» как граничную управляющую к работе «Планирование производства».

Добавьте стрелку «Заказ поставщику» как граничную стрелку выхода от работы «Планирование производства». Туннелируйте эти стрелки (*Resolve Border Arrow*).

На диаграмме А-0 туннелируйте стрелку (*Resolve Border Arrow*) «Собранные компьютеры» и свяжите ее на диаграмме А0 с выходом работы «Сборка продукта».

Результат выполнения перечисленных итераций представлен на рисунках 86 и 87.



Рис. 86 – Результат выполнения модификации – диаграмма А0

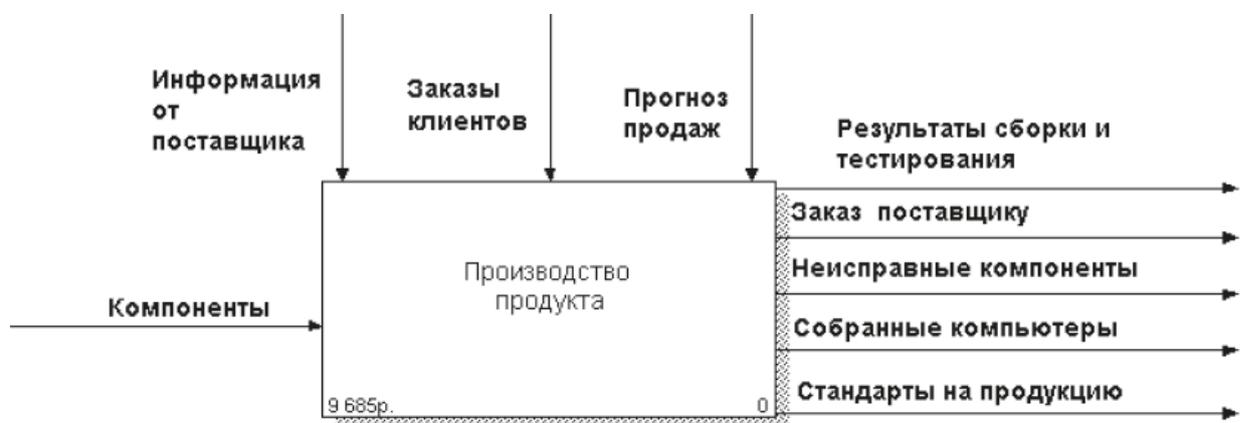


Рис. 87 – Результат выполнения модификации – диаграмма А-0

Слияние модели

Перейдите к работе «Производство продукта» в модели «Деятельность компании». Вызовите контекстное меню, щелкнув правой кнопкой мыши и выберите опцию *Merge Model*. В появившемся диалоге Merge Model установите опцию *Cut/Paste entire dictionaries*, опцию *Overwrite existing fields* и щелкните по ОК. Модели должны слиться.

На диаграмме A0 туннелируйте стрелки (*Resolve Border Arrow*) «Информация от поставщика» и «Заказ поставщику».

Направьте стрелку «Прогноз продаж» с выхода работы «Продажи и маркетинг» на управление работы «Производство продукта».

Направьте стрелку «Стандарты на продукцию» с выхода работы «Производство продукта» на управление работы «Продажи и маркетинг».

Удалите ветвь стрелки управления «Правила и процедуры» работы «Производство продукта». Закройте модель «Производство продукта». Результат выполнения выполненных итераций представлен на рис. 88 и 89.



Рис. 88 – Результат изменений на диаграмме A-0



Рис. 89 – Результат изменений на диаграмме А0

Использование Model Explorer для реорганизации дерева декомпозиции

Существуют причины, по которым работа «Разработать конфигурацию» должна быть на верхнем уровне, на диаграмме А0. Действительно, дизайнер разрабатывает стандарты на продукцию, включая правила сборки и тестирования, и список необходимых для закупки компонентов. Тем самым дизайнер управляет производством продукта в целом, кроме того, управляет работой «Продажи и маркетинг».

Было бы логично перенести эту работу на уровень выше.

Используя возможности *Model Explorer*, перенесите работу «Разработать конфигурацию» с диаграммы А2 «Производство продукта» на диаграмму А0.

Разрешите и перенаправьте стрелки. Результат изменения диаграммы А0 и А3 представлен на рисунках 90 и 91, соответственно.

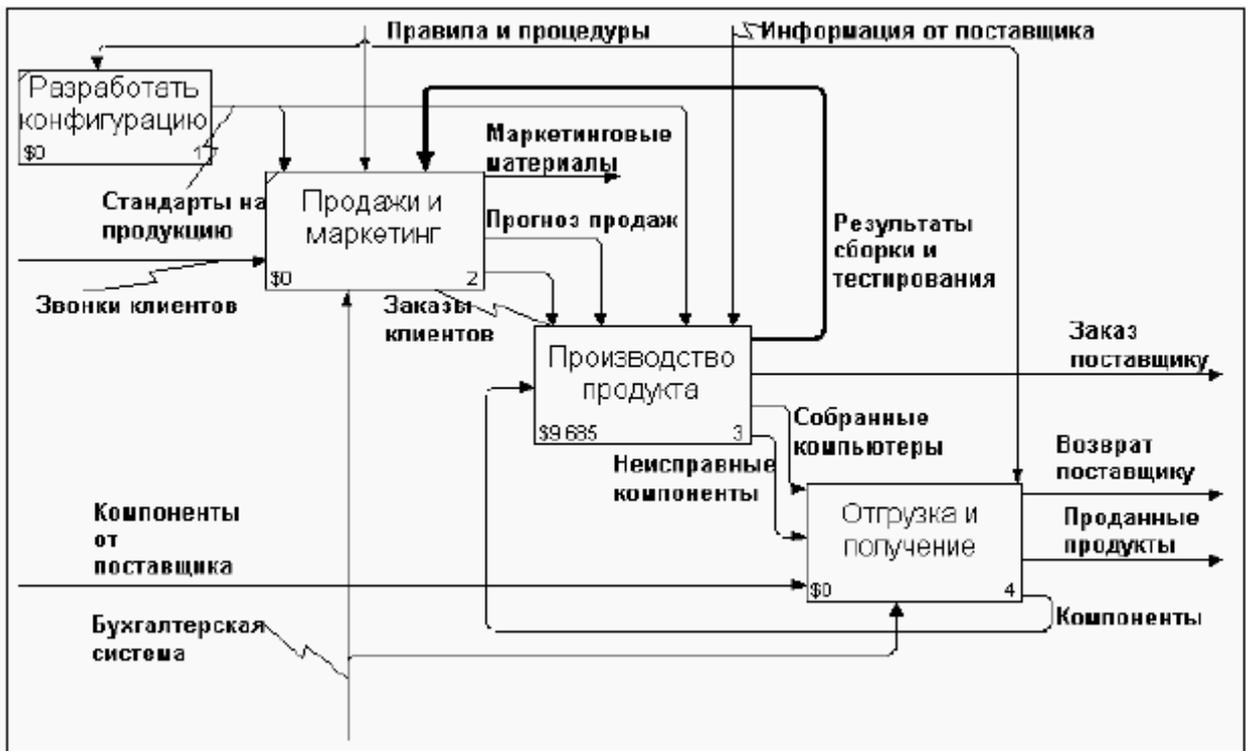


Рис. 90 – Результат выполнения итераций на диаграмме А0

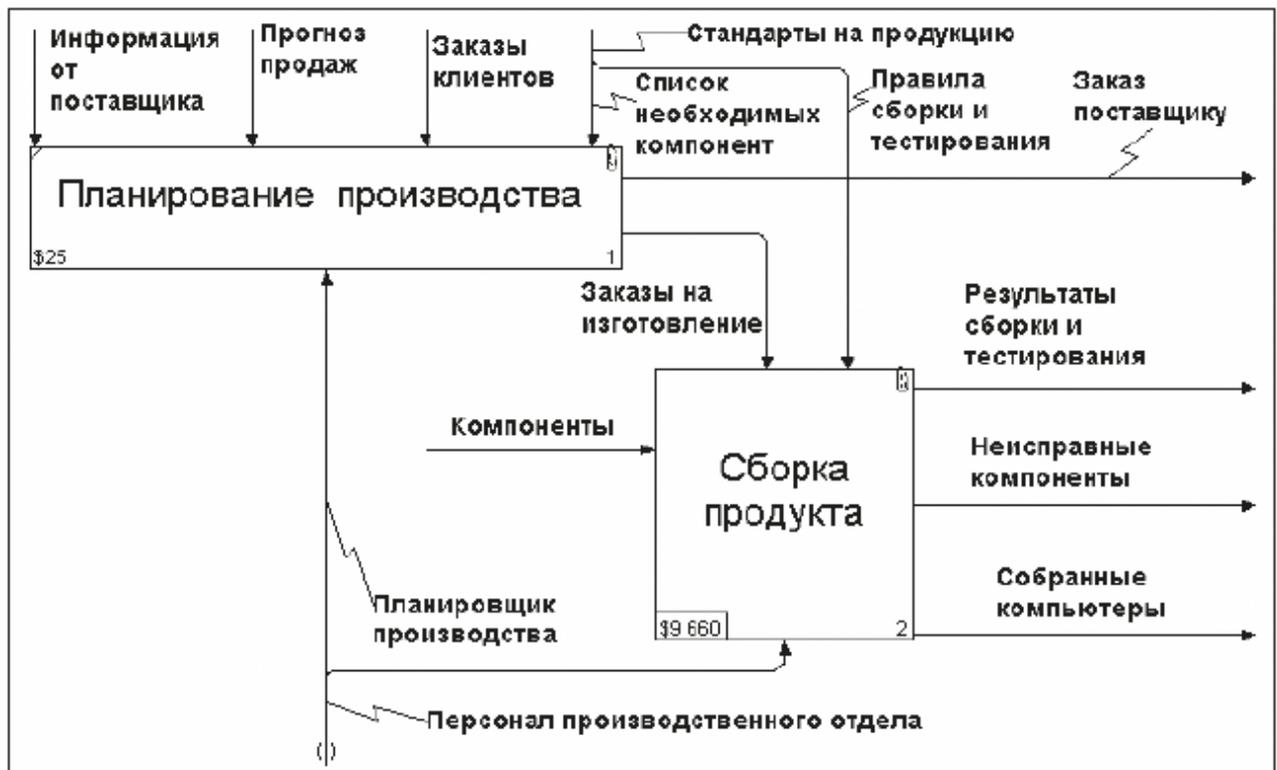


Рис. 91 – Результат выполнения итераций на диаграмме А3

Модификация диаграммы IDEF3 «Сборка продукта» с целью отображения новой информации

Так же как в модели «As-Is», сборка продукта состоит из сборки компонентов и установки программного обеспечения. Однако теперь в работу «Сборка продукта» включена работа «Тестирование компьютера».

Тестирование начинается после окончания процесса сборки компьютера и окончания процесса установки программного обеспечения. Если компьютер неисправен, в процессе тестирования у него заменяют компоненты, информация о неисправных компонентах может быть направлена на работу «Подготовка компонентов». Такая информация может помочь более тщательно подготавливать компоненты к сборке. Результатом процесса тестирования являются заказанные компьютеры и неисправные компоненты.

Модифицируйте диаграмму IDEF3 «Сборка продукта» в соответствии с приведенной информацией. Результат приведен на рисунке 92.

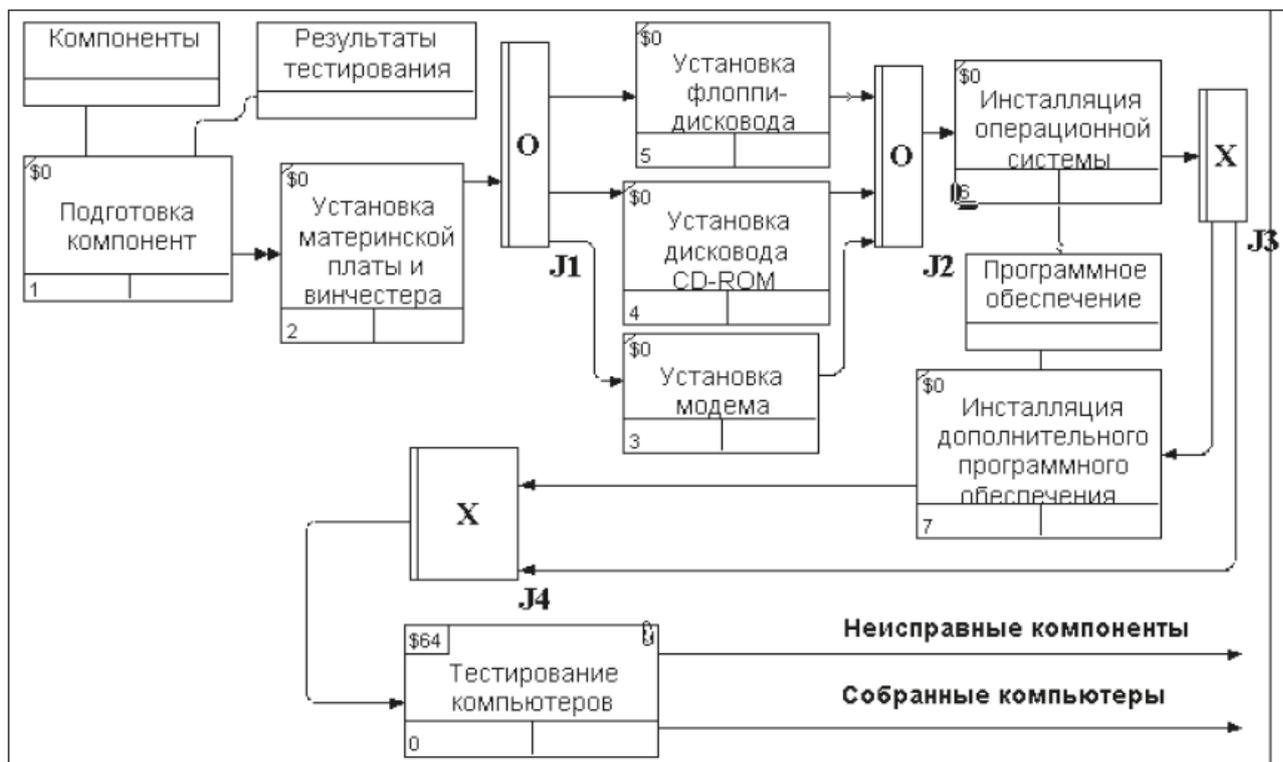


Рис. 92 – Результат модификации диаграммы А32.1

Контрольные вопросы:

1. Дайте определение понятия «инжиниринг бизнес-процессов».
2. Перечислите основные процедуры бизнес-инжиниринга.
3. Дайте определение понятия «реинжиниринг».
4. В чем заключается отличие инжиниринга от реинжиниринга бизнес-процессов.
5. В чем заключается цель перехода от модели «As-Is» в «To-Be» в процессе разработки архитектуры предприятия?
6. Перечислите этапы создания модели «To-Be» в среде Allfusion Process Modeler (BPWin).
7. Какие диаграммы можно преобразовать из модели «As-Is» в «To-Be» в среде Allfusion Process Modeler (BPWin)?
8. Использование Model Explorer для реорганизации дерева декомпозиции.
9. Раскройте цель и этапы модификации диаграммы IDEF3.

СПИСОК ЛИТЕРАТУРЫ

1. Любушин, Н. П. Архитектура предприятия: учебник / Н.П. Любушин, В. Ю. Карпычев, Н. Э. Бабичева; под общ. ред. Д.А. Ендовицкого. – Москва: КНОРУС, 2018. – 354 с.
2. Дязитдинова, А. Р. Архитектура предприятия как информационная основа корпоративной структуры телекоммуникационной компании: [Электронный ресурс] / А.Р. Дязитдинова, Е.А. Матвеева, О.Н. Ольховая // URL: <http://lab18.ipu.ru/projects/conf2012/1/5.htm>
3. Зиндер, Е. З. Архитектура предприятия в контексте бизнес-реинжиниринга. Ч. 1 / Е. З. Зиндер // Intelligent Enterprise. 2008. – № 4.
4. Кудрявцев, Д. В. Технологии бизнес-инжиниринга: учеб. пособие / Д. В. Кудрявцев, М. Ю. Арзуманян, Л. Ю. Григорьев. – СПб.: Изд-во Политехн. ун-та, 2014. – 427 с.
5. Когаловский, В. Происхождение ERP / В. Когаловский // Директор информационных систем. 2000. № 5. – С. 91.
6. Богомолова, М. А. Архитектура предприятия: учебное пособие / М. А. Богомолова. – Самара: ПГУТИ, 2016. – 155 с.
7. Сайт Всемирной организации корпоративной архитектуры GEAO. – URL: www.geao.org
8. Гриценко, Ю. Б. Архитектура предприятия: учеб. пособие / Ю.Б. Гриценко. – Томск: Томский государственный университет систем управления и радиоэлектроники, 2014. – 260 с.
9. Галактионов, В. Системная архитектура и ее место в архитектуре предприятия / В. Галактионов // Директор ИС. – 2002. – № 1(05.02.2002). – URL: <http://www.osp.ru/cio/2002/05/>
10. Данилин, А. Г. Архитектура и стратегия. «Инь» и «Янь» информационных технологий предприятия / А. Г. Данилин, А.В. Слюсаренко. – М.: Интернет ун-т инф. технологий, 2005. – 504 с.

11. Современный экономический словарь / Б. А. Райзберг [и др.]. – 5-е изд., перераб. и доп. – М.: ИНФРА-М, 2007. – 495 с.
12. Виханский, О.С. Стратегическое управление: учебник / О. С. Виханский. – 2-е изд., перераб. и доп. – М.: Гардарика, 1998. – 296 с.
13. Васильев, Р.Б. Управление развитием информационных систем: учеб. пособие / Р. Б. Васильев, Г. Н. Калянов, Г. А. Левочкина ; под ред. Г. Н. Калянова. – М.: Горячая линия-Телеком, 2009. – 376 с.
14. Быкова, А. Организационные структуры управления / А. Быкова. – М.: ОЛМА-ПРЕСС Инвест, 2003. – 160 с.
15. Григорьев, Л. Ю. Менеджмент по нотам. Технология построения эффективных компаний. / Л. Ю. Григорьев, С. Л. Горелик, Д. В. Кудрявцев. – М.: Альпина Паблишерз, 2010. – 692 с.
16. Кудрявцев, Д. В. Архитектура предприятия: переход от проектирования ИТ-инфраструктуры к трансформации бизнеса / Д.В. Кудрявцев, М.Ю. Арзуманян // Российский журнал менеджмента (Russian Management Journal). – 2017. – Том 15. – № 2. – С.193–224.
17. Всяких, Е. И. Практика и проблематика моделирования бизнес-процессов / Е. И. Всяких, А. Г. Зуева, Б. В. Носков, С. П. Киселев, Е. В. Сидоренко, А. И. Слюсаренко, И. А. Треско (общая редакция). – М.: ДМК Пресс; М.: Компания АйТи, 2008. – 246 с.
18. Калянов, Г. Н. Архитектура предприятия и инструменты ее моделирования / Г. Н. Калянов // Автоматизация в промышленности. – № 7. – 2004. – С. 9–12.
19. Тельнов, Ю. Ф. Инжиниринг предприятия и управление бизнес-процессами. Методология и технология / Ю. Ф. Тельнов, И. Г. Фёдоров. – М.: Юнити-Дана, 2015.
20. Carbone J. IT Architecture Toolkit. – Prentice Hall PTR, 2004. – 256 p.
21. Сайт консорциума OMG. – URL: <http://www.omg.org/>
22. Zachman J.A. Framework for Information System Architecture // IBM System Journal. – 1987. – vol. 26. – no. 3. – pp. 276–292.

23. Elizabeth, N. F. Information Management Directions: The Integration Challenge / N. F. Elizabeth, A. H. Goldfine // National Institute of Standards and Technology (NIST). – Special Publication. – 1989. – 167 p.
24. Spewak S. H., Steven C. H. Enterprise Architecture Planning: Developing a Blueprint for Data, Application and Technology / S. H. Spewak, C. H. Steven. – NY: A Wiley-QED Publication. – 1 ed., 1992. – 392 p.
25. Сайт компании ISACA. – URL: <https://www.isaca.org/Pages/default.aspx>
26. Сешнс, Р. Сравнение четырех ведущих методологий построения архитектуры предприятия. // Библиотека MSDN. – URL: <https://msdn.microsoft.com/ru-ru/library/ee914379.aspx>
27. Акт Клингера-Коэна 1996. // Библиотека Конгресса. – URL: <http://www.gpo.gov/fdsys/pkg/PLAW-104publ106/pdf/PLAW-104publ106.pdf>
28. GERAM: Generalised Enterprise Reference Architecture and Methodology: Version 1.6.3 1999. – URL: <http://www.cit.gu.edu.au/>
29. Йылмаз, О. А. Архитектура предприятия: учеб. пособие / О. А. Йылмаз. М.: Финансы и статистика, 2015. – 122 с.
30. Сайт корпорации The Open Group. – URL: <http://www.opengroup.org>
31. Зиндер, Е. З. «3D-предприятие» – модель трансформирующейся системы / Е. З. Зиндер // Электронный ресурс. – URL: http://iteam.ru/publications/it/section_53/article_1272
32. Сайт компании Gartner. – URL: <https://www.gartner.com/en>
33. Kruchten Ph. Architectural Blueprints «The 4+1 View Model of Software Architecture» // IEEE Software. – November 1995. – N. 12 (6). – P. 42-50.
34. Штейнгатт Е.А. Обзор и сравнительная характеристика методологий разработки архитектуры предприятий / Е.А. Штейнгатт, А.Н. Бурмистров. – Научно-технические ведомости СПбГПУ. Экономические науки. – № 3(245). – 2016. – С. 111-129.
35. Teale Ph., Jarvis R. Business Patterns for Software Engineering Use. P. 1 // MSDN: [Site]. 2004. – URL: <http://msdn.microsoft.com/ruru/library/aa480022.aspx>.

36. Методики фирмы Microsoft // Microsoft Architecture Journal: электронный журнал. – URL: <http://msdn.microsoft.com/architecture/journ/>.
37. COBIT 5: Бизнес-модель по руководству и управлению ИТ на предприятии. – М.: Московское подразделение ISACA®, 2012. – 94 с.
38. ГОСТ Р ИСО/МЭК 38500-2017 Информационные технологии (ИТ). Стратегическое управление ИТ в организации. – М.: Стандартинформ, 2019. – URL: <http://docs.cntd.ru/document/1200146826>
39. Черемных, С. В. Моделирование и анализ систем. IDEF-технологии: практикум / С. В. Черемных, И. О. Семенов, В. С. Ручкин. – М.: Финансы и статистика, 2006. – 192 с.
40. Калянов, Г. Н. Моделирование, анализ, реорганизация и автоматизация бизнес-процессов: учеб. пособие / Г. Н. Калянов. – М.: Финансы и статистика, 2006. – 240 с.
41. Молоткова, Н. В. Проектирование и реинжиниринг бизнес-процессов в коммерции и бизнес-информатике: учеб. пособие / Н. В. Молоткова, Д.Л. Хазанова, Т.И. Лапина. – Тамбов: ТГТУ, 2013. – 173 с.
42. Ануфриев, К. П. Практика и проблематика моделирования бизнес-процессов / К. П. Ануфриев. – М.: ДМК Пресс; АйТи, 2008. – 246 с.
43. Белов, А. В. Моделирование бизнес-процессов: учебно-методическое пособие / А. В. Белов, М. И. Нежурина, М. О. Утицких. – М.: МФТИ, 2009. – 184 с.
44. Лелюк, В.А. Совершенствование бизнес-систем. Методы, инструментарий, опыт: учеб. пособие / В. А. Лелюк, А. В. Лелюк, Харк. нац. акад. гор. хоз-ва. – Х.: ХНАГХ, 2011. – 438 с.
45. Брезгин, В. И. Моделирование бизнес процессов с AllFusion Process Modeler 4.1: лабораторный практикум. Ч. 2 / В. И. Брезгин. – Екатеринбург: Изд-во Урал. ун-та, 2015. – 52 с.
46. Цуканова, О. А. Методология и инструментарий моделирования бизнес-процессов: учеб. пособие / О. А. Цуканова. – СПб.: Университет ИТМО, 2015. – 100 с.

Гуменюк Наталья Владимировна

АРХИТЕКТУРА ПРЕДПРИЯТИЯ

Учебное пособие
для обучающихся образовательных учреждений
высшего профессионального образования
(на русском языке)

Ответственная за выпуск Н. Ф. Курган
Техническое редактирование и корректура Т. В. Чубучной

Подписано к печати 02.03.2020 г.
Формат 70×90/16. Бумага офисная.
Гарнитура «Times New». Печать – лазерная.
Усл. печ листов 17,67.
Тираж 100 экз. Заказ № 48.

ГОУВПО «ДОННТУ»
83001, ДНР, г. Донецк, ул. Артема, 58. Тел.: (062)301-03-04

Отпечатано в Автомобильно-дорожном институте ГОУВПО «ДОННТУ»
84646, г. Горловка, ул. Кирова, 51

Свидетельство о государственной регистрации ДНР
Серия АА03 № 029192 от 7 апреля 2016 г.