Владимирский государственный университет

Н. В. АБДУЛЛАЕВ И. Ю. КУЛИКОВА Н. В. МУРАВЬЕВА

АУДИТ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ КОМПАНИИ

Учебное пособие

Владимир 2026

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Н. В. АБДУЛЛАЕВ И. Ю. КУЛИКОВА Н. В. МУРАВЬЕВА

АУДИТ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ КОМПАНИИ

Учебное пособие

Электронное издание

Владимир 2026

ISBN 978-5-9984-2143-3

УДК 004.7:658.012.4:338.2(075.8)
ББК 65.290-93я73

Авторы: Н. В. Абдуллаев, И. Ю. Куликова, Н. В. Муравьева

Рецензенты:
Доктор экономических наук, доцент
зав. кафедрой менеджмента и маркетинга
Владимирского государственного университета
имени Александра Григорьевича и Николая Григорьевича Столетовых»
Н. Н. Ползунова

Кандидат экономических наук, доцент
зав. кафедрой экономики и финансов
Финансового университета при Правительстве Российской Федерации
(Владимирский филиал)
Д. В. Кузнецов

Абдуллаев, Н. В. Аудит цифровой инфраструктуры компании [Электронный ресурс] : учеб. пособие / Н. В. Абдуллаев, И. Ю. Куликова, Н. В. Муравьева ; Владим. гос. ун-т им. А. Г. и Н. Г. Столетовых. – Владимир : Изд-во ВлГУ, 2026. – 309 с. − ISBN 978-5-9984-2143-3. − Электрон. дан. (3,84 Мб). – 1 электрон. опт. диск (CD-ROM). – Систем. требования: Intel от 1,3 ГГц ; Windows XP/7/8/10 ; Adobe Reader ; дисковод CD-ROM. – Загл. с титул. экрана.

Содержит необходимый теоретический и справочный материал для изучения дисциплин «Информационная инфраструктура предприятия», «Аудит цифровой инфраструктуры», «Аудит информационных систем», а также других дисциплин, имеющих отношение к экономике и управлению информационными технологиями. В пособии рассматриваются эволюция, компонентный состав и архитектурные модели цифровой инфраструктуры, стратегии ее развития и выравнивания с бизнес-целями. Особое внимание уделяется вопросам обеспечения производительности, доступности и безопасности. Значительная часть издания посвящена методологии проведения аудита цифровой инфраструктуры, включая цели, виды, стандарты, этапы и особенности аудита облачных сред.

Предназначено для обучающихся по направлениям подготовки 02.03.03 − Математическое обеспечение и администрирование информационных систем, 38.03.05, 38.04.05 − Бизнес-информатика всех форм обучения, а также может быть полезно для аспирантов и преподавателей экономических и ИТ-направлений, практикующих специалистов в области управления информационными технологиями и внутреннего аудита.

Рекомендовано для формирования универсальных компетенций в соответствии с ФГОС ВО.

Табл. 61. Ил. 24. Библиогр.: 56 назв.

ISBN 978-5-9984-2143-3

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Современный этап развития экономики характеризуется углубляющейся цифровой трансформацией бизнеса, в рамках которой информационные технологии перестают выполнять исключительно вспомогательную функцию и приобретают статус стратегического актива, определяющего конкурентоспособность организаций. Цифровая инфраструктура компании, представляющая собой совокупность аппаратных, программных, сетевых и организационных ресурсов, выступает фундаментальной основой для реализации бизнес-процессов, хранения и обработки данных, а также предоставления цифровых сервисов как внутренним, так и внешним потребителям. В этих условиях эффективность управления цифровой инфраструктурой напрямую влияет на достижение стратегических целей организации, ее операционную устойчивость и способность адаптироваться к динамично меняющимся требованиям рынка.

Сложность и гетерогенность современных инфраструктурных решений, обусловленные внедрением облачных технологий, контейнеризации, микросервисной архитектуры, а также ростом объема обрабатываемых данных, предъявляют качественно новые требования к компетенциям специалистов, отвечающих за проектирование, эксплуатацию и оценку состояния ИТ-ландшафта предприятия. Особое значение приобретает аудит цифровой инфраструктуры как инструмент объективной оценки ее текущего состояния, выявления узких мест, оценки соответствия установленным стандартам и нормативным требованиям, а также формирования обоснованных рекомендаций по повышению эффективности ее функционирования. Системный подход к управлению и аудиту инфраструктуры позволяет не только минимизировать риски простоев и нарушений информационной безопасности, но и оптимизировать затраты на владение ИТ-активами.

Учебное пособие предназначено для формирования у обучающихся целостного представления о теоретических основах и прикладных аспектах управления и аудита цифровой инфраструктуры современного предприятия. Оно разработано с учетом формирования у студентов универсальных компетенций в соответствии с ФГОС ВО, в результате чего обучающиеся должны:

– *знать* теоретические и методологические основы аудита цифровой инфраструктуры, архитектурные, технологические компоненты цифровой инфраструктуры современной компании, а также экономические аспекты управления ее ключевыми компонентами;

– *уметь* планировать и проводить аудиторские процедуры в отношении цифровой инфраструктуры, оценивать ее соответствие критериям надежности, безопасности и эффективности, а также анализировать вклад цифровой инфраструктуры в реализацию бизнес-стратегии и платформенных моделей;

– *владеть* инструментарием аудита и мониторинга цифровой инфраструктуры компании, а также практическими навыками коммуникации и консалтинга для взаимодействия с техническими и бизнес-заказчиками в части обоснования предложений по оптимизации затрат, модернизации инфраструктуры и развитию ее платформенного потенциала в рамках стратегии компании.

В пособии последовательно раскрываются понятие, компоненты и эволюция цифровой инфраструктуры, особенности управления вычислительными, сетевыми и программными ресурсами, методология обеспечения производительности, доступности и безопасности. Значительное внимание уделяется сервисному подходу к управлению ИТ на основе библиотеки ITIL 4, а также современным практикам интеграции DevOps-культуры. Две завершающие главы посвящены методологии проведения аудита цифровой инфраструктуры, включая цели, виды, стандарты, этапы аудита облачных сред, а также вопросам построения комплексной системы управления и оценки эффективности инфраструктуры.

Методическая ценность пособия заключается в сочетании теоретического материала с практическими примерами, расчетными заданиями, вопросами для обсуждения и тестами для самоконтроля. Такой подход позволяет не только усвоить ключевые концепции, но и развить навыки их практического применения.

Глава 1. ЦИФРОВАЯ ИНФРАСТРУКТУРА КАК ОСНОВА БИЗНЕСА

1.1. Понятие, компоненты и эволюция цифровой инфраструктуры

Цифровая инфраструктура представляет собой комплекс взаимосвязанных аппаратных, программных и сетевых компонентов, обеспечивающих создание, передачу, хранение, обработку и потребление цифровых данных и сервисов. Она составляет тот технологический фундамент, который в современной экономике превратился из вспомогательного актива в критически важный базис, обеспечивающий функционирование всех секторов экономики, государственного управления и социальной сферы. Однако любой компонент цифровой инфраструктуры связан с классической ИТ-инфраструктурой, под которой следует понимать совокупность информационных ресурсов и технологий, программного и аппаратного обеспечения, сетевых компонентов, носителей данных и сервисов организации, которые обеспечивают эксплуатацию и управление IT-средой компании. В отличие от традиционной ИТ-инфраструктуры, цифровая инфраструктура носит экосистемный и сквозной характер, выступая платформой для взаимодействия множества субъектов (бизнесов, государства и отдельных граждан) и основой для инновационных бизнес-моделей, таких как платформенная экономика, интернет вещей (далее IoT) и индустрия 4.0. Её ключевыми атрибутами являются масштабируемость, высокая доступность, безопасность, гибкость и способность обеспечивать сквозные процессы, непосредственно обеспечивающие такие процессы как эксплуатация, развитие и сопровождение (рис. 1.1).

Основная цель как традиционной ИТ-инфраструктуры, так и цифровой – это обеспечивать бесперебойность и безопасность работы информационных сервисов, обработку, хранение и передачу информации.

В настоящее время выделяют два подхода к определению цифровой инфраструктуры:

1. Комплекс инфраструктур, отвечающих за протекание процессов, в основу которых положены цифровые технологии.

2. Комплекс цифровых сервисов, инструментов и технологий, а также набор созданных с их помощью продуктов, обеспечивающих сетевые, телекоммуникационные и вычислительные мощности.[1]

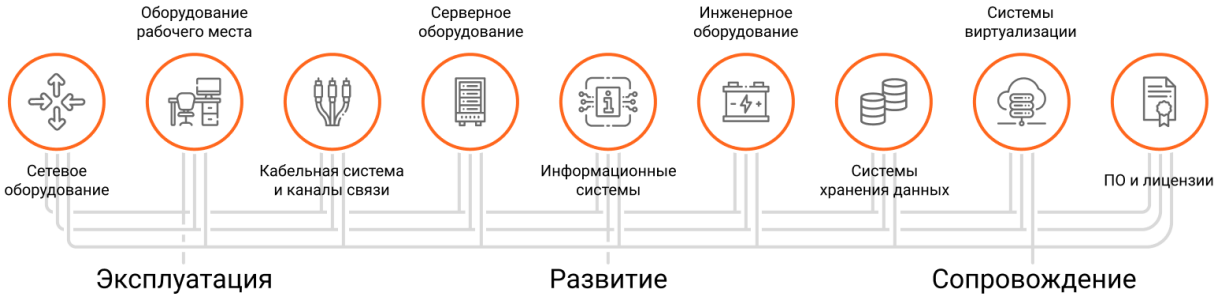

Рис. 1.1. Ключевые атрибуты и сопровождающие их процессы цифровой инфраструктуры

Следует отметить, что цифровая инфраструктура является многоуровневой системой. В настоящее время она имеет достаточно разнообразный состав, основные представители которой изображены ниже на рис 1.2.

Тем не менее, современная цифровая инфраструктуры может быть структурирована по ряду базовым компонентным группам (слоям):

1. Аппаратно-физический слой:

– Центры обработки данных (ЦОД): специализированные здания или помещения, где размещается серверное, сетевое и коммуникационное оборудование. Эволюционируют от локальных серверных комнат к гигантским облачным и гипермасштабируемым ЦОДам, характеризуемыми высочайшей энергоэффективностью и отказоустойчивостью.

– Сетевая инфраструктура: физические каналы передачи данных (волоконно-оптические линии связи, медные кабели, спутниковые каналы) и активное сетевое оборудование (маршрутизаторы, коммута-

[1] Хайруллина, А. Р. Цифровая инфраструктура как среда принятия управленческих решений в малом и среднем предпринимательстве / А. Р. Хайруллина // Экономика, предпринимательство и право. – 2021. – Т. 11, № 5. – С. 1151-1166. – DOI 10.18334/epp.11.5.112066

торы, шлюзы). Ключевое развитие - повсеместный переход на технологии 5G/6G и новейшие стандарты Wi-Fi для обеспечения высокой пропускной способности и низких задержек.

– Периферийные (граничные) вычисления (Edge Computing): распределённые микроЦОД/ЦХОДы и вычислительные узлы, размещаемые вблизи источников данных (например, на фабриках, в умных городах), что позволяет обрабатывать данные в реальном времени, снижая нагрузку на центральные облака и задержки.

– Клиентское оборудование: конечные устройства пользователей и систем (ПК, смартфоны, датчики IoT, промышленные контроллеры), которые являются точками входа и выхода данных.

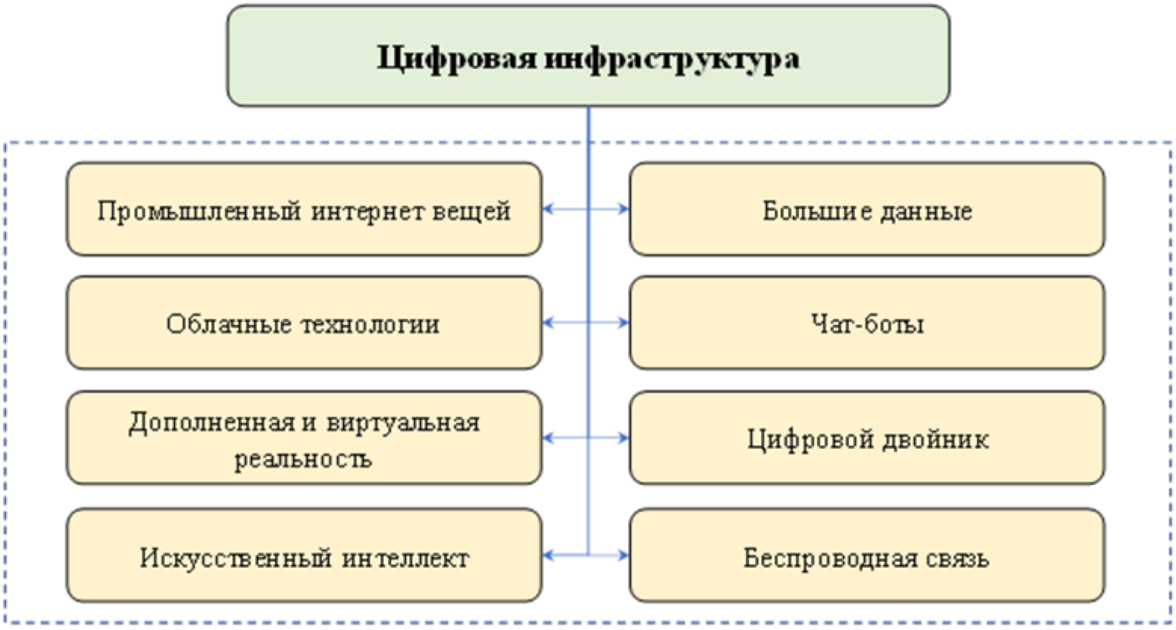

Рис. 1.2. Современный состав цифровой инфраструктуры[2]

2. Программно-абстрактный слой:

– Виртуализация и контейнеризация: технологии, абстрагирующие вычислительные ресурсы (серверы, сети, хранилища) от физического оборудования, позволяя создавать изолированные программные среды (виртуальные машины, контейнеры). Это основа гибкости и эффективного использования ресурсов.

[2] Пронин А.Ю. Основные тенденции развития цифровой инфраструктуры в интересах национальной экономики [Электронный ресурс]/ А.Ю. Пронин// Экономические исследования и разработки. – Режим доступа: http://edrj.ru/article/09-08-24 (дата обращения: 04.01.2026).

– Операционные системы и системы управления: специализированное программное обеспечение (далее ПО) для управления виртуализированными средами, кластерами и контейнерами, а также для оркестрации сложных распределённых приложений.

– Платформы (PaaS, aPaaS): среды, предоставляющие готовые инструменты для разработки, тестирования, развёртывания и управления приложениями, избавляя бизнес от необходимости управлять базовой инфраструктурой.

3. Слой сервисов и данных:

– Облачные сервисы (XaaS - «что-угодно как сервис»): модели предоставления услуг по требованию через интернет. Основные категории: IaaS (инфраструктура как сервис, например, аренда виртуальных серверов), PaaS (платформа как сервис), SaaS (программное обеспечение как сервис, например, современные CRM-системы).

– Системы хранения и управления данными: решения для структурированного (SQL-базы данных) и неструктурированного (Data Lakes («озера данных»), объектные хранилища) их типов, а также платформы для Big Data-аналитики, обеспечивающие извлечение знаний из больших массивов информации.

– Сервисы безопасности: комплексные решения, включая межсетевые экраны нового поколения (NGFW), системы предотвращения вторжений (IPS), средства шифрования и управления идентификацией и доступом (IAM), реализуемые как в виде аппаратных решений, так и облачных сервисов (SECaaS).

4. Интеграционный и коммуникационный слой:

– API (Application Programming Interface): стандартизированные интерфейсы, которые позволяют различным приложениям и сервисам обмениваться данными и функционалом, являясь «клеем» для создания цифровых экосистем.

– Платформы интеграции (iPaaS): специализированные среды для проектирования, выполнения и мониторинга потоков интеграции между разнородными локальными и облачными приложениями.

Современная цифровая инфраструктура всесьма сложное многоаспектное понятие, включающая ряд первостепенных слоев (табл. 1.1).

Таблица 1.1

Ключевые компоненты цифровой инфраструктуры и их функции

Слой/Группа	Ключевые компоненты	Основная функция
Аппаратно-физический	Центры обработки данных (ЦОД), волоконно-оптические сети, оборудование 5G, Edge-узлы, датчики IoT	Физическое размещение, передача и первичный сбор данных, обеспечение вычислительной мощности
Программно-абстрактный	Системы виртуализации и контейнеризации (Docker, Kubernetes), гипервизоры, ОС	Абстрагирование, изоляция и эффективное управление вычислительными ресурсами
Сервисов и данных	Облачные модели (IaaS, PaaS, SaaS), СУБД, Data Lakes, аналитические платформы (Hadoop, Spark)	Предоставление ИТ-ресурсов и инструментов по требованию, хранение и глубокая аналитика данных
Интеграционный	API (REST, GraphQL), платформы интеграции (iPaaS), сервисные шины (ESB)	Обеспечение взаимодействия и обмена данными между разнородными системами и сервисами

Следует заметить, что по уровню интеграции и современности цифровая инфраструктура может быть разделена на три вида:

1. Фрагментированная (разрозненная): создавалась стихийно, из несовместимых компонентов, приобретаемых под разные задачи в разное время. Слабая управляемость, высокие затраты на поддержку.

2. Конвергентная (Converged Infrastructure - CI). Представляет собой предварительно протестированный и подобранный комплект из серверов, СХД и сетевого оборудования от одного вендора (например, VxBlock от Dell EMC, Cisco UCS). Компоненты поставляются как единое решение.

Цель это вида инфраструктуры - упрощение развертывания и управления по сравнению с традиционной инфраструктурой.

3. Гиперконвергентная (Hyperconverged Infrastructure - HCI): представляет собой следующий шаг после CI. Все компоненты (вычисления, хранение, сети) виртуализованы и работают на стандартных серверах x86. Управление осуществляется через единый программный интерфейс. Программное обеспечение (например, VMware vSAN, Nutanix) абстрагирует все ресурсы и создает из пула серверов единый

вычислительный узел, ее несомненным плюсом является простота масштабирования (добавил сервер – получил сразу все ресурсы), высокая степень автоматизации, снижение TCO.

Инфраструктура, ориентированная на приложения и облачные сервисы (Cloud-Centric & Application-Led Infrastructure) Современная парадигма, где инфраструктура рассматривается не как набор железок, а как платформа для быстрого предоставления сервисов. Фокус смещается с управления физическими активами на управление рабочими нагрузками и приложениями. Эта модель тесно связана с облачными вычислениями и может реализовываться в различных формах:

- Частное облако (Private Cloud): Инфраструктура, предназначенная для исключительного использования одной организацией. Может размещаться on-premise (в собственном ЦОДе) или управляться внешним провайдером. Характеризуется самообслуживанием, эластичностью и учетом потребления ресурсов.

- Публичное облако (Public Cloud): Инфраструктура, предоставляемая сторонним провайдером (Amazon AWS, Microsoft Azure, Google GCP) в виде сервисов (IaaS, PaaS, SaaS) по модели pay-as-you-go («плати по факту использования»). Предприятие арендует виртуальные ресурсы, а провайдер отвечает за работу физического «железа».

- Гибридное облако (Hybrid Cloud): Модель, которая интегрирует частное и публичное облака, позволяя данным и приложениям портироваться между ними. Это обеспечивает гибкость, возможность сглаживания пиковых нагрузок за счет публичного облака и соблюдение требований к безопасности и регуляторики.

- Мультиоблако (Multi-Cloud): Стратегия использования услуг нескольких публичных облачных провайдеров одновременно для избежания вендор-локинга (зависимости от одного поставщика), выбора лучших сервисов и повышения отказоустойчивости.

Сравнительная таблица моделей цифровой инфраструктуры представлена в табл. 1.2.

Особого внимания заслуживает и эволюция цифровой инфраструктуры, которая носит нелинейный характер и представляет собой последовательность взаимодополняющих этапов, каждый из которых не отменяет предыдущий, а расширяет и трансформирует его.

Таблица 1.2

Сравнительная таблица моделей инфраструктуры

Критерий	Традиционная (On-Premise)	Конвергентная (CI)	Гиперконвергентная (HCI)	Облачная (IaaS)
Управление	Разрозненное, сложное	Единое, но для каждого компонента	Единое, централизованное	Полностью у провайдера
Масштабируемость	Сложно, дискретно (по компонентам)	Дискретно (блоками)	Просто, линейно (узлами)	Очень просто, мгновенно
Модель затрат	CAPEX (капитальные)	CAPEX	CAPEX (аппаратная часть)	OPEX (операционные)
Скорость развертывания	Месяцы	Недели	Дни	Минуты
Степень контроля	Полная	Высокая	Высокая	Ограниченная

Первый этап представляет централизованная мэйнфрейм-архитектура и сопутствующая ей и инфраструктура (1960-1980-е гг.). Характеризовался использованием крупных центральных компьютеров (мэйнфреймов) с терминальным доступом. Инфраструктура была монолитной, дорогостоящей и управлялась исключительно централизованно. Основная функция - автоматизация массовых транзакций (банковские операции, учет).

Второй этап заключается в появлении и в последствие широком использовании распределённых клиент-серверных систем, которые приходятся на 1980-1990-е гг. С появлением персональных компьютеров и локальных сетей (LAN) произошла децентрализация. Вычислительная нагрузка распределилась между серверами (хранение данных, бизнес-логика) и клиентскими рабочими станциями (интерфейс). Инфраструктура стала более гибкой, но сложной в управлении, породив

проблему «информационных silos» (изолированных хранилищ данных).

Третий этап приходится на эпоху Интернета и различных веб-технологий (конец 1990-х - 2000-е гг.). Глобальная сеть Интернет стала транспортной основой. Инфраструктура эволюционировала в сторону поддержки веб-приложений и первых облачных сервисов (SaaS). Появились крупные ЦОДы, предназначенные для хостинга. Ключевой тренд — начало консолидации ресурсов и предоставления услуг через сеть.

Сущность четвёртого этапа заключается в виртуализации и рождении облачных вычислений (конец 2000-х - 2010-е гг.). Технологии виртуализации позволили отказаться от жесткой привязки ПО к «железу», создав пулы абстрактных вычислительных ресурсов. Это привело к революционной модели облачных вычислений, где инфраструктура, платформы и ПО начали предоставляться как сервис по подписке (IaaS, PaaS, SaaS). Бизнес получил беспрецедентную масштабируемость, экономию на капитальных затратах (CAPEX) и переход к операционным расходам (OPEX).

Переход к пятому этапу позволил перейти к гибридной, мультиоблачной и периферийной инфраструктуре (2020-е гг. - настоящее время). Текущий этап определяется отказом от зависимости от единого провайдера и доминированием гибридных (комбинация приватных облаков и публичных сервисов) и мультиоблачных (одновременное использование услуг нескольких публичных облачных провайдеров) стратегий. Это позволяет оптимизировать стоимость, производительность и избежать vendor lock-in, т.е. «привязки к поставщику». Параллельно бурное развитие IoT, индустрии 4.0 и приложений, требующих минимальных задержек (автономный транспорт, дополненная реальность), стимулировало взрывной рост периферийных (граничных) вычислений, которые переносят обработку данных на «окраину» сети. Цифровая инфраструктура становится распределённой, интеллектуальной и контекстно-зависимой.

Сегодня общество оказывается на стадии шестого этапа, заключающегося в формировании следующего ключевого тренда: инфраструктура, управляемая искусственным интеллектом (AIOps) и квантовые вычисления. Эволюция продолжается в направлении полной автономии. AIOps (Artificial Intelligence for IT Operations) предполагает ис-

пользование машинного обучения и искусственного интеллекта для автоматического мониторинга, управления, устранения неисправностей и оптимизации инфраструктуры. В долгосрочной перспективе назревает конвергенция с квантовыми вычислениями, которые потенциально смогут решать задачи, непосильные для классической инфраструктуры (например, в области криптографии, молекулярного моделирования, оптимизации сложных систем).

Помимо эволюционного подхода, цифровую инфраструктуру компании или определенного вида бизнеса можно классифицировать и по другим признакам.

По масштабу и роли в компании она подразделяется на:[3]

- Инфраструктуру центрального офиса/корпоративного ЦОД: критичная, высокопроизводительная, отказоустойчивая среда для ключевых бизнес-приложений.

- Филиальную инфраструктуру (Branch Office): упрощенные решения для удаленных офисов, часто с минимальным локальным присутствием ИТ-персонала. Тренд — централизация управления и использование SD-WAN для подключения к центру.

- Периферийную инфраструктуру: распределенные микро-ЦОДы, размещаемые ближе к источникам данных (заводские цеха, розничные точки, устройства IoT). Необходима для обработки данных в реальном времени с минимальной задержкой (low latency).

По принципу управления и предоставления ресурсов классификация может быть следующей:

- Традиционная (аппаратная) инфраструктура: управление на уровне физических устройств.

- Программно-определяемая инфраструктура (Software-Defined Everything - SDx): абстрагирование функций инфраструктуры (сети - SDN, хранение - SDS, вычисления) от аппаратного уровня и централизованное управление ими через программное обеспечение. Это основа для современной гибкой и автоматизированной инфраструктуры.

- Инфраструктура как код (Infrastructure as Code - IaC): практика управления инфраструктурой через машиночитаемые файлы конфигу-

[3] Федоров А. Что такое ИТ-инфраструктура и из каких компонентов она состоит [Электронный ресурс]/ А. Федоров // Режим доступа: https://cyberprotect.ru/blog/it-infrastructure-intro?ysclid=mk1kblc7jg194849748 (дата обращения: 05.01.2026).

рации, а не через интерактивные инструменты ручного настройки. Позволяет автоматизировать, версионировать и повторять развертывание сложных сред.

В свою очередь, по степени критичности и требованиям к доступности цифровая инфраструктура может быть:

- Некритичной (Non-Mission-Critical): допускает простои (например, тестовые среды).

- Критичной (Mission-Critical): требует высокой доступности (High Availability - HA, 99.9% и выше), кластеризации и сложных решений по аварийному восстановлению (Disaster Recovery - DR).

Таким образом, цифровая инфраструктура представляет собой динамическую, многоуровневую систему, эволюционирующую от централизованных и жестких форм к распределённым, гибким и сервисно-ориентированным моделям. Её современное состояние характеризуется симбиозом облачных, периферийных и гибридных архитектур, где ключевыми драйверами развития являются потребности бизнеса в скорости, гибкости, экономической эффективности и способности поддерживать инновации. Понимание компонентного состава и логики эволюции цифровой инфраструктуры является обязательным условием для формирования эффективной цифровой стратегии любого современного предприятия, стремящегося к устойчивому развитию в условиях цифровой экономики.

1.2. Роль цифровой инфраструктуры в создании бизнес-ценности и цифровой трансформации

Цифровая инфраструктура перестала быть лишь вспомогательным технологическим активом, превратившись в фундаментальный стратегический ресурс, который непосредственно генерирует бизнес-ценность и является катализатором цифровой трансформации. Ее роль эволюционировала от обеспечения операционной эффективности до создания новых бизнес-моделей, конкурентных преимуществ и устойчивости в условиях неопределенности. Понимание этой роли требует анализа через призму многоуровневой архитектуры, которая обеспечивает переход от простой автоматизации к полномасштабной трансформации бизнес-процессов, продуктов и взаимодействия с клиентами.

1. Концептуальные основы: от затрат к стратегическим инвестициям. Традиционно расходы на ИТ-инфраструктуру рассматривались как операционные затраты, необходимые для поддержания работоспособности бизнеса. Современная цифровая инфраструктура (ЦИ) – это экосистема взаимосвязанных компонентов (вычислительные мощности, системы хранения данных, сети, платформенные решения, средства безопасности), управляемых преимущественно программным образом (Software-Defined). Ее ключевая характеристика – эластичность и гибкость, позволяющая масштабировать ресурсы в соответствии с динамикой бизнес-задач. Это превращает ЦИ из центра затрат в инвестиционную платформу, способность которой к быстрой адаптации и инновациям прямо коррелирует с создаваемой бизнес-ценностью. Ценность проявляется не только в экономии за счет оптимизации, но и в ускорении вывода продуктов на рынок (time-to-market), повышении качества обслуживания клиентов и открытии новых источников дохода.[4]

2. Механизмы генерации бизнес-ценности через цифровую инфраструктуру.

Создание бизнес-ценности происходит по нескольким взаимосвязанным направлениям, которые представлены ниже в табл.1.3.

Эти механизмы действуют синергетически. Например, облачная платформа (операционная эффективность) обеспечивает необходимое масштабирование для работы с большими данными (извлечение ценности), что, в свою очередь, позволяет запустить персонализированный сервис на основе микросервисов (ускорение инноваций).

3. Цифровая инфраструктура как драйвер и платформа цифровой трансформации

Цифровая трансформация (ЦТ) – это глубокая реорганизация бизнеса, направленная на кардинальное изменение операционных моделей и предложения ценности за счет цифровых технологий. ЦИ является не просто «техническим фундаментом» ЦТ, а ее активным драйвером и обязательным условием.

[4] Национальная цифровая инфраструктура. Выбираем модель управления [Электронный ресурс]// Режим доступа: https://plusworld.ru/journal/2022/plus-3-2022/natsionalnaya-tsifrovaya-infrastruktura-vybiraem-model-upravleniya/nalichnoe-denezhnoe-obrashchenie/?ysclid=mk1eoly9v78168091 (дата обращения: 05.01.2026).

Таблица 1.3

Механизмы создания бизнес-ценности цифровой инфраструктурой

Механизм	Ключевые технологии и подходы	Формы создаваемой бизнес-ценности
Операционная эффективность и снижение затрат	Виртуализация, облачные вычисления (IaaS), автоматизация оркестрации, контейнеризация.	Снижение CAPEX/OPEX, оптимизация использования ресурсов, ускорение развертывания сервисов, снижение энергопотребления.
Устойчивость и непрерывность бизнеса	Геораспределенные ЦОД, гибридные и мультиоблачные архитектуры, катастрофоустойчивые решения, резервное копирование.	Минимизация простоев, сохранение репутации, выполнение регуляторных требований, снижение операционных рисков.
Безопасность и управление рисками	Zero Trust Architecture, SIEM/SOAR-системы, аппаратное шифрование, защита периметра и конечных точек.	Защита интеллектуальной собственности, соответствие GDPR и другим стандартам, предотвращение финансовых и репутационных потерь от кибератак.
Ускорение инноваций и гибкость	Платформы для разработки (PaaS), микросервисная архитектура, DevOps/DataOps-практики, low-code/no-code платформы.	Сокращение цикла разработки продуктов, возможность быстрого тестирования гипотез (fail-fast), адаптация к изменениям рынка.
Извлечение ценности из данных	Big Data-платформы, хранилища данных, инструменты аналитики в реальном времени, edge-вычисления.	Принятие решений на основе данных, персонализация услуг, прогнозная аналитика, создание данных как продукта.
Интеграция и создание экосистем	API-менеджмент, шины данных (ESB), IoT-платформы, блокчейн.	Открытие новых каналов сбыта, партнерство, создание платформенных бизнес-моделей, повышение прозрачности цепочек поставок.

Здесь можно выделить два аспекта этой роли:

- Платформенный аспект. Современная ЦИ, построенная на принципах облачности и программно-определяемого управления, предоставляет «цифровую фабрику» или платформу для трансформации. Она абстрагирует сложность базовых аппаратных ресурсов, предоставляя бизнес-подразделениям и разработчикам самообслуживаемые (self-service) инструменты. Это демократизирует доступ к тех-

нологиям, позволяя бизнес-подразделениям самостоятельно реализовывать проекты, что ускоряет трансформационные инициативы по всей организации.

- Каталитический аспект. Появление новых возможностей ЦИ (например, сверхмалые задержки в сетях 5G, повсеместные датчики IoT, доступные высокопроизводительные вычисления для ИИ) создает сам импульс для трансформации. Эти технологические возможности ставят перед бизнесом вопрос: «Что мы можем делать теперь, чего не могли делать раньше?». Так, edge-вычисления делают возможной трансформацию в производстве через предиктивное обслуживание, а мощные облачные AI-сервисы – трансформацию в маркетинге через гиперперсонализацию.

4. Эволюция архитектурных парадигм и их влияние на бизнес Способность ЦИ создавать ценность напрямую зависит от выбранной архитектурной парадигмы. Происходит последовательный переход от монолитных, жестко связанных систем к модульным, гибким и распределенным архитектурам:

1) Традиционная (монолитная) инфраструктура: высокие капитальные затраты, длительный цикл развертывания, низкая гибкость. Бизнес-ценность сводится в основном к поддержке существующих критических операций.

2) Виртуализированная и консолидированная инфраструктура: повышение утилизации ресурсов, снижение затрат, улучшение управления. Бизнес получает ценность в виде операционной эффективности и основ для стандартизации.

3) Облачная (гибридная/мультиоблачная) и программно-определяемая инфраструктура: эластичность, самообслуживание, оплата по факту использования. Бизнес-ценность проявляется в гибкости, скорости внедрения инноваций и устойчивости.

4) Платформенная и бессерверная (serverless) архитектура: полная абстракция от инфраструктуры, фокус на бизнес-логике и коде. Максимизация ценности за счет фокусировки на уникальных бизнес-компетенциях и экспоненциального сокращения времени разработки. Эта эволюция позволяет бизнесу перейти от модели «проектирование под известные нагрузки» к модели «экспериментирование и адаптация в реальном времени».

5. Управление цифровой инфраструктурой как стратегической возможностью для реализации полного потенциала ЦИ необходимо соответствующее управление. Это требует смены парадигмы ИТ-менеджмента: от управления технологическими активами к управлению сервисами и бизнес-результатами. Ключевые компетенции смещаются в сторону:

- Финансового управления ИТ (FinOps): прозрачность затрат на облачные ресурсы и их оптимизация в соответствии с бизнес-целями.
- Product-centric подхода: рассмотрения инфраструктурных команд как поставщиков платформ и сервисов для внутренних бизнес-заказчиков.
- Безопасности по дизайну (Security by Design): интеграции инструментов и практик кибербезопасности на этапе проектирования архитектуры, а не постфактум.

Таким образом, роль цифровой инфраструктуры в современном бизнесе является определяющей. Она трансформировалась из пассивной поддерживающей функции в активный стратегический актив, который непосредственно формирует конкурентные преимущества и создает новые формы бизнес-ценности. Цифровая инфраструктура обеспечивает не только техническую возможность цифровой трансформации, но и выступает ее катализатором, задавая новый темп инновациям. Успех организации в цифровую эпоху все в большей степени зависит от ее способности проектировать, развертывать и эффективно управлять эластичной, безопасной и интеллектуальной цифровой инфраструктурой, которая становится основой для создания ценности, устойчивости и долгосрочного роста. Внедрение передовых архитектурных решений и управленческих практик, ориентированных на бизнес-результаты, превращает цифровую инфраструктуру из статьи расходов в ключевой драйвер капитализации компании.

1.3. Ключевые тренды в сфере информационных технологий и их влияние на цифровую инфраструктуру компании

Современная цифровая трансформация бизнеса детерминируется конвергенцией нескольких взаимосвязанных технологических трендов, которые не просто модернизируют, а фундаментально реконфигурируют архитектуру, принципы управления и требования к цифровой

инфраструктуре компаний. Ключевыми среди них являются облачные вычисления (Cloud Computing), Интернет вещей (Internet of Things, IoT), Большие данные (Big Data) и искусственный интеллект/машинное обучение (Artificial Intelligence/Machine Learning, AI/ML). Их синергетическое воздействие формирует новую парадигму, в которой инфраструктура перестает быть пассивной средой выполнения и становится активной, интеллектуальной и адаптивной основой для создания бизнес-ценности.

Облачные вычисления выступают инфраструктурным катализатором и операционной моделью для остальных трендов. Переход от статичных, капиталоемких (CAPEX) и on-premise решений к гибкой, операционной (OPEX) модели «как услуга» (XaaS - IaaS, PaaS, SaaS, FaaS) радикально меняет подход к планированию и владению ИТ-ресурсами. Цифровая инфраструктура компании приобретает свойства эластичности (способность к быстрому масштабированию), гетерогенности (использование лучших в своем классе сервисов от разных провайдеров — мульти- и гибридно-облачные стратегии) и распределенности (географически дисперсные центры обработки данных). Это влечет за собой сдвиг фокуса внутренних ИТ-подразделений от управления физическим оборудованием к оркестрации сервисов, управлению затратами (FinOps) и обеспечению безопасности и соответствия требованиям (Compliance) в распределенной среде. Инфраструктурный код (Infrastructure as Code, IaC) становится стандартной практикой, обеспечивая воспроизводимость, контроль версий и автоматизацию развертывания.

Интернет вещей (IoT) экспоненциально расширяет периферию цифровой инфраструктуры, интегрируя в информационный контур физический мир через сети датчиков, исполнительных механизмов и умных устройств. Это порождает требования к принципиально новым архитектурным слоям:

1) Периферийные вычисления (Edge Computing), необходимые для обработки данных вблизи источника генерации с целью снижения задержек (latency), экономии полосы пропускания и обеспечения работы в режиме offline.

2) Платформы IoT — специализированные middleware (промежуточное программное обеспечение, программное обеспечение среднего слоя, подпрограммное обеспечение, межплатформенное программное

обеспечение) для управления устройствами, их аутентификации, безопасного сбора, агрегации и трансляции данных.

3) Высокопроизводительные и устойчивые сети (5G, LPWAN). Таким образом, инфраструктура становится гибридной, распределенной по иерархии «устройство - шлюз - край - облако» (device-gateway-edge-cloud), что усложняет задачи обеспечения безопасности, отказоустойчивости и управления жизненным циклом огромного количества разнородных конечных точек.

Большие данные (Big Data) формируют вычислительно-аналитический контур цифровой инфраструктуры, предъявляя требования к хранению, обработке и анализу огромных объемов структурированных и неструктурированных данных, характеризующихся высокой скоростью поступления и многообразием форматов (три «V»: Volume, Velocity, Variety (объем, скорость, разнообразие)). Инфраструктурный ответ на эти вызовы — отказ от традиционных реляционных СУБД в пользу горизонтально масштабируемых распределенных систем. Это включает: кластеры на основе Hadoop/Spark для пакетной обработки, потоковые обработчики (Apache Kafka, Apache Flink) для обработки данных в реальном времени, NoSQL базы данных (ключ-значение, документные, колоночные, графовые) для хранения, а также Data Lakes как репозитории сырых данных произвольного формата. Инфраструктура Big Data требует специализированных инженерных компетенций (Data Engineer) и оптимизированных под вычислительные нагрузки конфигураций, часто развертываемых в виде управляемых облачных сервисов.

Искусственный интеллект и машинное обучение (AI/ML) представляют собой высший уровень цифровой инфраструктуры — интеллектуальный слой, превращающий данные в предиктивную аналитику, т.е. это направление анализа данных, которое прогнозирует будущие события или тренды на основе исторической информации и скрытых закономерностей, а также автоматизированные решения. Влияние AI/ML проявляется двояко.

Во-первых, эти технологии требуют собственной высокопроизводительной инфраструктуры для обучения сложных моделей: специализированные аппаратные ускорители (GPU, TPU, NPU), масштабируемые вычислительные кластеры и системы хранения для больших наборов обучающих данных.

Во-вторых, и это наиболее значимо, AI/ML начинают пронизывать и оптимизировать саму инфраструктуру, создавая феномен AIOps (Artificial Intelligence for IT Operations). Алгоритмы ML используются для прогнозного анализа отказов, динамического управления ресурсами (auto-scaling), интеллектуальной кибербезопасности (обнаружение аномалий), автоматизации обслуживания и повышения эффективности работы ЦОД. Таким образом, цифровая инфраструктура становится самообучающейся и самовосстанавливающейся.

Следует отметить, что какой бы тренд развития цифровой инфраструктуры компании не был использован на современном этапе ее развития, он будет направлен на реализацию мероприятий и направлений, указанных на рис.1.3.

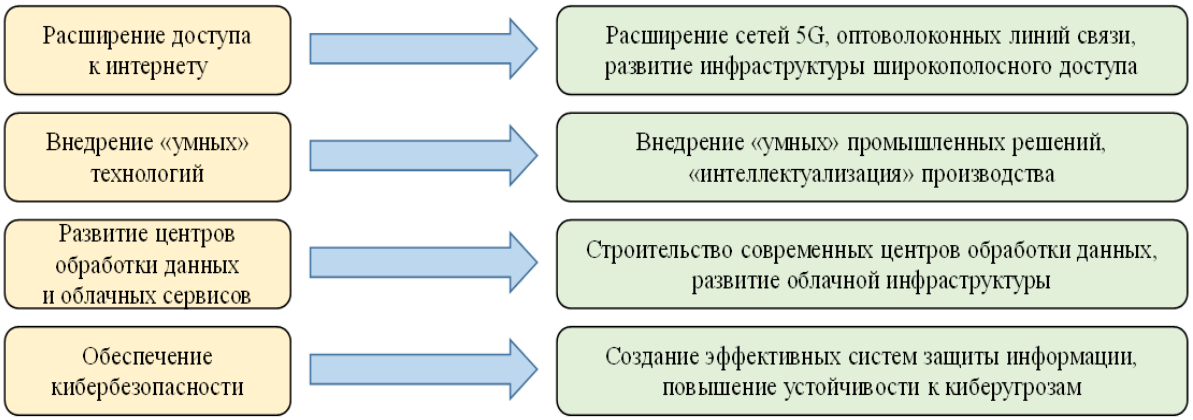

Рис. 1.3. Направления использования современных трендов развития цифровой инфраструктуры компании

Синергия этих трендов создает единый технологический стек, где IoT поставляет данные, облако предоставляет масштабируемые ресурсы для их консолидации и обработки, парадигма Big Data предоставляет инструменты для их организации, а AI/ML извлекает из них интеллектуальную ценность, одновременно оптимизируя работу всего стека. Ключевые инфраструктурные следствия этой конвергенции представлены в табл.1.4.

Таблица 1.4

Влияние ключевых технологических трендов на цифровую инфраструктуру компании

Тренд	Ключевые инфраструктурные требования	Архитектурные изменения	Операционные изменения
Облачные вычисления	Эластичность, глобальная доступность, API-управляемость, безопасность.	Гибридная и мульти-облачная архитектура, микросервисы, serverless.	Сдвиг к моделям OPEX, FinOps, DevOps/IaC, управление сервисами, а не железом.
IoT	Обработка на периферии (Edge), низкая задержка, высокая пропускная способность сети, безопасность «устройство-облако».	Многоуровневая архитектура (Edge-Fog-Cloud), платформы IoT, сетевая гетерогенность.	Управление и мониторинг распределенных систем, кибербезопасность для физического мира.
Big Data	Горизонтальная масштабируемость хранилищ и вычислителей, высокая пропускная способность сети и дисковых подсистем.	Data Lakes, Lambda/Kappa-архитектуры, кластерные развертывания (Hadoop/Spark).	Появление ролей Data Engineer, специализация на потоковой и пакетной обработке, управление жизненным циклом данных.
AI/ML	Высокопроизводительные вычисления (GPU/TPU), инфраструктура для обучения и эксплуатации (MLOps), хранение больших наборов данных.	Специализированные вычислительные кластеры, сервисы для ML pipeline (обучение, развертывание, мониторинг моделей).	Внедрение MLOps, использование AI для управления инфраструктурой (AIOps).

Совокупное влияние рассмотренных трендов трансформирует цифровую инфраструктуру из статичного затратного центра в динамичную, распределенную, интеллектуальную и сервисно-ориентированную платформу. Успех бизнеса в цифровую эпоху напрямую коррелирует со способностью организации проектировать, развертывать и управлять этой новой формой инфраструктуры, которая должна обеспечивать не только бесперебойность и безопасность базовых операций, но и обладать свойствами адаптивности, масштабируемости и способностью к быстрой интеграции инноваций. Будущее конкурентное преимущество будет принадлежать компаниям, чья инфраструктура построена как гибкая композитная платформа, органично объединяющая возможности облака, IoT, Big Data и AI/ML.

1.4. Введение в основные фреймворки: ITIL, COBIT, ISO 20000, TOGAF

В современной экономической парадигме цифровая инфраструктура перестала быть вспомогательным инструментом, превратившись в критический актив и основу бизнес-моделей. Ее стабильность, безопасность, гибкость и экономическая эффективность напрямую определяют конкурентоспособность организации. Однако сложность цифровых сред, их динамичность и глубокая интеграция с бизнес-процессами требуют системного подхода к управлению. Стихийное администрирование ИТ-ресурсов несет в себе риски простоя, несоответствия регуляторным требованиям, неоптимального использования бюджетов и, в конечном итоге, потери бизнес-возможностей.

Для решения этих задач был разработан ряд общепризнанных международных фреймворков и стандартов. Они представляют собой структурированные наборы передовых практик, принципов и моделей, обеспечивающих согласование целей цифровой инфраструктуры со стратегией бизнеса. Их внедрение позволяет перейти от реактивного устранения инцидентов к проактивному, процессно-ориентированному управлению услугами, архитектурой и рисками. Клевыми из них являются ITIL, COBIT, ISO/IEC 20000 и TOGAF. Они составляют фундамент для понимания методологических основ построения устойчивой и эффективной цифровой инфраструктуры.

Традиционно, в рамках исследования экономико-технических аспектов эффективности использования как классических ИТ-инфраструктур, так и цифровых их вариантов используют стандарт ITIL (Information Technology Infrastructure Library) и фреймворк управления ИТ-услугами (ITSM)

ITIL является де-факто мировым стандартом в области управления ИТ-услугами. Его эволюция, от библиотеки документов до современного набора практик (ITIL 4, 2019 год), отражает переход от управления ИТ-активами к созданию ценности для клиента через услуги. Центральным элементом ITIL 4 является «Модель ценности услуг (Service Value Model, SVS)», которая описывает, как различные компоненты организации взаимодействуют для создания ценности.

Ключевые компоненты ITIL 4 включают:

– Четырехмерная модель: рассмотрение любой услуги через призму организаций и людей, информационных и технологий, партнеров и поставщиков, потоков создания ценности и процессов.

– Система ценностей сервиса (SVS): интегрирует такие элементы, как управление услугами, принципы, руководство, непрерывное улучшение и 34 практики, объединенные в три категории: общие практики управления, практики управления услугами и технические практики.

– Цепочка создания ценности услуги (Service Value Chain): гибкая операционная модель, состоящая из шести взаимосвязанных видов деятельности (Планирование, Улучшение, Вовлечение, Проектирование и переход, Получение/создание, Оказание и поддержка), которую можно адаптировать под различные сценарии.

Основной вклад ITIL в управление цифровой инфраструктурой заключается в фокусе на процессы жизненного цикла услуг (от стратегии до непрерывного улучшения). Он обеспечивает общий язык и стандартизированные процедуры для управления инцидентами, проблемами, изменениями, уровнем услуг, что напрямую повышает надежность и предсказуемость инфраструктуры как основы бизнеса.

Еще одним стандартом, направленным на управление цифровой инфраструктурой компании, является COBIT (Control Objectives for Information and Related Technologies) или «Фреймворк управления и аудита ИТ».

Если ITIL фокусируется на «как» эффективно предоставлять услуги, то COBIT (актуальная версия COBIT 2019) отвечает на вопросы «что» и «для чего», обеспечивая комплексную систему корпоративного ИТ-управления (IT Governance). Его цель - поддержка руководства в выполнении требований заинтересованных сторон путем создания баланса между реализацией выгод от использования ИТ и оптимизацией уровней риска и ресурсов.

Структура COBIT 2019 базируется на нескольких ключевых элементах:

- Цели управления (Governance and Management Objectives): 40 целей (из которых 5 относятся к управлению, 35 – к менеджменту), охватывающих все аспекты ИТ-деятельности.

- Проектирование факторов (Design Factors): Контекстуальные элементы (стратегия предприятия, цели ИТ, модель управления, уровень зрелости и др.), которые определяют, как фреймворк должен быть адаптирован для конкретной организации.

- Фокусные области (Focus Areas): Наборы связанных между собой компонентов управления, решающих конкретные проблемы (например, безопасность, DevOps, управление данными).

- Процессы и компоненты: Каждая цель управления детализируется через практические процессы и набор из семи компонентов (принципы, политики, организационные структуры, процессы, информацию, культуру и людей).

COBIT обеспечивает сквозную систему контроля и управления над цифровой инфраструктурой, увязывая ИТ-цели с бизнес-целями через каскад метрик (Goals Cascade). Он предоставляет менеджменту инструменты для оценки рисков, обеспечения соответствия регуляторным требованиям и оптимизации инвестиций в ИТ, что делает цифровую инфраструктуру не просто рабочей, но и управляемой и подотчетной.

Помимо фреймворков и стандартов, указанных выше, также заслуживает внимания и ISO/IEC 20000: Международный стандарт системы управления ИТ-услугами (SMS).

ISO/IEC 20000 — это формальный международный стандарт, требования которого может быть сертифицирована организация. В отличие от фреймворков-рекомендаций (ITIL, COBIT), он задает четкие

и обязательные для выполнения критерии. Его цель - доказать способность организации планировать, проектировать, передавать, предоставлять и улучшать ИТ-услуги для удовлетворения согласованных требований.

Стандарт (часть 1: ISO/IEC 20000-1:2018) построен на основе модели PDCA (Plan-Do-Check-Act) и определяет структуру Системы менеджмента ИТ-услуг (SMS), включающую:

- Контекст организации и лидерство: определение заинтересованных сторон, политики, ролей.

- Планирование: оценка рисков и возможностей, постановка целей.

- Поддержка и операционная деятельность: обеспечение ресурсами, компетенциями, осведомленностью, коммуникациями, управление документацией и записями, а также выполнение ключевых процессов управления услугами (управление уровнем услуг, отношениями, инцидентами, запросами на обслуживание, проблемами, изменениями, конфигурациями, релизами, поставщиками).

- Оценка производительности и улучшение: мониторинг, анализ, внутренний аудит, анализ со стороны руководства, постоянное корректирующее действие.

Сертификация по ISO 20000 является доказательством зрелости процессов управления ИТ-услугами для внешних и внутренних заинтересованных сторон. Для цифровой инфраструктуры это означает гарантию того, что её эксплуатация и развитие осуществляются в рамках систематизированной, воспроизводимой и постоянно улучшаемой системы.

В условиях цифровой трансформации критически важным становится не только управление текущей эксплуатацией, но и проектирование будущего состояния цифровой инфраструктуры. Именно эту задачу решает архитектура предприятия (Enterprise Architecture, EA), а TOGAF является наиболее распространенным фреймворком для её разработки и управления.

Ядром TOGAF (версия 10, 2022) является «Метод разработки архитектуры (Architecture Development Method, ADM)» - итеративный, циклический процесс, состоящий из восьми фаз:

1. Предварительная: определение области, заинтересованных лиц, создание архитектурного видения.

2. Архитектура видения: формализация бизнес-целей, инициатив и ограничений.

3. Бизнес-архитектура: описание целевой бизнес-архитектуры.

4. Архитектура информационных систем: разработка архитектуры данных и приложений.

5. Технологическая архитектура: непосредственное проектирование целевой технологической (инфраструктурной) архитектуры.

6. Планирование и миграция: разработка планов реализации и миграции.

7. Управление реализацией: надзор за процессом внедрения.

8. Управление изменениями: управление изменениями в архитектуре, запуск нового цикла ADM.

Помимо ADM, TOGAF включает содержательную основу с ключевыми артефактами (документами, диаграммами, каталогами), модель зрелости и справочные модели. Его применение для цифровой инфраструктуры позволяет перейти от разрозненных технологических решений к целостной, модульной, масштабируемой и стратегически выверенной архитектуре, которая напрямую поддерживает бизнес-способности организации.

Общая концепция и направления использования рассмотренных выше стандартов и фреймворков при построении цифровой инфраструктуры компании представлены ниже на рис. 1.4.

Рис. 1.4. Обобщенная концепция построения цифровой инфраструктуры компании при использовании современных фреймворков и стандартов

Тем не менее, необходимо заметить, что ни один фреймворк создания и поддержания работоспособной цифровой инфраструктуры не является универсальным решением. Их эффективность проявляется в комплексном и взаимодополняющем использовании. Сравнительные характеристики рассмотренных стандартов и фреймворков представлены в табл. 1.5.

Таблица 1.5

Сравнительный анализ ключевых фреймворков и стандартов

Критерий	ITIL 4	COBIT 2019	ISO/IEC 20000-1	TOGAF 10
Основная направленность	Управление ИТ-услугами (ITSM), создание ценности	Управление и аудит ИТ (IT Governance), управление рисками и ресурсами	Стандарт для системы менеджмента ИТ-услуг (SMS)	Разработка и управление архитектурой предприятия (EA)
Основной объект управления	ИТ-услуга и её жизненный цикл	Информация и связанные технологии, процессы управления	Система менеджмента (процессы)	Архитектурные домены (бизнес, данные, приложения, технологии)
Статус	Фреймворк передовых практик (Best Practice)	Фреймворк управления и контроля	Международный стандарт (требования)	Фреймворк и методология
Ключевой результат	Эффективное и клиенто-ориентированное предоставление услуг	Обеспечение стратегического соответствия, управление рисками, оптимизация ресурсов	Сертифицируемая система менеджмента, гарантия качества услуг	Целостная, стратегически выверенная архитектура, дорожная карта трансформации
Роль для цифровой инфраструктуры	Операционная модель её эксплуатации и поддержки	Система контроля, аудита и управления ею как активом	Формализация процессов её эксплуатации	Инструмент её стратегического проектирования и эволюции

Синергетический эффект достигается при их совместном использовании.

Так, архитектурный фреймворк TOGAF определяет целевое состояние цифровой инфраструктуры (в частности фаза технологической

архитектуры в цикле ADM). В свою очередь, стандарт COBIT обеспечивает управление и контроль над реализацией архитектурных планов, оценку рисков и эффективности инвестиций, ITIL предоставляет операционные процессы для стабильной эксплуатации и поддержки развернутой цифровой инфраструктуры конкретного бизнеса как набора услуг. Кроме того, стандарт ISO 20000 формализует и сертифицирует эти процессы, обеспечивая внешнее подтверждение их зрелости.

Таким образом, современные фреймворки и стандарты ITIL, COBIT, ISO 20000 и TOGAF формируют комплексную многоуровневую систему управления цифровой инфраструктурой. Они охватывают стратегический (TOGAF, COBIT), тактический (COBIT, ITIL) и операционный (ITIL, ISO 20000) уровни, обеспечивая ее стратегическую направленность, управляемость, эффективность эксплуатации и способность к эволюции. Их изучение и грамотное комбинирование является необходимым условием для построения цифровой инфраструктуры, которая выступает не как центр затрат, а как устойчивый фундамент и драйвер бизнес-развития в цифровую эпоху.

Таким образом, цифровая инфраструктура компании представляет собой качественно новое явление по сравнению с традиционной ИТ-инфраструктурой. Если последняя ориентирована на обеспечение работоспособности внутренних систем организации, то цифровая инфраструктура носит экосистемный и сквозной характер, выступая в роли технологической платформы для взаимодействия множества субъектов (бизнеса, государства, потребителей) и основой для принципиально новых бизнес-моделей, таких как платформенная экономика, интернет вещей и индустрия 4.0. Её ключевыми атрибутами являются масштабируемость, высокая доступность, безопасность, гибкость и ориентация на предоставление сквозных сервисов.

Структурно цифровая инфраструктура представляет собой многоуровневую систему, включающую аппаратно-физический, программно-абстрактный, сервисно-данный и интеграционный слои. Эволюция ЦИ носит нелинейный и кумулятивный характер, пройдя путь от централизованных мэйнфреймов через клиент-серверные архитектуры и эпоху интернета к современной парадигме, определяемой виртуализацией, облачными вычислениями и их гибридными формами. Текущий этап характеризуется доминированием гибридных и мультиоблач-

ных стратегий, активным развитием периферийных (граничных) вычислений и началом внедрения практик AIOps, что свидетельствует о переходе к созданию распределенной, интеллектуальной и в значительной степени автономной инфраструктурной среды.

Важнейшим выводом является констатация трансформации роли ЦИ из вспомогательного технологического актива в стратегический ресурс, непосредственно генерирующий бизнес-ценность. Она перестала быть лишь центром затрат, превратившись в инвестиционную платформу, обеспечивающую операционную эффективность, устойчивость и непрерывность бизнеса, безопасность, ускорение инноваций и извлечение ценности из данных. Цифровая инфраструктура выступает не только техническим фундаментом, но и активным катализатором цифровой трансформации, создавая новые технологические возможности (например, за счет IoT и AI), которые стимулируют пересмотр бизнес-процессов и предложения ценности.

Анализ ключевых технологических трендов (облачные вычисления, IoT, Big Data, AI/ML) подтверждает их конвергентное и синергетическое влияние, приводящее к фундаментальной реконфигурации требований к ЦИ. Совокупный эффект от этих трендов формирует инфраструктуру нового типа - динамичную, эластичную, географически распределенную и управляемую через код (IaC), что требует от организаций развития новых компетенций в области оркестрации сервисов, управления затратами (FinOps) и обеспечения безопасности в сложных гетерогенных средах.

Для системного управления столь сложным активом необходима опора на проверенные международные фреймворки и стандарты. Как показано, фреймворки ITIL, COBIT, ISO 20000 и TOGAF образуют взаимодополняющую систему, покрывающую стратегический (TOGAF, COBIT), тактический (COBIT, ITIL) и операционный (ITIL, ISO 20000) уровни управления. Их комплексное применение позволяет обеспечить стратегическое соответствие ЦИ бизнес-целям, эффективное управление рисками и ресурсами, операционную надежность и формализованную зрелость процессов.

Таким образом, успех современной компании в условиях цифровой экономики напрямую зависит от его способности осознанно проектировать, развивать и управлять цифровой инфраструктурой как це-

лостной, гибкой и безопасной платформой. Понимание её компонентного состава, логики эволюции, механизмов создания бизнес-ценности и методологических основ управления, изложенное в данной главе, формирует необходимый концептуальный фундамент для формирования эффективной цифровой стратегии и достижения долгосрочной конкурентоспособности. Цифровая инфраструктура утвердилась в качестве несущего каркаса бизнеса, от качества которого зависит его устойчивость, адаптивность и потенциал роста.

Вопросы для обсуждения

1. Дайте определение традиционной ИТ-инфраструктуры и цифровой инфраструктуры компании.
2. Объясните, в чем заключаются ключевые различия между традиционной ИТ-инфраструктурой и современной цифровой инфраструктурой с точки зрения их роли в бизнесе?
3. Перечислите основные слои цифровой инфраструктуры и составляющие их элементы.
4. Представьте характеристики цифровых инфраструктур при их классификации по уровню интеграции.
5. Охарактеризуйте основные этапы эволюции цифровой инфраструктуры компании.
6. Укажите специфику классификации цифровой инфраструктуры по степени критичности и требованиям к доступности.
7. Поясните на конкретных примерах роль цифровой инфраструктуры в создании бизнес-ценности и цифровой трансформации.
8. Перечислите основные механизмы создания бизнес-ценности цифровой инфраструктурой.
9. Поясните, в чем заключается платформенный и каталитический аспекты усиления роли цифровой инфраструктуре в бизнесе.
10. Проанализируйте, каким образом цифровая инфраструктура трансформировалась из «центра затрат» в «стратегический инвестиционный актив».
11. Перечислите основные тренды в сфере информационных технологий, влияющих на развитие цифровой инфраструктуры бизнеса.

12. Поясните роль Интернет вещей (IoT) при построении цифровой инфраструктуры компании. Какие дополнительные слои инфраструктуры при этом необходимы.

13. Назовите интеллектуальный слой цифровой инфраструктуры, превращающий данные в предиктивную аналитику.

14. Поясните, каким образом Большие Данные (Big Data) формируют вычислительно-аналитический контур цифровой инфраструктуры.

15. Перечислите основные направления обеспечения кибербезопасности при использовании передовых информационных трендов построения цифровой инфраструктуры.

16. Дайте характеристику ключевым компонентам ITIL 4. Какова состоит их роль формировании цифрой инфраструктуры компании?

17. Поясните структуру COBIT 2019 при построении цифровой инфраструктуры компании.

18. Объясните, как фреймворки ITIL и COBIT дополняют друг друга, покрывая разные аспекты управления цифровой инфраструктурой бизнеса.

19. Поясните, в каких ситуациях бизнесу целесообразно стремиться к сертификации по ISO/IEC 20000 в рамках построения и использования цифровой его инфраструктуры.

20. Поясните роль современных стандартов, фреймворков и передовых информационных технологий для построения гибкой и масштабируемой цифровой инфраструктуры компании.

Практические задания

Задание 1. Выберите одну из небольших или средних компаний, относящихся к промышленных отраслям или непроизводственному сектору, для которой выполните следующее:

1. Проанализируйте бизнес-процессы и определите не менее трех ключевых сервисов, критически зависящих от цифровой инфраструктуры.

2. Разработайте в виде логико-структурной схемы модель цифровой инфраструктуры, необходимой для поддержки данных сервисов. Заполните таблицу, распределив необходимые технологии по четырем базовым слоям.

Слой инфраструктуры	Ключевые компоненты и технологии (не менее двух на слой)	Обоснование выбора (связь с бизнес-процессом/сервисом)
1. Аппаратно-физический		
2. Программно-абстрактный		
3. Сервисов и данных		
4. Интеграционный		

3. Обоснуйте выбор модели развертывания (традиционная, конвергентная, гиперконвергентная, облачная) для основных компонентов. Ответ представьте в виде краткого вывода (150-200 слов).

Задание 2. Разработка «дорожной карты» эволюции инфраструктуры с использованием фреймворков

Рассмотрите гипотетическую компанию из сферы логистики, которая использует устаревшую фрагментированную инфраструктуру и ставит стратегическую цель: в течение трех лет перейти к гибридной облачной модели для обеспечения сквозной видимости цепочек поставок в реальном времени.

Предложите дорожную карту использования различных фреймворков и стандартов для достижения этой цели:

1. Составьте хронологическую последовательность (план) вовлечения каждого из четырех фреймворков (TOGAF, COBIT, ITIL, ISO 20000) в проект трансформации. Для каждого этапа укажите:

– Фреймворк/стандарт.

– Конкретная задача в рамках проекта, решаемая с его помощью.

– Ожидаемый результат (артефакт или решение).

2. На основании предложенного плана постройте схему в виде блок-схемы или таблицы синергии, визуализирующую, как результаты применения одного фреймворка становятся входными данными или основой для работы другого. Сформулируйте вывод о том, какой управленческий пробел возникнет, если исключить из проекта один из предложенных фреймворков.

Тест для самоконтроля

1. В отличие от традиционной ИТ-инфраструктуры, цифровая инфраструктура носит экосистемный и сквозной характер?

а) Замкнутый характер.

б) Экосистемный и сквозной характер.

в) Развертываемый характер.

г) Расширяемый характер.

2. Какого слоя цифровой инфраструктуры не существует?

а) Аппаратно-физический.

б) Программно-абстрактный.

в) Сервисов и данных.

г) Итерационный.

3. Основное предназначение этого слоя цифровой инфраструктуры обеспечение взаимодействия и обмена данными между разнородными системами и сервисами. О каком ее слое идет речь.

а) Аппаратно-физический.

б) Интеграционный.

в) Сервисов и данных.

г) Программно-абстрактный.

4. На каком этапе развития цифровой инфраструктуры находится общество в данный момент?

а) На четвертом.

б) На пятом.

в) На шестом.

г) На переходном периоде.

5. По степени критичности и требованиям к доступности цифровая инфраструктура может быть…

а) Критичной и некритичной.

б) Стабильной и нестабильной.

в) Статичной и стабильной.

г) Динамичной и некритичной.

6. Принятие решений на основе данных, персонализация услуг, прогнозная аналитика, создание данных как продукта относится к механизму…

а) Ускорение инноваций и гибкость.

б) Извлечение ценности из данных.

в) Безопасность и управление рисками.

г) Интеграция и создание экосистем.

7. Укажите, на какие аспекты направлена разработка и эксплуатация современных цифровых инфраструктур

а) Платформенный и каталитический аспекты.

б) Синергетический и масштабируемый аспекты.

в) Платформенный и итерационный аспекты.

г) Интеграционный и синергетический аспекты.

8. Современная ЦИ, построенная на принципах облачности и программно-определяемого управления, предоставляет «цифровую фабрику» или платформу для трансформации характеризует…

а) Каталитический аспект.

б) Синергетический аспект.

в) Платформенный аспект.

г) Эволюционный аспект.

9. Высокие капитальные затраты, длительный цикл развертывания, низкая гибкость, бизнес-ценность сводится в основном к поддержке существующих критических операций характеризуют…

а) Традиционную (монолитную) инфраструктуру.

б) Виртуализированную и консолидированную инфраструктуру.

в) Облачную и программно-определяемую инфраструктуру.

г) Платформенную и бессерверную инфраструктуру.

10. Облачные вычисления выступают инфраструктурным … для остальных трендов развития цифровой инфраструктуры.

а) катализатором и операционной моделью;

б) направлением эволюции;

в) неотъемлемым элементом;

г) элементом свертывания.

11. Data Lake - это…

а) Особый вид реляционной базы данных.

б) Централизованное хранилище, которое позволяет хранить огромные объёмы только обработанных данных.

в) Централизованное хранилище, которое позволяет хранить огромные объёмы данных в их исходном, необработанном (сыром) формате

г) Децентрализованное хранилище, которое позволяет хранить огромные объёмы данных в их исходном, необработанном (сыром) формате

12. Горизонтальная масштабируемость хранилищ и вычислителей, высокая пропускная способность сети и дисковых подсистем характерна для следующего тренда цифровой инфраструктуры - …

а) IoT;

б) Big Data;

в) AI/ML;

г) облачные вычисления.

13. Искусственный интеллект и машинное обучение (AI/ML) представляют собой высший уровень цифровой инфраструктуры -….

а) аналитический слой;

б) интеллектуальный слой;

в) слой каталогизации;

г) слой интеграции.

14. Основной вклад ITIL в управление цифровой инфраструктурой заключается в фокусе ….

а) на процессы жизненного цикла услуг;

б) создание нового вида услуг;

в) на разделение услуг на определенные виды;

г) на свертывании и утилизации услуг.

15. Сколько практик содержит система ценностей сервиса (SVS) ITIL 4?

а) 30;

б) 32;

в) 34;

г) 36.

16. COBIT обеспечивает сквозную систему контроля и управления над цифровой инфраструктурой, увязывая ИТ-цели с бизнес-целями через …

а) каскад метрик;

б) показатели эффективности;

в) технические параметры конкретной инфраструктуры;

г) объемы инвестиций.

17. Ядром TOGAF (версия 10, 2022) является «Метод разработки архитектуры (ADM)», состоящий из…

а) 6 фаз;

б) 8 фаз;

в) 10 фаз;

г) 12 фаз.

18. 6 фазой «Метода разработки архитектуры (ADM)» ядра TOGAF является

а) Управление реализацией.

б) Планирование и миграция.

в) Бизнес-архитектура.

г) Архитектура информационных систем.

19. Основным объектом управления согласно международному стандарту ISO/IEC 20000-1 является…

а) Архитектурные домены.

б) Информация и связанные технологии, процессы управления.

в) ИТ-услуга и её жизненный цикл.

г) Система менеджмента (процессы).

20. Какие фреймворки охватывают стратегический уровень управления цифровой инфраструктурой компании?

а) COBIT, ITIL

б) TOGAF, COBIT;

в) ITIL, ISO 20000

г) Цикл ADM.

Глава 2. СТРАТЕГИЯ И АРХИТЕКТУРА ЦИФРОВОЙ ИНФРАСТРУКТУРЫ КОМПАНИИ

2.1. Выравнивание ИТ-стратегии с бизнес-целями

В современной конкурентной среде цифровая инфраструктура перестала быть вспомогательным инструментом, превратившись в ключевой стратегический актив, определяющий жизнеспособность и потенциал роста компании. Выравнивание (англ. strategic alignment) ИТ-стратегии с бизнес-целями представляет собой непрерывный и комплексный процесс целенаправленного согласования целей, приоритетов, инвестиций и операционной деятельности в области информационных технологий с тактическими и стратегическими задачами бизнеса. Это не просто техническое планирование, а управленческая дисциплина, обеспечивающая трансформацию бизнес-требований в конкретные архитектурные решения, сервисы и ИТ-процессы. Отсутствие такого выравнивания ведет к фундаментальным рискам: инвестиции в ИТ становятся затратными, а не создающими стоимость, возникают «информационные разрывы», снижается гибкость организации и её способность к инновациям. Таким образом, достижение стратегической конгруэнтности бизнеса и ИТ является критическим фактором для создания устойчивого конкурентного преимущества в цифровую эпоху.

Теоретической основой для практического выравнивания служит многоуровневая модель, связывающая корпоративное стратегическое планирование с проектированием и эксплуатацией цифровой инфраструктуры. Данный процесс может быть декомпозирован на четыре последовательных и взаимосвязанных уровня:[5]

- Уровень 1: Стратегический (Бизнес-стратегия → Цели ИТ).

На этом уровне происходит трансляция общих бизнес-целей компании (например, «увеличение доли рынка на 15%», «выход на новые географические рынки», «повышение лояльности клиентов») в стратегические цели ИТ-подразделения. Фокус смещается с поддержки теку-

[5] Бирюков А. ИТ-стратегии: какие бывают и как их использовать [Электронный ресурс]/ А. Бирюков // Режим доступа: https://habr.com/ru/companies/otus/articles/943210/ (дата обращения: 06.01.2026)

щих операций на создание возможностей для бизнеса. Ключевые вопросы: «Как ИТ могут обеспечить достижение этих бизнес-целей?» и «Какие цифровые возможности необходимо развить?». Результатом являются высокоуровневые ИТ-цели, такие как обеспечение платформы для омниокального взаимодействия (единая система взаимодействия через различные каналы: физические магазины, веб-сайт, мобильное приложение, социальные сети, мессенджеры) с клиентом, создание аналитической системы для прогнозирования спроса или обеспечение киберустойчивости в условиях расширения.

- Уровень 2: Тактический (Цели ИТ → Архитектура и портфель услуг).

Здесь стратегические цели ИТ трансформируются в конкретные архитектурные принципы, требования и портфель ИТ-сервисов. Формируется целевая архитектура цифровой инфраструктуры (англ. Target Architecture), определяющая стандарты, технологии, модели данных и интеграционные паттерны. Происходит планирование портфеля проектов и сервисов, приоритизация которых осуществляется на основе их вклада в бизнес-ценность. Критически важным становится использование архитектурных фреймворков (например, TOGAF) для обеспечения системности и согласованности решений.

- Уровень 3: Операционный (Архитектура и сервисы → Процессы и компетенции).

На данном уровне архитектурные решения реализуются через операционные ИТ-процессы (на базе лучших практик ITIL/ITSM), организационную структуру и компетенции команды. Выравнивание требует настройки процессов управления сервисами, инцидентами, изменениями и безопасностью таким образом, чтобы они надежно и эффективно поддерживали запланированные сервисы. Формируются ключевые показатели эффективности (KPI) и показатели результативности услуг (SLI/SLA), напрямую увязанные с бизнес-метриками.

- Уровень 4: Измерение и обратная связь (Мониторинг → Эволюция стратегии).

Процесс выравнивания является итеративным и требует постоянного мониторинга. Система метрик (как технических, так и бизнес-ориентированных, например, TCO, ROI, Time-to-Market, Net Promoter Score) позволяет оценить, насколько реализованные ИТ-решения и

сервисы действительно способствуют достижению бизнес-целей. Полученные данные служат основой для корректировки как ИТ-стратегии, так, в некоторых случаях, и бизнес-подходов (феномен «цифровой трансформации»).

Основные эволюционные этапы построения ИТ-стратегии разработки, внедрения и эксплуатации цифровой инфраструктуры представлены ниже на рис.2.1.

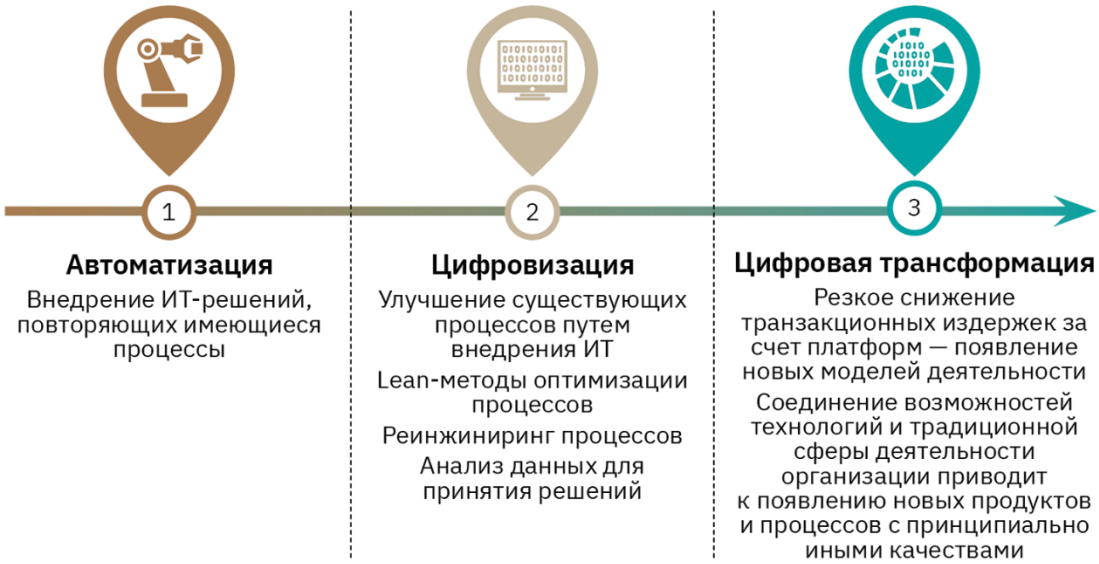

Рис. 2.1. Этапы построения ИТ-стратегии разработки, внедрения и эксплуатации цифровой инфраструктуры

Для практической реализации модели выравнивания используются специализированные управленческие и архитектурные фреймворки. Их комбинирование позволяет создать целостную систему управления. В частности, наиболее популярными для построения работоспособной цифровой инфраструктуры компании применяют:

- Модель зрелости стратегического выравнивания (SAMM): предоставляет методологию оценки текущего уровня интеграции бизнеса и ИТ по таким измерениям, как коммуникация, компетенции, управление и партнерство, помогая выявить слабые места.

- Сбалансированная система показателей (BSC) для ИТ: позволяет каскадировать корпоративные стратегические цели на уровень ИТ-подразделения, формируя систему взаимосвязанных показателей по четырем перспективам: финансы, клиенты (внутренние и внешние), внутренние процессы, обучение и рост.

- Бизнес-архитектура (например, на основе фреймворка BizBok): выступает в роли «переводчика» между бизнес-стратегией и ИТ-архитектурой. Она формализует бизнес-мотивацию (цели, стратегии), ключевые бизнес-процессы, организационную структуру и информацию, что создает однозначные требования для проектирования цифровой инфраструктуры.

- Портфельное управление ИТ-проектами (IT PPM): обеспечивает отбор, приоритизацию и контроль исполнения проектов на основе их стратегической ценности, рисков и ресурсных ограничений, гарантируя, что инвестиции направляются в наиболее значимые для бизнеса инициативы.

Связь бизнес-целей с элементами архитектуры построения и развития цифровой инфраструктуры компании представлен ниже в табл. 2.1.

Техническое выравнивание невозможно без соответствующей организационной поддержки, ключевыми условиями из которых являются:[6]

- Роль и вовлеченность руководства: активное участие бизнес-руководителей (вплоть до CEO) в принятии решений по ИТ-стратегии, а также наличие на уровне топ-менеджмента роли CIO или CDO, выступающего связующим звеном.

- Совместные кросс-функциональные команды: Создание структур, объединяющих бизнес-аналитиков, архитекторов, разработчиков и владельцев продуктов, для совместной работы над реализацией стратегических инициатив (модели Agile, DevOps, Product-centric).

- Эффективные механизмы коммуникации и управления требованиями: Установление прозрачных процессов сбора, анализа и приоритизации бизнес-требований, их регулярный пересмотр в свете изменяющихся условий рынка.

[6] Какие технологии помогают бизнесу построить единую ИТ-инфраструктуру [Электронный ресурс] // Режим доступа: https://digtlab.ru/tpost/rzfhfyfrr1-kakie-tehnologii-pomogayut-biznesu-postr?ysclid=mk2z328jiu499207780 (дата обращения: 06.01.2026).

Таблица 2.1

Связь бизнес-целей с элементами архитектуры цифровой инфраструктуры

Бизнес-цель / Стратегия	Стратегическая цель ИТ	Ключевые требования к архитектуре ЦИ	Ожидаемый бизнес-результат
Повышение клиентоцентрич-ности	Создание единой платформы взаимодействия с клиентом (CRM + омниканаль-ность)	Высокая доступность (99.95+%), интеграционный шина для подключения всех каналов, хранилище данных о клиентах (CDP), низкая задержка (latency) для вебсервисов	Рост лояльности (NPS), увеличение конверсии, снижение оттока клиентов.
Оптимизация операционных издержек	Автоматизация рутинных процессов и консолидация ИТ-ресурсов	Внедрение облачных моделей (IaaS/PaaS), виртуализация и контейнеризация, оркестрация, RPA-платформа, стандартизация и унификация компонентов.	Снижение ТСО ИТ-инфраструк-туры, увеличение производитель-ности, масштабируемость.
Ускорение вывода новых продуктов на рынок	Внедрение гибких методологий разработки (DevOps) и платформенного подхода	Микросервисная архитектура, CI/CD пайплайны, облачные платформы для разработки (PaaS), инфраструктура как код (IaC).	Сокращение цикла разработки (Time-to-Market), повышение частоты релизов, повышение устойчивости приложений.
Обеспечение соответствия регуляторным требованиям (напр., ФЗ-152, GDPR)	Создание системы управления информационной безопасностью и данными	Архитектура «безопасность по замыслу», сегментация сети, системы DLP и SIEM, централизованное управление доступом, шифрование данных, резервное копирование и аварийное восстановление	Снижение юридических и репутационных рисков, обеспечение непрерывности бизнеса, повышение доверия партнеров.

- Управление изменениями: осознание того, что выравнивание — это организационное изменение, требующее работы по преодолению сопротивления, развитию цифровой культуры и новых компетенций как в ИТ-подразделении, так и в бизнес-единицах.

Несмотря на очевидную важность, процесс выравнивания сталкивается с рядом системных барьеров:

- Разрыв в языках и ментальных моделях. Бизнес-руководители мыслят категориями рентабельности, рынка и клиентов, в то время как ИТ-специалисты - технологиями, надежностью и производительностью.

- Динамичность внешней среды. Быстрое изменение рынков и появление disruptive-технологий требует от процесса выравнивания высочайшей гибкости и адаптивности, что противоречит традиционным долгосрочным циклам стратегического планирования.

- Наследие (Legacy Systems). Устаревшие, монолитные и слабодокументированные информационные системы создают технологический долг, ограничивающий скорость инноваций и повышающий стоимость изменений.

- Раздельное бюджетирование и оценка эффективности.

- Традиционное восприятие ИТ как центра затрат, а не инвестиций, и отсутствие прозрачных моделей расчета возврата на инвестиции (ROI) для ИТ-проектов.

Таким образом, следует заключить, что выравнивание ИТ-стратегии с бизнес-целями – это не разовое мероприятие, а циклический и динамичный процесс управления, требующий интеграции стратегического планирования, архитектурного проектирования и операционного совершенства. Успех в его реализации определяется не столько выбором конкретных технологий, сколько качеством управленческих процессов, зрелостью архитектурной практики и глубиной взаимопонимания между бизнес- и ИТ-лидерами. В конечном итоге, именно это выравнивание позволяет трансформировать цифровую инфраструктуру из пассивного «объекта расходов» в активный стратегический драйвер, способный генерировать новую ценность, обеспечивать устойчивость и создавать фундамент для долгосрочного конкурентного преимущества компании в цифровой экономике.

2.2. Принципы проектирования современной инфраструктуры

Стратегическое проектирование цифровой инфраструктуры предприятия в условиях цифровой трансформации требует строгого следования набору взаимосвязанных императивных принципов. Эти

принципы формируют архитектурный каркас, обеспечивающий не только операционную жизнеспособность ИТ-среды, но и ее способность быть драйвером бизнес-развития. Ключевыми из них являются масштабируемость, отказоустойчивость, безопасность и экономическая эффективность, реализуемые не изолированно, а в рамках целостной архитектурной модели.

1. Масштабируемость (Scalability) – это системное свойство инфраструктуры увеличивать или уменьшать свои вычислительные мощности и производительность в ответ на изменение нагрузки с минимальными издержками и без нарушения функционирования. В современных условиях масштабируемость является антитезой традиционным монолитным системам, требующим длительных и дорогостоящих апгрейдов «железа». Принцип декомпозирует на два основных вектора:

- Горизонтальное масштабирование (Scale-out): увеличение общей производительности за счет добавления однотипных, стандартизированных узлов (серверов, контейнеров, экземпляров виртуальных машин) в пул ресурсов. Данный подход, фундаментальный для облачных и гибридных сред, обеспечивает высокую гибкость и отказоустойчивость.

- Вертикальное масштабирование (Scale-up): увеличение мощности отдельного узла (например, добавление процессоров, памяти, дискового пространства). Часто связано с остановкой сервиса и имеет физические и экономические ограничения, поэтому в чистом виде применяется для специфических рабочих нагрузок (например, СУБД с большим объемом оперативной памяти).

Архитектурными паттернами, обеспечивающими масштабируемость, являются микросервисная архитектура (замена монолита на множество независимо развертываемых сервисов), использование контейнеризации (Docker) и оркестрации (Kubernetes) для эффективного управления распределенными нагрузками, а также проектирование stateless-сервисов (не хранящих состояние сессии на конкретном инстансе), что позволяет свободно добавлять или удалять экземпляры.

2. Отказоустойчивость (Fault Tolerance и High Availability) – это способность системы продолжать корректное функционирование (полностью или с допустимой деградацией) при возникновении отказа одного или нескольких ее компонентов. В контексте стратегии цифровой

инфраструктуры этот принцип трансформируется из задачи минимизации простоев (MTTR – Mean Time To Repair) в задачу обеспечения постоянной доступности и целостности данных. Реализация требует многоуровневого подхода:

- Резервирование (Redundancy): создание избыточности на уровне компонентов (диски в RAID-массивах, блоки питания), серверов (кластеризация, отказоустойчивые пары), центров обработки данных (Geo-Redundancy, распределенные AZ – Availability Zones) и каналов связи.

- Автоматическое восстановление (Self-healing): архитектурная способность системы автоматически обнаруживать сбой и перераспределять нагрузку на исправные компоненты без вмешательства человека. Это достигается за счет использования балансировщиков нагрузки, оркестраторов и предопределенных сценариев failover.

- Устойчивое проектирование (Resilience Engineering): парадигма, выходящая за рамки технического резервирования. Она включает практики хаос-инжиниринга (целенаправленное внесение сбоев в продакшн-среду для проверки устойчивости), проектирование для деградации (graceful degradation) и использование идемпотентных операций, позволяющих безопасно повторять транзакции.

Уровни доступности цифровой инфраструктуры, их бизнес-интерпретация и классы критичности представлены ниже в табл. 2.2.

3. Безопасность (Security by Design) – это принцип интеграции мер и механизмов кибербезопасности на каждом этапе проектирования и на каждом уровне архитектуры, а не их последующее наложение на готовую систему. В современной распределенной инфраструктуре периметр безопасности размыт, что требует перехода от моделей «крепости с замком» к моделям «недоверия» (Zero Trust).

Таблица 2.2

Уровни доступности инфраструктуры и их бизнес-интерпретация

Уровень доступности (%)	Максимальное время простоя в год	Класс критичности	Типовые требования
99.0 (two nines)	3 дня 15.6 часа	Базовый	Некритичные внутренние сервисы, тестовые среды.
99.9 (three nines)	8 часов 46 минут	Корпоративный	ERP, CRM-системы, электронная почта.
99.99 (four nines)	52 минуты 36 секунд	Высокий	Системы онлайн-транзакций, большинство публичных веб-сервисов.
99.999 (five nines)	5 минут 15 секунд	Критичный	Телекоммуникационные платформы, биржевые торги, системы управления критическими процессами.

Ключевые аспекты включают:

- Идентификация, аутентификация и авторизация (IAAA): строгое управление доступом на основе ролей (RBAC) с обязательной многофакторной аутентификацией (MFA) для всех пользователей и сервисов.

- Защита данных: шифрование данных как в состоянии покоя (at rest) на дисках и в базах данных, так и в состоянии передачи (in transit) между компонентами. Регулярное резервное копирование с проверкой целостности и изолированным хранением копий.

- Сегментация сети: логическое разделение инфраструктуры на изолированные сегменты (VLAN, VXLAN, микросетевое сегментирование) для ограничения горизонтального перемещения злоумышленника в случае компрометации.

- Неизменяемая инфраструктура (Immutable Infrastructure): практика, при которой после развертывания компоненты (серверы, контейнеры) не модифицируются, а заменяются на новые, развернутые из предварительно собранного и проверенного на безопасность образа. Это минимизирует дрейф конфигураций и снижает поверхность атаки.

- Единое управление безопасностью и соответствием (Compliance): Автоматизированный сбор логов, мониторинг событий безопасности (SIEM) и инструменты для автоматического аудита на

соответствие регуляторным требованиям (GDPR, ФЗ-152, PCI DSS и др.).

4. Экономическая эффективность (Cost-Efficiency, FinOps) – это оптимизация совокупной стоимости владения (TCO) цифровой инфраструктурой через осознанное управление и распределением ресурсов, выравнивание ИТ-расходов с реальными бизнес-потребностями.[7]

В облачных и гибридных моделях данный принцип эволюционирует в дисциплину FinOps (Financial Operations), а именно:

- От операционных расходов (OpEx) к эффективности: модель облачных услуг позволяет перейти от крупных капитальных затрат (CapEx) на покупку оборудования к более гибким операционным расходам, оплачивая только фактически потребленные ресурсы. Ключевой задачей становится предотвращение «распухания» облачных счетов,

- Управление потреблением и оптимизация: регулярный аудит задействованных ресурсов, автоматическое масштабирование (автоскейлинг) для соответствия пиковой и минимальной нагрузке, выбор оптимальных классов услуг (например, использование spot-инстансов или preemptible VMs для не критичных к прерыванию фоновых задач), архивирование неиспользуемых данных.

- Методики Total Cost of Ownership (TCO): при стратегическом выборе между on-premise, облачной или гибридной моделью необходим комплексный расчет TCO, включающий не только прямую стоимость оборудования/аренды, но и затраты на электроэнергию, охлаждение, физическую безопасность ЦОД, зарплаты персонала, лицензии на ПО, риски простоя.[8]

Сравнительный анализ влияния архитектурных принципов на атрибуты цифровой инфраструктуры представлен в табл. 2.3.

---

[7] Бурцев Д.С. Инфраструктура и ресурсное обеспечение цифровой экономики: учеб. Пособие/ Д. С. Бурцев [и др.]. - СПб: Университет ИТМО, 2021. – 190 с. - Режим доступа: https://books.ifmo.ru/file/pdf/3020.pdf (дата обращения: 16.01.2026).

[8] Пронин А.Ю. Основные тенденции развития цифровой инфраструктуры в интересах национальной экономики [Электронный ресурс]/ А.Ю. Пронин// Экономические исследования и разработки. – Режим доступа: http://edrj.ru/article/09-08-24 (дата обращения: 04.01.2026).

Таблица 2.3

Сравнительный анализ влияния архитектурных принципов на атрибуты инфраструктуры

Архитектурный принцип	Ключевые технологии и практики	Влияние на доступность	Влияние на гибкость	Влияние на TCO
Масштабируемость	Микросервисы, контейнеризация, облачные сервисы, автоскейлинг.	Косвенное (через управление нагрузкой)	Кардинальное повышение	Снижение за счет оплаты по факту; риск роста при неконтролируемом потреблении.
Отказоустойчивость	Кластеризация, гео-распределение, балансировка нагрузки, хаос-инжиниринг.	Прямое кардинальное повышение	Повышение (за счет абстракции от железа)	Повышение (необходимость избыточных ресурсов).
Безопасность	Zero Trust, шифрование, сегментация, SIEM, Immutable Infrastructure.	Повышение (защита от инцидентов, ведущих к простоям)	Снижение (введение контролей)	Повышение (стоимость решений и эксплуатации).
Экономическая эффек-	FinOps-практики, автоматизация, TCO-анализ, гибридные архитектуры.	Косвенное (инвестиции в надежность)	Повышение (более гибкое распределение бюджета)	Прямое кардинальное снижение.

Современная цифровая инфраструктура – это динамическая, программно-определяемая экосистема, для которой масштабируемость, отказоустойчивость, безопасность и экономическая эффективность являются не отдельными техническими задачами, а интегрированными стратегическими целями. Их сбалансированная реализация,

часто требующая компромиссов (например, между абсолютной безопасностью и простотой использования или между максимальной отказоустойчивостью и бюджетом), определяет конкурентную устойчивость и операционную зрелость компании в цифровой экономике. Проектирование такой инфраструктуры – это непрерывный итеративный процесс, основанный на глубоком понимании бизнес-процессов, архитектурных паттернах и строгом управлении жизненным циклом ИТ-ресурсов.

2.3. Модели архитектуры: монолитная, сервис-ориентированная, микросервисная

Выбор базовой архитектурной модели цифровой инфраструктуры является фундаментальным стратегическим решением, определяющим гибкость, масштабируемость, стоимость владения и скорость разработки компании. Эволюция архитектурных подходов от монолитной к сервис-ориентированной и далее к микросервисной отражает ответ индустрии на растущие требования к адаптивности, сложности систем и необходимости непрерывной интеграции и доставки (CI/CD). Каждая модель представляет собой компромисс между связанностью (coupling) и связностью (cohesion) компонентов системы, что в свою очередь влияет на все уровни цифровой инфраструктуры: от развертывания и эксплуатации до организационной структуры команд разработки.

Монолитная архитектура исторически является исходной и наиболее простой моделью. В ней все функциональные компоненты приложения (логика представления, бизнес-логика, уровень доступа к данным) тесно интегрированы в единую кодовую базу и развертываются как единый процесс или исполняемый файл. Такая архитектура характеризуется высокой связностью и сильной связанностью модулей, которые взаимодействуют через вызовы функций или методов в рамках общего пространства памяти. На начальных этапах развития продукта монолит обладает неоспоримыми преимуществами: простота разработки, тестирования и развертывания (единая сборка), легкость отладки и высокая производительность внутри процессного взаимодействия. Однако по мере роста сложности и объема кодовой базы про-

являются его системные недостатки. Любое изменение, даже незначительное, требует перекомпиляции и полного повторного развертывания всего приложения, что увеличивает риски и снижает частоту обновлений. Масштабирование возможно только путем репликации всего монолита на несколько серверов (горизонтальное масштабирование), что неэффективно с точки зрения использования ресурсов, если «узким местом» является лишь одна функция. Кроме того, технологический стек оказывается «замороженным» на одном выборе, а долгосрочная поддержка кодовой базы становится все более сложной из-за накопления взаимозависимостей. Монолитная архитектура часто сопряжена с единой, централизованной моделью данных, что еще более усиливает связанность.

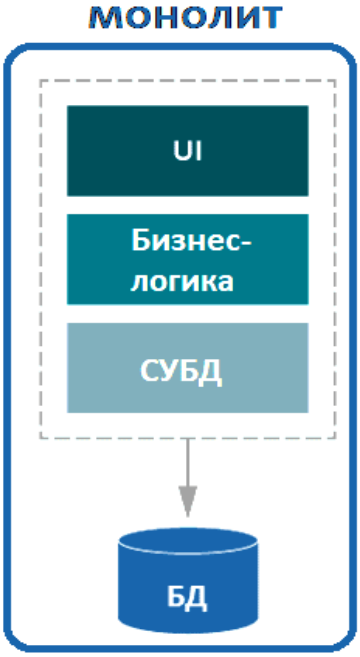

Рис. 2.2. Монолитная архитектура

Сервис-ориентированная архитектура (SOA) возникла как эволюционный ответ на ограничения монолита, предложив парадигму построения системы как набора слабосвязанных, повторно используемых сервисов, каждый из которых инкапсулирует четко определенную бизнес-функцию (например, «Управление заказами», «Расчет налогов»). Эти сервисы взаимодействуют друг с другом по стандартным, обычно сетевых протоколах, через формально определенные интерфейсы и контракты (рис. 2.3).

Рис. 2.3. Схема сервис-ориентированной архитектуры (SOA)

Это позволяет интегрировать разнородные системы, часто унаследованные (legacy), в рамках единого инфраструктурного стека. Сервисы в SOA, как правило, являются относительно крупными (крупнозернистыми) и ориентированы на реализацию целостных бизнес-процессов. Преимущества SOA включают: повышение гибкости за счет повторного использования сервисов, возможность независимого обновления и масштабирования отдельных бизнес-компонентов, улучшенную интеграцию разнородных систем. Однако внедрение SOA часто приводит к высокой сложности самой шины ESB, которая становится централизованным «узким местом» и точкой отказа. Развертывание и управление ESB требуют значительных экспертизы и ресурсов, а общая производительность системы может снижаться из-за накладных расходов на многократную трансформацию данных в шине.

Микросервисная архитектура представляет собой дальнейшее развитие и радикализацию идей сервис-ориентированности, но с акцентом на полную декомпозицию и автономность. Система строится как совокупность мелкозернистых, слабосвязанных сервисов (микросервисов), организованных вокруг конкретных бизнес-возможностей (например, «Каталог товаров», «Корзина покупок», «Служба уведомлений»). Каждый микросервис является независимым единицей развертывания: имеет собственную кодовую базу, базу данных (принцип «сервис владеет своими данными») и процесс. Он может быть разработан, развернут и масштабирован независимо от других, что позволяет использовать различные технологические стеки (полиглотное программирование и хранение данных) в рамках одного приложения (рис. 2.4).

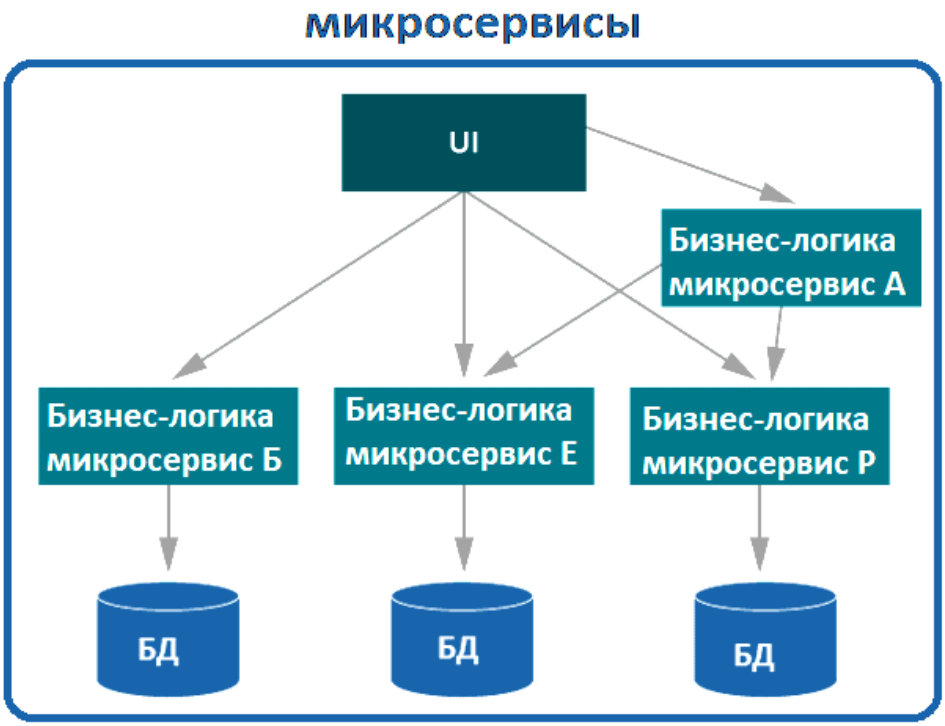

Рис. 2.4. Микросервисная архитектура

Взаимодействие между микросервисами осуществляется через легковесные механизмы, чаще всего через RESTful API или асинхронный обмен сообщениями (message brokers, например, RabbitMQ, Apache Kafka). Централизованная шина ESB устраняется, а ее функции (маршрутизация, балансировка нагрузки, безопасность) выносятся в инфраструктурный уровень, часто реализуемый с помощью технологий контейнеризации (Docker) и оркестрации (Kubernetes), а также шаблонов API Gateway и Service Mesh. Это позволяет достичь исключительной гибкости, устойчивости и скорости внесения изменений, делает систему пригодной для практик DevOps и CI/CD. Однако микросервисная архитектура привносит значительную распределенную сложность. Усложняются операции: мониторинг, трассировка запросов, управление конфигурациями и обеспечение согласованности данных (требуются паттерны Saga, CQRS). Требуются зрелые DevOps-культура и автоматизированные процессы сборки, тестирования и развертывания. Накладные расходы на межсервисную сетевую коммуникацию могут влиять на производительность.

Сравнительный анализ ключевых характеристик трех архитектурных моделей представлен в табл. 2.4.

Таблица 2.4

Сравнительный анализ архитектурных моделей

Критерий	Монолитная архитектура	Сервис-ориентированная архитектура (SOA)	Микросервисная архитектура
Гранулярность	Крупная (единое приложение)	Крупная/средняя (бизнес-сервисы)	Мелкая (одна функция/ возможность)
Связность (Cohesion)	Высокая	Средняя (в рамках сервиса)	Высокая (в рамках микросервиса)
Связанность (Coupling)	Очень сильная (внутрипроцессная, компиляционная)	Слабая (через контракты, часто при посредничестве ESB)	Очень слабая (через API, асинхронные сообщения)
Управление данными	Единая, централизованная БД	Глобальная схема данных, возможна интеграция через ESB	Распределенная, каждый сервис владеет своей схемой
Механизм интеграции	Внутрипроцессные вызовы (функции)	Централизованная шина (ESB), стандартные протоколы (SOAP и др.)	Легковесные протоколы (REST, gRPC, асинхрон. сообщения)
Единица развертывания	Целое приложение	Отдельные сервисы (но часто зависящие от ESB)	Независимый микросервис (контейнер)
Масштабируемость	Горизонтальное, неэффективное (масштабируется все)	Горизонтальное на уровне сервисов	Точное, независимое масштабирование каждого сервиса
Технологическая гибкость	Единый стек технологий	Ограниченная гибкость в рамках поддержки ESB	Полиглотность (разные стеки для разных сервисов)
Сложность разработки	Низкая на старте, экспоненциально растет с ростом	Высокая (особенно интеграционная)	Высокая (распределенная системная сложность)
Сложность эксплуатации	Низкая (единый процесс)	Очень высокая (управление ESB, мониторинг шины)	Высокая (оркестрация, мониторинг распределенной системы)
Подход к отказам	Единая точка отказа	ESB - потенциальная точка отказа	Устойчивость за счет изоляции отказов

Стратегический выбор модели должен основываться на комплексном анализе контекста компании. Монолит остается рациональным выбором для небольших проектов с четко очерченными границами, ограниченными командами и необходимостью быстрого старта.

SOA может быть оправдана в крупных корпоративных экосистемах с множеством унаследованных систем, где критически важна стандартизированная интеграция на уровне предприятия. Микросервисы наиболее соответствуют стратегиям, ориентированным на цифровую трансформацию, высокие темпы выпуска изменений, независимость продуктовых команд и эксплуатацию в облачных средах с высокой динамикой нагрузки. Важно отметить, что эволюционный переход от монолита к микросервисам через промежуточные состояния (модульный монолит, SOA) часто является более безопасным и управляемым путем, чем радикальная перестройка.[9]

Таким образом, архитектура цифровой инфраструктуры компании не является статичной; она должна эволюционировать в соответствии с меняющимися бизнес-целями, технологическим ландшафтом и организационной зрелостью, а выбор модели служит основой для построения адаптивной и эффективной ИТ-экосистемы.

2.4. Роль архитектора цифровой инфраструктуры компании и его взаимодействие с бизнес-заказчиками

В современной цифровой экономике и экономики данных архитектор цифровой инфраструктуры (Architect of Digital Infrastructure, ADI) трансформировался из технического специалиста в стратегического игрока, обеспечивающего трансляцию бизнес-целей в технологически реализуемые и экономически эффективные решения. Его роль заключается в проектировании, описании, управлении и эволюции фундаментальных компонентов информационно-технологического ландшафта компании — вычислительных ресурсов, систем хранения, сетевых коммуникаций, платформ виртуализации и облачных сервисов, — рассматриваемых как единая целостная система. Ключевым аспектом успешного выполнения этой миссии является эффективное, структурированное и непрерывное взаимодействие с бизнес-заказчиками (стейкхолдерами, владельцами бизнес-процессов). Это взаимодействие выступает связующим звеном между стратегическим видением бизнеса и операционно-технологической реализацией.

[9] Федоров А. Что такое ИТ-инфраструктура и из каких компонентов она состоит [Электронный ресурс]/ А. Федоров // Режим доступа: https://cyberprotect.ru/blog/it-infrastructure-intro?ysclid=mk1kblc7jg194849748 (дата обращения: 05.01.2026).

Стратегический контекст и зоны ответственности ADI
Роль ADI определяется на стыке трех стратегических плоскостей: бизнес-стратегии компании, ИТ-стратегии и собственно стратегии развития цифровой инфраструктуры. ADI не является пассивным исполнителем запросов, а активным со-проектировщиком бизнес-возможностей. Его ответственность включает:

1. Стратегическое выравнивание (Strategic Alignment): обеспечение того, что каждый элемент инфраструктуры вносит вклад в достижение ключевых бизнес-показателей (KPI), таких как ускорение вывода продукта на рынок (time-to-market), обеспечение непрерывности бизнеса, снижение операционных рисков.

2. Архитектурное проектирование и стандартизация: разработка принципов, стандартов и шаблонов (patterns) для инфраструктуры, обеспечивающих ее масштабируемость, отказоустойчивость, безопасность и экономическую эффективность.

3. Управление технологическим портфелем: оценка, отбор и внедрение новых технологий (гибридные облака, контейнеризация, edge-вычисления, платформы для анализа данных), формирующих конкурентные преимущества.

4. Управление жизненным циклом и эволюцией: планирование модернизации и вывода из эксплуатации устаревших систем, минимизация технического долга инфраструктуры.

5. Управление затратами и оптимизация: анализ стоимостных моделей (CAPEX/OPEX), контроль за эффективностью использования ресурсов, перевод затрат из категории накладных расходов в категорию управляемых инвестиций.[10] Ключевая разница между этими моделями состоит в том, что CAPEX - это инвестиции в будущее, которые позволяют компании расти, масштабироваться и приобретать долгосрочные активы. Эти расходы отражаются в балансе и влияют на финансовое состояние в течение нескольких лет. В свою очередь, OPEX - это расходы на поддержание текущей деятельности. Они регулярны, напрямую влияют на прибыль, требуют постоянного контроля и оптимизации.

[10] Что такое CAPEX и OPEX и зачем их считать[Электронный ресурс]// Режим доступа: https://sales-generator.ru/blog/capex-i-opex/ (дата обращения: 08.01.2026).

Для эффективного финансового управления цифровой инфраструктурой важно чётко разделять операционные и капитальные расходы. В табл. 2.5 приведено подробное сравнение CAPEX и OPEX по основным критериям.

Таблица 2.5

Сравнительный анализ моделей CAPEX и OPEX[11]

Параметр	OPEX (операционные расходы)	CAPEX (капитальные затраты)
Цель затрат	Обеспечить ежедневную работу компании, поддерживать текущие процессы, генерировать выручку	Инвестировать в долгосрочные активы, расширять и развивать бизнес
Характер расходов	Повторяющиеся, регулярные, с краткосрочным эффектом	Единовременные или крупные, с расчётом на долгосрочную выгоду
Период действия	Затраты относятся к текущему периоду (месяц, квартал)	Актив используется на протяжении нескольких лет
Отражение в учете	Списываются на расходы в отчёте о прибылях и убытках (P&L)	Учёт в балансе как актив с последующей амортизацией
Влияние на прибыль	Уменьшают чистую прибыль текущего периода	Увеличивают активы, но не влияют напрямую на прибыль в момент покупки
Влияние на финансовую устойчивость	Высокая доля OPEX может свидетельствовать о высокой нагрузке на бизнес	Рост CAPEX — признак инвестиционной активности и долгосрочной стратегии
Отражение в отчетности	Отчёт о прибылях и убытках, управленческий отчёт по затратам	Отчёт о движении денежных средств (инвестиционная деятельность), бухгалтерский баланс
Примеры	Зарплата, аренда, реклама, коммунальные услуги, закупка сырья	Покупка зданий, оборудования, земельных участков, лицензий, дорогостоящего ПО

[11] CAPEX и OPEX: что это, разница, как рассчитать и анализировать [Электронный ресурс]// Режим доступа: https://vladimir.1cbit.ru/blog/capex-i-opex-chto-eto-raznitsa-kak-rasschitat-i-analizirovat/?utm_referrer=https%3A%2F%2Fya.ru%2F (дата обращения: 08.01.2026).

Двухуровневая модель взаимодействия с бизнес-заказчиками взаимодействия ADI с бизнесом носит многоуровневый характер, который можно структурировать в виде двух взаимосвязанных контуров: стратегического и тактико-операционного (табл. 2.6).

Для успешной работы в данной модели ADI должен обладать гибридным набором компетенций:

1. Бизнес-компетенции: понимание основ бизнес-модели компании, ее финансовых показателей, отраслевой специфики и конкурентной среды. Способность говорить на языке бизнес-ценностей, а не технологических спецификаций.

2. Коммуникативные и фасилитационные навыки: умение вести диалог, выявлять скрытые потребности, проводить интервью и воркшопы, визуализировать сложные концепции для нетехнической аудитории.

3. Системное мышление: способность видеть компанию как целостную систему, где изменение в одном бизнес-процессе влечет изменения в требованиях к инфраструктуре.

4. Технологическая экспертиза: глубокое знание технологий, рыночных предложений, трендов, что позволяет предлагать адекватные и современные решения.

5. Управление ожиданиями и компромиссами: четкое обозначение границ возможного, объяснение последствий выбора между, например, максимальной доступностью и минимальной стоимостью.

Таблица 2.6

Двухуровневая модель взаимодействия архитектора цифровой инфраструктуры с бизнес-заказчиками

Уровень взаимодействия	Цель и фокус	Ключевые механизмы и артефакты	Результат
Стратегический контур	Выравнивание и совместное формирование долгосрочной технологической дорожной карты, отвечающей бизнес-стратегии.	Участие в стратегических сессиях и воркшопах. Анализ бизнес-стратегии и рыночных трендов. Разработка архитектурных видений и сценариев будущего (future-state scenarios). Представление технологических возможностей как драйверов бизнес-инноваций.	Дорожная карта развития цифровой инфраструктуры. Архитектурные принципы, утвержденные на уровне бизнес-руководства. Бюджетные заявки стратегического характера.
Тактико-операционный контур	Перевод конкретных бизнес-требований (проектов, инициатив) в технические спецификации и контроль их реализации.	Работа с бизнес-требованиями (functional) и атрибутами качества (non-functional requirements: производительность, доступность, безопасность). Участие в проектных комитетах и рабочих группах. Разработка архитектурных решений (Solution Design) и моделей развертывания. Управление компромиссами (trade-off analysis).	Согласованные технические задания и архитектурные спецификации для проектов. Баланс между инновационностью, стоимостью, риском и временем реализации. Обеспечение соблюдения стандартов и принципов в реализуемых проектах.

Взаимодействие можно представить в виде итеративного процесса, состоящего из нескольких фаз, представленных в табл. 2.7.

Таблица 2.7

Фазы процесса взаимодействия ADI с бизнес-заказчиком в рамках проекта/инициативы

Фаза	Действия ADI	Инструменты и артефакты	Критерий завершения
1. Выявление и анализ требований	Диалог с заказчиком для перевода бизнес-идеи в структурированные требования. Акцент на атрибуты качества (производительность, доступность 99.99%, время восстановления).	Шаблоны для сбора требований. Матрица трассируемости требований. Описание бизнес-сценариев (use cases).	Согласованный и приоритизированный перечень функциональных и нефункциональных требований.
2. Разработка и оценка вариантов решения	Проектирование одного или нескольких архитектурных вариантов, удовлетворяющих требованиям. Оценка каждого по критериям стоимость, риск, сложность, соответствие стандартам.	Диаграммы развертывания и компонентные модели. Модели затрат (TCO-анализ). Оценка рисков и узких мест.	Презентация вариантов решения бизнес-заказчику и ИТ-руководству с четким обоснованием рекомендаций.
3. Согласование и принятие решения	Организация обсуждения, объяснение технических компромиссов на языке бизнес-последствий. Например, «повышение доступности на 0.1% увеличит стоимость на X, но снизит риски простоя, оцениваемые в Y у.ден.ед./час».	Совещания по принятию архитектурных решений (Architecture Review Board). Финализированная спецификация решения	Подписанный архитектурный решений документ (Architecture Decision Record, ADR), утвержденный всеми сторонами.
4. Контроль реализации и обратная связь	Консультации команд разработки и эксплуатации, контроль ключевых точек реализации на соответствие утвержденному проекту.	Участие в статус-совещаниях проекта. Ревизия ключевых конфигураций.	Успешный ввод решения в эксплуатацию, соответствующий исходным требованиям.
5. Мониторинг ценности и эволюция	Анализ того, как внедренное решение повлияло на целевые бизнес-показатели. Сбор обратной связи для будущих улучшений.	Отчеты о выполнении KPI инфраструктуры. Регулярные встречи по развитию сервиса.	Подтверждение достижения ожидаемой бизнес-ценности, актуализация дорожной карты.

Взаимодействие ADI и бизнес-заказчика часто сопряжено с рядом системных проблем:

- Разрыв в терминологии: бизнес оперирует понятиями рентабельности, клиентской удовлетворенности, рыночной доли, ADI - пропускной способностью, задержками, IOPS.

Путь преодоления состоит в разработке глоссария и использование метафор, создание перекрестных ссылок между бизнес-возможностями и технологическими компонентами.

- Неполные или меняющиеся требования: бизнес часто формулирует требования в общих чертах или меняет их в процессе.

Путь преодоления заключается в использовании итеративных подходов (Agile), прототипирование, явное управление изменениями требований через формальные процедуры.

- Конфликт приоритетов: бизнес стремится к скорости и инновациям, ADI - к стабильности, безопасности и стандартизации.

Путь преодоления состоит во внедрении гибких, но управляемых платформ (Platform Engineering), которые предоставляют бизнесу стандартизированные «строительные блоки» для быстрой сборки решений, сохраняя контроль.

- Отсутствие доверия и прозрачности: бизнес может воспринимать ИТ, и ADI, в частности, как «черный ящик! с высокими затратами.

Путь преодоления заключается в подготовке регулярной отчетности на языке бизнес-ценности, прозрачные модели расчета затрат (например, показ стоимости на бизнес-единицу или транзакцию), демонстрация успешных кейсов.

Таким образом, архитектор цифровой инфраструктуры выступает в роли ключевого транслятора и интегратора, чья деятельность направлена на создание не просто технологически совершенной, но бизнес-релевантной, адаптивной и экономически обоснованной инфраструктурной основы. Его эффективное взаимодействие с бизнес-заказчиками, построенное на принципах стратегического партнерства, прозрачности и совместного проектирования, является критическим фактором успеха цифровой трансформации и обеспечения долгосрочной конкурентоспособности компании. Переход от пассивного обслуживания запросов к активному формированию технологической повестки бизнеса является отличительной чертой ADI нового поколения,

превращающей цифровую инфраструктуру из центра затрат в драйвер роста и инноваций.

Подводя итог сказанному выше, необходимо заключить, что цифровая инфраструктура эволюционировала из пассивной поддерживающей среды в активный стратегический актив, чья эффективность напрямую определяет операционную жизнеспособность, инновационный потенциал и адаптивность организации.

Ключевым императивом, обеспечивающим эту трансформацию, является процесс стратегического выравнивания ИТ-стратегии с бизнес-целями. Как показано, это не разовое мероприятие, а циклическая управленческая дисциплина, реализуемая на четырех взаимосвязанных уровнях: от трансляции бизнес-стратегии в цели ИТ через тактическое архитектурное проектирование и операционную реализацию до постоянного измерения результативности и обратной связи. Успех данного процесса детерминирован не столько технологическим выбором, сколько качеством организационных механизмов: вовлеченностью руководства, работой кросс-функциональных команд, эффективными процессами управления требованиями и изменениями. Преодоление системных барьеров, таких как терминологический разрыв, наследие устаревших систем и раздельное бюджетирование, выступает необходимым условием для перевода ИТ из центра затрат в источник создания стоимости.

Стратегическое выравнивание находит свою конкретную материализацию в принципах проектирования современной инфраструктуры. Анализ подтвердил, что масштабируемость, отказоустойчивость, безопасность и экономическая эффективность представляют собой не изолированные технические требования, а комплекс взаимозависимых и зачастую конфликтующих атрибутов, требующих сбалансированных архитектурных компромиссов. Достижение требуемых уровней доступности (SLA) или соблюдение регуляторных норм безопасности напрямую коррелирует с увеличением совокупной стоимости владения (TCO), что требует от управления взвешенного, экономически обоснованного подхода, такого как дисциплина FinOps в облачных средах. Таким образом, проектирование инфраструктуры является непрерывным итеративным процессом оптимизации, основанным на глубоком понимании бизнес-процессов и их требований к производительности, надежности и затратам.

Выбор фундаментальной архитектурной модели - монолитной, сервис-ориентированной (SOA) или микросервисной представляет собой стратегическое решение, определяющее долгосрочную гибкость и эволюционный потенциал компании. Исследование выявило, что каждая модель является закономерным ответом на определенный этап зрелости бизнеса и ИТ. Монолит сохраняет рациональность для небольших, стабильных систем, SOA служит для интеграции сложных корпоративных экосистем с унаследованными системами, в то время как микросервисная архитектура становится оптимальным выбором для организаций, ориентированных на высокие темпы цифровых инноваций, независимость команд и эксплуатацию в динамичных облачных средах. При этом эволюционный переход между моделями является более предпочтительным и управляемым путем, чем радикальная перестройка.

Центральной фигурой, обеспечивающей синтез стратегических установок, архитектурных принципов и технологического выбора, выступает архитектор цифровой инфраструктуры (ADI). Его роль трансформировалась от узкого технического специалиста к стратегическому партнеру бизнеса и интегратору. Эффективность ADI определяется способностью к двустороннему переводу: бизнес-требований в архитектурные спецификации и технологических возможностей — в язык бизнес-ценности. Как показано, критическим фактором успеха является выстраивание структурированного, многоуровневого взаимодействия с бизнес-заказчиками через стратегический и тактико-операционный контуры, основанного на гибридном наборе бизнес-компетенций, системном мышлении и навыках управления компромиссами.

Таким образом, стратегия и архитектура цифровой инфраструктуры формируют единый контур управления, где стратегическое видение, инженерные принципы, технологический выбор и организационные роли находятся в тесной взаимозависимости. Успешная реализация этого контура обеспечивает переход от реактивного обслуживания запросов к проактивному формированию цифровых возможностей бизнеса. В конечном итоге, именно зрелость в управлении стратегией и архитектурой цифровой инфраструктуры позволяет компании трансформировать свои технологические инвестиции в драйвер роста, устойчивости и долгосрочной конкурентоспособности.

Вопросы для обсуждения

1. Дайте определение выравниванию ИТ-стратегии с бизнес-целями.

2. Укажите ступени (уровни) модели, связывающей корпоративное стратегическое планирование с проектированием и эксплуатацией цифровой инфраструктуры.

3. Объясните связь бизнес-целей с элементами архитектуры цифровой инфраструктуры

4. Поясните, с какими системными барьерами сталкивается процесс выравнивания цифровой инфраструктуры.

5. Укажите, какие специализированные управленческие и архитектурные фреймворки используются для практической реализации модели выравнивания бизнес-стратегий и цифровой инфраструктуры.

6. Поясните основные принципы проектирования современной инфраструктуры

7. Дайте определение масштабируемости, перечислите ее основные виды и формы.

8. Объясните принципы отказоустойчивости и безопасности при проектировании цифровой инфраструктуры и ее отдельных элементов

9. Поясните направления использования методологии управления облачными финансами FinOps (Financial Operations) .

10. Охарактеризуйте уровни доступности инфраструктуры и их бизнес-интерпретация.

11. Укажите направления влияния архитектурных принципов на атрибуты инфраструктуры.

12. Дайте определение монолитной архитектуре, укажите достоинства и недостатки ее применения.

13. Укажите направления применения сервис-ориентированной архитектуры (SOA) при разработке элементов цифровой инфраструктуры.

14. Поясните достоинства и «узкие места» микросервисной архитектуры при проектировании элементов цифровой инфраструктуры.

15. Укажите стратегический контекст и зоны ответственности модели ADI.

16. Поясните направления применения модели CAPEX

17. Объясните специфику использования модели OPEX при проектировании элементов цифровой инфраструктуры.

18. Охарактеризуйте двухуровневую модель взаимодействия архитектора цифровой инфраструктуры с бизнес-заказчиками

19. Перечислите гибридный набор компетенций для успешной работы архитектора в модели ADI при построении цифровой инфраструктуры компании.

20. Перечислите фазы процесса взаимодействия ADI с бизнес-заказчиком в рамках проекта/инициативы.

Практические задания

Задание 1. Проанализируйте четырехуровневую модель стратегического выравнивания ИТ-стратегии с бизнес-целями. Для каждого уровня:

- стратегический;
- тактический;
- операционный;
- измерение и обратная связь.

Приведите по одному конкретному примеру деятельности или решения по изменению элементов или параметров инфраструктуры, которое должно быть реализовано на данном уровне в компании, стремящейся к цифровой трансформации.

Обоснуйте выбор примеров, опираясь размер этой компании, ее отраслевую принадлежность, специфику работу, номенклатуру выпускаемой продукции или оказываемых услуг.

Задание 2. Разработайте сравнительную таблицу современной компании на основе принципов проектирования современной цифровой инфраструктуры:

- масштабируемость;

- отказоустойчивость;
- безопасность;
- экономическая;
- эффективность.

В таблице для каждого принципа укажите:

1. Два ключевых технологических решения или практики для его реализации.

2. Один потенциальный конфликт с другим принципом и возможный компромисс для его разрешения. Аргументируйте предложенные технологические решения и компромиссы, данные о компании (отраслевая принадлежность, ее размер, номенклатуру выпускаемой продукции или оказываемых услуг).

Задание 3. Сравните варианты внедрения архитектурных моделей (монолитная, сервис-ориентированная, микросервисная) в деятельность небольшой производственной компании (отраслевая принадлежность не имеет значения). Выберите одну из бизнес-стратегий: «Ускорение вывода новых продуктов на рынок» или «Обеспечение соответствия регуляторным требованиям».

Определите, какая архитектурная модель наиболее соответствует выбранной стратегии для этой компании?

Обоснуйте свой выбор, указав не менее трех причин, опираясь на характеристики моделей и роль архитектора цифровой инфраструктуры в их реализации.

Тест для самоконтроля

1. Что представляет собой выравнивание ИТ-стратегии с бизнес-целями?

а) Разовое техническое мероприятие по обновлению оборудования.

б) Непрерывный процесс согласования целей, приоритетов и деятельности в области ИТ со стратегическими задачами бизнеса.

в) Процесс сокращения ИТ-бюджета.

г) Автоматизация всех бизнес-процессов.

2. Какой из перечисленных уровней не входит в многоуровневую модель выравнивания ИТ и бизнеса?

а) Тактический.

б) Стратегический.

в) Финансовый.

г) Операционный.

3. Для какой бизнес-цели ключевым требованием к архитектуре является «микросервисная архитектура и инфраструктура как код (IaaC)»?

а) Обеспечение соответствия регуляторным требованиям.

б) Оптимизация операционных издержек.

в) Ускорение вывода новых продуктов/ услуг на рынок.

г) Повышение клиентоцентричности.

4. Что из перечисленного является системным барьером для процесса выравнивания?

а) Использование облачных технологий.

б) Разрыв в языках и ментальных моделях между бизнесом и ИТ.

в) Внедрение гибких методологий.

г) Наличие кросс-функциональных команд.

5. Что означает горизонтальное масштабирование?

а) Увеличение мощности отдельного узла (сервера).

б) Уменьшение количества сервисов.

в) Увеличение производительности за счёт добавления однотипных узлов в пул ресурсов.

г) Консолидация всех данных в единое хранилище.

6. Какой уровень доступности инфраструктуры (примерно 8 часов 46 минут простоя в год) соответствует классу «Корпоративный»?

а) 99.0%

б) 99.9%

в) 99.99%

г) 99.999%

7. Какой принцип проектирования подразумевает интеграцию мер кибербезопасности на каждом этапе, а не их наложение на готовую систему?

а) Экономическая эффективность.

б) Масштабируемость.

в) Безопасность по замыслу.

г) Отказоустойчивость.

8. Какая дисциплина фокусируется на оптимизации совокупной стоимости владения (TCO) в облачных средах через осознанное управление ресурсами?

а) ITIL.

б) DevOps.

в) FinOps.

г) COBIT.

9. Какая характеристика не является типичной для монолитной архитектуры?

а) Высокая связность компонентов.

б) Простота начальной разработки и развертывания.

в) Независимое масштабирование отдельных функций.

г) Единая кодовая база.

10. Что является центральным элементом и потенциальной точкой отказа в сервис-ориентированной архитектуре (SOA)?

а) Балансировщик нагрузки.

б) Шина предприятия.

в) Контейнер.

г) API-шлюз.

11. Какое утверждение верно для микросервисной архитектуры?

а) Все сервисы используют единую, централизованную базу данных.

б) Каждый микросервис имеет собственную кодовую базу и может быть развёрнут независимо.

в) Взаимодействие между сервисами происходит только через центральную шину.

г) Она характеризуется самой низкой распределённой сложностью по сравнению с монолитом и сервис-ориентированной архитектурой.

12. Что из перечисленного является ключевой стратегической ответственностью архитектора цифровой инфраструктуры?

а) Ручная настройка сетевого оборудования.

б) Стратегическое выравнивание инфраструктуры с бизнес-показателями .

в) Написание кода для бизнес-приложений.

г) Ежедневная техническая поддержка пользователей.

13. CAPEX, в отличие от OPEX...

а) Регулярно списываются на расходы в отчёте о прибылях и убытках.

б) Учитываются в балансе как актив с последующей амортизацией.

в) Имеют краткосрочный характер.

г) Напрямую уменьшают чистую прибыль текущего периода.

14. На каком уровне взаимодействия с бизнесом ADI участвует в стратегических сессиях и разрабатывает архитектурное видение будущего?

а) Тактико-операционный контур.

б) Стратегический контур.

в) Контур технической поддержки.

г) Контрольный контур.

15. В какой фазе процесса взаимодействия ADI с заказчиком происходит подписание архитектурного документа решения?

а) Выявление и анализ требований.

б) Разработка и оценка вариантов решения.

в) Согласование и принятие решения.

г) Мониторинг ценности и эволюция.

16. Что из перечисленного не является рекомендуемым путём преодоления разрыва в терминологии между бизнесом и ADI?

а) Разработка общего глоссария.

б) Использование метафор и перекрёстных ссылок.

в) Полный отказ от использования технических терминов.

г) Объяснение технологических концепций на языке бизнес-ценности.

17. Какой архитектурный принцип оказывает прямое кардинальное влияние на повышение доступности системы?

а) Экономическая эффективность.

б) Масштабируемость.

в) Отказоустойчивость.

г) Безопасность.

18. Согласно тексту, какой фактор является критически важным для успеха выравнивания ИТ и бизнеса?

а) Выбор самого дорогого и современного оборудования.

б) Качество управленческих процессов и глубина взаимопонимания между бизнес- и ИТ-лидерами.

в) Полный отказ от унаследованных систем.

г) Максимальная централизация всех ИТ-решений.

19. Какой подход к масштабированию связан с добавлением процессоров или памяти к существующему серверу и часто требует его остановки?

а) Горизонтальное масштабирование.

б) Диагональное масштабирование.

в) Вертикальное масштабирование.

г) Автоматическое масштабирование.

20. Какая модель архитектуры, согласно тексту, наиболее подходит для крупных корпоративных экосистем с множеством унаследованных систем, где критически важна стандартизированная интеграция?

а) Монолитная архитектура.

б) Сервис-ориентированная архитектура.

в) Микросервисная архитектура.

г) Бессерверная архитектура.

Глава 3. КОМПОНЕНТЫ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ И ИХ УПРАВЛЕНИЕ

3.1. Вычислительные ресурсы: серверы и системы хранения данных

В основе современной цифровой инфраструктуры лежат вычислительные ресурсы и системы хранения данных, которые обеспечивают выполнение прикладных задач, обработку и сохранение информации. Их рациональная организация, выбор архитектуры и эффективное управление являются критическими факторами для обеспечения производительности, надежности, масштабируемости и экономической целесообразности ИТ-систем. В этой связи необходимо рассмотреть ключевые компоненты этих ресурсов: от эволюции серверных платформ (от физических серверов к виртуализации и контейнеризации) до специализированных систем хранения данных (SAN и NAS).

Сервер представляет собой специализированный вычислительный ресурс, предназначенный для непрерывного выполнения служб и обработки запросов от клиентских устройств или других серверов. За последние два десятилетия произошла фундаментальная трансформация в способах развертывания и предоставления серверных мощностей.

Различают физические серверы (Bare-Metal) и виртуальные серверы (виртуальные машины или ВМ)

Физический сервер — это автономная аппаратная единица, состоящая из процессора(ов), оперативной памяти, подсистем ввода-вывода и накопителей, на которой напрямую, без промежуточного слоя абстракции, установлена операционная система и прикладное программное обеспечение. Их архитектурные особенности состоят в высокой степени детерминированности и производительности, так как ресурсы не разделяются с другими workloads, а конфигурация жестко привязана к аппаратному обеспечению.

Несомненными преимуществами Bare-Metal являются максимальная производительность для ресурсоемких задач (высокопроизводительные вычисления - HPC, СУБД с интенсивным вводом-выводом), простота управления для изолированных систем, отсутствие наклад-

ных расходов на гипервизор. В свою очередь, они не лишены недостатков, состоящих в низкой эффективности использования ресурсов (загрузка ЦПУ редко превышает 10-15% в среднем), длительный процесс развертывания и масштабирования («закупка-установка-настройка»), высоких затрат на электроэнергию и охлаждение, привязке приложения к конкретному аппаратному обеспечению.

Основная область применения физических серверов - это критичные к задержкам и производительности системы, унаследованные приложения, требующие специфичного аппаратного обеспечения, высокопроизводительные СУБД.

Виртуализация серверов стала прорывной технологией, позволившей преодолеть недостатки физической модели. В ее основе лежит гипервизор (Type 1 - «голый металл», например, VMware vSphere, Microsoft Hyper-V, а также Type 2 — хостовая операционная система (далее ОС)), который абстрагирует физические ресурсы сервера (ЦПУ, память, диски, сеть) и создает на их основе изолированные виртуальные экземпляры — виртуальные машины. Каждая ВМ содержит собственную гостевую операционную систему.

Принцип работы виртуальных сервером заключается в следующем: гипервизор осуществляет планирование и выделение физических ресурсов между конкурирующими ВМ, обеспечивая изоляцию на уровне оборудования (рис. 3.1).

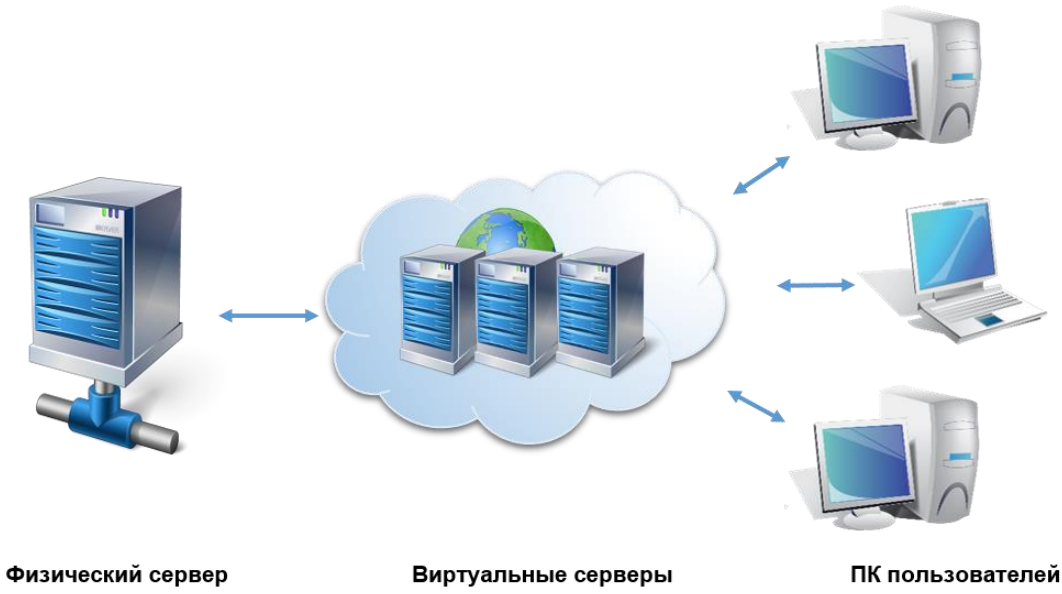

Рис. 3.1. Схема работы виртуального сервера

Преимущества ВМ состоит в ряде параметров их работы, а именно:

- Консолидация: высокая степень утилизации ресурсов за счет одновременной работы множества ВМ на одном физическом хосте.

- Изоляция: сбой или конфигурационная ошибка в одной ВМ не затрагивает другие.

- Агностицизм оборудования: ВМ представляет собой набор файлов (образ диска, конфигурация), что обеспечивает мобильность, быстрое развертывание, клонирование и миграцию между физическими хостами без простоя.

- Управляемость: централизованное управление пулом ресурсов кластера.

«Узкими местами» в работе виртуальных серверов являются высокие накладные расходы на работу гипервизора и дублирование гостевых ОС в каждой ВМ (потребление памяти, затраты на лицензирование и патчинг), относительно больший объем виртуального образа (десятки ГБ), не всегда оптимальная производительность для ultra-low-latency задач (т.е. задач, для которых минимальна задержка между действием и реакцией оборудования).

Основная область применения виртуальных серверов – это использование в универсальных платформах для подавляющего большинства корпоративных приложений, тестовых и изолированных сред, систем, требующих различных версий ОС.

Сравнительная характеристика физических и виртуальных серверов представлена в табл. 3.1.

Контейнеризация представляет собой следующую ступень абстракции — виртуализацию на уровне операционной системы. В отличие от ВМ, контейнеры разделяют ядро хостовой ОС, но изолируют пользовательское пространство (процессы, файловую систему, сеть). Технология основана на возможностях ядра Linux (namespaces, cgroups), а стандартом де-факто является Docker с экосистемой оркестраторов (Kubernetes).[12]

[12] Какие технологии помогают бизнесу построить единую ИТ-инфраструктуру [Электронный ресурс] // Режим доступа: https://digtlab.ru/tpost/rzfhfyfrr1-kakie-tehnologii-pomogayut-biznesu-postr?ysclid=mk2z328jiu499207780 (дата обращения: 06.01.2026).

Таблица 3.1

Сравнительная характеристика физических и виртуальных серверов

Критерий	Физический сервер (Bare-Metal)	Виртуальный сервер (ВМ)
Уровень абстракции	Аппаратное обеспечение	Аппаратное обеспечение + гипервизор
Единица развертывания	Аппаратный сервер	Виртуальная машина (файл-образ)
Изоляция	Физическая	Аппаратно-виртуальная (гипервизором)
Эффективность использования ресурсов	Низкая (15-20%)	Высокая (60-80% и выше)
Время развертывания	Дни/недели	Минуты
Мобильность (миграция)	Сложная, с простоем	Простая, без простоя (live migration)
Накладные расходы	Минимальные	Гипервизор, гостевая ОС в каждой ВМ
Ключевое преимущество	Максимальная производительность и детерминизм	Гибкость, консолидация, управляемость

Архитектурные особенности этого элемента цифровой инфраструктуры состоят в том, что контейнер содержит только приложение и его зависимости (библиотеки, переменные среды), но не включает полную ОС. Все контейнеры на хосте выполняются поверх общего ядра ОС.

Преимущества контейнеров заключаются в следующих аспектах:

- Экстремальная легкость и скорость: образы контейнеров на порядок меньше образов ВМ (МБ против ГБ), запуск происходит за секунды.

- Высокая плотность размещения: на одном хосте можно запустить в десятки раз больше экземпляров приложений, чем в модели ВМ.

- Идемпотентность и переносимость: обеспечивает идентичное исполнение приложения в любой среде (разработка, тестирование, производство).

Микросервисная архитектура: идеально подходят для декомпозиции монолитных приложений на независимые, слабосвязанные микросервисы.

Недостатки контейнеров состоит в менее строгой изоляции, чем у ВМ (уязвимость на уровне ядра ОС затрагивает все контейнеры). Кроме того, все контейнеры на хосте должны быть совместимы с одной версией/типом ядра ОС (Linux).

Областью применения контейнеров являются микросервисные приложения, DevOps-практики (CI/CD), облачно-нативные приложения, пакетная обработка данных, системы с требованием быстрого горизонтального масштабирования.

Для обеспечения работы серверных платформ и сохранения данных используются специализированные системы хранения, которые по методу предоставления доступа делятся на два основных класса:

- SAN (Storage Area Network);
- NAS (Network Attached Storage).

SAN – это выделенная высокопроизводительная сеть, которая предоставляет блочный уровень доступа к данным. SAN абстрагирует физические дисковые массивы и представляет их серверам в виде «логических единиц» (LUN — Logical Unit Number), которые система воспринимает как локальные диски (рис. 3.2).

В основе SAN лежат сети на основе протоколов Fibre Channel (FC-SAN, высокая производительность, низкая задержка) или iSCSI (IP-SAN, использование стандартных Ethernet-сетей, более экономичное решение). Ключевые компоненты: дисковые массивы, коммутаторы (FC или Ethernet), адаптеры на серверах (HBA — Host Bus Adapter). Модель доступа SAN – блочный уровень (block-level), когда сервер получает сырые блоки данных и самостоятельно управляет файловой системой на предоставленной LUN.[13]

Преимущества SAN состоят в высочайшей производительности и низкой задержке, централизованном управлении и резервном копированиие на уровне блоков, возможности создания кластеров высокого уровня доступности (например, отказоустойчивые кластеры Microsoft, VMware vSphere HA/FT), поддержке advanced-функций (моментальные снимки – snapshots, репликация на уровне блоков).

В свою очередь, слабые места этого элемента цифровой инфраструктуры состоят в высокой сложности и стоимости развертывания и

[13] Федоров А. Что такое ИТ-инфраструктура и из каких компонентов она состоит [Электронный ресурс]/ А. Федоров // Режим доступа: https://cyberprotect.ru/blog/it-infrastructure-intro?ysclid=mk1kblc7jg194849748 (дата обращения: 05.01.2026).

администрирования, когда требуется отдельная экспертиза и инфраструктура (FC-сети).

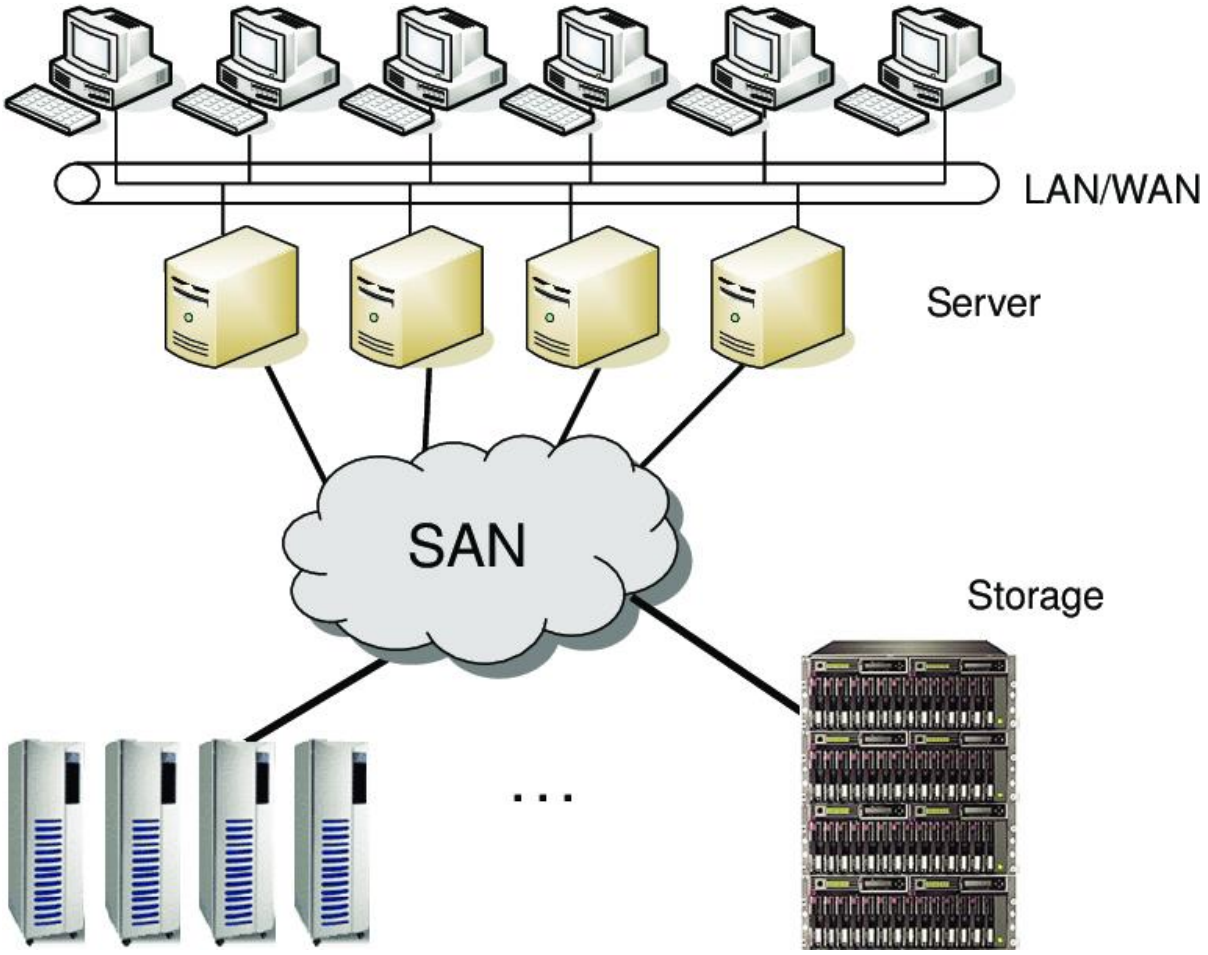

Рис. 3.2. Схема работы виртуального сервера SAN (Storage Area Network)

NAS – это специализированное сетевое устройство (или кластер), которое предоставляет доступ к данным на уровне файлов по стандартным сетевым протоколам. NAS представляет собой законченную файловую систему. EE архитектура представляет собой автономное устройство (или программно-определяемое решение) со своей ОС, процессорами, памятью и дисковыми накопителями, подключаемое к стандартной IP-сети (Ethernet).[14]

[14] Емельянов В.А., ИТ-инфраструктура организации: учебное пособие/ В.А. Емельянов. - Москва : КноРус, 2022. - 144 с. - ISBN 978-5-406-09892-9. - URL: https://book.ru/book/943918 (дата обращения: 18.01.2026).

Модель доступа NAS – это файловый уровень (file-level). Доступ осуществляется по сетевым протоколам, таким как NFS (для Unix/Linux) и SMB/CIFS (для Windows). Сервер обращается не к блокам, а к файлам и каталогам (рис. 3.3).

Преимущества этих специализированных устройств состоит в простоте развертывания и администрирования, использовании существующей сетевой инфраструктуры, эффективного общий доступ к файлам для множества клиентов, встроенных функциях репликации и снапшотов на уровне файлов, как правило, более низкой стоимости владения.

Недостатки NAS заключаются в том, что производительность ограничена сетью Ethernet и накладными расходами файловых протоколов, задержка которых выше, чем у SAN. Это устройства не подходят для приложений, требующих прямого блочного доступа.

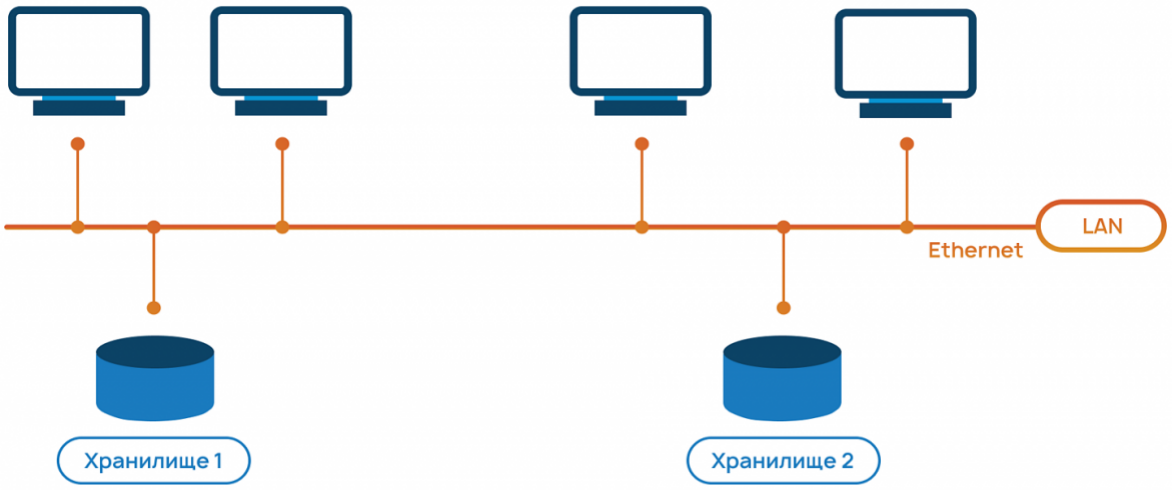

Рис. 3.3. Схема работы NAS (Network Attached Storage)

Сравнение архитектур систем хранения SAN и NAS представлено ниже в табл. 3.2.

Эволюция вычислительных ресурсов и систем хранения демонстрирует четкий вектор от жесткой привязки к аппаратному обеспечению к все более высоким уровням абстракции, программной определимости и гибкости. Физические серверы, виртуальные машины и контейнеры не являются взаимоисключающими технологиями, а образуют многоуровневый стек, где каждый уровень решает свои специфические задачи.

Таблица 3.2

Сравнение архитектур систем хранения SAN и NAS

Параметр	SAN (Storage Area Network)	NAS (Network Attached Storage)
Уровень доступа	Блочный (Block-level)	Файловый (File-level)
Предоставляемый ресурс	«Сырой» диск (LUN)	Файловый ресурс, директория
Сетевой протокол	Fibre Channel (FC), iSCSI, NVMe-oF	NFS, SMB/CIFS, FTP, SFTP
Тип сети	Выделенная сеть хранения (FC) или VLAN (iSCSI)	Общая LAN/WAN сеть (Ethernet)
Управление файловой системой	На стороне сервера (инициатора)	На стороне устройства хранения (NAS-головки)
Основная задача	Высокопроизводительный доступ для серверов и приложений (СУБД, ВМ)	Совместный доступ к файлам для рабочих групп и серверов
Производительность	Очень высокая, низкая задержка	Зависит от нагрузки сети и протокола, задержка выше
Сложность и стоимость	Высокая	От средней до низкой

Аналогично, границы между SAN и NAS размываются с появлением унифицированных систем хранения (Unified Storage), способных предоставлять данные одновременно и по блочным, и по файловым протоколам из единого пула дисков. Современные подходы, такие как гиперконвергентная инфраструктура (HCI), еще дальше интегрируют вычисления и хранение, объединяя их на уровне программно-определяемых модулей.

Таким образом. Именно понимание архитектурных особенностей, преимуществ и ограничений каждого из рассмотренных компонентов является обязательным для проектирования, управления и оптимизации цифровой инфраструктуры, отвечающей текущим и будущим бизнес-требованиям.

3.2. Сетевые ресурсы цифровой инфраструктуры

Сетевые ресурсы представляют собой фундаментальный компонент цифровой инфраструктуры, обеспечивающий взаимодействие вычислительных систем, сервисов и конечных пользователей. Их эволюция от жестко сконфигурированных аппаратных систем к программно-определяемым и виртуализированным платформам определяет современные подходы к построению, управлению и эксплуатации информационных сред. Ключевыми элементами в этой иерархии являются локальные (LAN) и глобальные (WAN) сети, а также технологии программно-конфигурируемых сетей (SDN) и балансировщики нагрузки, которые совместно формируют гибкую, масштабируемую и отказоустойчивую транспортную основу.

Локальные вычислительные сети (LAN) представляют собой инфраструктуру, ограниченную географическими рамками одного здания или комплекса зданий. Их основное назначение — обеспечение высокоскоростного и низколатентного обмена данными между устройствами внутри ограниченного периметра. Физическую основу LAN традиционно составляют кабельные системы (витая пара, оптическое волокно) и коммутаторы (L2/L3), образующие проводную магистраль, дополняемую беспроводным доступом через точки доступа WI-FI, интегрированные в общую архитектуру. Ключевыми характеристиками LAN являются высокая пропускная способность (до сотен Гбит/с и более в магистрали), использование технологий Ethernet и IP на канальном и сетевом уровнях модели OSI/ TCP/IP, а также централизованное управление политиками безопасности (через межсетевые экраны, системы обнаружения вторжений) и качества обслуживания (QoS). В современных условиях локальная сеть трансформируется из изолированного сегмента в часть гибридной облачной инфраструктуры, где критически важными становятся интеграция с облачными сервисами, микросегментация трафика для безопасности и поддержка интернета вещей (IoT).

Глобальные вычислительные сети (WAN) обеспечивают соединение географически распределенных сетевых сегментов, в том числе локальных сетей филиалов и центров обработки данных, с удаленными облачными платформами и интернетом. Если LAN оптимизированы

для производительности в ограниченной области, то WAN проектируются для надежной и безопасной передачи данных на большие расстояния, часто с использованием арендованных каналов связи у телекоммуникационных операторов. Традиционные WAN-технологии, такие как MPLS (Multiprotocol Label Switching), обеспечивали гарантированную пропускную способность и низкие потери данных, но отличались высокой стоимостью и недостаточной гибкостью. Современная эволюция WAN связана с переходом к программно-определяемым глобальным сетям (SD-WAN), которые используют дешевые и доступные каналы интернет (включая DSL, 4G/5G, спутниковую связь) в качестве транспорта, интеллектуально управляя потоками трафика на основе централизованной политики. SD-WAN абстрагирует управление от физических каналов, обеспечивая динамическую маршрутизацию, шифрование трафика, объединение каналов и приоритезацию критически важных приложений (например, VoIP, видеоконференцсвязи) в реальном времени, что приводит к снижению затрат и повышению гибкости инфраструктуры.

Программно-конфигурируемые сети (SDN) представляют собой парадигму архитектурного разделения плоскости управления (control plane) и плоскости данных (data plane) в сетевых устройствах. В традиционных сетях каждый коммутатор или маршрутизатор самостоятельно принимает решения о передаче трафика на основе локальных таблиц. В архитектуре SDN логика управления централизована в отдельном программном компоненте – контроллере SDN, который глобально «видит» всю сеть и программирует потоковые таблицы на физических или виртуальных коммутаторах через открытые протоколы, такие как OpenFlow. Это разделение обеспечивает беспрецедентную гибкость, автоматизацию и программный контроль над сетевыми ресурсами. Администратор может оперативно изменять конфигурацию, маршрутизацию и политики безопасности для всей сети через единый программный интерфейс (API), не настраивая каждое устройство отдельно. SDN является технологическим фундаментом для концепций виртуализации сетевых функций (NFV), когда такие сервисы, как межсетевые экраны, балансировщики нагрузки или шлюзы, реализуются в виде программных модулей, работающих на стандартных серверах, что снижает зависимость от специализированного аппаратного обеспечения и ускоряет развертывание сервисов (рис. 3.4).

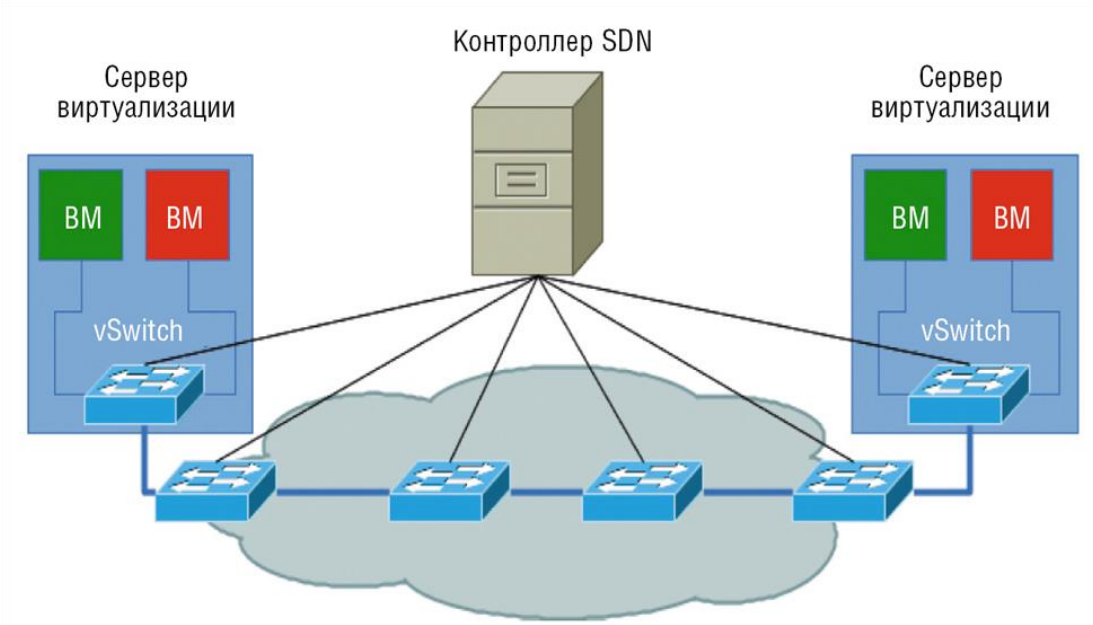

Рис. 3.4. Схема Реализация SDN на базе специальных коммутаторов[15]

Балансировщики нагрузки – это специализированные устройства или программные компоненты, предназначенные для оптимального распределения входящего сетевого трафика между несколькими серверами или сетевыми каналами. Их основная цель – повышение доступности, отказоустойчивости и производительности приложений путем предотвращения перегрузки отдельных узлов. Балансировщики работают на различных уровнях сетевой модели: на транспортном уровне (L4), распределяя соединения на основе IP-адресов и портов, и на прикладном уровне (L7), где могут анализировать содержимое HTTP/HTTPS-запросов и принимать решения на основе URL, cookies или типа контента. Современные балансировщики нагрузки являются ключевым элементом архитектуры центров обработки данных и облачных платформ, обеспечивая масштабируемость веб-сервисов, терминацию SSL-шифрования, поддержку сессий (session persistence) и выполнение функций обратного прокси-сервера.

[15] Смелянский Р. Технологии реализации программно конфигурируемых сетей: Overlay vs OpenFlow [Электронный ресурс]/ Руслан Смелянский // Режим доступа: https://www.osp.ru/lan/2014/04/13040709 (дата обращения: 18.01.2026).

Их эволюция тесно связана с SDN и облачными технологиями, где они существуют в виде виртуальных сетевых функций, динамически масштабируемых и управляемых через API в соответствии с изменяющейся нагрузкой.

Взаимосвязь и конвергенция этих компонентов определяют современную сетевую инфраструктуру. Локальные сети обеспечивают высокопроизводительный доступ на уровне периметра, глобальные сети — их безопасную и интеллектуальную интеграцию в распределенную среду. Парадигма SDN вносит в эту инфраструктуру гибкость и программируемость, а балансировщики нагрузки оптимизируют распределение ресурсов на уровне сервисов. Управление таким комплексом требует комплексного подхода, использующего системы оркестрации и аналитики для обеспечения надежности, безопасности и эффективности всего комплекса сетевых ресурсов цифровой организации. Сравнительная характеристика ключевых сетевых ресурсов цифровой инфраструктуры представлена в табл. 3.3.

Концептуально сетевые ресурсы корпоративной цифровой инфраструктуры можно структурировать по уровням, отражающим их функциональное назначение и место в общей архитектуре. Данная структура не является жесткой и зависит от масштаба компании, отраслевой специфики и принятых архитектурных принципов (например, сходимость или дивергенция сетей передачи данных, голоса и видео). Однако базовые элементы остаются инвариантными.

Физический уровень (Physical Layer) образует материальную основу сети. К нему относятся кабельные системы, пассивное и активное коммутационное оборудование, беспроводные точки доступа, а также специализированные носители, такие как радиочастотные каналы в спутниковой или сотовой связи. Качество и правильность проектирования физической инфраструктуры определяют базовые характеристики всей сети: пропускную способность, задержки, устойчивость к помехам и масштабируемость.[16]

[16] Заводцев И.В. Программные и программно-аппаратные средства защиты информации в объектах информационной инфраструктуры: учебное пособие/ И.В. Заводцев, А.В. Крупенин, С.В. Скрыль. – М: ACADEMIA, 2023. – 288 с. - 978-5-0054-0439-8

Таблица 3.3

Сравнительная характеристика ключевых сетевых ресурсов

Критерий	Локальные сети (LAN)	Глобальные сети (WAN)	Программно-конфигрируемые сети (SDN)	Балансировщики нагрузки
Основное назначение	Высокоскоростной обмен внутри ограниченного периметра.	Соединение распределенных сетей на больших расстояниях.	Централизованное программное управление сетевой инфраструктурой.	Распределение трафика между серверами для отказоустой-чивости и производительности.
Географический охват	Ограничен (здание, кампус).	Глобальный (город, страна, континент).	Не зависит от географии, архитектурный принцип.	Не зависит от географии, функциональный компонент.
Ключевые технологии	Ethernet (IEEE 802.3), Wi-Fi (IEEE 802.11), коммутаторы L2/L3.	MPLS, IPsec, SD-WAN, выделенные линии, интернет-каналы.	Контроллер SDN, протокол OpenFlow, API (Northbound/Southbound).	Алгоритмы round-robin, least connections, L4/L7-балансировка, health checks.
Уровень управления	Децентрализованное (традиционно) или централизованное (через контроллеры).	Часто децентрализованное; централизация в SD-WAN.	Полностью централизованное (плоскость управления).	Централизованное в рамках пула серверов/сервисов.
Гибкость и автоматизация	Умеренная, зависит от аппаратной конфигурации.	Низкая в классических WAN; высокая в SD-WAN.	Крайне высокая, основана на программных политиках и API.	Высокая, особенно в виртуальных и облачных реализациях.
Критически важные атрибуты	Пропускная способность, задержка, безопасность периметра.	Надежность, доступность, безопасность передачи, задержка.	Программируемость, абстракция, согласованность политик.	Доступность, время отклика, эффективность распределения.
Тенденции развития	Конвергенция с Wi-Fi 6/7, микросегментация, IoT, интеграция с облаком.	Доминирование SD-WAN, интеграция с SASE (Secure Access Service Edge).		

Современные подходы, такие как структурированные кабельные системы (СКС), предусматривают стандартизацию и унификацию физических компонентов, что упрощает управление и развитие сети.

Уровень коммутации и маршрутизации (Switching & Routing Layer) является ключевым для организации логической структуры сети. На данном уровне функционируют маршрутизаторы, коммутаторы различного уровня (L2, L3) и шлюзы. Их основная задача – эффективная доставка пакетов данных от источника к получателю на основе заданных правил и алгоритмов. Логическая топология сети (звезда, кольцо, ячеистая структура), сегментация на виртуальные локальные сети (VLAN), обеспечение отказоустойчивости за счет протоколов избыточности (например, STP, EtherChannel) реализуются именно на этом уровне. Важнейшим аспектом является разграничение между внутренней (интрасеть) и внешней сетями, а также организация безопасного межсетевого взаимодействия.

Уровень транспортных и сервисных ресурсов (Transport & Services Layer) включает в себя комплекс средств, обеспечивающих не просто передачу данных, а предоставление сетевых сервисов. Сюда входят:

1. Сервисы идентификации и контроля доступа: подсистемы аутентификации, авторизации и учёта (AAA), часто реализуемые через протоколы RADIUS или TACACS+ в интеграции с корпоративными каталогами (например, Microsoft Active Directory).
2. Сервисы адресации и управления конфигурацией: серверы DHCP (Dynamic Host Configuration Protocol) для автоматического назначения IP-адресов и параметров сети, а также системы управления IP-адресным пространством (IPAM).
3. Сервисы разрешения имён: иерархия DNS-серверов (Domain Name System), обеспечивающих преобразование доменных имён в IP-адреса, что является критически важным для работы практически всех приложений.
4. Сервисы времени: протокол NTP (Network Time Protocol) для синхронизации системного времени на всех сетевых устройствах и серверах, что необходимо для корректной работы протоколов безопасности, аудита и координации распределённых систем.
5. Сервисы мониторинга и управления: системы, использующие протоколы SNMP, NetFlow, sFlow для сбора телеметрии, выявления аномалий и оценки производительности сети.

Уровень периметральной безопасности и подключения (Perimeter Security & Connectivity Layer) отвечает за защиту внутренней сети от

угроз извне и за организацию безопасных каналов связи с удалёнными объектами. Основными компонентами являются межсетевые экраны (брандмауэры) нового поколения (NGFW), системы обнаружения и предотвращения вторжений (IDS/IPS), шлюзы безопасности веб- и почтового трафика (Secure Web Gateway, Email Gateway). Отдельным критически важным ресурсом являются средства построения виртуальных частных сетей (VPN), которые обеспечивают шифрование и целостность данных, передаваемых через публичные сети, формируя защищённые туннели для удалённых сотрудников (клиент-сеть) и для объединения распределённых филиалов (сеть-сеть).

Уровень программно-определяемых сетей (Software-Defined Networking, SDN) и виртуализации сетевых функций (Network Functions Virtualization, NFV) представляет собой современную парадигму управления сетевыми ресурсами. SDN отделяет плоскость управления (control plane) от плоскости данных (data plane), централизуя логику управления в контроллере. Это позволяет программировать сетевое поведение, динамически перенастраивать потоки данных и реализовывать сложные политики на уровне всего инфраструктурного пула, а не отдельных устройств.

NFV дополняет эту концепцию, переводя традиционные сетевые функции (маршрутизацию, балансировку нагрузки, межсетевые экраны) с проприетарного аппаратного обеспечения на стандартные серверы, где они работают в виде виртуальных машин или контейнеров. Это повышает гибкость, ускоряет развёртывание новых сервисов и снижает капитальные затраты. Совместное использование SDN и NFV формирует основу для создания облакоподобной, самообслуживаемой и автоматизированной сетевой среды.[17]

Так, взаимосвязь между сетевыми ресурсами, предоставляемыми сервисами и используемыми для управления ими протоколами и технологиями представлена ниже в табл. 3.4.

[17] Сетевые технологии в России – 2025: разбор трендов [Электронный ресурс]/ А. Федоров // Режим доступа: https://dzen.ru/a/aDbUmk3nFC8zwWey?ysclid=mkgxclbq18962166339 (дата обращения: 16.01.2026).

Таблица 3.4

Классификация сетевых ресурсов и предоставляемых сервисов

Уровень/ Категория ресурсов	Ключевые аппаратные/ программные компоненты	Предоставляемые сервисы и функции	Ключевые протоколы и стандарты
Физический	Кабели (витая пара, оптоволокно), патч-панели, медиаконвертеры, точки доступа Wi-Fi	Обеспечение физической среды для передачи сигналов; беспроводной доступ	IEEE 802.3 (Ethernet), 802.11 (Wi-Fi), стандарты на СКС (TIA-568)
Коммутации и маршрутизации	Маршрутизаторы, коммутаторы L2/L3, контроллеры беспроводной сети	Сегментация (VLAN), маршрутизация между сетями, отказоустойчивость, базовое QoS	IP, OSPF, BGP, STP/RSTP, 802.1Q
Транспортный и сервисный	DHCP-серверы, DNS-серверы, NTP-серверы, системы IPAM, RADIUS-серверы	Динамическая адресация, разрешение имён, синхронизация времени, централизованная аутентификация	DHCP, DNS, NTP, RADIUS/TACACS+
Периметра безопасности	Межсетевые экраны (NGFW), шлюзы VPN, системы IDS/IPS, прокси-серверы	Защита периметра, фильтрация трафика, предотвращение вторжений, шифрование удалённого доступа	IPsec, SSL/TLS, OpenVPN, протоколы анализа трафика
Программно-определяемый	Контроллеры SDN, гипервизоры, виртуальные машины, контейнеры	Централизованное управление, автоматизация конфигурации, виртуализация сетевых функций, оркестрация	OpenFlow, NETCONF/YANG, API (REST, gRPC)

Эффективное управление сетевыми ресурсами требует реализации цикла PDCA (Plan-Do-Check-Act) и опирается на несколько фундаментальных процессов.

Проектирование и архитектура предполагают разработку масштабируемой, отказоустойчивой и безопасной сетевой топологии, соответствующей текущим и прогнозируемым бизнес-потребностям.

Оперативное управление конфигурацией (Network Configuration Management, NCM) направлено на контроль за изменениями параметров сетевых устройств, обеспечение их соответствия политикам и быстрое восстановление в случае сбоев. [18]

Мониторинг производительности и доступности в реальном времени позволяет выявлять узкие места, прогнозировать проблемы и планировать модернизацию. Управление инцидентами и проблемами нацелено на минимизацию времени простоя и устранение коренных причин сбоев. Всё большее значение приобретает сетевая аналитика (Network Analytics), которая на основе больших данных и методов машинного обучения позволяет перейти от реактивного к проактивному и предиктивному управлению, прогнозируя аномалии и оптимизируя распределение ресурсов. Некоторые направления сетевой аналитики:

- Непрерывный мониторинг в реальном времени — постоянный сбор данных о состоянии сети, позволяющий мгновенно идентифицировать аномалии.

- Предиктивная аналитика — использование исторических данных и машинного обучения для предсказания потенциальных проблем до их возникновения.

- Автоматизация реагирования — настройка автоматических действий при обнаружении определённых паттернов трафика или сбоев.

- Сквозная видимость (end-to-end visibility) — комплексное отслеживание всего пути данных от источника до получателя.

Таким образом, сетевые ресурсы цифровой инфраструктуры представляют собой сложную, многоуровневую и динамичную систему, интегрирующую физические и логические компоненты в единую транспортно-сервисную платформу. Их эволюция в сторону программной определимости, виртуализации и глубокой автоматизации трансформирует сеть из статической коммуникационной артерии в интеллектуальный, гибкий и программируемый слой, способный адаптивно поддерживать цифровую трансформацию бизнеса. Управление этими ресурсами, следовательно, требует от специалистов не только глубоких знаний классических сетевых технологий, но и компетенций

[18] Грибанов Ю. И., Руденко М. Н., Аленина К. А. Современные подходы к формированию цифровой инфраструктуры // Управленческое консультирование. 2020. № 8. С. 88–98. DOI 10.22394/1726-1139-2020-8-88-98

в области программирования, анализа данных и интеграции с облачными средами.

3.3. Платформенное и системное программное обеспечение
3.4. Управление конфигурациями и автоматизация цифровой инфраструктуры компании

Современная цифровая инфраструктура представляет собой динамичную и высоко распределенную систему, состоящую из множества взаимосвязанных компонентов: физических и виртуальных серверов, сетевого оборудования, контейнеров, микросервисов, приложений, сред исполнения и зависимостей между ними. Эффективное управление таким комплексом требует перехода от администрирования отдельных устройств к системному контролю над всей совокупностью единиц инфраструктуры, их состояниями и взаимосвязями. Ключевую роль в этом процессе играет система управления конфигурациями (Configuration Management Database, CMDB).

CMDB представляет собой централизованное хранилище, аккумулирующее информацию обо всех значимых объектах инфраструктуры (Configuration Items, CI – элементы конфигурации) и отношениях между ними. Каждый CI описывается набором атрибутов (версия ОС, IP-адрес, объем памяти, установленное программное обеспечение, владелец) и его жизненным циклом. Основная функция CMDB – обеспечение «единого источника истины» (Single Source of Truth), что является фундаментом для процессов управления услугами (ITSM), оценки воздействия изменений, управления проблемами и соблюдения требований безопасности и регуляторных норм. Точность и актуальность данных в CMDB являются критическим условием для принятия обоснованных управленческих решений.

Рост масштабов и сложности инфраструктуры сделал ручное администрирование неэффективным и подверженным ошибкам. Это привело к развитию парадигмы Infrastructure as Code (IaC – инфраструктура как код), которая рассматривает конфигурацию инфраструктуры как программный код, подлежащий версионному контролю, тестированию и воспроизведению. IaC реализуется через два взаимодополняющих подхода:

- Императивная автоматизация (конфигурационное управление) фокусируется на определении последовательности команд для приведения системы в желаемое состояние. Инструменты этого класса (Ansible, Puppet, Chef) отвечают на вопрос «как» достичь конфигурации. Они идеально подходят для управления конфигурацией операционных систем, установки пакетов, настройки служб и обеспечения их постоянного соответствия политике (конфигурационный дрейф).

- Декларативная автоматизация (оркестрация предоставления) фокусируется на описании «чего» – конечного состояния инфраструктуры, предоставляя платформе самой определить необходимую последовательность действий для его достижения. Инструменты этого класса (Terraform, Kubernetes API) используются для создания и управления облачными ресурсами (виртуальные машины, сети, хранилища) и контейнеризованными средами.

Интеграция данных из инструментов автоматизации в CMDB позволяет поддерживать ее актуальность, создавая динамическую и достоверную модель инфраструктуры.

Рассмотрим ключевые инструменты в контексте их функционального назначения и взаимодействия с CMDB.

Ansible – инструмент императивного конфигурационного управления, использующий модель без агентов и опирающийся на язык YAML для описания плейбуков (playbooks). Ansible обеспечивает идемпотентность – многократное выполнение плейбука приводит систему к одному и тому же конечному состоянию. Его роль заключается в детальной настройке CI, управлении их конфигурацией и деплое приложений. Сведения о фактически примененной конфигурации могут быть экспортированы в CMDB, обеспечивая связь между желаемым состоянием (код) и реальным.

Terraform – ведущий инструмент декларативной оркестрации, использующий собственный декларативный язык HCL (HashiCorp Configuration Language). Terraform оперирует абстракцией провайдеров (AWS, Azure, Google Cloud, VMware и др.) и позволяет определять, планировать и создавать облачные ресурсы. Его ключевая функция – управление внешней, часто облачной, инфраструктурой. План выполнения Terraform служит источником информации для CMDB о плани-

руемых к созданию, изменению или уничтожению ресурсах, а состояние (state file) – актуальном их наборе, что критично для управления зависимостями.

Kubernetes (K8s) – система оркестрации контейнеризованных приложений, представляющая собой платформу декларативного управления для микросервисных архитектур. Пользователь декларативно описывает желаемое состояние рабочих нагрузок (deployments), сервисов, правил сетевого доступа и хранения данных в YAML- или JSON-манифестах. Kubernetes постоянно отслеживает фактическое состояние кластера и автоматически приводит его в соответствие с декларируемым. Для CMDB кластер Kubernetes становится сложным CI, включающим в себя такие подчиненные элементы, как ноды (Nodes), пространства имен (Namespaces), поды (Pods), сервисы (Services), что требует специализированных интеграций для корректного отражения его внутренней структуры и состояния.

Сравнительная характеристика инструментов автоматизации по ключевым критериям представлена в табл. 3.6.

Эффективное управление цифровой инфраструктурой достигается не использованием одного инструмента, а их комбинацией в рамках сквозного жизненного цикла:

- Проектирование и предоставление (Provisioning). Terraform используется для запроса «сырых» вычислительных, сетевых и хранящих ресурсов у облачного провайдера или платформы виртуализации. На этом этапе в CMDB регистрируются новые CI с базовыми атрибутами.

- Конфигурация и деплой (Configuration & Deployment). Ansible применяется для начальной настройки созданных Terraform ресурсов: настройки ОС, установки middleware, конфигурации безопасности. Далее Ansible или специализированные инструменты CI/CD развертывают прикладное программное обеспечение. Данные о примененной конфигурации и версиях ПО обновляются в CMDB.

Таблица 3.6

Сравнительная характеристика инструментов автоматизации

Критерий	Ansible	Terraform	Kubernetes
Основная парадигма	Императивная (процедурная)	Декларативная	Декларативная
Предмет управления	Конфигурация ПО и ОС на существующих ресурсах	Жизненный цикл облачных и виртуальных ресурсов (создание, изменение, удаление)	Жизненный цикл контейнеров и сопутствующих ресурсов в кластере
Ключевая задача	«Как» настроить систему	«Что» создать в инфраструктуре	«Что» должно работать в кластере
Состояние системы	Управляется через идемпотентность плейбуков	Хранится во внешнем файле состояния (state file)	Хранится в etcd (распределенное хранилище кластера)
Интеграция с CMDB	Фактическая конфигурация после применения плейбуков	Планируемые и фактически развернутые ресурсы (из state file)	Топология кластера, состояние подов, сервисов, развертываний

- Оркестрация и эксплуатация (Orchestration & Operation). Для контейнеризованных приложений Kubernetes берет на управление их жизненный цикл, обеспечивая масштабирование, отказоустойчивость и обновления. Интеграция CMDB с Kubernetes позволяет отслеживать состояние подов, распределение по нодам и актуальные версии образов контейнеров.

- Мониторинг и соответствие (Monitoring & Compliance). Информация из CMDB о взаимосвязях CI используется для настройки систем мониторинга и анализа логов. Инструменты конфигурационного управления могут выполнять периодические проверки на соответствие

политикам безопасности, исправляя отклонения и обновляя статусы в CMDB.

Таким образом, инструменты формируют технологический стек, где Terraform создает фундамент, Ansible его настраивает и поддерживает, а Kubernetes управляет современными контейнеризованными приложениями поверх этого фундамента. CMDB выступает интегрирующим информационным ядром, связывающим данные о планируемом (код), фактическом (состояние инструментов) и нормативном (политики) состоянии инфраструктуры.

Внедрение автоматизированного управления на основе CMDB и инструментов IaC представляет собой не только технологическую, но и организационную трансформацию. Ключевыми аспектами являются:

- Версионность и контроль изменений. Все артефакты (плейбуки Ansible, модули Terraform, манифесты Kubernetes) должны храниться в системах контроля версий (Git), что обеспечивает трассируемость, возможность отката и совместную разработку.

- Идемпотентность и детерминизм. Повторяемость результатов – краеугольный камень автоматизации. Конфигурация, примененная к идентичным системам, должна приводить к идентичному состоянию, что обеспечивается принципами, заложенными в инструментах.

- Управление секретами (Secrets Management). Автоматизация требует безопасного обращения с учетными данными, ключами API, сертификатами. Интеграция со специализированными системами (HashiCorp Vault, Azure Key Vault) становится обязательным компонентом.

- Сложность интеграции и актуальность данных. Поддержание синхронизации между CMDB, динамической облачной инфраструктурой и несколькими инструментами автоматизации требует продуманной архитектуры интеграций (через API, веб-хуки, специализированные коннекторы) для предотвращения устаревания данных в CMDB.

Таким образом, управление современной цифровой инфраструктурой эволюционировало от ручного администрирования к комплексной, кодово-ориентированной модели. CMDB служит центральным информационным реестром и моделью взаимосвязей, в то время как комбинированное использование инструментов автоматизации обеспечивает эффективное и надежное управление жизненным циклом

всех компонентов. Их синергия позволяет реализовать принципы гибкости, масштабируемости, отказоустойчивости и безопасности, являющиеся критически важными для цифровой трансформации предприятий. Успех зависит от системного подхода, рассматривающего технологические инструменты в неразрывной связи с процессами управления и качеством данных в CMDB.

Подводя итог сказанному, цифровая инфраструктура представляет собой сложную, многоуровневую и динамичную систему, эволюция которой демонстрирует четкий вектор от жесткой привязки к аппаратному обеспечению в сторону возрастающих уровней абстракции, программной определимости и виртуализации. Вычислительные ресурсы эволюционировали от изолированных физических серверов через платформы виртуализации к контейнеризации и микросервисным архитектурам. Каждый из этих уровней — физические серверы, виртуальные машины, контейнеры — не является взаимоисключающим, а формирует комплексный стек, оптимальный для различных классов задач: от высокопроизводительных вычислений до гибкого развертывания облачно-нативных приложений. Аналогичная трансформация наблюдается в подсистемах хранения данных, где наблюдается конвергенция блочных (SAN) и файловых (NAS) моделей в рамках унифицированных решений, а также интеграция с вычислительными ресурсами в гиперконвергентных системах.

Сетевые ресурсы трансформируются из набора статически сконфигурированных аппаратных устройств в программно-определяемую, интеллектуальную транспортно-сервисную платформу. Ключевыми трендами выступают повсеместное внедрение принципов SDN и NFV, обеспечивающих централизованное программируемое управление и виртуализацию сетевых функций, а также доминирование архитектур SD-WAN для построения гибких и экономичных глобальных сетей. Сеть перестает быть пассивной коммуникационной средой, становясь активным, программируемым слоем, способным динамически адаптироваться к требованиям приложений и политикам безопасности.

Фундаментальную роль в обеспечении функционирования прикладного уровня играет платформенное и системное программное обеспечение, включающее операционные системы, промежуточное ПО (middleware) и системы управления базами данных. Данный слой

обеспечивает абстракцию аппаратных ресурсов, предоставляет стандартизированные сервисы исполнения, взаимодействия и хранения данных. Современная практика управления этим стеком смещается от администрирования отдельных экземпляров к использованию методологий «инфраструктуры как кода» (IaC), контейнеризации и оркестрации, что обеспечивает воспроизводимость, согласованность и автоматизацию жизненного цикла.

Именно принципы IaC и автоматизации становятся центральными для современного управления всей цифровой инфраструктурой в целом. Комбинированное использование инструментов императивного и декларативного управления позволяет реализовать сквозной автоматизированный жизненный цикл — от предоставления ресурсов и их конфигурации до оркестрации сложных распределенных приложений. Критически важным элементом управленческой модели выступает система управления конфигурациями, выполняющая роль единого источника истины о всех элементах инфраструктуры и их взаимосвязях. Актуальность данных в CMDB, поддерживаемая интеграцией с инструментами автоматизации, формирует основу управления изменениями, безопасностью и соответствием требованиям.

Таким образом, эффективное управление компонентами цифровой инфраструктуры требует перехода от изолированного, реактивного администрирования к целостному, проактивному и основанному на коде подходу. Это предполагает глубокое понимание архитектурных особенностей и эволюционных трендов каждого компонентного слоя, а также владение технологиями и практиками их интеграции в единую, гибкую, масштабируемую и безопасную платформу, способную поддерживать цифровую трансформацию бизнеса. Успех определяется не отдельными технологиями, а системной зрелостью процессов управления, обеспечивающей синергию между вычислительными, сетевыми, программными ресурсами и автоматизированными системами контроля.

Вопросы для обсуждения

1. Дайте определение физическому и виртуальному серверу. В чем состоят принципы их работы?

2. Охарактеризуйте контейнеризацию как неотъемлемого элемента цифровой инфраструктуры. Поясните достоинства, недостатки, архитектуру и область применения контейнеров.

3. Поясните принципы работы и сферу применения SAN (Storage Area Network).

4. Дайте определение NAS (Network Attached Storage). Поясните модель доступа NAS, ее основные достоинства и недостатки.

5. Сравните архитектуры систем хранения SAN и NAS по следующим параметрам: уровень доступа, тип сети, производительность и предоставляемым ресурсам.

6. Перечислите основные сетевые ресурсы цифровой инфраструктуры компании.

7. Объясните принципы и условия работы локальных вычислительных сетей (LAN).

8. Дайте характеристику глобальным вычислительным сетям (WAN) как одному из ключевых элементов современных цифровых инфраструктур.

9. Поясните направления использования программно-конфигурируемых сетей (SDN).

10. Дайте определение балансировщикам нагрузки. В чем состоит их функции при построении цифровой инфраструктуры компании.

11. Поясните классификацию сетевых ресурсов и предоставляемых сервисов по разбиению их на уровни

12. Перечислите основные категории платформенного и системного программного обеспечения цифровой инфраструктуры.

13. Дайте определение СУБД. Какие элемементы входят в состав современных СУБД?

14. Приведите основные характеристики основных моделей современных СУБД.

15. Поясните, какие задачи включает управление парком СУБД.

16. Дайте определение системе управления конфигурациями (Configuration Management Database, CMDB).

17. Объясните сущность парадигмы Infrastructure as Code (IaC – инфраструктура как код). Какие два подхода она реализует?

18. Поясните специфику работы инструмента императивного конфигурационного управления (Ansible).

19. Охарактеризуйте инструмент декларативной оркестрации (Terraform). Какой декларативный язык он использует?

20. Поясните специфику работы системы оркестрации контейнеризованных приложений (Kubernetes). Сравните направления ее применения с Ansible и Terraform.

Практические задания

Задание 1. Составьте и заполните сравнительную таблицу, систематизирующую ключевые архитектурные, эксплуатационные и экономические характеристики физических серверов (Bare-Metal), виртуальных машин (ВМ) и контейнеров, в которой должны быть отражены не менее семи рассматриваемых критериев: уровень абстракции, единицу развертывания, время развертывания, накладные расходы, основную область применения и др.

На основе составленной таблицы сформулируйте и обоснуйте рекомендации по выбору платформы виртуализации для трех типовых сценариев:

- Высоконагруженная транзакционная СУБД.
- Микросервисное веб-приложение с динамическим масштабированием.
- Унаследованное приложение, требующее специфичной версии операционной системы.

Задание 2. Спроектируйте логическую архитектуру сегмента локальной сети (LAN) для офиса разработки численностью 50 человек, для чего следует определить состав и назначение ключевых аппаратно-программных компонентов для каждого из пяти уровней.

Для каждого компонента необходимо указать предоставляемый сервис и обосновать выбор ключевого протокола/стандарта.

Изобразите полученную архитектуру в виде схемы, отображающей взаимосвязи между основными компонентами.

Задание 3. На основе принципов Infrastructure as Code (IaC), составьте декларативное описание для развертывания типового трехзвенного веб-приложения (frontend, backend, database).

Описание должно включать три части:

- Часть A (Terraform): Декларативный манифест для создания базовых облачных ресурсов: одной виртуальной частной сети, трех виртуальных машин и группы правил межсетевого экрана.

- Часть B (Ansible): Фрагмент плейбука для императивной настройки на одной из ВМ (роль «backend»): установка необходимых пакетов, копирование конфигурационного файла приложения.

- Часть C (Kubernetes): Фрагмент манифеста для декларативного описания развертывания и сервиса для контейнеризованного frontend-компонента, указав запросы ресурсов и лимиты для CPU и памяти.

Для каждой части кратко поясните, как ее выполнение влияет на состояние системы управления конфигурациями (CMDB).

Тест для самоконтроля

1. Выделенная высокопроизводительная сеть, которая предоставляет блочный уровень доступа к данным – это…

а) Виртуальный сервер.

б) Физический сервер.

в) SAN (Storage Area Network).

г) NAS (Network Attached Storage).

2. Специализированное сетевое устройство (или кластер), которое предоставляет доступ к данным на уровне файлов по стандартным сетевым протоколам – это…

а) NAS (Network Attached Storage).

б) LAN (локальная вычислительная сеть)

в) Контейнер.

г) Bare-Metal (физический сервер)

3. Модель доступа NAS – это ….

а) Сетевой уровень.

б) Файловый уровень.

в) Физический уровень.

г) Сервисный уровень.

4. Автономная аппаратная единица, состоящая из процессора, оперативной памяти, подсистем ввода-вывода и накопителей, на которой напрямую установлена операционная система и прикладное программное обеспечение – это…

а) Физический сервер.

б) Виртуальный сервер.

в) Локальная вычислительная сеть.

г) Программно-конфигурируемая сеть.

5. Представляет собой особую ступень абстракции — виртуализация на уровне операционной системы.

а) Контейнеризация.

б) Виртуальная машина.

в) Глобальная вычислительная сеть.

г) IaC (инфраструктура как код).

6. Для этого вида сетей характерно обеспечение высокоскоростного и низколатентного обмена данными между устройствами внутри ограниченного периметра.

а) Локальная сеть.

б) Глобальная сеть.

в) Региональная сеть.

г) Программно-конфигурируемая сеть.

7. Эти сети представляют собой парадигму архитектурного разделения плоскости управления и плоскости данных в сетевых устройствах.

а) Локально-вычислительные сети.

б) Глобальные сети.

в) Программно-конфигурируемые сети.

г) Локальные сети.

8. К какому уровню цифровой инфраструктуры относятся сервисы идентификации и контроля?

а) Физический уровень.

б) Логический уровень.

в) Уровень транспортных и сервисных ресурсов.

г) Уровень программно-определяемых сетей.

9. Маршрутизаторы, коммутаторы L2/L3, контроллеры беспроводной сети относятся к …

а) уровню периметра безопасности;

б) к транспортному и сервисному уровню;

в) физическому уровню;

г) логическому уровню.

10. Комплексное отслеживание всего пути данных от источника до получателя – это…

а) Непрерывный мониторинг в реальном времени.

б) Сквозная видимость.

в) Предиктивная аналитика.

г) Автоматизация реагирования.

11. Занимает концептуальное положение между операционной системой и прикладным бизнес-логикой.

а) Специализированное программное обеспечение.

б) Промежуточное программное обеспечение.

в) Общее программное обеспечение.

г) Мобильное приложение.

12. Брокеры объектных запросов (Object Request Brokers, ORB) относятся к …

а) операционным системам;

б) специализированному программному обеспечению;

в) промежуточному программному обеспечению;

г) общему программному обеспечению.

13. Представляют собой развитую форму middleware, выступающую в качестве централизованного узла для интеграции множества приложений.

а) Интеграционные шины предприятия (ESB).

б) Мобильные приложения

в) Коммутаторы.

г) Брокеры объектных запросов.

14. Управление файловой системой и устройствами ввода-вывода относится к функциям…

а) middleware;

б) операционных систем;

в) коммутатора;

г) общему программному обеспечению.

15. Что не относится к типам (моделям) СУБД?

а) Реляционные (SQL).

б) Нереляционные (NoSQL).

в) Новые SQL (NewSQL).

г) Каталогизированные СУБД.

16. Представляет собой централизованное хранилище, аккумулирующее информацию обо всех значимых объектах инфраструктуры и отношениях между ними.

а) CMDB.

б) версия ОС.

в) IP-адрес.

г) Configuration Items (CI).

17. Парадигма, которая рассматривает конфигурацию инфраструктуры как программный код, подлежащий версионному контролю, тестированию и воспроизведению.

а) SaaS

б) IaC

в) PaaS

г) FaaS

18. Инструмент императивного конфигурационного управления, использующий модель без агентов и опирающийся на язык YAML для описания плейбуков (playbooks).

а) Ansible.

б) Kubernetes (K8s).

в) Terraform.

г) CMDB.

19. Система оркестрации контейнеризованных приложений, представляющая собой платформу декларативного управления для микросервисных архитектур – это…

а) Kubernetes (K8s).

б) Terraform.

в) IaC

г) Ansible.

20. Декларативная парадигма характерна для…

а) Kubernetes (K8s).

б) Ansible.

в) IaC

г) Terraform.

Глава 4. МОДЕЛИ РАЗВЕРТЫВАНИЯ ИНФОРМАЦИОННОЙ ИНФРАСТРУКТУРЫ КОМПАНИИ

4.1. Модели локального развёртывания программного обеспечения

Локальное (on-premise, «внутри площадки») развёртывание программного обеспечения представляет собой классическую модель организации информационной инфраструктуры, при которой все аппаратные ресурсы, системное и прикладное программное обеспечение физически размещаются и эксплуатируются в пределах контролируемой заказчиком территории, такой как собственные или арендованные дата-центры, серверные помещения. Эта модель исторически предшествовала облачным подходам и продолжает оставаться критически важной для широкого спектра задач, где приоритетами являются полный контроль над данными и системами, гарантированная производительность, строгое соответствие регуляторным требованиям или специфическая интеграция с физическим оборудованием. Локальное развёртывание предполагает, что организация самостоятельно осуществляет капитальные затраты (CAPEX) на приобретение инфраструктуры, несёт все операционные расходы (OPEX) на её содержание, обслуживание, модернизацию и обеспечение информационной безопасности, а также формирует штат квалифицированных специалистов для администрирования.

Следует отметить, что до 2010 года локальное развертывание было стандартной практикой, и термин «On-Premise» не использовался. С развитием облачных технологий он стал обозначением традиционного подхода, противопоставленного современным моделям. Короткая форма «On-Prem» также широко используется для удобства.

Фундаментом локальной модели является принцип полного суверенитета и ответственности. Организация владеет или долгосрочно арендует вычислительные серверы, системы хранения данных (СХД), сетевую аппаратуру (коммутаторы, маршрутизаторы, межсетевые экраны) и обеспечивает их бесперебойную работу. Это требует создания специализированной инженерной инфраструктуры: систем электропитания с гарантированным качеством электроэнергии и источни-

ками бесперебойного питания (ИБП), климат-контроля (прецизионного кондиционирования), физической безопасности и мониторинга. Программный стек, начиная от операционных систем и виртуализаторов (например, VMware vSphere, Microsoft Hyper-V, KVM) и заканчивая базами данных, веб-серверами и бизнес-приложениями, устанавливается, настраивается и обновляется силами внутренних ИТ-отделов или привлечённых интеграторов. Такой подход обеспечивает детерминированную производительность, поскольку ресурсы не делятся с другими потребителями, и позволяет проводить тонкую настройку оборудования и ПО под эксклюзивные нужды конкретных задач.

Ключевым драйвером выбора модели on-premise выступают требования к безопасности и соответствию нормативным актам. Для организаций, работающих с персональными данными (подпадающими под действие 152-ФЗ в РФ, GDPR в ЕС), государственной тайной, финансовой информацией (требования ЦБ РФ, PCI DSS), критически важно физическое размещение данных на территории, подконтрольной юрисдикции государства. Локальное развёртывание позволяет реализовать изолированный периметр безопасности, недоступный из публичных сетей, и применять собственные, часто более жёсткие, политики контроля доступа, аудита и шифрования. Кроме того, данная модель незаменима для работы со сложными промышленными системами (АСУ ТП, SCADA[21]), требующими сверхнизких и предсказуемых задержек (детерминизм реального времени) и прямой интеграции с технологическим оборудованием через специализированные интерфейсы, что в публичных облаках часто технически невозможно или экономически нецелесообразно.

С экономической точки зрения модель характеризуется высокой первоначальной капиталоёмкостью и длительным циклом развёртывания. Затраты носят капитальный характер и часто амортизируются в течение нескольких лет. Это создаёт финансовую нагрузку и требует долгосрочного планирования. При этом организация сталкивается с рисками неэффективного использования ресурсов (недогрузки или,

[21] SCADA (англ. Supervisory Control And Data Acquisition) — программно-аппаратный комплекс для диспетчерского управления и сбора данных в реальном времени. SCADA-система собирает данные от датчиков и контроллеров (температуры, давления, потока и т. д.), отображает их на экранах операторов и позволяет дистанционно отдавать команды (например, включить насос, закрыть клапан и др).

наоборот, их нехватки в пиковые периоды), поскольку масштабирование связано с процедурами закупки, поставки и ввода в эксплуатацию нового оборудования, занимающими недели и месяцы.

Ответственность за резервирование и аварийное восстановление (Disaster Recovery) также полностью ложится на организацию, вынуждая создавать и содержать дополнительную инфраструктуру в резервном дата-центре, что удваивает затраты.

Процесс внедрения моделей on-premise сопровождается следующими основными этапами:[22]

1. Покупка лицензионного программного обеспечения.
2. Установка на физические или виртуальные серверы компании.
3. Интеграция с внутренними системами (CRM, ERP, БД, настройка логирования и аудиоархива).
4. Настройка доступа, мониторинга, шифрования и резервного копирования.
5. Аудит безопасности и тестирование на уязвимости.
6. Обучение сотрудников работе с решением.
7. Поддержка, масштабирование и обновление — на стороне клиента или выбранного подрядчика.

Эволюция локальных моделей под влиянием облачных принципов привела к появлению гибридных архитектур и концепций приватного облака. Приватное облако представляет собой логическое развитие on-premise-подхода, внедряя внутри собственной инфраструктуры облачные парадигмы: самообслуживание пользователей через портал, автоматизированное оркестрирование ресурсов, эластичность и учёт потребления. Это позволяет ИТ-подразделениям трансформироваться во внутренних провайдеров услуг, повышая гибкость и эффективность использования существующих инвестиций в оборудование.

Сравнительный анализ ключевых характеристик различных моделей локального развёртывания представлен в табл. 4.1.

[22] On-Premise: что это, чем отличается от облака и кому подходит [Электронный ресурс]// Режим доступа: https://www.fromtech.ru/blog/chto-znachit-on-premise/ (дата обращения: 19.01.2026).

Таблица 4.1

Сравнительные характеристики моделей локального развёртывания

Критерий	Классическая локальная модель (On-Premise)	Модель управляемого хостинга (Managed Hosting)	Модель приватного облака (On-Premise Private Cloud)
Размещение инфраструктуры	Собственные помещения заказчика.	Площадка провайдера (выделенные стойки/серверы).	Собственные помещения заказчика или выделенный центр данных провайдера
Владение активами	Заказчик владеет всем оборудованием.	Оборудование может принадлежать как провайдеру, так и заказчику.	Заказчик, как правило, владеет оборудованием.
Управление инфраструктурой	Полностью силами заказчика	Физическое обслуживание, мониторинг и обеспечение доступности - зона ответственности провайдера.	Полностью или преимущественно силами заказчика с использованием платформ оркестрации.
Управление ОС и ПО	Полностью силами заказчика.	Может варьироваться: от только ОС до полного управления приложениями (по SLA)	Силами заказчика с автоматизацией через портал самообслуживания и шаблоны.
Экономическая модель	Капитальные затраты (CAPEX).	Операционные расходы или комбинированная модель.	Преимущественно CAPEX на оборудование и ПО оркестрации
Гибкость и масштабируемость	Низкая, требует длительного цикла закупок.	Средняя, масштабирование осуществляется запросом провайдеру на выделение новых ресурсов.	Высокая внутри пределов физически развёрнутой инфраструктуры за счёт автоматизации.
Основное преимущество	Максимальный контроль и безопасность.	Снятие с заказчика задач физического обслуживания при сохранении выделенности.	Сочетание контроля on-premise с гибкостью облачной модели.

Критерий	Классическая локальная модель (On-Premise)	Модель управляемого хостинга (Managed Hosting)	Модель приватного облака (On-Premise Private Cloud)
Типичный use-case	Государственные органы, оборонные предприятия, финансовые организации со строгим регулированием.	Корпоративные веб-проекты, ERP-системы, требующие высокой доступности без роста штата.	Крупные предприятия, стремящиеся к оптимизации и трансформации внутреннего ИТ в сервисную модель.

Следует сделать важное замечание, что on-premise не является универсальным решением, так как выбор модели развёртывания зависит от специфики бизнеса, размера компании и ее отраслевой принадлежности. Например, локальные решения подходят для компаний с особыми требованиями к безопасности (банки, медицинские организации), крупного бизнеса, который не хочет зависеть от провайдера, или компаний с жёсткими требованиями к скорости и доступности данных.

Таким образом, модели локального развёртывания не являются устаревшим артефактом, но представляют собой стратегический выбор, обусловленный техническими, нормативными и экономическими соображениями. Их развитие идёт по пути адаптации лучших практик из облачных парадигм, что приводит к созданию гибридных сред, где критически важные, чувствительные системы с предсказуемой нагрузкой остаются on-premise, а менее критичные или пиковые нагрузки выносятся в публичное облако. Понимание архитектурных, управленческих и финансовых аспектов локального развёртывания остаётся необходимым фундаментом для проектирования современной, сбалансированной и отвечающей бизнес-требованиям информационной инфраструктуры.

4.2. Модели развёртывания облачных инфраструктур компании

В рамках эволюции подходов к построению информационной инфраструктуры организации, облачные модели развёртывания представляют собой закономерный этап перехода от капиталоёмкой собственности физическими активами к операционной модели потребления ИТ-ресурсов как сервиса. В отличие от традиционной on-premise инфраструктуры, где компания владеет и управляет всем аппаратно-

программным комплексом, облачные модели предполагают выделение ресурсов (вычислительных мощностей, хранилищ, сетей, платформ и приложений) провайдером через сеть, чаще всего интернет, по запросу и с возможностью самообслуживания. Ключевыми атрибутами облачных вычислений, согласно определению NIST (Национального института стандартов и технологий США), являются самообслуживание по требованию, широкий доступ по сети, объединение ресурсов в пулы, эластичность и измеряемость сервиса. На их основе сформировались несколько фундаментальных моделей развёртывания, выбор среди которых является стратегическим решением, определяющим баланс контроля, безопасности, гибкости и экономических затрат компании.

Основные модели развёртывания облачных инфраструктур можно систематизировать по критерию владения, управления и локализации инфраструктуры. К ним относятся публичное облако, частное облако, гибридное облако и мультиоблако (являющееся скорее архитектурным подходом на базе предыдущих моделей). Каждая из этих моделей обладает уникальными характеристиками, делающими её предпочтительной для определённых классов задач и бизнес-контекстов.

Публичное облако (Public Cloud) представляет собой модель, в которой инфраструктура предоставляется облачным провайдером (таким как Amazon Web Services, Microsoft Azure, Google Cloud Platform) для общего использования множеством независимых клиентов («мультитенантная» архитектура).

Все базовые компоненты - центры обработки данных, серверы, системы хранения и сетевые устройства - принадлежат и эксплуатируются провайдером. Компания-клиент арендует виртуализированные ресурсы, оплачивая фактически потреблённый объём по подписочной модели (pay-as-you-go). Преимущества данной модели заключаются в беспрецедентной экономической эффективности для переменных или растущих нагрузок, высокой эластичности, позволяющей мгновенно масштабировать ресурсы вверх или вниз, а также в отсутствии капитальных затрат (CapEx) и операционных расходов на поддержку физической инфраструктуры. Однако публичное облако накладывает определённые ограничения в части кастомизации базового оборудования и программного обеспечения, может вызывать озабоченность по поводу

безопасности данных, размещаемых за пределами периметра компании, а также потенциально вести к зависимости от конкретного провайдера (vendor lock-in) и росту затрат при неоптимальном управлении ресурсами.

Частное облако (Private Cloud) — это модель развёртывания, при которой облачная инфраструктура предназначена для эксклюзивного использования одной организацией. Она может быть развернута на собственных площадках компании (on-premise private cloud), в стороннем дата-центре (hosted private cloud) или находиться под управлением специализированной третьей стороны. Ключевое отличие от классической on-premise инфраструктуры — внедрение технологий облачной автоматизации, виртуализации и самообслуживания, что обеспечивает повышенную гибкость и эффективность использования внутренних ресурсов. Частное облако обеспечивает максимальный уровень контроля, безопасности и соответствия жёстким отраслевым регуляторным требованиям (например, в финансах, госсекторе, здравоохранении). Оно позволяет глубоко кастомизировать инфраструктуру под специфичные приложения. Недостатками модели являются высокая первоначальная стоимость развёртывания и поддержки, потребность в собственных квалифицированных кадрах для управления облаком, а также ограниченная, по сравнению с публичным облаком, эластичность, зависящая от доступных физических мощностей.

Гибридное облако (Hybrid Cloud) представляет собой композицию из двух или более различных облачных инфраструктур (частного, публичного или сообщества), остающихся уникальными объектами, но объединённых между собой стандартизированными или проприетарными технологиями, обеспечивающими переносимость данных и приложений. Данная модель является доминирующей в современной корпоративной практике, так как позволяет оптимально распределить рабочие нагрузки между средами. Критически важные системы с жёсткими требованиями к безопасности и низкой задержке могут оставаться в частном облаке, в то время как системы с переменной или пиковой нагрузкой (например, веб-сайты во время маркетинговых акций, системы аналитики big data, среды разработки и тестирования) могут динамически расширяться (bursting) в публичное облако. Гибридная модель даёт бизнесу баланс между контролем и гибкостью, оптимизи-

рует затраты и снижает риски. Однако её построение и управление требуют сложной интеграции, единой платформы оркестрации (например, на базе Kubernetes), обеспечения согласованной политики безопасности и идентичности во всех средах, что повышает операционную сложность (рис. 4.1).

Рис. 4.1. Схема гибридного облака (Hybrid Cloud)

Мультиоблако (Multi-Cloud) — это стратегия использования облачных сервисов от двух и более публичных провайдеров одновременно, часто в сочетании с частными средами. В отличие от гибридного облака, фокус смещается не только на связь между приватной и публичной средой, но и на распределение нагрузок между различными публичными облаками (например, использование SaaS-сервисов от Salesforce, платформенных услуг от Azure и IaaS от AWS). Целями такой стратегии являются избегание зависимости от одного поставщика, минимизация рисков простоя, выбор оптимальных по цене и характеристикам сервисов у разных вендоров для различных задач, а также выполнение требований по резидентности данных. Мультиоблачная архитектура максимально сложна в управлении, требует высокого уровня экспертизы и использования кросс-облачных инструментов мониторинга, управления затратами и обеспечения безопасности.

Основными барьерами для внедрения мультиоблака является необходимость обеспечения совместимости между двумя облачными платформами. Но на практике этот барьер довольно легко преодолевается, если у провайдера есть команда экспертов, которая помогает и с построением инфраструктуры, и с настройкой, и с переносом данных в облако с минимальными рисками и без простоев.

Для наглядного сопоставления ключевых характеристик представленных моделей целесообразно использовать сравнительную табл. 4.2.

Таблица 4.2

Сравнительный анализ моделей развёртывания облачных инфраструктур

Критерий	Публичное облако	Частное облако	Гибридное облако	Мультиоблако
Владелец инфраструктуры	Облачный провайдер	Организация или хостинг-провайдер	Комбинация владельцев	Комбинация владельцев
Локализация	Дата-центры провайдера	Собственный ЦОД, размещённый ЦОД	Распределённая между локациями	Распределённая между провайдерами и определенными локациями
Капитальные затраты	Низкие (операционные, OpEx)	Высокие (CapEx)	Переменные	Переменные
Эластичность и масштаби-руемость	Очень высокие	Ограничены мощностью собственной инфра-структуры	Высокие за счёт использова-ния публичного облака	Максимальные за счёт выбора лучших сервисов
Уровень контроля и кастомизации	Ограниченный, в рамках предложений провайдера	Максимальный	Высокий в частной части, ограниченный в публичной	Дифференци-рованный по средам
Безопасность и соответствие	Зависит от модели ответственности провайдера, возможны риски	Максимальный контроль, идеально для строгих требований	Требует сложной интеграции политик безопасности	Максимально сложная задача управления доступом и данными
Операционная сложность	Низкая (управляется провайдером)	Высокая (требует собственных специалистов)	Очень высокая	Крайне высокая
Ключевой драйвер использования	Экономическая эффективность, скорость выхода на рынок	Безопасность, контроль, соблюдение регуляторных норм	Баланс, гибкость, оптимизация затрат и рисков	Избегание lock-in, отказоустой-чивость и выбор лучших в класс сервисов

Благодаря доступности платформенных и инфраструктурных сервисов, построить нужную инфраструктуру в облаке и даже подключить нужный стек довольно просто — во многих случаях основные этапы этих процессов можно переложить на команду облачной платформы.

В настоящее время имеет место два подхода, связанных с моделями развёртывания облачных инфраструктур компании.

Первый из них заключается со сменой платформы (replatforming). Реплатформинг (replatforming) – это подход, при котором в приложения вносятся минимальные изменения для успешной адаптации к облачной среде. Например, можно перейти на управляемые базы данных или использовать облачные сервисы для части функционала. [23]

Преимущества этого подхода состоят в следующих аспектах:

- возможность воспользоваться возможностями облака без полной переработки приложений;

- умеренные затраты времени и ресурсов.

Реплатформинг актуален, если нужно быстро перенести приложение в облако с минимальными изменениями, и/или в ситуациях, когда нужно улучшить производительность без кардинальных изменений архитектуры.

Второй подход связан со сменой архитектуры (rearchitecting).

Реархитектинг (rearchitecting) — радикальный подход, при котором приложение полностью переосмысливается и переписывается под облако.

К примерам реализации подхода можно отнести переход на микросервисную архитектуру, использование контейнеризации и внедрение DevOps-практик.

У подхода два ключевых достоинства: [24]

- максимальное использование всех преимуществ облачной среды;

---

[23] Мультиклауд: как строить распределенную инфраструктуру в 2025 году [Электронный ресурс] // Режим доступа: https://companies.rbc.ru/news/FEb0OQ7BLn/multiklaud-kak-stroit-raspredelennuyu-infrastrukturu-v-2025-godu/ (дата обращения: 19.01.2026).

[24] Затраты на IT инфраструктуру. Сравнение облака и on premise [Электронный ресурс] // Режим доступа: https://blog.cortel.cloud/2023/02/28/zemlya-ili-oblako-ekonomika-vladeniya/?ysclid=mki95pgb9q242240226 (дата обращения: 17.01.2026).

- получение высокой гибкости, масштабируемости и отказоустойчивости — приложение можно переделать так, как пожелается.

Тем не менее, реархитектинг – сложный, трудозатратный, длительный процесс. Более того, часто нельзя гарантировать, что смена архитектуры будет успешной или оправданной.

Таким образом, выбор оптимальной модели развёртывания облачной инфраструктуры является для компании многофакторной задачей, решение которой должно основываться на тщательном анализе бизнес-требований, нормативных ограничений, экономических расчетов (TCO – совокупная стоимость владения) и долгосрочной ИТ-стратегии. Современный тренд свидетельствует о движении от чистых моделей к гибридным и мультиоблачным подходам, которые обеспечивают необходимую бизнесу гибкость и устойчивость. Успешная реализация таких сложных инфраструктур невозможна без внедрения принципов DevOps, использования технологий контейнеризации и оркестрации (Kubernetes), а также инфраструктуры как кода (IaC), которые обеспечивают согласованность, воспроизводимость и управляемость распределённых сред. Таким образом, модели развёртывания облачных инфраструктур представляют собой не статичный выбор, а динамичный фундамент для построения цифровой трансформации предприятия.

4.3. Сравнительный анализ моделей On-Premise, IaaS, PaaS, SaaS, FaaS

Эволюция моделей развертывания информационной инфраструктуры, от традиционной On-Premise до спектра облачных сервисов (IaaS, PaaS, SaaS) и более специализированной модели FaaS, отражает поступательный процесс абстрагирования потребителя от управления физическими ресурсами и низкоуровневыми компонентами стека информационных технологий. Сравнительный анализ этих моделей позволяет выявить фундаментальные различия в распределении ответственности между поставщиком и потребителем, уровне управляемости, экономической модели и операционной гибкости, что является критически важным для обоснованного выбора архитектуры в рамках цифровой трансформации.

Ключевым системообразующим признаком для классификации и сравнения моделей является разделение ответственности за компоненты ИТ-стека.

В модели On-Premise организация владеет и управляет всей цепочкой: от физической инфраструктуры (здания, электропитание, охлаждение, серверы, сетевое оборудование, системы хранения данных) до операционных систем, сред исполнения, промежуточного программного обеспечения, данных и прикладного кода. Данная модель обеспечивает максимальный контроль и возможность глубокой кастомизации, но одновременно налагает полное бремя капитальных затрат (CAPEX), эксплуатационных расходов (OPEX) на содержание штата специалистов, а также ответственность за обеспечение безопасности, отказоустойчивости и обновления всех компонентов.

Модель «Инфраструктура как услуга» (IaaS) представляет собой первый уровень абстракции в облаке. Поставщик услуги берет на себя ответственность за физический уровень (дата-центры, оборудование) и базовую виртуализацию, предоставляя потребителю в аренду виртуальные вычислительные ресурсы (виртуальные машины), хранилища данных, сетевые функции и балансировку нагрузки. Потребитель сохраняет полный контроль над операционными системами, установленным ПО, конфигурациями и приложениями, развернутыми на арендованной виртуальной инфраструктуре. Таким образом, IaaS устраняет необходимость в капитальных вложениях в железо и трансформирует затраты в операционные, но оставляет за клиентом сложные задачи по управлению и сопровождению операционных систем и вышележащего стека.

Следующей ступенью абстракции является модель «Платформа как услуга» (PaaS). В этой модели поставщик управляет не только физической инфраструктурностью, но и операционными системами, серверами приложений, системами управления базами данных, middleware и другими инструментами, необходимыми для жизненного цикла разработки и развертывания ПО. Потребитель фокусируется исключительно на разработке, размещении и управлении собственным прикладным кодом и данными. PaaS значительно ускоряет процессы разработки и вывода продукта на рынок, автоматизируя рутинные операции по обеспечению среды исполнения, ее масштабированию и об-

новлению. Однако эта модель ограничивает свободу выбора конкретных версий системного ПО и глубокой низкоуровневой настройки среды.

Модель «Программное обеспечение как услуга» (SaaS) представляет собой наиболее законченный вариант облачной услуги, при котором потребителю предоставляется доступ к готовому прикладному программному обеспечению, работающему в облачной инфраструктуре поставщика. Управление всей нижележащей платформой, инфраструктурой и кодом приложения полностью осуществляется поставщиком. Пользователь взаимодействует с приложением, как правило, через веб-браузер или тонкий клиент, и отвечает лишь за управление своими данными внутри приложения и настройку пользовательских параметров (конфигураций).

SaaS полностью снимает с организации бремя установки, обновления, резервного копирования и технической поддержки ПО, но при этом предлагает наименьшую степень кастомизации и контроля над приложением (рис. 4.2).

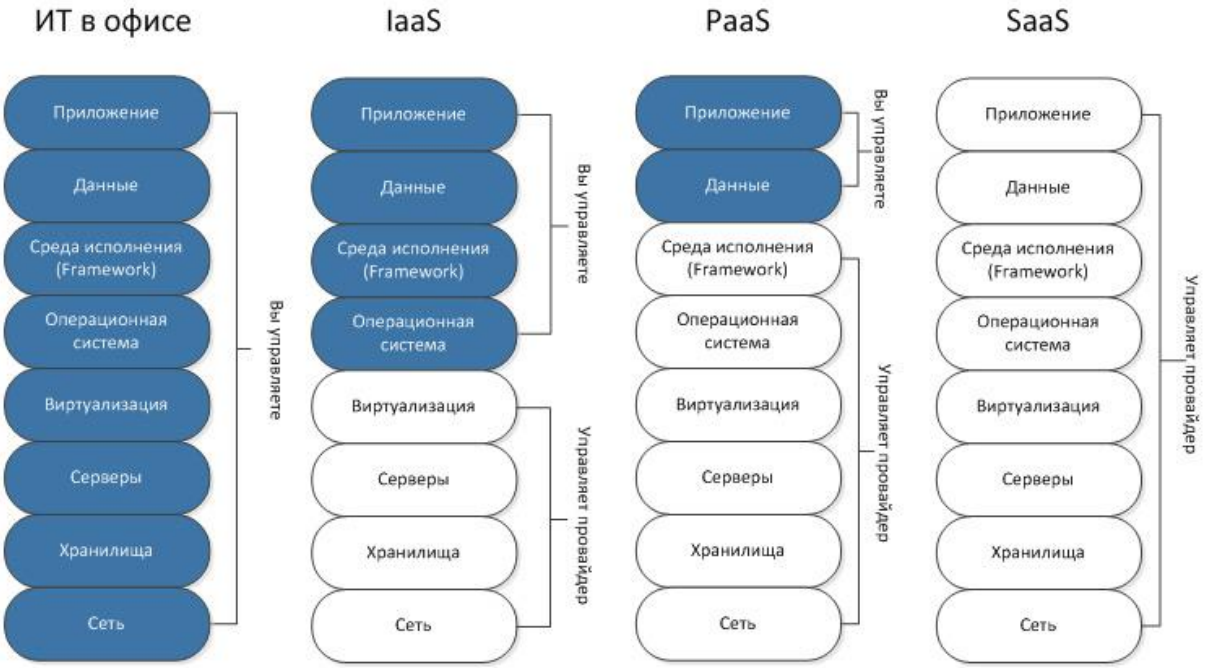

Рис. 4.2. Зоны ответственности компании и провайдера в моделях On-Premise, IaaS, PaaS, SaaS при построении ее цифровой инфраструктуры

Сравнительные характеристики и распределение зон ответственности в основных моделях представлены в табл. 4.3.[25]

Отдельное место в этом континууме занимает модель «Функция как услуга» (FaaS), являющаяся эволюционным развитием и специализацией концепций PaaS и контейнеризации в рамках парадигмы бессервисных вычислений. FaaS предоставляет платформу, на которой потребитель может развертывать отдельные функции (логические блоки кода), выполняемые в ответ на определенные события (HTTP-запрос, добавление файла в хранилище, сообщение в очереди). Ключевое отличие от классической PaaS заключается в полной абстракции от понятия постоянно работающего сервера (даже виртуального).

Таблица 4.3

Сравнительный анализ моделей On-Premise, IaaS, PaaS, SaaS

Компонент стека	On-Premise	IaaS	PaaS	SaaS
Физическая инфраструктура (дата-центр, железо, сети)	Клиент	Поставщик	Поставщик	Поставщик
Виртуализация и абстракция ресурсов	Клиент	Поставщик	Поставщик	Поставщик
Операционные системы и их обновления	Клиент	Клиент	Поставщик	Поставщик
Промежуточное ПО (серверы приложений, СУБД)	Клиент	Клиент	Поставщик	Поставщик
Прикладное программное обеспечение	Клиент	Клиент	Клиент	Поставщик
Данные, контент, идентификационная информация	Клиент	Клиент	Клиент	Клиент

Поставщик FaaS-платформы динамически управляет выделением и масштабированием вычислительных ресурсов для исполнения кода функции, взимая плату исключительно за время фактического выполнения кода и количество потребленных ресурсов (рис. 4.3).

[25] Рынок облачных технологий в России: импортозамещение, безопасность данных и перспективы роста [Электронный ресурс]// Режим доступа: https://delprof.ru/press-center/open-analytics/rynok-oblachnykh-tekhnologiy-v-rossii-importozameshchenie-bezopasnost-dannykh-i-perspektivy-rosta/?ysclid=mki93a78ef979797749 (дата обращения: 17.01.2026).

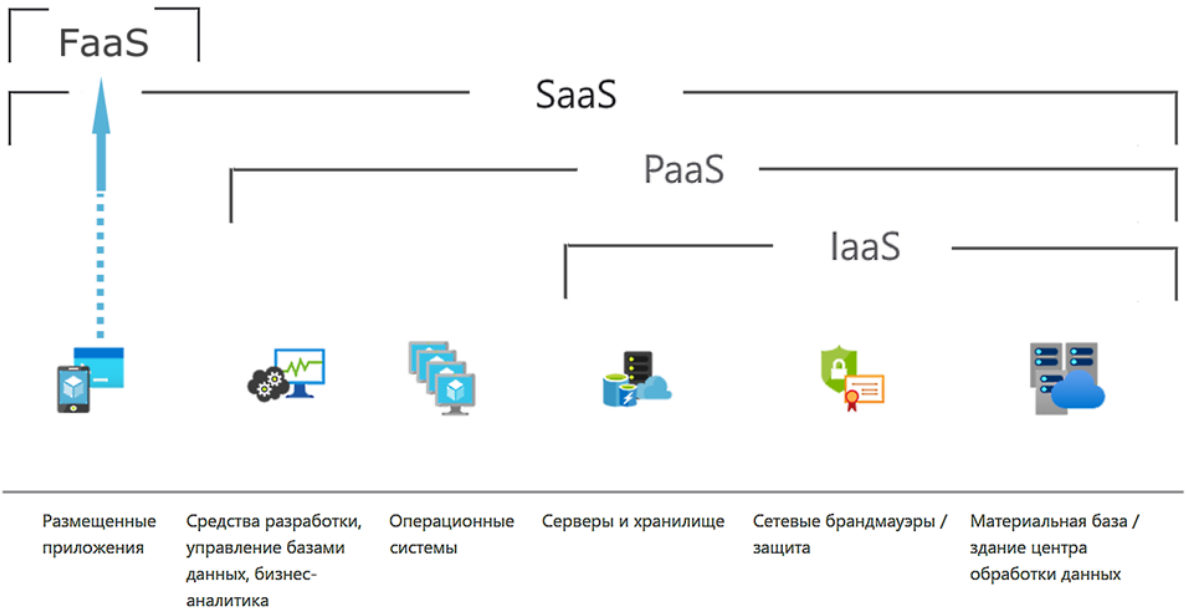

Рис. 4.3. Схема работы модели «Функция как услуга» (FaaS)

Потребитель ответственен только за код функции и ее конфигурацию, в то время как все аспекты инфраструктуры, операционной системы, среды исполнения и горизонтального масштабирования полностью инкапсулированы у поставщика. Это позволяет достичь чрезвычайно высокой эффективности использования ресурсов и автоматического масштабирования до нуля, что идеально подходит для событийно-ориентированных, непостоянных рабочих нагрузок. Однако данная модель налагает архитектурные ограничения (например, ограничения по времени исполнения и сложности) и может порождать проблемы, связанные с «холодным стартом».

В экономическом аспекте модели демонстрируют переход от модели капитальных затрат (CAPEX) к модели операционных расходов (OPEX). On-Premise требует значительных первоначальных инвестиций в оборудование и лицензии, а также постоянных затрат на обслуживание и амортизацию. Облачные модели IaaS, PaaS, SaaS функционируют по принципу оплаты по мере использования (pay-as-you-go или pay-per-use), что повышает операционную гибкость и позволяет оптимизировать расходы, масштабируя потребляемые ресурсы в соответствии с текущей нагрузкой. FaaS доводит эту модель до крайности, реализуя оплату за фактическое время выполнения с точностью до миллисекунд, минимизируя затраты на простой.

С точки зрения управляемости и контроля наблюдается обратная зависимость: чем выше уровень сервиса (и абстракции), тем ниже степень контроля со стороны потребителя над нижележащими слоями. On-Premise предлагает максимальный контроль, сопряженный с максимальной сложностью управления. SaaS предлагает минимальный контроль, но и минимальную операционную нагрузку. IaaS и PaaS занимают промежуточное положение, предлагая баланс между контролем и избавлением от рутинных задач. FaaS, будучи высокоуровневой моделью, предлагает уникальный баланс: почти нулевую управляемость инфраструктурой при сохранении полного контроля над бизнес-логикой в коде функции.

Вопросы безопасности и соответствия требованиям также распределяются по принципу разделенной ответственности. В облачных моделях поставщик гарантирует безопасность облака (безопасность физической инфраструктуры и гипервизора в IaaS, платформы в PaaS), в то время как потребитель отвечает за безопасность в облаке (конфигурация гостевых ОС и приложений в IaaS, настройки доступа к данным и приложениям в PaaS/SaaS). Для моделей SaaS и FaaS доля ответственности потребителя за безопасность смещается в область управления доступом, конфигурации приложения и защиты данных.

Выбор оптимальной модели развертывания является стратегическим решением и должен основываться на комплексном анализе требований к приложению или сервису. Критическими факторами являются: необходимость глубокой кастомизации и контроля (склоняет к On-Premise или IaaS), скорость разработки и вывода на рынок (склоняет к PaaS, SaaS), характер рабочей нагрузки (постоянная или событийно-управляемая, пиковая), экономическая модель организации, а также требования регуляторов к хранению и обработке данных. Современные гибридные и мультиоблачные стратегии зачастую предусматривают использование комбинации этих моделей в рамках единой информационной экосистемы, что позволяет извлекать преимущества из каждой в зависимости от решаемой бизнес-задачи.

4.4. Гибридная и мультиоблачная стратегия: преимущества, сложности и модели управления

В эволюции моделей развертывания информационной инфраструктуры гибридная и мультиоблачная стратегии представляют собой закономерный этап, отражающий стремление организаций к оптимизации, гибкости и снижению рисков. Эти модели не являются взаимоисключающими: мультиоблачный подход часто становится архитектурной основой для гибридной среды. Гибридная облачная модель предполагает интеграцию ресурсов частного облака (on-premise или стороннего) с одним или несколькими публичными облаками, объединенными скоординированными процессами управления. Мультиоблачная стратегия подразумевает использование услуг нескольких публичных облачных провайдеров (таких как AWS, Microsoft Azure, Google Cloud Platform) для распределения рабочих нагрузок, часто без обязательной глубокой интеграции с частным ЦОД. Переход к этим моделям обусловлен не технологической модой, а прагматичным ответом на комплексные бизнес-требования, однако он сопровождается значительным усложнением архитектуры и процессов управления.

Преимущества гибридных и мультиоблачных сред носят стратегический и тактический характер. Ключевым драйвером является избегание зависимости от единственного поставщика (vendor lock-in), что повышает переговорную способность компании, снижает долгосрочные риски и обеспечивает свободу выбора лучшего в своем классе сервиса для каждой конкретной задачи. Вторая группа преимуществ связана с оптимизацией затрат и производительности. Организация может размещать предсказуемые, чувствительные к задержкам или регулируемые рабочие нагрузки в частной инфраструктуре, а для пиковых, экспериментальных или глобально распределенных нагрузок эластично использовать публичные облака, применяя модель «оплаты по факту использования». Мультиоблако позволяет выбирать наиболее экономичные или высокопроизводительные сервисы у разных провайдеров для различных компонентов единого приложения.

Третье существенное преимущество — повышение отказоустойчивости и надежности. Распределение систем между географически и платформенно разрозненными средами минимизирует риск катастро-

фического отказа из-за сбоя у одного провайдера или в одном дата-центре. Наконец, эти модели поддерживают поэтапную трансформацию, позволяя переносить унаследованные (устаревшие) системы в облако постепенно, без большой миграции, и соблюдать регуляторные требования, оставляя чувствительные данные в юрисдикционно контролируемом частном сегменте.

Сложности и вызовы, порождаемые гибридными и мультиоблачными средами, носят фундаментальный характер и требуют пересмотра ИТ-операций. Основная проблема — высокая степень архитектурной и операционной сложности. Неоднородность сред приводит к фрагментации инструментов мониторинга, управления безопасностью, развертывания и резервного копирования. Администраторам необходимо обладать экспертизой в нескольких, зачастую несовместимых, технологических стеках. Управление безопасностью и комплаенсом становится крайне трудной задачей, так как периметр защиты размывается, а единые политики необходимо применять к разнородным средам с разными механизмами контроля. Управление затратами, вопреки потенциальной экономии, может выйти из-под контроля из-за сложности отслеживания расходов по множеству счетов от разных провайдеров, эффекта «скрытых» затрат на передачу данных и неоптимального размещения ресурсов. Проблема переносимости данных и приложений упирается в отсутствие полной совместимости API, форматов данных и сервисов у разных облачных вендоров, что может нивелировать преимущества от отсутствия привязки к поставщику. Наконец, возникает дефицит кадров с необходимым кросс-платформенным опытом и культурой работы в распределенных гетерогенных средах.

Эффективное управление гибридной и мультиоблачной инфраструктурой невозможно без внедрения специальных моделей и платформ управления. Центральное место среди них занимают концепции единой плоскости управления (unified management plane) и унифицированных операционных моделей. Реализуются они через несколько ключевых технологических подходов.

1. Использование платформ контейнеризации и оркестрации, прежде всего Kubernetes. Kubernetes, особенно в его дистрибутивах, способных работать поверх любой инфраструктуры (например, Red Hat OpenShift, VMware Tanzu, SUSE Rancher), становится абстрактным слоем, нивелирующим различия между облаками. Он обеспечивает

единый механизм для развертывания, масштабирования и управления контейнеризированными приложениями, декларируя инфраструктуру как код (IaC). Это резко повышает переносимость и согласованность сред.

2. Внедрение решений для мультиоблачного управления (Multicloud Management Platforms, CMP) и облачных брокеров услуг (Cloud Service Brokerage, CSB). Эти платформы предоставляют единую консоль для предоставления ресурсов, мониторинга, управления затратами и обеспечения безопасности в различных облаках. Они автоматизируют жизненный цикл ресурсов, применяют политики управления доступом и затратами, предоставляя аналитику по использованию и оптимизации.

3. Применение принципов Infrastructure as Code (IaC) и GitOps. Использование декларативных инструментов, таких как Terraform, Ansible или Pulumi, позволяет описывать и версионировать желаемое состояние инфраструктуры в любом облаке из единого репозитория. GitOps-практики распространяют модель CI/CD на инфраструктуру, обеспечивая ее воспроизводимость, аудируемость и быстрое восстановление.

4. Архитектура, ориентированная на сервисы (Service Mesh). Для сложных распределенных приложений, компоненты которых работают в разных облаках, становится критичным управление межсервисной коммуникацией, безопасностью и наблюдаемостью. Сервис-меши, такие как Istio или Linkerd, создают выделенный инфраструктурный слой для этого, обеспечивая сквозное шифрование, балансировку нагрузки, контроль доступа и телеметрию трафика между сервисами независимо от их физического размещения.

Сравнительный анализ ключевых аспектов управления представлен в табл. 4.4.

Таблица 4.4

Сравнительные аспекты управления в гибридной и мультиоблачной средах

Аспект управления	Традиционный подход (разрозненные инструменты)	Современный подход (унифицированная платформа)
Подготовка инфраструктуры	Ручная работа в консолях каждого облака; скрипты, специфичные для провайдера.	Автоматизация через Infrastructure as Code (Terraform, Crossplane) с поддержкой мультиоблака из единого шаблона.
Оркестрация приложений	Разные механизмы в каждом окружении (VMware, Cloud-native сервисы).	Единый оркестратор (Kubernetes), абстрагирующий underlying-инфраструктуру.
Наблюдаемость (Monitoring)	Несвязанные стеки мониторинга, отсутствие единой картины, сложность корреляции.	Единая платформа для агрегации логов, метрик и трассировок со всех сред (например, Grafana stack, Datadog).
Безопасность и комплаенс	«Ручное» применение политик в каждой среде, фрагментированный аудит.	Политика как код (Policy as Code), централизованное управление доступом (IAM), сканирование конфигураций.
Управление затратами	Анализ множества счетов, ручное выявление неоптимальных ресурсов.	Использование специализированных инструментов (CloudHealth, CloudCheckr) для кросс-облачного анализа и оптимизации.
Аварийное восстановление	Сложные, уникальные для каждого сайта и облака процедуры.	Унифицированные планы, автоматизируемые через IaC и оркестрацию, возможность миграции workloads между облаками.

В заключение необходимо подчеркнуть, что успешная реализация гибридной и мультиоблачной стратегии зависит не столько от технологий, сколько от зрелости процессов и организационной культуры. Она требует перехода от управления инфраструктурой к управлению услугами и продуктами, где единицей планирования становится приложение, а не сервер или облачный регион. Необходимо формирование кросс-функциональных команд (по модели DevOps/Platform

Engineering), ответственных за полный жизненный цикл сервиса в любой среде. Таким образом, гибридные и мультиоблачные модели представляют собой не просто техническую архитектуру, а стратегическую парадигму, которая при грамотном управлении, основанном на унификации, автоматизации и глубокой абстракции, позволяет достичь беспрецедентного уровня гибкости, устойчивости и эффективности информационной инфраструктуры в условиях цифровой трансформации.

4.5. Вендоры и выбор облачного провайдера

В контексте эволюции моделей развертывания информационной инфраструктуры от on-premise решений к облачным парадигмам, вопрос выбора вендора и конкретного облачного провайдера перестает быть сугубо техническим и трансформируется в стратегическую задачу, определяющую долгосрочную эффективность, безопасность и конкурентную устойчивость организации. Данный выбор представляет собой комплексную многофакторную проблему, лежащую на пересечении технологических, экономических, юридических и операционных аспектов. Его анализ требует системного подхода, учитывающего как текущее состояние ИТ-ландшафта предприятия, так и траекторию его будущего развития.

Фундаментальным отличием облачной модели от традиционной on-premise является переход от владения капиталоемкими активами (железо, системы хранения, ЦОД) к потреблению услуг (IaaS, PaaS, SaaS) на основе контракта с внешним провайдером. Это смещает фокус с компетенций по управлению физической инфраструктурой к навыкам управления взаимоотношениями с вендором, контролю качества услуги (SLA) и интеграции разнородных сервисов. Вендор в облачной экосистеме выступает не просто как поставщик оборудования или лицензий, а как партнер, от архитектуры и политик которого напрямую зависит работоспособность бизнес-процессов клиента.

Современный рынок облачных услуг характеризуется выраженной стратификацией. На верхнем уровне находятся глобальные гиперскейлеры (hyperscalers) – компании, обладающие распределенной сетью дата-центров планетарного масштаба и предлагающие максимально широкий и глубокий портфель сервисов. Лидерами этого сегмента являются Amazon Web Services (AWS), Microsoft Azure и Google

Cloud Platform (GCP). Их ключевые преимущества заключаются в беспрецедентной эластичности, инновационности (быстрый вывод новых сервисов, интеграция искусственного интеллекта и машинного обучения), глобальном покрытии и развитой партнерской экосистеме. Однако взаимодействие с гиперскейлерами может создавать риски «замыкания» на одном поставщике (зависимости от вендора) из-за использования проприетарных API и сервисов, а также требовать высокой квалификации внутренних команд.

Параллельно существует сегмент нишевых и региональных провайдеров, которые фокусируются на конкретных вертикалях, строгих требованиях к резидентности данных или специализированных сервисах (например, управляемый Kubernetes).

Отдельную категорию составляют поставщики телекоммуникационных услуг, предлагающие облачные решения, часто с акцентом на гибридные сценарии и сети доставки контента (CDN). Важным явлением в ряде регионов, особенно с учетом требований регуляторов, является развитие национальных или суверенных облаков, физически и юридически локализованных на территории определенной страны.

Для структурированного анализа при выборе провайдера целесообразно оценивать критерии по следующим ключевым группам:

1. Технологические и архитектурные критерии: полнота и зрелость сервисного каталога (вычислительные мощности, хранение, сети, базы данных, аналитика, AI/ML); совместимость со стеками технологий, используемых в организации (например, поддержка конкретных ОС, СУБД, middleware); возможности для контейнеризации и оркестрации (Kubernetes); качество и пропускная способность глобальной сети; уровень автоматизации предоставления и управления ресурсами (API, CLI, Terraform); стратегия в области устойчивого развития (зеленые технологии).

2. Экономические критерии: прозрачность и гибкость моделей ценообразования (pay-as-you-go, резервирование инстансов, spot-инстансы, лицензирование ПО); инструменты для детального мониторинга затрат, бюджетного контроля и прогнозирования (FinOps); отсутствие скрытых платежей; общая стоимость владения (TCO) в сравнении с on-premise или другими облачными моделями.

3. Операционные критерии и безопасность: гарантированный уровень обслуживания, формализованный в Соглашении об уровне

услуг (SLA) по доступности, производительности и времени восстановления; прозрачность инцидент-менеджмента и процедур уведомления; соответствие международным и отраслевым стандартам безопасности и комплаенса (ISO 27001, SOC 1/2/3, PCI DSS, HIPAA, GDPR); модель ответственности (Shared Responsibility Model) и инструменты безопасности, предоставляемые провайдером.

Стратегические и юридические критерии: география и юрисдикция регионов и зон доступности, соответствие требованиям о резидентности данных; репутация и финансовая устойчивость провайдера; открытость к гибридным и мультиклаудным сценариям, поддержка стандартов и инструментов, минимизирующих lock-in; качество технической поддержки и документации; зрелость партнерской и интеграционной экосистемы.

Следует особо отметить, что стратегия многих организаций сегодня эволюционирует в сторону мультиклауда – преднамеренного использования услуг двух или более облачных провайдеров. Данный подход позволяет избежать монопольной зависимости, оптимизировать затраты, выбирая лучшие в своем классе сервисы у разных вендоров, а также повысить отказоустойчивость.

Однако он существенно увеличивает сложность управления, требуя наличия единой платформы для оркестрации, мониторинга и обеспечения безопасности (cloud management platform) и соответствующих компетенций (табл. 4.5).

Процесс выбора должен быть итеративным и включать этапы: формирования рабочей группы с участием технических специалистов, финансового отдела и представителей бизнеса; определения функциональных и нефункциональных требований; проведения предварительного отбора 2-3 провайдеров; реализации пилотного проекта (Proof of Concept, PoC) для проверки ключевых гипотез на реальной нагрузке; детальной оценки экономической модели и правового анализа договорных условий, особенно SLA и политик ответственности.

Таблица 4.5

Сравнительный анализ ключевых аспектов выбора облачного провайдера

Критерий	Гиперскейлеры (AWS, Azure, GCP)	Нишевые / Региональные провайдеры	Национальные / Суверенные облака
Сервисный портфель	Максимально широкий и глубокий, включая инновационные сервисы (AI/ML, IoT, квантовые вычисления).	Сфокусированный на специфических услугах (HPC, хостинг игр, CDN) или ограниченный набор базовых IaaS/PaaS.	Часто ограниченный набор базовых IaaS и PaaS-сервисов, соответствующий требованиям регуляторов.
Глобальное присутствие	Обширная сеть регионов и зон доступности по всему миру.	Ограниченное одним или несколькими регионами, часто в пределах одной страны или континента.	Локализовано строго в пределах национальной юрисдикции.
Комплаенс и безопасность	Поддерживают широкий спектр международных сертификатов; клиент сам конфигурирует многие параметры безопасности.	Могут предлагать специализированные отраслевые решения; более персонализирован-ный подход к требованиям безопасности.	Приоритет – полное соответствие национальному законодательству о данных; часто строго регламентирован-ные настройки безопасности.
Экономическая модель	Сложные, но гибкие схемы ценообразования; мощные инструменты анализа затрат; требуют высокой квалификации для оптимизации бюджета.	Часто более простые и предсказуемые тарифные планы; возможны индивидуальные условия для крупных клиентов.	Цены могут быть выше из-за меньшей масштабируемости и специфических требований; финансирование может быть государственным.
Риск vendor lock-in	Высокий из-за активного использования уникальных проприетарных сервисов и API.	Может варьироваться; ниже при использовании стандартных технологий (например, VMware, OpenStack).	Часто высокий, обусловленный не только технологиями, но и нормативными требованиями.
Ключевое преимущество	Масштаб, инновации эластичность, глобальность.	Специализация, гибкость, персональный сервис, соответствие специфическим требованиям.	Гарантированное соблюдение требований резидентности данных и национального законодательства.

Таким образом, выбор облачного провайдера в современной парадигме развертывания информационной инфраструктуры является комплексным стратегическим решением, определяющим технологическую и операционную гибкость компании на годы вперед. Отказ от универсальных рецептов в пользу тщательного анализа собственного контекста, взвешенной оценки технологических возможностей, экономических моделей и долгосрочных рисков, включая зависимость от вендора, является необходимым условием для успешной цифровой трансформации и построения эффективной, надежной и безопасной информационной инфраструктуры.

4.6. Управление затратами в облаке (FinOps)

Переход от традиционной модели информационной инфраструктуры on-premise к облачной парадигме кардинально трансформирует подходы к финансированию и учету затрат на ИТ. Капитальные затраты (CapEx), характерные для приобретения и содержания собственного оборудования, заменяются операционными расходами (OpEx), основанными на потреблении облачных ресурсов по принципу «плати только за то, что используешь». Данное изменение, обеспечивающее гибкость и масштабируемость, порождает и принципиально новую проблему: сложность прогнозирования, контроля и оптимизации постоянно изменяющихся расходов в условиях самообслуживания и практически неограниченной ресурсной емкости. В результате в корпоративной практике сформировалась дисциплина FinOps (Financial Operations) - культурная парадигма и набор практик, направленных на повышение финансовой ответственности и подотчетности в процессе потребления облачных услуг для достижения оптимального соотношения ценности и затрат.

FinOps не является исключительно технологическим процессом или инструментом; это в первую очередь культурная трансформация, требующая коллаборации между финансистами, техническими специалистами и бизнес-подразделениями. Ее цель — создание единой системы взаимодействия, где каждый участник облачного цикла несет ответственность за свои решения, влияющие на стоимость. Финансовые команды получают инструменты и данные для точного бюджетирования и отчетности, инженеры и разработчики — прозрачность о

стоимости своих архитектурных решений в реальном времени, а бизнес-лидеры — возможность принимать взвешенные решения о балансе между производительностью, скоростью выхода на рынок и затратами. Ключевые принципы FinOps включают: коллективную ответственность за облачные расходы; централизованное принятие решений на основе достоверных, своевременных данных; необходимость отчетности и бенчмаркинга; и, наконец, непрерывную итеративную оптимизацию потребления ресурсов. Внедрение FinOps включает 4 аспекта, представленных на рис. 4.4.

Рис. 4.4. Направления внедрения FinOps

Процесс FinOps носит циклический и итерационный характер, состоящий из трех взаимосвязанных фаз: информирования, оптимизации и операций.

- Фаза 1: информирование (Inform). Данная фаза направлена на достижение полной видимости (Visibility) и распределения (Allocation) затрат. В облачной модели с ее виртуализированными, эфемерными и динамически создаваемыми ресурсами классический учет неприменим. Требуется инструментарий для сбора, агрегации и категоризации данных о расходах из различных облачных сервисов (вычислительные мощности, хранение данных, сетевой трафик, лицензии ПО). Ключевая практика — распределение затрат по центрам финансовой ответственности (ЦФО) через систему тегирования ресурсов (например, по про-

екту, отделу, окружению, приложению). Это позволяет ответить на вопрос «кто и на что потратил деньги». Важным элементом является также создание прогнозов (Forecasting) на основе исторических данных и планов развертывания, что формирует основу для бюджетирования. Бенчмаркинг, сравнение своих показателей удельной стоимости (например, стоимость на транзакцию, на пользователя) с внутренними или отраслевыми нормами, позволяет оценить эффективность.

- Фаза 2: оптимизация (Optimize). На этапе оптимизации фокус смещается на повышение экономической эффективности использования уже развернутых ресурсов. Практики включают:

1. Устранение неиспользуемых ресурсов: автоматическое выявление и удаление «осиротевших» дисков (orphaned volumes), неиспользуемых IP-адресов, остановленных виртуальных машин, что является одним из самых быстрых способов снижения затрат.

2. Выбор оптимального типа и размера ресурсов (Right-sizing): анализ фактической нагрузки на вычислительные инстанции (использование CPU, памяти, дискового ввода-вывода) и их последующий масштаб (downsizing) или изменение типа на более подходящий и дешевый (например, переход с инстансов общего назначения на инстансы, оптимизированные для вычислений или памяти).

3. Использование моделей дисконтирования: облачные провайдеры предлагают значительные скидки за предсказуемую нагрузку. Ключевыми инструментами являются резервирование инстансов (Reserved Instances) на 1 или 3 года с предоплатой или без нее для регулярно работающих сервисов, а также использование спотовых (прерываемых) инстансов (Spot Instances) для fault-tolerant и гибких рабочих нагрузок (например, пакетная обработка, CI/CD), стоимость которых может быть на 60-90% ниже.

4. Оптимизация архитектуры: пересмотр архитектуры приложений в сторону использования полностью управляемых бессерверных сервисов (Serverless, такие как AWS Lambda, Azure Functions), которые позволяют перейти от оплаты за время работы инстанса к оплате за фактическое выполнение кода, или применение автоматического масштабирования (Autoscaling) для адаптации ресурсов к текущей нагрузке.

- Фаза 3: операции (Operate). На этой фазе происходит формализация и интеграция лучших практик в ежедневные бизнес-процессы организации. Это включает настройку механизмов управления (Governance) — политик и лимитов расходов, внедрение утверждения бюджетов на новые облачные инициативы, создание кросс-функциональных FinOps-команд.[26]

Автоматизация играет критическую роль: от автоматического применения тегов и уведомлений о превышении бюджетных порогов до автоматического масштабирования и остановки неиспользуемых ресурсов в нерабочее время.

Сравнительная таблица практик оптимизации затрат представлена ниже в табл. 4.6.

Реализация FinOps невозможна без специализированных инструментов. К их числу относятся нативные сервисы облачных провайдеров (AWS Cost Explorer, Azure Cost Management, Google Cloud Billing), а также платформы сторонних вендоров (Flexera, CloudHealth, Apptio Cloudability), предоставляющие кросс-облачную аналитику, рекомендации по оптимизации и расширенные возможности управления политиками.

Эффективность внедрения FinOps измеряется не только абсолютным снижением счета за облако, но и бизнес-метриками: стоимостью на единицу продукции (Unit Economics), процентом использования зарезервированных инстансов, скоростью обнаружения и исправления аномальных расходов (Anomaly Detection), а также показателем возврата инвестиций в облако (Cloud ROI).

Таким образом, FinOps представляет собой эволюционный ответ на финансовые вызовы облачной модели развертывания. Он синтезирует финансовый менеджмент, технологическую экспертизу и бизнес-стратегию, превращая облачные расходы из непредсказуемой переменной в управляемый источник конкурентного преимущества. Внедрение культуры FinOps позволяет организациям в полной мере реализовать гибкость и инновационный потенциал облаков, избегая при этом ловушек неконтролируемого роста затрат.

---

[26] Какие технологии помогают бизнесу построить единую ИТ-инфраструктуру [Электронный ресурс] // Режим доступа: https://digtlab.ru/tpost/rzfhfyfrr1-kakie-tehnologii-pomogayut-biznesu-postr?ysclid=mk2z328jiu499207780 (дата обращения: 06.01.2026).

Таблица 4.6

Обобщенная характеристика практик оптимизации затрат

Практика	Сущность	Тип рабочей нагрузки	Экономический эффект
Ликвидация неиспользуемых ресурсов	Поиск и удаление ресурсов, не связанных с работающими сервисами	Любая	Немедленное снижение затрат без влияния на производитель-ность
Правильный выбор размера (Right-sizing)	Корректировка конфигурации вычислительных инстансов в соответствии с фактическими метриками использования.	Устойчивые нагрузки с предсказуемыми паттерном, но с неоптимальным первоначальным выбором.	Снижение ежемесячных затрат на 10-40% за счет отказа от переплаты за избыточную мощность.
Зарезервированные инстансы (RI)	Предоплата или обязательство по использованию инстанса определенного типа на 1 или 3 года.	Стабильные, долгосрочные, непрерывные рабочие нагрузки (базы данных, бэкенд-сервисы).	Скидка до 60-70% по сравнению с оплатой по факту использования (On-Demand).
Спотовые инстансы (Spot)	Использование излишков вычислительной мощности облака по аукционной модели с возможностью прерывания.	Отказоустойчивые, прерываемые, гибкие нагрузки (HPC, рендеринг, CI/CD, анализ данных).	Скидка до 90% по сравнению с On-Demand, но с риском прерывания работы.
Автоматическое масштабирование	Динамическое добавление или удаление ресурсов в ответ на изменение нагрузки.	Рабочие нагрузки с переменной, предсказуемой или непредсказуемой активностью (веб-сервисы).	Оптимизация затрат за счет соответствия предоставленных ресурсов реальной необходимости.
Архитектурная оптимизация (Serverless)	Переход на event-driven, бессерверные вычисления, где оплата взимается за время выполнения кода.	Событийно-управляемые, кратковременные или нерегулярные задачи (API-бэкенды, обработка файлов).	Устранение затрат на простой ресурсов (idle time), оплата только за полезную работу.

В контексте эволюции от on-premise к облачным моделям FinOps становится неотъемлемым компонентом стратегического управления информационной инфраструктурой, обеспечивая баланс между скоростью разработки, стабильностью работы и финансовой эффективностью в цифровую эпоху.

Подводя итоги сказанного выше, необходимо отметить, что анализ моделей развертывания информационной инфраструктуры от локальных (on-premise) до облачных демонстрирует эволюционный переход от парадигмы владения капиталоемкими физическими активами к парадигме потребления ИТ-ресурсов как сервиса. Этот переход обусловлен не только технологическим прогрессом, но и изменением экономических моделей, требований к гибкости бизнеса и скорости внедрения инноваций. Каждая из рассмотренных моделей — классическая локальная, частное, публичное, гибридное и мультиоблако, а также спектр сервисных моделей (IaaS, PaaS, SaaS, FaaS) - представляет собой компромисс между уровнем контроля, безопасностью, операционной гибкостью и экономической эффективностью.

Локальное развертывание сохраняет свою актуальность для задач, требующих максимального суверенитета, детерминированной производительности, глубокой кастомизации и соблюдения жестких регуляторных требований, однако сопряжено с высокими капитальными затратами, длительным циклом обновления и ограниченной эластичностью. Облачные модели, напротив, обеспечивают преобразование капитальных затрат в операционные, беспрецедентную масштабируемость и скорость предоставления ресурсов, но вводят новые проблемы в области безопасности по модели разделенной ответственности, управления затратами и потенциальной зависимости от вендора.

Доминирующим трендом современной корпоративной практики является движение к гибридным и мультиоблачным средам. Эти архитектуры позволяют организациям реализовать стратегически взвешенный подход, размещая критические и регулируемые рабочие нагрузки в контролируемом частном сегменте, а переменные, инновационные или глобально распределенные нагрузки — в публичных облаках.

Мультиоблачная стратегия дополнительно снижает риски, повышает отказоустойчивость и позволяет выбирать оптимальные сервисы у разных провайдеров. Однако реализация этих преимуществ сопряжена со значительным ростом операционной сложности, требующим внедрения единых плоскостей управления на базе технологий контейнеризации,

инфраструктуры как кода (IaC), сервис-мешей и специализированных платформ мультиоблачного управления.

Ключевым выводом является то, что выбор модели развертывания перестает быть единовременным техническим решением и превращается в динамическую стратегию, требующую постоянной адаптации. Успешность ее реализации определяется не столько технологическим выбором, сколько зрелостью организационных процессов, культурой коллаборации между бизнесом, разработкой и эксплуатацией (DevOps/Platform Engineering), а также наличием компетенций в области финансового управления облачными расходами (FinOps).

Таким образом, современная информационная инфраструктура представляет собой сложную, адаптивную и комбинированную экосистему, эффективное управление которой основано на глубоком понимании взаимосвязей между архитектурными моделями, экономическими принципами и операционными практиками в контексте непрерывной цифровой трансформации предприятия.

Вопросы для обсуждения

1. Дайте определение и перечислите основные модели локального развёртывания программного обеспечения.
2. Охарактеризуйте основные этапы процесса внедрения моделей on-premise.
3. Укажите ключевые драйверы выбора модели on-premise.
4. Перечислите основные модели развёртывания облачных инфраструктур компании.
5. Охарактеризуйте особенности публичных, частных и гибридных облаков.
6. Укажите основные характеристики и направления использования мультиоблачной архитектуры (Multi-Cloud).
7. Поясните сущность подходов реплатформинга (replatforming) и реархитектинга (rearchitecting).
8. Укажите сущность и особенности использования моделей IaaS, PaaS, SaaS. В чем состоят их достоинства и недостатки?
9. Поясните схему работы модели «Функция как услуга» (FaaS).
10. Объясните, какими параметрами и свойствами модель FaaS отличается от классической PaaS.
11. Дайте определение мультиоблачной стратегии, а также отметьте ее достоинства и недостатки.

12. Поясните сложности и вызовы, порождаемые гибридными и мультиоблачными средами.

13. Объясните сущность концепции единой плоскости управления и унифицированных операционных моделей.

14. Охарактеризуйте роль вендора в облачной экосистеме.

15. Укажите основные характеристики современного рынка облачных услуг характеризуется выраженной стратификацией

16. Поясните, какие критерии и ключевые группы целесообразно оценивать при выборе провайдера.

17. Дайте определение FinOps и поясните направления внедрения этой парадигмы.

18. Поясните, из каких взаимосвязанных фаз состоит процесс FinOps.

19. Дайте краткую характеристику основным практикам оптимизации затрат на создание и совершенствование цифровой инфраструктуры компании.

20. Поясните тенденции развития моделей развертывания цифровой инфраструктуры на современном этапе их развития.

Практические задания

Задание 1. На основании исходных данных к заданию, проанализируйте и предложите оптимальную модель развертывания информационной инфраструктуры (локальное, облачное или гибридное) и уровня обслуживания (IaaS, PaaS, SaaS).

Исходные данные: Региональный банк, работающий в юрисдикции РФ, планирует цифровую трансформацию.

Ключевые требования:

1) разработка и запуск нового мобильного приложения для розничных клиентов;

2) миграция унаследованной (legacy) системы аналитической отчетности, построенной на специализированном стеке ПО;

3) организация защищенного хранилища для персональных данных и документов, подпадающих под действие 152-ФЗ.

Для каждого направления определите и обоснуйте рекомендуемую модель развертывания и уровень облачной услуги. Результат оформите в виде аналитической таблицы с итоговыми рекомендациями и развернутым пояснением по каждому пункту.

Задание 2. Необходимо модель распределения ответственности для гибридной инфраструктуры условного промышленного предприятия, архитектура которого включает следующие параметры:

1) критическую SCADA-систему, развернутую в локальном дата-центре (частное облако на базе OpenStack);

2) корпоративный портал и систему электронного документооборота, размещенные в публичном облаке (Yandex Cloud) по модели IaaS;

3) систему бизнес-аналитики на основе SaaS-решения от стороннего вендора.

Задание: разработать детализированную таблицу, в которой для каждого из трех компонентов системы определены зоны ответственности поставщика услуги и потребителя (ИТ-службы предприятия). В таблице должны быть отражены следующие аспекты: физическая инфраструктура, виртуализация, операционная система, промежуточное ПО, приложение, данные, идентификация и доступ. Таблицу завершите выводами, описывающими ключевые риски и необходимые меры контроля.

Тест для самоконтроля

1. Что выступает ключевым драйвером выбора модели «on-premise»?

а) Требования к наполнению ИТ-отделов компании.

б) Требования к безопасности и соответствию нормативным актам.

в) Требования к квалификации персонала.

г) Требования к размеру компанию.

2. Как осуществляется управление инфраструктурой в модели «on-premise»?

а) Полностью силами заказчика.

б) Полностью или преимущественно силами заказчика с использованием платформ оркестрации.

в) Физическое обслуживание, мониторинг и обеспечение доступности - зона ответственности провайдера.

г) Полностью силами провайдера.

3. Каким образом осуществляется владение активами в модели «on-premise»?

а) Провайдер владеет всем оборудованием.

б) Провайдер выделяет стойки/серверы.

в) Заказчик владеет всем оборудованием.

г) Физическое обслуживание и обеспечение доступности - зона ответственности провайдера.

4. Эластичность и масштабируемость в публичном облаке…

а) Ограничены мощностью собственной инфраструктуры.

б) Очень высокие.

в) Высокие.

г) Максимальные за счет выбора лучших сервисов.

5. Операционная сложность в гибридном облаке…

а) Высокая, так как требует собственных специалистов

б) Очень высокая.

в) Крайне высокая.

г) Низкая, так как управляется провайдером.

6. Подход, при котором в приложения вносятся минимальные изменения для успешной адаптации к облачной среде – это…

а) Реплатформинг

б) DevOps-практика.

в) Реархитиринг.

г) TCO.

7. Подход, при котором приложение полностью переосмысливается и переписывается под облако называется…

а) TCO;

б) реархитирингом;

в) DevOps-практикой;

г) реплатформингом.

8. Как называется модель развёртывания, при которой облачная инфраструктура предназначена для эксклюзивного использования одной организацией.

а) Гибридное облако.

б) Частное облако.

в) Мультиоблако.

г) Публичное облако.

9. Какая модель представляет собой первый уровень абстракции в облаке?

а) IaaS;

б) On-Premise;

в) PaaS;

г) SaaS.

10. Модель, при котором потребителю предоставляется доступ к готовому прикладному программному обеспечению, работающему в облачной инфраструктуре поставщика.

а) PaaS;

б) IaaS;

в) SaaS;

г) FaaS.

11. Модель, являющаяся эволюционным развитием и специализацией концепций PaaS – это…

а) On-Premise;

б) SaaS;

в) IaaS;

г) FaaS.

12. Мультиоблачная стратегия подразумевает использование услуг…

а) нескольких публичных облачных провайдеров;

б) одного публичного провайдера;

в) только двух публичных облачных провайдеров;

г) не предполагает использование публичных облачных провайдеров.

13. Подготовка инфраструктуры при традиционном подходе предполагает..

а) Ручную работа в консолях каждого облака; скрипты, специфичные для провайдера.

б) Автоматизацию через Infrastructure as Code.

в) Ручную работу в консолях каждого облака; скрипты, неспецифичные для провайдера.

г) Автоматизацию через Infrastructure as Code и скрипты, неспецифичные для провайдера.

14. Безопасность и комплаенс для унифицированных платформ означает...

а) Применение политики как код (Policy as Code), централизованное управление доступом (IAM), сканирование конфигураций.

б) Применение только сканирования конфигураций.

в) «Ручное» применение политик в каждой среде, фрагментированный аудит.

г) Применение фрагментированного аудита.

15. Что представляет собой аварийное восстановление при традиционном подходе представляет?

а) Унифицированные планы, автоматизируемые через IaC и оркестрацию.

б) Возможность миграции workloads между облаками.

в) Сложные, уникальные для каждого сайта и облака процедуры.

г) Простые и унифицированные для каждого сайта и облака процедуры.

16. Фундаментальным отличием облачной модели от традиционной является… .

а) переход от владения капиталоемкими активами к потреблению на основе контракта с внешним провайдером;

б) переход от потребления на основе контракта с внешним провайдером к владению капиталоемкими активами;

в) управление капиталоёмкими ИТ-активами;

г) управление на основе контракта с внутренним провайдером.

17. Как называются компании, обладающие распределенной сетью дата-центров планетарного масштаба и предлагающие максимально широкий и глубокий портфель сервисов.

а) Глобальные гиперскейлеры.

б) Локальные гиперскейлеры.

в) Региональные гиперскейлеры.

г) Местные гиперскейлеры.

18. Парадигма и набор практик, направленных на повышение финансовой ответственности в процессе потребления облачных услуг для достижения оптимального соотношения ценности и затрат – это …

а) FinOps;

б) CapEx;

в) OpEx;

г) On-premise.

19. Сколько фаз содержит процесс FinOps?

а) 3 фазы;

б) 4 фазы;

в) 5 фаз;

г) 6 фаз;

20. Как называется вторая фаза FinOps.

а) Идентификация.

б) Оптимизация.

в) Операции.

г) Информирование.

Глава 5. УПРАВЛЕНИЕ ПРОИЗВОДИТЕЛЬНОСТЬЮ И ДОСТУПНОСТЬЮ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ

5.1. Ключевые метрики производительности

Управление производительностью и доступностью цифровой инфраструктуры требует объективного, количественного подхода к измерению и оценке ее работы. Эффективный менеджмент невозможен без четко определенной системы метрик, которые служат основой для установки целей, анализа состояния и планирования развития систем. В современной практике SRE (Site Reliability Engineering) и ИТ-менеджмента сформировалась иерархическая концепция, связывающая сырые измерения, индикаторы, целевые уровни обслуживания и юридические обязательства. Эта концепция включает в себя такие ключевые элементы, как SLI (Service Level Indicator), SLO (Service Level Objective), SLA (Service Level Agreement).

Кроме того, в основе любого анализа производительности лежат две первичные и взаимосвязанные метрики: латентность (задержка) и пропускная способность. Понимание их сущности, взаимосвязей и методов применения является критически важным для обеспечения надежности и эффективности бизнес-сервисов.

Итак, латентность (Latency) - это мера времени, затрачиваемого на выполнение отдельной операции или доставку пакета данных от источника к получателю. Важно различать различные компоненты латентности:

– Время обработки (Service Time): время, непосредственно затрачиваемое сервером на обработку запроса.

– Время ожидания (Wait Time/Queueing Delay): время, которое запрос проводит в очередях (на сетевом оборудовании, в балансировщике, в очереди приложения или СУБД).

– Сетевая задержка (Network Latency): время распространения сигнала по сетевым магистралям, включая задержки на маршрутизаторах (propagation + transmission + processing delay).

Латентность является векторной величиной и для ее анализа необходимо применять статистические агрегаты: среднее значение, процентили (p50, p95, p99, p99.9), максимальное значение. Для пользовательского восприятия критически важны высокие процентили (p95,

p99), так как они отражают наихудший, но все еще значительный опыт части пользователей.

В свою очередь, пропускная способность (Throughput) - это мера количества операций или объема данных, которые система способна обработать за единицу времени (например, запросов в секунду — RPS/QPS, бит в секунду - bps, транзакций в секунду - TPS). Пропускная способность характеризует производительную мощность системы. Между латентностью и пропускной способностью существует нелинейная зависимость, описываемая, в частности, законом Литтла (Little's Law), математически описываемым выражением (5.1):

$$L = \lambda * W, \quad (5.1)$$

где L – среднее число запросов в системе;

λ – средняя скорость поступления (throughput);

W – среднее время отклика (latency).

При приближении нагрузки к пределу пропускной способности системы латентность начинает нелинейно и резко возрастать.

Связь между латентностью и пропускной способностью в условиях роста нагрузки иллюстрирует табл. 5.1.

Таблица 5.1

Взаимосвязь латентности и пропускной способности при возрастании нагрузки

Уровень нагрузки (относительно предела)	Пропускная способность (Throughput)	Латентность (Latency)	Состояние системы
Низкая (70%)	Растет линейно	Растет нелинейно, но стабильно	Стабильная работа, формирование очередей
Высокая (>85%)	Стабилизируется у предела	Резко и нелинейно растет	Насыщение, длинные очереди
Перегрузка (>100%)	Падает (коллапс)	Стремится к бесконечности, таймауты	Отказ в обслуживании

Service Level Indicator (SLI) — это непосредственно измеряемая количественная характеристика аспекта предоставляемого сервиса. SLI является фундаментальным измерительным инструментом, который превращает сырые данные мониторинга в значимый индикатор. SLI всегда определяется для конкретного сервиса и его критически важного сценария использования.

Типичные примеры SLI:

- Доля успешных HTTP-запросов.
- Латентность запросов, измеряемая как доля запросов, выполненных быстрее заданного порога (например, доля запросов с latency < 300 мс).
- Частота ошибок, выраженная в процентах от общего числа запросов.
- Пропускная способность системы в условиях сохранения целевой латентности.

Важнейший принцип при выборе SLI - его релевантность для потребителя сервиса (пользователя или другой системы). Например, для веб-сервиса латентность следует измерять не на стороне сервера, а от момента отправки запроса браузером до момента получения последнего байта ответа, чтобы учесть всю цепочку.

Service Level Objective, SLO - это целевое значение или диапазон значений для SLI, которое определяется как внутренняя цель команды разработки и эксплуатации. SLO представляет собой формальное, количественное определение того, насколько надежным должен быть сервис в течение определенного периода (обычно 28-30 дней). SLO является ключевым инструментом управления надежностью, так как устанавливает баланс между скоростью разработки (внедрением новых функций) и стабильностью работы. SLO всегда должен быть чуть ниже максимально возможного уровня надежности системы, чтобы создать «бюджет ошибок» (Error Budget), который позволяет командам вносить изменения без страха нарушить абсолютную стабильность. Пример SLO: «Доля успешных HTTP-запросов к API сервиса «X» должна составлять не менее 99.9% за расчетный период в 28 дней».

Service Level Agreement (SLA) - это юридический или формальный договор между поставщиком и потребителем сервиса, который включает в себя одно или несколько SLO. Ключевое отличие SLO от SLA - наличие финансовых или иных юридических последствий (штрафов, компенсаций) в случае невыполнения условий, оговоренных в SLA.

SLA являются внешними документами и, как правило, устанавливаются на менее строгих значениях, чем внутренние SLO, чтобы со-

здать операционный буфер и минимизировать риски штрафных санкций. Например, при внутреннем SLO в 99.9% доступности, SLA для клиентов может быть установлено на уровне 99.5%.

Иерархическая взаимосвязь между техническими метриками, SLI, SLO и SLA представлена в табл. 5.2.

Таблица 5.2

Иерархия метрик производительности и доступности

Уровень	Концепция	Сущность	Пример	Контекст и назначение
Уровень 1: измерения	Latency, Throughput, Error Count	Сырые данные, первичные метрики	Время отклика: 150 мс. Кол-во запросов: 1000 RPS.	Основа для вычисления SLI. Мониторинг состояния узлов и сетей.
Уровень 2: индикатор	SLI (Service Level Indicator)	Нормализованный, пользователь-центричный показатель, рассчитанный на основе измерений.	Доля HTTP-запросов с временем ответа < 300 мс.	Отражает качество обслуживания с точки зрения пользователя. Измеряемый факт.
Уровень 3: цель	SLO (Service Level Objective)	Внутренняя, количественная цель для SLI.	SLI (запросы <300 мс) >= 99% за 28 дней.	Инструмент баланса между инновациями и надежностью. Основа для расчета бюджета ошибок.
Уровень 4: обязательство	SLA (Service Level Agreement)	Формальный договор с последствиями при невыполнении.	Гарантированный уровень доступности API – 99.5% в месяц. При нарушении – возврат 10% от стоимости услуги.	Правовая и финансовая основа отношений с клиентом. Детерминанта для установки внутренних SLO.

Таким образом, управление производительностью цифровой инфраструктуры представляет собой системную деятельность, основанную на строгой иерархии метрик. Фундаментальные технические показатели латентность и пропускная способность позволяют проводить глубокий анализ и тюнинг компонентов инфраструктуры. Их осмысление в контексте пользовательского опыта приводит к формулированию SLI. На основе SLI устанавливаются внутренние цели SLO, которые являются не догмой, а инструментом для принятия взвешенных решений о рисках и темпах развития. Внешние SLA, в свою очередь, фик-

сируют минимальные гарантии для потребителей. Корректное определение, измерение и согласование этих метрик на всех уровнях позволяет перейти от реактивного устранения инцидентов к проактивному, научно обоснованному управлению надежностью и эффективностью цифровых сервисов, что напрямую влияет на удовлетворенность пользователей и финансовые результаты бизнеса.

5.2. Мониторинг, сбор и анализ логов и метрик

Эффективное управление производительностью и доступностью цифровой инфраструктуры невозможно без системного подхода к сбору, хранению и анализу операционных данных. Эти данные, представленные в двух основных формах — метрики (metrics) и логи (logs), образуют основу для принятия информированных инженерных и управленческих решений. Метрики — это числовые измерения, характеризующие состояние системы в конкретный момент времени (например, загрузка процессора, потребление памяти, скорость обработки запросов). Логи — это хронологически упорядоченные текстовые записи о событиях, произошедших в системе или приложении, содержащие контекстную информацию для детального расследования инцидентов. Современным ответом на задачу консолидации и интерпретации этих разнородных данных стали открытые технологические стеки, среди которых доминирующие позиции занимают связки ELK (Elasticsearch, Logstash, Kibana) для работы с логами и Prometheus с Grafana для работы с метриками. Их интеграция формирует целостную систему наблюдаемости (Observability), выходящую за рамки классического мониторинга.

Система Prometheus, ставшая де-факто стандартом для мониторинга в облачных и контейнеризированных средах, архитектурно построена на модели pull-запросов. Её ядром является сервер сбора и хранения временных рядов, который периодически опрашивает (scrapes) заданные конечные точки (endpoints), экспортирующие метрики в специфическом текстовом формате. Данные хранятся в эффективном внутреннем хранилище на диске с использованием алгоритмов сжатия. Важнейшей концепцией Prometheus является многомерная модель данных: каждая метрика идентифицируется именем и набором пар ключ-значение (labels), что позволяет осуществлять гибкую фильтрацию и

агрегацию. Мощный язык запросов PromQL (Prometheus Query Language) предоставляет возможности для выполнения сложных аналитических операций над временными рядами в реальном времени, включая прогнозирование, вычисление перцентилей и многоуровневую агрегацию (табл. 5.3).

Таблица 5.3

Ключевые компоненты экосистемы Prometheus

Компонент	Назначение и функциональные особенности
Prometheus Server	Основной модуль, выполняющий сбор, хранение временных рядов на локальном диске и обработку запросов с использованием PromQL.
Client Libraries	Библиотеки для инструментирования прикладного кода (Java, Go, Python и др.), позволяющие определять и экспортировать пользовательские метрики.
Exporters	Специализированные агенты или службы, преобразующие метрики из форматов, не поддерживаемых Prometheus "из коробки" (например, состояние аппаратной части, метрики ОС, базы данных, сторонних приложений), в понятный для него формат.
Service Discovery	Механизмы автоматического обнаружения целей для сбора метрик в динамических средах (Kubernetes, Consul, AWS EC2), что критически важно для микросервисных архитектур.
Alertmanager	Отдельный компонент, отвечающий за обработку, группировку, подавление и маршрутизацию алертов, сгенерированных сервером Prometheus на основе правил Alerting Rules.

Для визуализации метрик, собранных Prometheus, а также данных из множества других источников (реляционные и NoSQL базы данных, облачные провайдеры, системы ITSM) используется Grafana - открытая платформа для мониторинга и визуализации данных, ориентированная на данные систем ИТ-мониторинга. Её ключевым преимуществом является декларативный подход к созданию интерактивных информационных панелей через веб-интерфейс. Пользователь может комбинировать на одном холсте различные типы графиков (линейные, гистограммы, тепловые карты), таблицы, текстовые аннотации, выбирая данные через встроенные редакторы запросов для каждого подключенного источника. Панели поддерживают использование переменных, что позволяет создавать параметризированные, многократно используемые дашборды, адаптированные под конкретные сервисы или

среду. Grafana выполняет роль единого окна визуализации, агрегируя информацию не только о метриках производительности, но и о событиях, что обеспечивает оператору контекстное понимание поведения системы (рис. 5.1).

Рис. 5.1. Рабочий интерфейс Grafana

Параллельно с миром метрик задача централизованного анализа логов решается с помощью стека ELK (Elasticsearch, Logstash, Kibana), который эволюционировал в более широкую экосистему Elastic. Его ядро – распределенная поисковая и аналитическая машина Elasticsearch, построенная на базе Apache Lucene. Elasticsearch индексирует структурированные и неструктурированные лог-данные, обеспечивая чрезвычайно высокую скорость полнотекстового и контекстного поиска, а также агрегаций. Для приема и преобразования лог-потоков традиционно использовался Logstash - конвейер обработки данных на Java, способный принимать события из множества источников, трансформировать их с помощью фильтров (парсинг, обогащение, дедупликация) и направлять в различные хранилища. В современных высоконагруженных сценариях его роль на этапе сбора часто переходит к более легкому и производительному агенту Beats (например, Filebeat для сбора лог-файлов). Задача сложной трансформации данных при этом делегируется вычислительному узлу Elasticsearch Ingest Node (табл. 5.4).

Таблица 5.4

Функциональное разделение компонентов стека ELK для обработки логов

Этап обработки	Компонент	Описание процесса
Сбор и буферизация	Beats (Filebeat, Metricbeat и др.)	Легковесные агенты, устанавливаемые на источник данных, выполняют сбор лог-файлов или метрик, осуществляют первичную обработку (мультистрочный парсинг, фильтрация) и буферизированную отправку далее по конвейеру. Могут использовать очереди (например, Kafka) для обеспечения надежности.
Трансформация и обогащение	Logstash или Elasticsearch Ingest Pipelines	Logstash выполняет сложные ETL-операции: парсинг неструктурированных логов по Grok-шаблонам, геообогащение по IP-адресам, анонимизацию, объединение событий. Ingest Pipelines предлагают аналогичный, но более легковесный функционал внутри кластера Elasticsearch.
Хранение, поиск и анализ	Elasticsearch	Распределенное хранилище индексирует поступившие структурированные документы. Обеспечивает масштабируемость и отказоустойчивость за счет шардирования и репликации. Предоставляет REST API и язык запросов Query DSL для сложного поиска и агрегаций (выявление статистических закономерностей, построение временных рядов из логов).
Визуализация и исследование	Kibana	Веб-интерфейс для работы с данными в Elasticsearch. Позволяет строить дашборды с графиками, таблицами, картами; выполнять ад-hoc анализ через интерфейс Discover; создавать мониторинговые панели для логов (Logs app) и реализовывать превентивный мониторинг с помощью машинного обучения (Machine Learning features).

Синергия рассмотренных инструментов достигается за счет их интеграции. Grafana может использовать Elasticsearch как источник данных для визуализации логов совместно с метриками из Prometheus

на единой панели, что критически важно для корреляции инцидентов. Для комплексного мониторинга приложений, развернутых в контейнерах, используется подход, при котором Prometheus собирает метрики производительности Docker и Kubernetes, а агенты Filebeat или Fluentd собирают журналы работы контейнеров, направляя их в Elasticsearch. [27]

Настройка алертинга в такой интегрированной системе также становится двухуровневой: Prometheus Alertmanager генерирует алерты на основе нарушений числовых порогов (например, рост 95-го перцентиля времени ответа), в то время как в Kibana или Elasticsearch можно настроить правила для выявления аномалий в логических паттернах (например, учащение ошибок определенного класса или появление сообщений об атаках в журналах безопасности).

Таким образом, совместное применение стека ELK для анализа логов и связки Prometheus/Grafana для работы с метриками формирует технологический фундамент управления производительностью и доступностью. Это позволяет инженерным командам перейти от реактивного устранения инцидентов к проактивному управлению системами на основе данных, прогнозированию трендов, оптимизации использования ресурсов и обеспечению соблюдения соглашений об уровне обслуживания (SLA). Постоянное развитие данных платформ, их адаптация к работе в гибридных и мультиоблачных средах подтверждают их статус неотъемлемых элементов жизненного цикла современной цифровой инфраструктуры.

5.3. Построение отказоустойчивых систем и планов обеспечения непрерывности бизнеса

В контексте управления производительностью и доступностью цифровой инфраструктуры, достижение высокой доступности (High Availability, HA) является необходимым, но не всегда достаточным условием для обеспечения долгосрочной устойчивости бизнес-процессов. Экстремальные события, такие как масштабные стихийные бед-

[27] Какие технологии помогают бизнесу построить единую ИТ-инфраструктуру [Электронный ресурс] // Режим доступа: https://digtlab.ru/tpost/rzfhfyfrr1-kakie-tehnologii-pomogayut-biznesu-postr?ysclid=mk2z328jiu499207780 (дата обращения: 06.01.2026).

ствия, кибератаки, техногенные катастрофы или длительные общесистемные сбои, могут вывести из строя даже избыточные системы в пределах первичного центра обработки данных (ЦОД). Поэтому целостная стратегия управления доступностью должна интегрировать два взаимодополняющих направления: построение отказоустойчивых систем, минимизирующих вероятность сбоя, и разработку комплексных планов обеспечения непрерывности бизнеса (Business Continuity Planning, BCP) и аварийного восстановления (Disaster Recovery, DR), направленных на минимизацию последствий реализовавшихся катастрофических событий.

Отказоустойчивость (Resilience) цифровой инфраструктуры - это ее свойство противодействовать сбоям, поглощать их воздействие и сохранять или оперативно восстанавливать ключевые функции в условиях деградации компонентов, внешнего давления или атак. Это свойство достигается не за счет отдельной технологии, а через системную архитектурную философию, основанную на принципах избыточности, разнообразия, автоматизации и декомпозиции.

Ключевые архитектурные паттерны включают: активный-активный и активный-пассивный кластеризации критических приложений и баз данных; георазпределение узлов инфраструктуры для исключения единой точки отказа (SPOF) на уровне площадки; использование механизмов автоматического переключения на основе постоянного мониторинга здоровья компонентов; сегментацию сети (микросегментацию) для сдерживания последствий инцидентов безопасности; а также проектирование stateless-сервисов, которые не сохраняет информацию о предыдущих состояниях или сеансах, упрощающая при этом их перезапуск и репликацию. При этом проектирование отказоустойчивости должно носить экономически обоснованный характер, где стоимость реализации механизмов резервирования соотносится с вероятностью и потенциальным ущербом от простоя.

Однако архитектурная отказоустойчивость имеет практические и экономические пределы. Целью планов обеспечения непрерывности бизнеса (BCP) и аварийного восстановления (DR) является формализация действий, необходимых для защиты критических бизнес-процессов организации от воздействия значительных разрушительных событий и обеспечения их восстановления в приемлемые сроки. BCP имеет

более широкий охват, фокусируясь на поддержании операционной деятельности компании в целом (включая персонал, связь, цепочки поставок), в то время как DR является его подмножеством, сконцентрированным исключительно на восстановлении ИТ-инфраструктуры, данных и приложений после катастрофы. Управление доступностью в этом ракурсе трансформируется из задачи чистой инженерии в управленческую дисциплину, основанную на оценке рисков и анализе воздействия на бизнес (Business Impact Analysis, BIA).

BIA является краеугольным камнем для построения эффективных DR/BCP-планов. В его рамках проводится идентификация и приоритизация бизнес-процессов, определяются критические ИТ-системы и данные, их поддерживающие, а также устанавливаются целевые метрики восстановления. Двумя ключевыми метриками являются:

- Целевое время восстановления (Recovery Time Objective, RTO) – максимально допустимая длительность простоя системы или бизнес-процесса после инцидента.

- Целевая точка восстановления (Recovery Point Objective, RPO) – максимальный объем данных (измеряемый во времени), допустимый к потере в результате инцидента.

Значения RTO и RTO для каждого критического актива напрямую определяют технологический и организационный дизайн решений по аварийному восстановлению. Строгие значения (близкие к нулю) требуют значительных инвестиций в полностью синхронную репликацию данных и инфраструктуру «горячего» резерва в географически удаленном ЦОД, в то время как более мягкие значения допускают использование асинхронной репликации или даже восстановления из резервных копий.

Современные подходы к DR/BCP эволюционируют от традиционных моделей, ориентированных на физическое восстановление оборудования в резервном ЦОД, в сторону облачно-ориентированных стратегий. Использование публичных, частных или гибридных облаков предлагает гибкость и масштабируемость для размещения резервных мощностей, реализацию модели DRaaS (Disaster Recovery as a Service) и возможность быстрого развертывания инфраструктуры «как код» (Infrastructure as Code, IaC). Это позволяет существенно сократить RTO по сравнению с ручными процедурами.

Важнейшим аспектом является не разработка, а постоянное поддержание актуальности и проверка работоспособности планов DR/BCP. Регулярное тестирование в различных форматах – от настольных упражнений до полномасштабных отработок переключения на резервный сайт — выявляет пробелы в документации, технические несовместимости и проблемы координации между командами. Таким образом, отказоустойчивость и BCP/DR формируют непрерывный цикл совершенствования: практические уроки, извлеченные из тестов и реальных инцидентов, должны напрямую влиять на архитектурные решения и обновление процедур восстановления.

Сравнительная характеристика уровней стратегий аварийного восстановления в зависимости от целевых показателей RTO и RPO представлена ниже в табл. 5.5.

Таблица 5.5

Уровни стратегий аварийного восстановления ИТ-систем

Уровень стратегии	Характерные RTO/RPO	Технологический подход	Преимущества	Недостатки и ограничения
Резервное копирование и восстановление (Backup & Restore)	RTO: часы – дни; RPO: 24 часа и более	Периодическое создание резервных копий на съемных носителях или в облако. Восстановление выполняется вручную после поставки оборудования.	Низкая стоимость, простота реализации.	Длительное время простоя, высокий риск потери данных, значительные трудозатраты, сложность проверки целостности.
Резервный ЦОД («холодный»/«теплый»)	RTO: часы; RPO: от нескольких часов до минут	Наличие подготовленной площадки с базовой инфраструктурой. Восстановление из резервных копий или асинхронной репликации данных.	Более предсказуемое RTO по сравнению с первым уровнем, относительно умеренная стоимость.	Требует времени на активацию и конфигурирование систем, возможно расхождение данных (при асинхронной репликации).
Синхронная репликация с «теплым» резервом	RTO: десятки минут – час; RPO ≈ 0	Синхронная репликация данных между основным и резервным ЦОД. Резервные серверы находятся в готовности к запуску.	Минимальная потеря данных (RPO≈0), ускоренное восстановление.	Высокая стоимость из-за необходимости полной инфраструктуры и каналов связи с низкой задержкой для синхронной репликации.

Уровень стратегии	Характерные RTO/RPO	Технологический подход	Преимущества	Недостатки и ограничения
Кластерное решение «активный-активный» между ЦОД	RTO: близко к 0; RPO = 0	Георасределенный кластер, где нагрузка балансируется между двумя и более активными центрами. Данные реплицируются синхронно.	Практически непрерывная доступность, автоматическое переключение без потери данных и транзакций.	Наибольшая сложность и стоимость реализации, требования к производительности и задержкам в сети, сложность поддержки.
Облачно-ориентированное восстановление (DRaaS)	RTO: минуты – часы; RPO: минуты – часы	Использование облачной платформы как резервной среды. Репликация данных в облако, автоматическое развертывание инфраструктуры по сценарию.	Гибкость, масштабируемость, операционная модель «как услуга», снижение капитальных затрат (CAPEX).	Зависимость от провайдера и каналов связи, потенциальные затраты на исходящий трафик (egress fees), вопросы соответствия нормативным требованиям.

В итоге, управление производительностью и доступностью цифровой инфраструктуры на системном уровне требует двойного фокуса. Во-первых, это проактивное инженерное проектирование отказоустойчивости для обработки сбоев на уровне компонентов и узлов, что напрямую поддерживает показатели доступности в рамках SLA. Во-вторых, это стратегическое планирование устойчивости на уровне бизнеса через формализованные процессы BCP/DR, которые признают возможность катастрофических событий и обеспечивают восстановление в рамках согласованных с бизнесом параметров RTO и RPO. Интеграция этих двух направлений в единый цикл управления, подкрепленный регулярным аудитом и тестированием, формирует основу для истинной надежности и конкурентной устойчивости современной организации в цифровую эпоху.

Операционные процессы обеспечения доступности и восстановления заключаются в реализации следующего алгоритма и его ключевых направлений.

1. Концептуальные основы. Обеспечение высокой доступности (High Availability, HA) и эффективное восстановление цифровой инфраструктуры после сбоев представляют собой не набор разрозненных

технических мер, а целостную систему строго формализованных операционных процессов. Эти процессы образуют циклическую модель управления, в которой предиктивная аналитика, проактивные действия, реактивное восстановление и пост-инцидентный анализ неразрывно связаны. Управление доступностью направлено на максимизацию времени предоставления регламентированного уровня сервиса (Service Level Agreement, SLA), в то время как процессы восстановления (Recovery) фокусируются на минимизации времени недоступности (Downtime) и объема потерь данных (Recovery Point Objective, RPO) при наступлении инцидента. Совокупность этих процессов является критическим элементом управления производительностью, так как каждый инцидент, снижающий доступность, напрямую нарушает нормальное функционирование системы и ведет к потерям в производительности бизнес-процессов.

2. Проактивные операционные процессы обеспечения доступности, которые нацелены на предотвращение инцидентов и создание отказоустойчивой среды. Их основу составляет архитектурная и операционная избыточность, на практике обеспечиваемая реализацией следующих процессов:

- Процесс проектирования и поддержания отказоустойчивой архитектуры. Он непрерывен и начинается на этапе проектирования инфраструктуры. Он включает в себя внедрение принципов N+1 или 2N резервирования для критических компонентов (электропитание, охлаждение, сетевое оборудование, серверные узлы), использование распределенных и кластеризованных систем, балансировку нагрузки. Ключевым подпроцессом является регулярная проверка и актуализация схем резервирования в соответствии с эволюцией инфраструктуры и бизнес-требований.

- Процесс мониторинга и предиктивной аналитики представляет собой пассивный сбор метрик (CPU, память, диск) трансформируется в активный процесс анализа трендов. Использование систем мониторинга, основанных на машинном обучении, позволяет выявлять аномалии в поведении систем, прогнозировать исчерпание ресурсов и потенциальные сбои до их фактического возникновения. На основе этого процесса формируются упреждающие уведомления (proactive alerts), инициирующие корректирующие действия.

- Процесс управления изменениями и конфигурациями (Change Management). Статистически значимая доля инцидентов вызвана некорректными изменениями. Формализованный процесс управления изменениями, включающий стадии запроса, планирования, тестирования, утверждения, реализации и пост-релизного контроля, минимизирует риски, вносимые в стабильную среду. Автоматизированное управление конфигурациями (Configuration Management) обеспечивает консистентность и документированность состояния всех компонентов инфраструктуры, что является основой для быстрого восстановления.

- Процесс проведения регламентных работ и тестирования отказоустойчивости. Плановое техническое обслуживание (патчинг, обновления) должно выполняться по формализованному графику без прерывания сервиса, что требует процессов «горячего» обновления или switchover. Краеугольным камнем является процесс регулярного тестирования отказоустойчивости, включающий в себя плановое, контролируемое выведение из строя компонентов (Chaos Engineering) для верификации реальной работоспособности резервных решений и корректности процедур восстановления.

3. Реактивные операционные процессы восстановления

Когда инцидент предотвратить не удалось, в действие вступают реактивные процессы, структура и скорость выполнения которых напрямую определяют ключевые метрики RTO и RPO, а именно:

- Процесс детектирования и классификации инцидента. Автоматизированные системы мониторинга и оповещения (Alerting) должны быть настроены на генерацию инцидентов с четкой классификацией по степени критичности (Severity Level: Critical, Major, Minor и т.д.). Каждому уровню соответствует свой процесс эскалации, регламентирующий временные рамки реакции и круг ответственных лиц или команд. Отсутствие формализованной классификации ведет к хаосу и потерям времени.

- Процесс локализации и диагностики. Целью является точное определение корневой причины (Root Cause) и границ воздействия инцидента. Этот процесс опирается на централизованное логирование (ELK-стек, аналоги), сбор и корреляцию событий (SIEM-системы), трассировку распределенных транзакций (Distributed Tracing). Использование runbooks – детализированных пошаговых инструкций для диагностики типовых проблем – ускоряет этот этап.

- Процесс устранения и восстановления работоспособности. На этом этапе применяются заранее разработанные и оттестированные сценарии. В зависимости от ситуации процесс может включать: автоматическое или ручное переподключение на резервный узел/центр обработки данных; откат последних изменений конфигурации или кода; перезапуск отказавших служб или виртуальных машин; изоляцию проблемного сегмента сети или оборудования. Приоритетом здесь является восстановление сервиса, даже временными средствами, с последующим поиском окончательного решения.

- Процесс восстановления данных (Data Recovery). Если инцидент привел к потере или повреждению данных, запускается отдельный процесс восстановления из резервных копий. Его эффективность определяется регулярностью и надежностью процесса резервного копирования и его валидации. Критически важным является не просто факт создания бэкапа, а регулярное тестирование процедуры его восстановления на изолированном стенде для гарантии достижения целевых показателей RPO и RTO.

4. Процессы непрерывного улучшения (Post-Incident Review)

Цикл управления завершается процессами анализа и интеграции полученного опыта, что отличает зрелую операционную модель, а именно:

- Процесс анализа инцидентов (Postmortem / Blameless Retrospective). После стабилизации ситуации проводится формализованный разбор инцидента с фокусом на системные причины, а не на поиск виновных.

Целью является ответ на вопросы:

1. «Что произошло?»

2. «Почему это произошло?»

3. «Какие действия мы предпримем, чтобы это не повторилось?».

Результатом является документ, содержащий хронологию, первопричины, влияние и, главное, план корректирующих действий.

- Процесс обновления операционной документации. На основе выводов анализа инцидента обязательным шагом является актуализация всех связанных процедур: инструкций по мониторингу, диагностике, восстановлению, а также внесение изменений в архитектурную документацию.

- Процесс уточнения метрик и требований SLA/SLO: Статистика по инцидентам, реально достигнутые показатели времени восстановления (MTTR) и доступности анализируются на соответствие целевым значениям (SLO). Это может привести к пересмотру требований к инфраструктуре, выделению дополнительных ресурсов или корректировке ожиданий бизнеса.

5. Интеграция процессов и матрица ответственности

Для успешной реализации описанных процессов необходима четкая организационная структура и интеграция с общепринятыми фреймворками, такими как ITIL.

Взаимосвязь процессов и распределение зон ответственности между командами (например, Service Desk, DevOps/SRE, инженеры по хранению данных, сетевым инженерам) может быть отображена в матрице RACI (табл. 5.6).

Таблица 5.6

Связь операционных процессов с ключевыми метриками доступности и восстановления

Категория процессов	Конкретный процесс	Основные используемые метрики (KPI)	Влияние на целевые показатели
Проактивные	Мониторинг и предиктивная аналитика	Среднее время наработки на отказ (MTBF), уровень утилизации ресурсов, тренды аномалий	Увеличивает MTBF, предотвращает инциденты, поддерживает SLA
Проактивные	Тестирование отказоустойчивости	Процент успешных автоматических тестов, время переключения	Прямо улучшает RTO, подтверждает архитектурную надежность
Проактивные/ Реактивные	Управление резервными копиями	Полнота бэкапов, время создания бэкапа, успешность тестового восстановления	Определяет достижимость RPO и RTO для данных
Реактивные	Детектирование и реагирование	Среднее время обнаружения (MTTD), среднее время реагирования	Сокращает начальную задержку, влияющую на общее MTTR
Реактивные	Восстановление работоспособности	Среднее время восстановления (MTTR), процент инцидентов, решенных с первого раза	Непосредственно определяет RTO и соблюдение SLA

Категория процессов	Конкретный процесс	Основные используемые метрики (KPI)	Влияние на целевые показатели
Улучшающие	Анализ инцидентов	Количество реализованных корректирующих действий по итогам отладки, частота повторения аналогичных инцидентов	Снижает частоту инцидентов (MTBF), системно улучшает все предыдущие процессы

Таким образом, управление доступностью и восстановлением цифровой инфраструктуры представляет собой сложную, но строго формализуемую систему взаимосвязанных операционных процессов. От проактивного проектирования и мониторинга до реактивного восстановления и последующего глубокого анализа — каждый этап должен быть документирован, автоматизирован там, где это возможно, и постоянно совершенствоваться на основе данных. Интеграция этих процессов в общую систему управления производительностью инфраструктуры позволяет перейти от ситуативного «тушения пожаров» к прогнозируемому, управляемому и надежному предоставлению цифровых сервисов, что является фундаментальным требованием для современного бизнеса. Эффективность всей системы количественно оценивается через достижение согласованных уровней сервиса (SLA) и непрерывное улучшение ключевых метрик, таких как MTBF, MTTR, RTO и RPO.

5.4. Использование передовых технологий обеспечения доступности и восстановления цифровой инфраструктуры

В контексте управления производительностью и доступностью цифровой инфраструктуры, задача обеспечения её бесперебойного функционирования и оперативного восстановления после сбоев является критической. Современный подход к решению данной задачи основывается на интеграции передовых технологий, которые трансформируют традиционные реактивные модели в проактивные и адаптивные системы. Эти технологии позволяют не только минимизировать время простоя (MTTR — Mean Time To Recovery), но и предсказывать потенциальные отказы, предотвращая их воздействие на бизнес-процессы.

Фундаментальной концепцией, лежащей в основе современных систем обеспечения доступности, является отказоустойчивость. В отличие от простой надежности, ориентированной на предотвращение сбоев, отказоустойчивость делает акцент на способности системы поглощать воздействия, адаптироваться к изменениям и быстро восстанавливать работоспособность. Реализация этой концепции невозможна без комплексной автоматизации и оркестрации процессов восстановления. Платформы оркестрации, такие как Kubernetes для контейнеризированных сред или специализированные решения для Disaster Recovery (DR), позволяют декларативно описывать желаемое состояние инфраструктуры и автоматически приводить систему к этому состоянию при отклонениях, реализуя принципы «Инфраструктура как код» (IaC) и «GitOps».

Ключевую роль в прогнозировании сбоев играют технологии искусственного интеллекта и машинного обучения, применяемые в рамках платформ AIOps (Artificial Intelligence for IT Operations). Эти системы анализируют огромные объемы телеметрических данных (метрики, логи, трассировки) в реальном времени, выявляя скрытые аномалии и корреляции, неочевидные для человеческого оператора. Модели машинного обучения способны предсказывать исчерпание ресурсов, деградацию оборудования или аномальное поведение приложений, инициируя превентивные действия до возникновения инцидента, что напрямую повышает доступность сервисов.

Технологии неизменяемой инфраструктуры и контейнеризации значительным образом меняют подход к восстановлению. Вместо трудоемкого устранения неисправностей на работающей системе происходит её полная замена на заранее подготовленный и протестированный образ. Этот подход, часто реализуемый в связке с системами непрерывной интеграции и доставки (CI/CD), гарантирует идентичность развертываемых экземпляров, устраняя дрейф конфигураций и значительно ускоряя процесс развертывания и отката.

Для обеспечения доступности на уровне данных и приложений критически важны передовые решения в области резервного копирования и аварийного восстановления. Современные системы переходят от периодического полного копирования к непрерывной защите данных (CDP – Continuous Data Protection), позволяющей восстановить инфор-

мацию на момент, непосредственно предшествующий сбою, с минимальной потерей данных (RPO — Recovery Point Objective близкий к нулю). Георасределенные и гибридные архитектуры, включая мультиоблачные стратегии, позволяют создавать активные-активные или активные-пассивные кластеры, распределенные между географически разнесенными дата-центрами, обеспечивая доступность даже при выходе из строя целого региона.

Важным аспектом является использование технологий наблюдаемости (Observability), выходящих за рамки классического мониторинга. Инструменты, агрегирующие метрики, логи и распределенные трассировки, обеспечивают сквозную видимость работы всех компонентов сложных распределенных систем (микросервисов). Это позволяет не только быстро детектировать, но и точно локализовать корневую причину инцидента, что является необходимым условием для эффективного восстановления.

Направления взаимодействия ключевых технологий в контексте управления доступностью представлены ниже в виде табл. 5.7.

Таблица 5.7

Вклад передовых технологий в ключевые аспекты обеспечения доступности и восстановления

Технология / Подход	Основная функция в обеспечении доступности	Влияние на метрики доступности
Оркестрация и автоматизация (Kubernetes, Ansible, Terraform)	Автоматическое развертывание, масштабирование, самовосстановление компонентов инфраструктуры.	Сокращение MTTR за счет автоматизации реагирования; повышение общей стабильности.
AIOps и предикативная аналитика	Анализ и выявление аномалий, прогнозирование инцидентов, автоматическое определение корневых причин.	Увеличение MTBF за счет превентивных действий; снижение количества инцидентов.
Неизменяемая инфраструктура и контейнеры	Устранение дрейфа конфигураций, гарантированная идентичность окружений, быстрый откат и развертывание.	Сокращение времени на восстановление; упрощение процедур DR за счет использования готовых образов.
Непрерывная защита данных (CDP) и репликация	Минимальная потеря данных при сбое (RPO → 0), быстрое переключение на резервный сайт.	Минимизация потерь данных (RPO); сокращение времени восстановления.

Технология / Подход	Основная функция в обеспечении доступности	Влияние на метрики доступности
Сквозная наблюдаемость (Observability)	Глубокая видимость во все компоненты системы, быстрое понимание состояния и зависимостей.	Сокращение времени на диагностику и локализацию проблемы (MTTD), что снижает общий MTTR.
Георазпределенные и гибридные архитектуры	Обеспечение доступности сервиса на уровне приложения и данных при отказе целого дата-центра или облачного региона.	Повышение общего времени доступности, приближение к целям уровня обслуживания (SLA) в 99,99% и выше.

Внедрение данных технологий требует системного подхода и трансформации не только технических процессов, но и организационной культуры в сторону DevOps и SRE (Site Reliability Engineering). Принципы SRE, такие как управление ошибками (Error Budget), постулируют баланс между инновациями и стабильностью, формализуя допустимый уровень недоступности, что позволяет объективно оценивать эффективность принимаемых мер.

Эволюция технологий обеспечения доступности движется в сторону создания полностью автономных (самоцелевых) систем (Autonomous Systems), способных самостоятельно прогнозировать, адаптироваться и восстанавливаться без вмешательства человека. Однако достижение этой цели сопряжено с вызовами, включая возрастающую сложность управления самими алгоритмами ИИ, вопросы безопасности автоматизированных систем, а также этические и нормативные аспекты принятия решений автономными агентами.

Таким образом, использование передовых технологий формирует новую парадигму управления доступностью цифровой инфраструктуры, в которой упор смещается с реагирования на инциденты к их предотвращению, а восстановление становится быстрым, детерминированным и максимально автоматизированным процессом. Это позволяет организациям обеспечивать соответствие жестким требованиям к доступности в условиях роста сложности и динамичности современных ИТ-ландшафтов.

5.5. Экономические аспекты и аудит доступности и восстановления цифровой инфраструктуры

Управление доступностью и восстановлением цифровой инфраструктуры (ЦИ) неразрывно связано с комплексом экономических решений, определяющих как текущую операционную эффективность, так и долгосрочную устойчивость организации. Экономический анализ в данной сфере выходит за рамки простого учета затрат на оборудование и программное обеспечение, трансформируясь в стратегическое управление стоимостью владения (Total Cost of Ownership, TCO) и стоимостью бизнес-простоя (Cost of Downtime, CoD). Эти две категории формируют контур экономической целесообразности инвестиций в отказоустойчивость, резервирование и системы аварийного восстановления (Disaster Recovery, DR).

Стоимость владения цифровой инфраструктурой высокой доступности включает капитальные (CAPEX) и операционные (OPEX) расходы, распределенные по всему жизненному циклу. К CAPEX относятся затраты на приобретение серверного и сетевого оборудования повышенной надежности, системы хранения данных с функциями репликации (SAN/NAS), лицензии на кластерное программное обеспечение, резервные центры обработки данных (ЦОД). OPEX составляют расходы на энергопотребление и охлаждение, часто возрастающие из-за дублирования систем, оплату высококвалифицированного персонала для поддержки сложных инфраструктур, ежегодные лицензионные отчисления, затраты на тестирование планов восстановления и регулярный аудит. При этом экономически неоправданное стремление к максимально возможным показателям доступности (например, 99,999% против 99,9%) может привести к экспоненциальному росту TCO, непропорциональному реальным потребностям бизнеса.

В противовес TCO, стоимость простоя является метрикой потенциальных убытков. CoD - комплексная величина, включающая прямые и косвенные потери. Прямые потери: упущенная выручка от транзакций, штрафы за неисполнение контрактных обязательств и SLA, затраты на ликвидацию инцидента. Косвенные потери: репутационный ущерб, снижение лояльности клиентов, отток капитала, падение курса акций публичных компаний. Расчет CoD является фундаментом для обоснования инвестиций в системы восстановления.

Аудит доступности и восстановления ЦИ представляет собой систематическую, независимую и документированную процедуру оценки способности инфраструктуры обеспечивать требуемый уровень сервиса в условиях сбоев и катастроф. Его цель — верификация соответствия фактического состояния инфраструктуры, процессов и документации установленным внутренним политикам, отраслевым стандартам (ISO 22301, ISO 27031) и внешним регуляторным требованиям. Экономическая роль аудита заключается в минимизации финансовых рисков, обеспечении контроля за эффективностью инвестиций и предотвращении скрытых издержек, связанных с невыявленными уязвимостями.

Содержательно аудит охватывает несколько ключевых областей.

1. Аудит стратегии и политик. Проверяется наличие и адекватность формализованной политики доступности и аварийного восстановления, ее согласованность с бизнес-целями организации. Анализируются результаты оценки бизнес-воздействия (Business Impact Analysis, BIA), на основе которых определены критические приложения, целевые показатели времени восстановления (RTO) и точки восстановления (RPO) для каждого бизнес-процесса. Аудитор оценивает, как экономические параметры (приемлемый уровень потерь) трансформированы в технические требования.

2. Аудит архитектуры и инфраструктуры. Исследуется техническая реализация принципов отказоустойчивости: дублирование критических компонентов (N+1, 2N), корректность настройки кластерных решений, наличие и актуальность систем резервного копирования, включая географически распределенные, надежность систем бесперебойного питания и автоматического пожаротушения. Проводится анализ соответствия фактической конфигурации заявленным RTO/RPO.

3. Аудит процессов и документации. Проверяется полнота и релевантность операционных процедур, планов аварийного восстановления (DRP) и инструкций по эскалации инцидентов. Особое внимание уделяется процессу управления изменениями (Change Management), так как неконтролируемые изменения — одна из основных причин снижения доступности. Аудируется регулярность и методика тестирования планов восстановления, анализ результатов тестов и выполнение корректирующих действий.

4. Аудит организационной структуры и компетенций. Оценивается четкость распределения ролей и ответственности в рамках процессов управления инцидентами и восстановления, наличие подготовленных сотрудников (включая дублирование ключевых компетенций), эффективность программ обучения и повышения осведомленности.

Экономическая эффективность аудита проявляется в его превентивной функции. Своевременное выявление «узких мест» позволяет перераспределить инвестиции на наиболее уязвимые участки инфраструктуры, избегая избыточных затрат на избыточное резервирование ненекритичных систем. Аудит также снижает риски финансовых санкций со стороны регуляторов и контрагентов за несоблюдение гарантированных уровней доступности.

Для формализации оценки часто применяются сводные таблицы, позволяющие сопоставить экономические параметры с техническими решениями (табл. 5.8).

Таблица 5.8

Соответствие уровней доступности экономическим и техническим требованиям

Уровень доступности (%)	Допустимое время простоя в год	Бизнес-критичность	Примерная модель развертывания инфраструктуры	Прогнозируемое влияние на ТСО
99 (две девятки)	87 часов 36 минут	Низкая	Стандартное оборудование, локальные бэкапы	Базовое
99.9 (три девятки)	8 часов 46 минут	Средняя	Локальное резервирование (N+1), горячее резервирование СХД	Умеренный рост
99.99 (четыре девятки)	52 минуты 36 секунд	Высокая	Распределенные кластеры, автоматическое переключение, geo-бэкапы	Значительный рост
99.999 (пять девяток)	5 минут 15 секунд	Критичная	Активно-активный ЦОД, синхронная репликация данных, полная избыточность всех путей	Экспоненциальный рост

В заключение, управление экономическими аспектами доступности и восстановления представляет собой непрерывный итеративный процесс балансировки между затратами на предотвращение простоя и потенциальными убытками от его реализации. Систематический аудит выступает в роли механизма обратной связи, обеспечивающего объективную проверку этой балансировки. Только через интеграцию строгого экономического анализа, основанного на точных метриках TCO и CoD, и регулярного всестороннего аудита организация может построить цифровую инфраструктуру, которая является не только технически надежной, но и экономически оптимальной, способной поддерживать непрерывность бизнеса в условиях растущих киберугроз и технологических сбоев.

Таким образом, производительность и доступность цифровой инфраструктуры необходимо рассматривать не как совокупность изолированных технических параметров, а как комплекс взаимосвязанных атрибутов, определяющих качество предоставления цифрового сервиса в целом. Достижение целевых показателей в этой области базируется на строгой иерархии метрик — от первичных измерений латентности и пропускной способности до индикаторов уровня обслуживания (SLI), внутренних целей (SLO) и формальных соглашений (SLA). Эта иерархия устанавливает четкую связь между технической реализацией и бизнес-ценностью, переводя инженерные задачи в плоскость управленческих решений.

Технологическим фундаментом управления выступают комплексные системы сбора и анализа операционных данных, объединенные в концепцию наблюдаемости. Симбиоз специализированных стеков для работы с метриками (Prometheus, Grafana) и логами (ELK/Elastic) формирует единое информационное пространство, необходимое для проактивного мониторинга, оперативной диагностики и глубокого пост-инцидентного анализа. Эффективность использования этих инструментов напрямую зависит от зрелости операционных процессов, среди которых критически важным является четкое организационное и методологическое разделение управления инцидентами и управления проблемами. Первое обеспечивает оперативное восстановление, второе — системное устранение коренных причин, что в совокупности создает цикл непрерывного улучшения.

Архитектурные принципы проектирования высокодоступных и производительных систем, такие как отказоустойчивость, горизонтальное масштабирование, георазпределение и декомпозиция, получают практическое воплощение через внедрение передовых технологий. Контейнеризация, оркестрация, инфраструктура как код, неизменяемые развертывания и стратегии на основе искусственного интеллекта (AIOps) трансформируют реактивную парадигму восстановления в проактивную и адаптивную модель. Это позволяет автоматизировать реакции на сбои и прогнозировать инциденты, минимизируя человеческий фактор и время простоя.

Системный подход завершается интеграцией технических решений в рамки стратегического управления непрерывностью бизнеса и аварийным восстановлением (BCP/DR). Установление целевых показателей RTO и RPO на основе анализа воздействия на бизнес (BIA) обеспечивает экономическое обоснование инвестиций в избыточность и резервирование. Экономический аспект является определяющим, так как он задает баланс между стоимостью владения инфраструктурой высокой доступности и потенциальными убытками от ее простоя. Регулярный аудит выступает механизмом верификации этого баланса, обеспечивая соответствие архитектуры, процессов и документации установленным требованиям и выявляя области для оптимизации.

Таким образом, управление производительностью и доступностью представляет собой целостную дисциплину, синтезирующую инженерные практики, процессный менеджмент, экономический анализ и стратегическое планирование. Успех в этой области достигается не за счет единичных технологических решений, а через создание сбалансированной и эволюционирующей системы, в которой метрики, процессы, архитектура и экономика согласованы для достижения единой цели — гарантированного предоставления цифровых сервисов, отвечающих потребностям бизнеса и пользователей в условиях постоянной изменчивости и роста сложности ИТ-ландшафта.

Вопросы для обсуждения

1. Дайте определение латентности (задержки) и перечислите ее основные компоненты. В чем состоит ее учет в качестве меры производительности цифровой инфраструктуры.

2. Представьте определение пропускной способности и ее основные характеристики.

3. Объясните закон Литтла, показывающий связь между латентностью и пропускной способностью. Какие компоненты входят в математическое выражение, описывающего данный закон?

4. Представьте основные характеристики инструментов Service Level Indicator (SLI), Service Level Objective (SLO) и Service Level Agreement (SLA).

5. Объясните иерархию метрик производительности и доступности.

6. Дайте определение метрикам и логам при оценке эффективности работы цифровой инфраструктуры компании.

7. Перечислите ключевые компоненты экосистемы Prometheus. В чем состоят их назначение и функциональные особенности?

8. Укажите основное предназначение и функциональные возможности открытой платформы Grafana.

9. Дайте определение отказоустойчивости и доступности цифровой инфраструктуры компании.

10. Объясните сущность использования ключевых метрик подхода Business Impact Analysis (BIA)

11. Перечислите уровни стратегий аварийного восстановления ИТ-систем.

12. Поясните, в реализации какого алгоритма участвуют операционные процессы доступности и восстановления.

13. Объясните, связь операционных процессов с ключевыми метриками доступности и восстановления.

14. Поясните вклад передовых технологий в ключевые аспекты обеспечения доступности и восстановления цифровой инфраструктуры.

15. Охарактеризуйте направления движения эволюции обеспечения доступности цифровой инфраструктуры.

16. Объясните экономические особенности управления доступностью и восстановлением цифровой инфраструктуры.

17. Перечислите, какие категории формируют контур экономической целесообразности инвестиций в отказоустойчивость, резервирование и системы аварийного восстановления.

18. Поясните, что представляет собой аудит доступности и восстановления цифровой инфраструктуры компании.

19. Охарактеризуйте основную цель проведения аудита доступности и восстановления цифровой инфраструктуры компании. Какие ключевые области он охватывает?

20. Укажите направления, по которым происходит соответствие уровней доступности экономическим и техническим требованиям.

Практические задания

Задание 1. Разработайте систему метрик для оценки качества предоставления корпоративного сервиса ведения проектов (Project Management Office) промышленного предприятия, отраслевую принадлежность студент выбирает самостоятельно. В ходе выполнения задания необходимо:

1. Определить перечень критически значимых с точки зрения потребителя (пользователя) сценариев использования сервиса (не менее трех).

2. Для каждого сценария сформулировать один–два количественных показателя уровня обслуживания (Service Level Indicator, SLI), обосновав выбор измеряемой характеристики и метода её сбора.

3. Установить для каждого SLI целевые значения уровня обслуживания (Service Level Objective, SLO) на расчетный период 30 дней с обоснованием выбранных пороговых величин.

4. Предложить проект соглашения об уровне обслуживания (Service Level Agreement, SLA), включающий не менее двух гарантируемых метрик, описание метода их измерения, периодичность расчета, а также финансовые или иные санкции за нарушение установленных обязательств.

5. Оценить бюджет ошибок (Error Budget) для предложенных SLO и описать, каким образом рассчитанный бюджет может использоваться командой разработки и эксплуатации для принятия решений о внедрении изменений.

Результат должен быть представлен в форме аналитической записки, содержащей обоснование выбора метрик, а также таблицы соответствия SLI–SLO–SLA и расчета бюджета ошибок.

Задание 2. Разработай архитектуру системы наблюдаемости для георасределенной платформы электронной коммерции, состоящей из микросервисных компонентов, развернутых в среде оркестрации контейнеров Kubernetes в трех регионах. Для это необходимо:

1. Обосновать выбор стека технологий для сбора, хранения и визуализации метрик производительности, а также для централизованного сбора и анализа логов с учетом требований требования к масштабируемости, отказоустойчивости и временным задержкам.
2. Разработать схему потоков данных, отражающую маршруты движения метрик и логов от источников (приложения, узлы кластера, сетевые устройства) до систем хранения и визуализации.
3. Определить перечень ключевых метрик, подлежащих сбору на каждом уровне архитектуры (инфраструктурном, платформенном, прикладном), с указанием метода их получения.
4. Описать конфигурацию правил оповещения для трех различных сценариев развития инцидента (деградация производительности, частичная недоступность сервиса, исчерпание ресурсов) с указанием пороговых значений и механизмов маршрутизации уведомлений.
5. Предложить структуру единой панели визуализации, позволяющей оператору получить комплексное представление о состоянии платформы, включая корреляцию метрик и логов.

Задание оформляется в виде технического решения, содержащего текстовое описание архитектуры, блок-схем потоков данных, спецификацию метрик и правил алертинга, и макета информационной панели с перечнем отображаемых элементов.

Тест для самоконтроля

1. Мера времени, затрачиваемого на выполнение отдельной операции или доставку пакета данных от источника к получателю – это…

а) Пропускная способность.

б) Латентность.

в) Время задержки.

г) Время отклика.

2. В чем измеряется латентность?

а) Процентили.

б) Проценты.

в) Доли единиц.

г) Килобайты.

3. Как называется мера количества операций или объема данных, которые система способна обработать за единицу времени?

а) Время ожидания.

б) Латентность.

в) Пропускная способность.

г) Время обработки.

4. Зависимость между латентностью и пропускной способностью описывает …

а) Закон Зипфа.

б) Закон больших чисел.

в) Законы де Моргана.

г) Закон Литтла.

5. В иерархии метрик производительности и доступности SLA находится…

а) на уровне 1: измерения;

б) на уровне 2: индикатор;

в) на уровни 3: цель;

г) на уровне 4: обязательство.

6. Числовые измерения, характеризующие состояние системы в конкретный момент времени, называются… .

а) метриками;

б) логами;

в) операциями;

г) стеками.

7. Хронологически упорядоченные текстовые записи о событиях, произошедших в системе или приложении, содержащие контекстную информацию для детального расследования инцидентов – это…

а) метрики;

б) пропускная способность;

в) логи;

г) время ожидания.

8. Сколько этапов обработки содержит функциональное разделение компонентов стека ELK для мониторинга логов?

а) 3;

б) 4;

в) 5;

г) 6.

9. Свойство инфраструктуры противодействовать сбоям, поглощать их воздействие и сохранять или оперативно восстанавливать ключевые функции в условиях деградации компонентов, внешнего давления или атак.

а) Отказоустойчивость.

б) Достаточность.

в) Работоспособность.

г) Активность.

10. Максимально допустимая длительность простоя системы или бизнес-процесса после инцидента в рамках концепции BIA - это

а) Целевое время восстановления.

б) Целевая точка восстановления.

в) Целевая пропускная способность.

г) Целевое время отклика.

11. Целевая точка восстановления – это…

а) свойство противодействовать сбоям, поглощать их воздействие и сохранять или оперативно восстанавливать ключевые функции;

б) хронологически упорядоченные текстовые записи о событиях, произошедших в системе или приложении;

в) максимальный объем данных (измеряемый во времени), допустимый к потере в результате инцидента;

г) максимально допустимая длительность простоя системы или бизнес-процесса после инцидента.

12. Низкая стоимость, простота реализации характеризуется для какого уровня стратегии аварийного восстановления ИТ-системы?

а) Резервный ЦОД.

б) Резервное копирование и восстановление.

в) Синхронная репликация с «теплым» резервом.

г) Кластерное решение «активный-активный» между ЦОД.

13. Что не содержат реактивные процессы восстановления цифровой инфраструктуры

а) Процесс детектирования и классификации инцидента.

б) Процесс локализации и диагностики.

в) Процесс проведения регламентных работ и тестирования отказоустойчивости.

г) Процесс устранения и восстановления работоспособности.

14. Какие вопросы не ставятся при реализации процесса анализа инцидентов (Postmortem / Blameless Retrospective)?

а) «Что произошло?»

б) «Почему это произошло?»

в) «Когда это произошло»?

г) «Какие действия мы предпримем, чтобы это не повторилось?»

15. Фундаментальной концепцией, лежащей в основе современных систем обеспечения доступности, является … .

а) отказоустойчивость;

б) время ожидания;

в) время простоя;

г) диагностика.

16. Основная функция в обеспечении доступности при применении технологий оркестрации и автоматизации является…

а) Устранение дрейфа конфигураций, гарантированная идентичность окружений, быстрый откат и развертывание.

б) Автоматическое развертывание, масштабирование, самовосстановление компонентов инфраструктуры.

в) Анализ и выявление аномалий, прогнозирование инцидентов, автоматическое определение корневых причин.

г) Минимальная потеря данных при сбое, быстрое переключение на резервный сайт.

17. В чем состоит основная функция применения технологии сквозной наблюдаемости?

а) Анализ и выявление аномалий, прогнозирование инцидентов, автоматическое определение корневых причин.

б) Глубокая видимость во все компоненты системы, быстрое понимание состояния и зависимостей

в) Обеспечение доступности сервиса на уровне приложения и данных при отказе целого дата-центра или облачного региона.

г) Анализ и выявление аномалий, прогнозирование инцидентов, автоматическое определение корневых причин.

18. К CAPEX относятся затраты…

а) на приобретение серверного и сетевого оборудования повышенной надежности, системы хранения данных с функциями репликации;

б) на энергопотребление и охлаждение, часто возрастающие из-за дублирования систем;

в) на оплату высококвалифицированного персонала для поддержки сложных инфраструктур, ежегодные лицензионные отчисления;

г) на тестирование планов восстановления и регулярный аудит.

19. Представляет собой независимую и документированную процедуру оценки способности инфраструктуры обеспечивать требуемый уровень сервиса в условиях сбоев и катастроф – это… .

а) аудит доступности и восстановления информационной системы;

б) аудит отказоустойчивости цифровой инфраструктуры;

в) аудит доступности и восстановления цифровой инфраструктуры;

г) аудит работоспособности цифровой инфраструктуры.

20. Уровень доступности в 99,99% характеризует допустимое время простоя в год…

а) 87 часов 36 минут.

б) 8 часов 46 минут.

в) 52 минуты 36 секунд.

г) 5 минут 15 секунд.

Глава 6. БЕЗОПАСНОСТЬ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ

6.1. Модель угроз для цифровой инфраструктуры

В теории информационной безопасности под моделью угроз понимается формализованное описание свойств информационной системы, совокупности актуальных угроз безопасности информации, а также сценариев их реализации нарушителями. Моделирование угроз выступает фундаментальным этапом построения системы защиты, позволяющим перейти от абстрактных принципов обеспечения безопасности к верифицируемым требованиям к механизмам защиты.

Необходимость построения модели угроз обусловлена фундаментальным положением теории безопасности: невозможно создать эффективную систему защиты без идентификации объектов защиты и источников опасности. Попытки реализации защиты «в целом» без предварительной систематизации угроз приводят либо к избыточности мер защиты и неоправданным экономическим затратам, либо к формированию «слепых зон» — неконтролируемых каналов реализации угроз.

В российской юрисдикции методологический базис моделирования угроз задан нормативными документами регуляторов. Ключевым актом выступает «Методика оценки угроз безопасности информации», утвержденная ФСТЭК России 5 февраля 2021 года, устанавливающая единый порядок определения актуальности угроз для государственных информационных систем, объектов критической информационной инфраструктуры и систем персональных данных.

Начальным этапом построения модели угроз выступает инвентаризация и классификация объектов защиты. Под объектом защиты понимается информация, носитель информации или информационный процесс, в отношении которого возможна реализация угроз безопасности. Классификация объектов осуществляется по следующим категориям:

1. Информационные активы: сведения, подлежащие защите в соответствии с законодательством (персональные данные, государственная тайна, коммерческая тайна), а также информация, критически важная для функционирования организации.

2. Программные компоненты: системное и прикладное программное обеспечение, включая исходные коды и исполняемые модули.

3. Аппаратные компоненты: серверное оборудование, автоматизированные рабочие места, сетевая инфраструктура, физические носители.

4. Технологические процессы: процедуры сбора, обработки, хранения и передачи информации.

Определение границ системы требует формальной спецификации всех интерфейсов взаимодействия: точки сопряжения с внешними сетями, каналы удаленного доступа, регламенты взаимодействия с подрядными организациями. Полнота описания границ системы детерминирует адекватность последующей оценки угроз.

Нарушитель безопасности информации представляет собой субъект (физическое лицо, юридическое лицо, группа лиц), реализующий угрозу безопасности информации. В современной теории информационной безопасности принята многоуровневая классификация нарушителей.

По признаку наличия доступа:

- Внешние нарушители — не имеющие права доступа к защищаемой системе;

-Внутренние нарушители — легитимные пользователи, обладающие определенными правами доступа.

По уровню возможностей (градация ФСТЭК):

- Н1 (базовый уровень) — нарушители, использующие общедоступные инструменты и методы;

- Н2 (базовый повышенный уровень) — нарушители, обладающие специальными знаниями и возможностью создания вредоносного ПО;

- Н3 (средний уровень) — организованные группы, имеющие доступ к специализированным средствам;

- Н4 (высокий уровень) — нарушители, располагающие неограниченными ресурсами (спецслужбы) .

По мотивационному фактору:

- Финансово мотивированные (киберпреступные группы);

- Политически мотивированные (хактивисты);

- Разведывательные (конкуренты, иностранные государства);

- Немотивированные деструктивные действия (вандализм);

- Инсайдеры (действия в личных интересах или под внешним давлением).

Идентификация угроз безопасности информации осуществляется на основе анализа источников угроз, факторов, способствующих их реализации, и потенциальных последствий. В российской практике основным источником информации об угрозах выступает Банк данных угроз безопасности информации ФСТЭК России, аккумулирующий формализованные описания угроз с присвоением уникальных идентификаторов.

Международным стандартом описания поведения нарушителей является матрица MITRE ATT&CK, систематизирующая тактики (целевые этапы атаки) и техники (конкретные методы реализации). Интеграция отечественных и международных классификаторов позволяет построить многоуровневую модель, учитывающую как требования регуляторов, так и актуальную практику противодействия атакам.

Формально угроза признается актуальной при одновременном выполнении трех условий:

1. Существование потенциального нарушителя, способного реализовать угрозу.

2. Наличие уязвимостей, обеспечивающих техническую возможность реализации.

3. Возникновение неприемлемых последствий для деятельности организации в случае реализации угрозы.

Модель угроз безопасности информации представляет собой динамический документ, требующий регулярной актуализации по мере изменения как архитектуры защищаемой системы, так и внешних условий функционирования. Корректно построенная модель позволяет обеспечить селективность применения защитных мер — концентрацию ресурсов на противодействии наиболее вероятным и опасным угрозам при минимизации избыточных затрат. Данное обстоятельство является фундаментом при разработке базовых принципов формирования информационной безопасности цифровой инфраструктуры современной компании независимо от ее размера и отраслевой принадлежности.

6.2. Базовые принципы информационной безопасности

Информационная безопасность (ИБ) цифровой инфраструктуры представляет собой системную и непрерывную деятельность, направленную на обеспечение устойчивого функционирования информационных систем, защиту информационных активов и управление связанными с ними рисками. В основе этой деятельности лежат фундаментальные, или базовые, принципы, которые носят универсальный характер и должны учитываться на всех этапах жизненного цикла цифровых систем – от проектирования и разработки до эксплуатации и вывода из использования. Эти принципы формируют концептуальный каркас, определяющий цели, подходы и методы построения надежной системы защиты (рис. 6.1).

Рис.6.1. Основные принципы информационной безопасности

Центральной концепцией, интегрирующей базовые принципы, является триада КИА (CIA Triad), которая традиционно рассматривается как ядро целей информационной безопасности. Конфиденциальность обеспечивает ограничение доступа к информации и информационным активам только для авторизованных субъектов (пользователей, процессов, систем). Целостность гарантирует, что информация и процессы ее обработки защищены от несанкционированного или непред-

намеренного искажения или разрушения. Доступность означает обеспечение авторизованного доступа к информации и связанным с ней активам в требуемое время. Данная триада не является исчерпывающей, но задает базовые ориентиры. Важно отметить, что в современных условиях принципы аутентичности, подотчетности (неотказуемости) и достоверности также рассматриваются как фундаментальные дополнения к классической триаде. Аутентичность обеспечивает проверку и подтверждение того, что субъект или ресурс является тем, за кого он себя выдает. Подотчетность предполагает, что действия субъекта в системе могут быть однозначно прослежены до этого субъекта, что исключает возможность отказа от совершенных действий. Достоверность означает, что информация и процессы ее обработки являются технически корректными и обоснованными.

Принцип гарантированности (assurance) утверждает, что меры безопасности, их реализация и функционирование должны быть обоснованными, проверяемыми и достоверными. Это требует применения формализованных методов верификации, тестирования, сертификации и оценки соответствия заданным требованиям и стандартам. Без гарантированности сама система защиты не может считаться надежной, так как ее эффективность не подтверждена объективными доказательствами.

Принцип адекватности (соразмерности) средств защиты уровню угроз и ценности активов является краеугольным камнем экономически обоснованного управления рисками. Реализация этого принципа требует проведения регулярной оценки рисков, в рамках которой идентифицируются активы, анализируются уязвимости и угрозы, оценивается потенциальный ущерб. Меры защиты должны быть соразмерны вероятности реализации угрозы и величине возможных потерь. Недопустимо как применение избыточных, экономически нецелесообразных мер, так и пренебрежение необходимыми мерами защиты для критически важных активов.

Принцип непрерывности и эшелонированности защиты (защита в глубине, defence-in-depth) предполагает, что безопасность не может обеспечиваться единичным техническим средством или решением. Он требует создания многоуровневой системы защиты, в которой применяются разнородные, взаимодополняющие меры (организационные, физические, технические, правовые) на различных рубежах (периметр,

сеть, узел, приложение, данные). Нарушение одного рубежа или средства защиты должно компенсироваться и обнаруживаться на следующем уровне. Данный принцип также подразумевает непрерывность процесса защиты во времени, включая мониторинг, реагирование и восстановление.

Принцип минимизации привилегий (наименьших прав) является ключевым для ограничения потенциального ущерба от ошибок или злонамеренных действий. Каждому субъекту системы (пользователю, процессу, службе) должны предоставляться минимально необходимые права доступа и полномочия, требуемые для выполнения его законных задач, и только на минимально необходимое время. Это существенно сокращает поверхность атаки и ограничивает горизонтальное перемещение злоумышленника в случае компрометации.

Принцип разделения обязанностей и ротации кадров направлен на предотвращение мошенничества и злоупотреблений. Критически важные функции, операции или знания должны быть разделены между несколькими субъектами таким образом, чтобы для совершения потенциально опасного действия требовалось сговор нескольких лиц. Ротация персонала, ответственного за выполнение таких функций, снижает риски длительных скрытых нарушений.

Принцип простоты контроля и понимания механизмов защиты исходит из того, что сложная, не поддающаяся анализу система защиты с большой вероятностью содержит скрытые уязвимости и ошибки. Архитектура безопасности должна быть максимально прозрачной, понятной для специалистов, осуществляющих ее проектирование, внедрение и аудит. Чрезмерная сложность является самостоятельным источником угроз.

Принцип слабейшего звена гласит, что стойкость всей системы защиты в целом определяется стойкостью ее наиболее уязвимого компонента. Злоумышленник, как правило, атакует не самые защищенные элементы, а ищет наиболее легкий путь. Следовательно, система защиты должна проектироваться и оцениваться комплексно, с особым вниманием к потенциально слабым местам, таким как человеческий фактор, устаревшее оборудование или непропатченное программное обеспечение.

Принцип постоянного мониторинга и реагирования на инциденты отражает динамическую природу угроз. В условиях постоянно

эволюционирующего ландшафта киберугроз невозможно создать статическую, раз и навсегда заданную систему защиты. Требуется непрерывный сбор и анализ данных о событиях безопасности (посредством систем SIEM, IDS/IPS), оперативное выявление аномалий и инцидентов, а также наличие заранее отработанных планов реагирования и восстановления (процедур CSIRT и SOC).

Принцип осведомленности пользователей и формирования культуры безопасности признает, что человек, являясь неотъемлемым элементом цифровой инфраструктуры, часто представляет собой наиболее значимый фактор риска. Технические меры могут быть нивелированы ошибками, небрежностью или социальной инженерией. Поэтому обязательным элементом системы ИБ является постоянное обучение и повышение осведомленности всех пользователей и персонала в вопросах политик безопасности, распознавания угроз и безопасных практик работы.

Принцип комплексного подхода (холистичности) интегрирует все вышеперечисленное, требуя одновременного и взаимосвязанного рассмотрения всех аспектов безопасности: юридического, организационного, кадрового, технического, технологического и операционного. Безопасность не является отдельной функцией или дополнением к системе; она должна быть «встроена» (security by design) в архитектуру и бизнес-процессы на этапе проектирования и сопровождать систему на всем протяжении ее жизненного цикла.

Табл. 6.1 обобщает взаимосвязь базовых принципов с практическими целями и реализующими их мерами.

Таблица 6.1

Соответствие базовых принципов ИБ целям и категориям мер защиты

Базовый принцип	Ключевая цель	Категории практических мер (примеры)
Конфиденциальность	Защита от несанкционированного раскрытия информации	Шифрование данных (на хранении, в передаче), управление доступом (RBAC, ABAC, ACL), экранирование сетей (брандмауэры, сегментация), DLP-системы, маскирование и анонимизация данных.

Базовый принцип	Ключевая цель	Категории практических мер (примеры)
Целостность	Защита от несанкционированного изменения информации	Контрольные суммы, хэш-функции, электронная цифровая подпись (ЭЦП), журналирование и аудит изменений, механизмы транзакционной целостности в СУБД, применение средств контроля целостности файлов (HIDS).
Доступность	Обеспечение своевременного доступа к ресурсам	Резервирование и кластеризация оборудования, отказоустойчивые архитектуры, балансировка нагрузки, планирование ресурсов, защита от атак типа «отказ в обслуживании» (DDoS-продукты), создание и тестирование планов восстановления (DRP).
Аутентичность и Подотчетность	Установление подлинности субъектов и регистрация их действий	Многофакторная аутентификация (MFA), инфраструктура открытых ключей (PKI), единая точка входа (SSO), системы централизованного аудита и журналирования (SIEM), корреляция событий.
Защита в глубине	Создание многоуровневой и устойчивой системы защиты	Комбинация периметровых (брандмауэры) и сетевых (IDS/IPS) средств, защита конечных точек (антивирус, EDR), безопасность приложений (WAF, SAST/DAST), изоляция критических сегментов, физическая охрана ЦОД.
Минимизация привилегий	Ограничение масштаба возможного ущерба	Строгое следование модели наименьших привилегий при настройке прав пользователей и сервисных аккаунтов, изоляция контейнеров и виртуальных машин, применение технологий типа Just-In-Time (JIT) административного доступа, сегментация сети по ролям.
Непрерывный мониторинг и реакция	Своевременное обнаружение и нейтрализация инцидентов	Внедрение SOC (Security Operations Center), использование платформ SIEM для сбора и анализа логов, развертывание систем EDR/XDR на конечных точках, проведение регулярных тестов на

Базовый принцип	Ключевая цель	Категории практических мер (примеры)
		проникновение и анализа уязвимостей, наличие регламентированных процедур CSIRT.
Осведомленность и обучение	Снижение рисков, связанных с человеческим фактором	Регулярные обязательные тренинги по ИБ для сотрудников всех уровней, проведение кампаний по фишинговому тестированию, разработка и доведение до сведения понятных политик безопасности, создание позитивной культуры безопасности.

В свою очередь, вместе базовыми принципами информационной безопасности рассматривают и принципы защиты данных при разработке проектов внедрения и модернизации цифровой инфраструктуры (рис. 6.2).

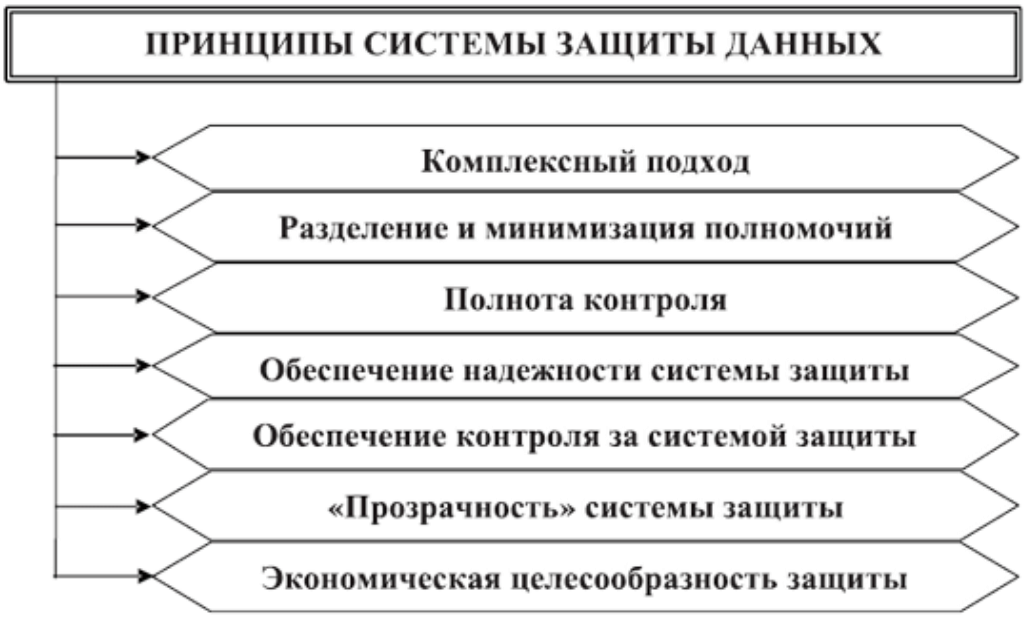

Рис.6.2. Принципы системы защиты данных

В заключение необходимо подчеркнуть, что базовые принципы информационной безопасности носят императивный и взаимозависимый характер. Их реализация не может быть выборочной или фрагментарной. Пренебрежение одним принципом, например, минимизацией привилегий или обучением пользователей, может свести на нет эффект от многомиллионных вложений в технические средства защиты, реализующие принципы конфиденциальности или целостности.

Формирование безопасной цифровой инфраструктуры – это непрерывный процесс, управляемый рисками и основанный на системном применении всей совокупности описанных принципов. Они служат не только руководством для архитекторов и инженеров, но и критерием для оценки зрелости и эффективности существующих систем защиты в условиях постоянно меняющихся технологических ландшафтов и угроз.

6.3. Управление уязвимостями и исправлениями

Управление уязвимостями и исправлениями (Vulnerability and Patch Management) представляет собой непрерывный, циклический и систематический процесс идентификации, анализа, оценки, приоритизации и устранения уязвимостей в программном и аппаратном обеспечении цифровой инфраструктуры. В контексте современной кибербезопасности данный процесс трансформировался из периодической административной задачи в стратегическую функцию управления рисками. Его цель заключается не в достижении абсолютной безопасности (что является недостижимым идеалом), а в минимизации окон эксплуатации и сокращении поверхности атаки до уровней, приемлемых для конкретной организации. Эффективность процесса напрямую определяет устойчивость инфраструктуры к известным угрозам, поскольку подавляющее большинство успешных кибератак эксплуатируют уже описанные и часто давно исправленные уязвимости.

Фундаментальным для понимания предмета является четкое разграничение сопряженных понятий. Уязвимость (Vulnerability) – это слабость в системе, ее процедурах, внутреннем контроле или реализации, которую может использовать источник угрозы для нарушения политики безопасности. Угроза (Threat) – это потенциальная причина нежелательного инцидента, которая может нанести ущерб системе или организации. Эксплойт (Exploit) – это метод или код, намеренно использующий уязвимость для реализации угрозы. Исправление (Patch) – это выпущенное производителем программное обеспечение изменение, предназначенное для устранения уязвимости, исправления ошибки или обновления функциональности. Заплатка (Hotfix, Workaround) – это временное или экстренное решение, позволяющее

смягчить последствия уязвимости до применения полноценного исправления.

Управление уязвимостями базируется на нескольких ключевых принципах: непрерывность (постоянный мониторинг), комплексность (охват всех активов), приоритизация (основанная на рисках) и интеграция в общий процесс управления ИТ-сервисами и киберрисками.

Стандартизированный жизненный цикл описывает последовательность этапов, обеспечивающих полноту и замкнутость процесса.

- Этап 1: инвентаризация активов и определение контекста. Процесс начинается с создания и поддержания в актуальном состоянии полного реестра всех компонентов инфраструктуры: серверов (физических, виртуальных, облачных), рабочих станций, сетевого оборудования, устройств Интернета вещей (IoT), прикладного и системного ПО с указанием версий. Без точной инвентаризации процесс управления уязвимостями теряет смысл, так как невозможно оценить воздействие на неизвестный актив.

- Этап 2: сканирование и обнаружение уязвимостей. На данном этапе осуществляется активный и/или пассивный поиск известных уязвимостей в активах. Для этого используются специализированные средства – сканеры уязвимостей, которые опираются на базы данных общеизвестных уязвимостей (таких как CVE – Common Vulnerabilities and Exposures) и их оценок по шкале CVSS (Common Vulnerability Scoring System). Сканирование может быть аутентифицированным (с предоставлением учетных данных для глубокого анализа) и неаутентифицированным (оценка с точки зрения внешнего злоумышленника). Ключевой метрикой является полнота охвата и частота сканирований.

- Этап 3: анализ и верификация результатов. Автоматически полученные данные сканирования содержат шум и ложные срабатывания. Задача аналитика – провести триангуляцию данных, сопоставив результаты различных инструментов, и верифицировать критичные уязвимости, чтобы подтвердить их реальное наличие и эксплуатабельность в конкретной среде. Это предотвращает ненужное расходование ресурсов на ложные угрозы.

- Этап 4: оценка рисков и приоритизация. Это наиболее сложный и ответственный этап, требующий аналитического подхода. Приоритизация на основе исключительно базового балла CVSS часто недостаточна, так как не учитывает контекст бизнеса.

Современные методологии, такие как EPSS (Exploit Prediction Scoring System) или собственные модели оценки рисков организации, интегрируют дополнительные факторы:

а) Контекст бизнеса: Критичность актива для бизнес-процессов (например, сервер с базой данных клиентов vs. тестовый стенд).

б) Контекст угроз: Наличие работающего эксплойта в открытом доступе (Exploit Available), активность эксплуатации в дикой природе (Active Exploitation), включение уязвимости в популярные эксплойт-паки.

в) Контекст безопасности: Наличие и эффективность компенсирующих контрмер (межсетевые экраны, системы WAF, сегментация сети), которые могут блокировать векторы атаки.
На выходе этапа формируется ранжированный перечень уязвимостей для устранения.

- Этап 5: Устранение (Remediation). Устранение – это процесс ликвидации уязвимости. Он может реализовываться несколькими путями, выбор которых зависит от оценки риска и операционной осуществимости:

а) Применение исправления (Patching). Наиболее надежный и рекомендуемый метод.

б) Внедрение компенсирующих мер (Compensating Control): если немедленное применение исправления невозможно (например, из-за риска нарушения работы критичного приложения), внедряются временные меры безопасности (правила брандмауэра, изоляция сегмента, изменение конфигураций).

в) Принятие риска (Risk Acceptance). Формальное документированное решение руководства принять на себя риск, связанный с уязвимостью, если стоимость устранения превышает потенциальный ущерб. Такой подход требует регулярного пересмотра.

г) Деактивация или замена актива (Decommissioning). Удаление уязвимого компонента из среды, если он больше не нужен или может быть заменен на безопасный аналог.

- Этап 6: Подтверждение и отчетность. После выполнения мероприятий по устранению необходимо повторно просканировать активы для подтверждения эффективности примененных мер. Не менее важна отчетность перед руководством и регуляторами, демонстрирующая метрики снижения рисков, такие как среднее время на исправление

(Mean Time to Remediate – MTTR), количество открытых критических уязвимостей, тенденции их появления и устранения.

Управление исправлениями является критической подсистемой в общем цикле устранения уязвимостей. Это регламентированный процесс тестирования, утверждения, развертывания и проверки исправлений для ПО и систем.

Классификация исправлений и источники обновлений Исправления классифицируются по критичности, источнику и типу. Основными источниками являются официальные репозитории вендоров ОС, приложений и производителей оборудования. Мониторинг этих источников должен быть формализован. По срочности и типу выделяют:

- Критические / Безопасностные обновления (Security Updates). Устраняют конкретные уязвимости безопасности.

- Накопительные пакеты обновлений (Cumulative Updates). Включают все предыдущие обновления и новые исправления.

- Обновления функций (Feature Updates). Добавляют новый функционал, но также могут содержать исправления безопасности.

- Обновления драйверов (Driver Updates).

Стандартизированный рабочий процесс (Workflow) Эффективный процесс управления исправлениями включает строгую последовательность действий, представленную в табл. 6.2.

Таблица 6.2

Типовые этапы рабочего процесса управления исправлениями

Этап	Цель и ключевые действия	Участники / Ответственные
Оповещение и получение	Мониторинг официальных источников на предмет выхода новых исправлений. Загрузка и проверка цифровой подписи.	Команда безопасности, ИТ-администраторы
Оценка и приоритизация	Анализ описания исправления (CVE, CVSS), оценка критичности и срочности применения в контексте среды организации.	Команда безопасности, владельцы бизнес-приложений
Тестирование	Развертывание исправления в изолированной тестовой среде, максимально приближенной к промышленной. Проверка на функциональную совместимость и отсутствие регрессионных ошибок.	ИТ-администраторы, тестировщики

Этап	Цель и ключевые действия	Участники / Ответственные
Утверждение	Формальное утверждение плана развертывания на основе результатов тестирования и оценки рисков.	CAB (Change Advisory Board), ИТ-руководство
Развертывание	Плановое каскадное внедрение исправления в промышленную среду, начиная с наименее критичных групп. Использование систем централизованного управления (WSUS, SCCM, сторонние решения).	ИТ-администраторы
Верификация и откат	Подтверждение успешной установки и отсутствия сбоев. Разработка и готовность к выполнению процедуры отката на предыдущую стабильную конфигурацию в случае возникновения проблем.	ИТ-администраторы, команда поддержки
Документирование	Внесение записей о примененном исправлении в реестр изменений (Change Log), обновление инвентаризационных данных активов.	Все участники процесса

Эффективность процессов управления уязвимостями и исправлениями подлежит количественной оценке. Ключевые показатели эффективности (KPI) позволяют объективно оценить производительность, выявить узкие места и обосновать ресурсные потребности. Основные метрики сведены в табл. 6.3.

Таблица 6.3.

Ключевые метрики для оценки эффективности управления уязвимостями и исправлениями

Категория метрик	Конкретная метрика	Формула / Описание	Целевое значение
Своевременность	Среднее время на исправление (MTTR)	Σ(Время обнаружения – Время устранения) / Число уязвимостей	Минимизация, специфично для класса риска (напр., <7 дней для критичных)
	Охват сканированием	(Число отсканированных активов / Общее число активов) * 100%	Стремление к 100%
Качество процесса	Процент ложных срабатываний	(Число ложных срабатываний / Общее число обнаружений) * 100%	Минимизация
	Соотношение способов устранения	Распределение уязвимостей по способам: патч / заплатка / принятие риска	Рост доли устранения патчами

Категория метрик	Конкретная метрика	Формула / Описание	Целевое значение
Состояние защищенности	Количество открытых уязвимостей по критичности	Абсолютное число открытых уязвимостей с высоким/критичным уровнем риска	Постоянная тенденция к снижению
	Возраст самой старой критической уязвимости	Время с момента публикации CVE до текущей даты для самой старой неисправленной критической уязвимости	Минимизация

Управление уязвимостями не существует изолированно. Оно тесно интегрировано с другими процессами безопасности:

- Управление ИТ-активами (ITAM) - является источником достоверных данных для инвентаризации.

- Управление конфигурациями (CM) обеспечивает соблюдение безопасных базовых стандартов (hardening), снижающих исходную уязвимость систем.

- Управление инцидентами безопасности (SIEM, SOAR). Данные об уязвимостях используются для корреляции и расследования инцидентов; информация об активных атаках помогает в приоритизации.

- Управление рисками кибербезопасности. Процесс является основным операционным механизмом для обработки рисков, связанных с программными уязвимостями.

Кроме того, процесс является критически важным для соблюдения требований отраслевых и международных стандартов, таких как ГОСТ Р 57580, PCI DSS (требование 6), ISO 27001 (контроль A.12.6.1), NIST Cybersecurity Framework (функция «Защита», категория PR.IP-12). Наличие отлаженного, документированного и измеряемого процесса управления уязвимостями является обязательным элементом зрелой системы информационной безопасности любой современной цифровой инфраструктуры.

6.4. Идентификация и доступ (IAM), защита периметра и сегментация сети

Безопасность цифровой инфраструктуры основывается на фундаментальном принципе минимизации доверия и применения стратегий глубокой (эшелонированной) обороны. В этом контексте три взаимосвязанных концепции – управление идентификацией и доступом (Identity and Access Management, IAM), защита сетевого периметра и сегментация сети – формируют базис для построения контролируемой и устойчивой к угрозам ИТ-среды. Их комплексная реализация позволяет не только противостоять внешним атакам, но и эффективно ограничивать ущерб от инцидентов, инициированных внутри системы, будь то действия скомпрометированной учетной записи или вредоносное программное обеспечение. Данный параграф рассматривает эволюцию, архитектурные принципы и практические аспекты реализации этих механизмов в современных условиях гибридных и облачных инфраструктур.

IAM представляет собой дисциплину и совокупность технологий, обеспечивающих корректную идентификацию, аутентификацию, авторизацию и учет действий (Audit) субъектов (пользователей, сервисов, устройств) в отношении объектов (данных, приложений, систем). Это центральный элемент кибербезопасности, определяющий принцип «кто, что и при каких условиях может делать».

Современная модель IAM эволюционировала от простых локальных списков пользователей к распределенным, федеративным и гибридным системам. Ключевыми компонентами являются:

- Идентификация и Аутентификация. Процесс подтверждения заявленной идентичности субъекта. Современные системы требуют перехода от единичных факторов (пароль) к многофакторной аутентификации (MFA), основанной на комбинации знания (пароль), владения (токен, мобильное устройство) и свойства (биометрия).

- Авторизация - процесс определения прав доступа аутентифицированного субъекта к ресурсам. Здесь доминируют модели на основе ролей (Role-Based Access Control, RBAC) и, все чаще, на основе атрибутов (Attribute-Based Access Control, ABAC). ABAC обеспечивает бо-

лее гранулированный и контекстно-зависимый контроль, учитывая атрибуты пользователя, ресурса, действия и среды (время, местоположение, уровень безопасности устройства).

- Управление жизненным циклом идентификаций. Автоматизированные процессы создания, удаления и изменения прав в соответствии с бизнес-процессами (прием на работу, смена роли, увольнение).

- Учет и мониторинг (Audit & Logging). Неизменяемая регистрация всех событий, связанных с доступом, для последующего анализа, расследования инцидентов и обеспечения соответствия требованиям регуляторов.

Особую сложность в современных условиях представляет управление доступом не только для людей, но и для машин (Machine-to-Machine, M2M), сервисов и приложений. Для этого используются сервисные аккаунты и принципы наименьших привилегий (Principle of Least Privilege, PoLP), когда субъекту предоставляется ровно тот доступ, который необходим для выполнения конкретной задачи, и не более.

Традиционно концепция защиты периметра (Perimeter Security) строилась на модели «крепости» с надежными внешними стенами и условно доверенной внутренней средой. Основным инструментом являлся межсетевой экран (брандмауэр), осуществляющий фильтрацию трафика между сетями с разным уровнем доверия на основе статических правил (адреса, порты, протоколы).

Данная модель оказалась неадекватной в условиях современного ландшафта угроз и цифровой трансформации. Стирание четких границ сети из-за распространения мобильных устройств, удаленного доступа, облачных сервисов и партнерских подключений привело к концепции «размытого периметра». Современная защита периметра трансформировалась в набор распределенных контролирующих точек доступа, которые применяют политики безопасности независимо от физического местоположения пользователя или ресурса.

Ключевые технологические сдвиги в этой области:

- Next-Generation Firewalls (NGFW). Интегрируют традиционные функции брандмауэра с возможностями глубокой инспекции пакетов (Deep Packet Inspection, DPI), системой предотвращения вторжений (IPS), фильтрацией на уровне приложений (Application Awareness) и контролем на основе идентификации пользователей (User-ID).

- Экранирование веб-приложений (Web Application Firewall, WAF). Специализированный защитный механизм, анализирующий HTTP/HTTPS-трафик для защиты веб-приложений от атак, таких как SQL-инъекции, межсайтовый скриптинг (XSS) и др.

- Программно-определяемая периметральная безопасность (Software-Defined Perimeter, SDP). Архитектура, реализующая принцип «скрытия» инфраструктуры от неавторизованных пользователей. Доступ к ресурсам предоставляется только после успешной аутентификации и авторизации, а сетевое соединение устанавливается динамически на индивидуальной основе, минуя общие точки входа.

Таким образом, современная защита периметра – это не единая линия обороны, а совокупность интеллектуальных, identity-aware решений, распределенных по всей инфраструктуре.

Сегментация сети (Network Segmentation) – это практика разделения компьютерной сети на изолированные подсети (сегменты, зоны) для ограничения горизонтального перемещения угроз. Если IAM контролирует вертикальный доступ «от субъекта к ресурсу», то сегментация регулирует горизонтальную коммуникацию «между ресурсами» внутри инфраструктуры.

Цели сегментации:

1. Сокращение поверхности атаки путем изоляции критически важных систем.

2. Сдерживание и локализация инцидентов безопасности (например, распространения ransomware) в пределах одного сегмента.

3. Соответствие требованиям регуляторов к изоляции данных (например, ПДн, платежные данные).

4. Оптимизация сетевого трафика и производительности.

Традиционная сегментация на основе VLAN и ACL на маршрутизаторах является статической и сложной в управлении в масштабируемых динамических средах. На смену ей приходит микросетевая сегментация (табл. 6.4).

Таблица 6.4

Сравнение классической и микросетевой сегментации

Критерий	Классическая сегментация (на уровне сети)	Микросетевая сегментация (на уровне рабочей нагрузки)
Гранулярность	Грубая (подсети, VLAN). Все хосты в сегменте неявно доверяют друг другу.	Тонкая (отдельная рабочая нагрузка, виртуальная машина, контейнер, группа приложений).
Базовый принцип	Контроль на основе IP-адресов, портов и протоколов (L3-L4).	Контроль на основе идентификации рабочей нагрузки, контекста приложения, тегов (L3-L7).
Границы политики	Привязаны к сетевой топологии и IP-адресации.	Отвязаны от сетевой инфраструктуры, привязаны к самой рабочей нагрузке.
Адаптивность	Низкая. Изменения требуют ручного переконфигурирования сетевого оборудования.	Высокая. Политики динамически применяются и обновляются вместе с жизненным циклом рабочей нагрузки.
Реализация	Сетевое оборудование (маршрутизаторы, коммутаторы, брандмауэры).	Агенты или гипервизорные средства в сочетании с программно-определяемыми сетевыми (SDN) решениями.

Микросетевая сегментация реализует принцип нулевого доверия (Zero Trust) на сетевом уровне, явно определяя, какие рабочие нагрузки могут общаться друг с другом и какие сервисы они могут использовать, независимо от их расположения в сети.

Изолированное применение рассмотренных механизмов не обеспечивает комплексной безопасности. Их максимальная эффективность достигается в рамках архитектуры нулевого доверия (Zero Trust Architecture, ZTA). ZTA отвергает концепцию автоматического доверия к чему-либо внутри или вне периметра и требует постоянной проверки каждого запроса на доступ.

В данной парадигме:

- IAM становится стратегической плоскостью управления, предоставляющей контекст (идентификацию, атрибуты, права) для всех решений о доступе.

- Защита периметра трансформируется в распределенные шлюзы доступа (на уровне сети, приложения, данных), которые запрашивают решения у централизованной системы управления политиками, основанной на данных IAM.

- Сегментация (особенно микросетевая) выступает как тактическая плоскость исполнения внутри центра обработки данных или облака, применяющая детализированные политики доступа между рабочими нагрузками на основе того же контекста.

Пример интегрированного сценария - попытка доступа пользователя к внутреннему приложению:

1. Пользователь аутентифицируется в системе IAM с использованием MFA.

2. Шлюз удаленного доступа (современный периметровый контроллер) запрашивает у системы авторизации решение о доступе к конкретному приложению, передавая контекст (идентификатор, устройство, его состояние).

3. Получив разрешение, пользователь подключается к приложению. Система микросетевой сегментации, будучи синхронизированной с IAM, разрешает трафик только между конкретным экземпляром этого приложения и нужной базой данных в бэкенде, блокируя любые другие попытки связи.

Ключевые направления реализации принципов Zero Trust представлены ниже в табл. 6.5.

Идентификация и доступ, защита периметра и сегментация сети представляют собой триединую основу безопасности цифровой инфраструктуры. Их эволюция движется в сторону большей гранулярности, динамичности, ориентации на идентичность и приложения, а также глубокой интеграции. Внедрение этих механизмов по отдельности дает лишь частичный эффект.

Стратегическая цель – построение адаптивной, контекстно-зависимой системы безопасности, в которой они работают согласованно в рамках единой архитектурной модели, такой как Zero Trust. Это позволяет организациям не только защищаться от внешних и внутренних угроз, но и безопасно ускорить цифровую трансформацию, используя гибридные и облачные среды без компромиссов в области безопасности.

Таблица 6.5

Вклад компонентов в реализацию принципов Zero Trust

Принцип Zero Trust	Реализация через IAM	Реализация через защиту периметра	Реализация через сегментацию
Явная проверка	Строгая многофакторная аутентификация.	Инспекция всего трафика, валидация сессий.	Проверка идентификаторов и контекста рабочих нагрузок для любого внутреннего соединения.
Принцип наименьших привилегий	RBAC/ABAC, JIT-доступ (Just-In-Time).	Правила «запретить по умолчанию», открывающие доступ только к конкретным сервисам.	Политики «запретить по умолчанию» между сегментами/рабочими нагрузками.
Предположение о компрометации	Постоянный мониторинг аномалий в поведении учетных записей.	Изоляция и анализ подозрительного трафика, песочницы (sandboxing).	Минимизация «взрывной радиуса» — изоляция инцидента в пределах сегмента.

Будущие исследования и разработки в этой области, вероятно, будут сосредоточены на дальнейшей автоматизации, использовании искусственного интеллекта для анализа поведения и динамического формирования политик, а также на создании единых стандартов для обеспечения совместимости компонентов от различных вендоров.

Подводя итог сказанному выше, необходимо заключить, что фундаментом построения защищенной цифровой инфраструктуры выступает формализованная модель угроз, динамически актуализируемая в соответствии с изменением архитектуры объекта защиты и ландшафта угроз. Корректная идентификация источников угроз, классификация нарушителей и учет условий актуальности угроз позволяют перейти от абстрактного обеспечения безопасности к верифицируемым требованиям и экономически обоснованной селективности защитных мер.

В сою очередь, система обеспечения безопасности базируется на императивной и взаимозависимой совокупности базовых принципов. Классическая триада «конфиденциальность, целостность, доступность» дополняется принципами аутентичности, подотчетности, а

также концепциями гарантированности, минимизации привилегий и эшелонированной защиты.

Таким образом, обеспечение контролируемости цифровой инфраструктуры достигается синергией механизмов управления идентификацией и доступом (IAM), адаптивной защиты периметра и глубокой сегментации сети. В условиях размытия традиционных границ периметра архитектура нулевого доверия (Zero Trust) выступает интегрирующей парадигмой, в рамках которой IAM предоставляет стратегический контекст для принятия решений, а микросетевая сегментация и распределенные шлюзы обеспечивают гранулированное и динамичное исполнение политик безопасности на уровне рабочих нагрузок. Изолированное применение указанных механизмов не позволяет достичь требуемого уровня защиты в современных гибридных средах.

Вопросы для обсуждения

1. Раскройте содержание понятия «модель угроз безопасности информации». Какова ее роль в процессе построения системы защиты?
2. Перечислите основные этапы построения модели угроз. Какие действия должны быть выполнены на каждом из этапов?
3. Охарактеризуйте категории объектов защиты цифровой инфраструктуры, которые подлежат идентификации при моделировании угроз.
4. Объясните, по каким причинам необходимо четко определять границы анализируемой системы.
5. Охарактеризуйте основные этапы жизненного цикла управления уязвимостями.
6. Поясните, на каком этапе, по какой причине, а также с использованием каких критериев производится приоритизация уязвимостей?
7. Перечислите и охарактеризуйте основные категории объектов защиты, подлежащих инвентаризации при построении модели угроз в соответствии с российскими методологиями.
8. Сформулируйте условия, при которых угроза безопасности информации признается актуальной.
9. Назовите ключевые условия, при которых угроза безопасности информации признается актуальной. Какую роль в этом процессе играет Банк данных угроз ФСТЭК России?

10. Перечислите основные факторы, которые необходимо учитывать при оценке последствий реализации угрозы. Приведите классификацию последствий (нарушение конфиденциальности, целостности, доступности).

11. Объясните, в чем состоит специфика учета внутренних нарушителей при построении модели угроз. Какие категории персонала представляют наибольшую опасность и по какой причине?

12. Объясните, по каким причинам модель угроз требует регулярной актуализации. Какие события в организации или во внешней среде являются основанием для пересмотра модели?

13. Дайте определите понятию «управление идентификацией и доступом» и охарактеризуйте его ключевые компоненты.

14. Назовите ключевые метрики эффективности (KPI) процессов управления уязвимостями и исправлениями и поясните, что измеряет каждая из них.

15. Поясните различия между моделями авторизации на основе ролей (RBAC) и на основе атрибутов (ABAC).

16. Объясните, какие цели преследует сегментация сети в контексте обеспечения информационной безопасности цифровой инфраструктуры.

17. Опишите гипотетический сценарий попытки доступа пользователя к внутреннему приложению, демонстрирующий интегрированное взаимодействие механизмов IAM, современной защиты периметра и микросетевой сегментации.

18. Раскройте содержание типового рабочего процесса управления исправлениями, указав цели и ключевые действия на каждом этапе.

19. Объясните, каким образом в рамках архитектуры нулевого доверия распределяются функции между системами IAM, защиты периметра и сегментации сети для обеспечения явной проверки каждого запроса доступа.

20. Поясните тезис о том, что идентификация и доступ, защита периметра и сегментация сети представляют собой «триединую основу» безопасности, а их изолированное применение не обеспечивает комплексной защиты.

Практические задания

Задание 1. В компании после увольнения сотрудника его учётная запись не была заблокирована. Через две недели зафиксирована ночная выгрузка базы клиентов из CRM на внешний носитель.

1. Идентифицируйте объекты воздействия, использованные уязвимости и последствия инцидента.

2. Найдите в БДУ ФСТЭК не менее трёх угроз, соответствующих инциденту (указать ID).

3. Классифицируйте нарушителя (тип, уровень возможностей).

4. Предложите две меры защиты (организационную и техническую).

Задание 2. Клиника имеет: 5 АРМ администраторов (Windows, доступ в интернет), 3 АРМ врачей (без интернета), сервер с МИС (персональные данные пациентов), файловый сервер, раздельные Wi-Fi сети (гостевая и административная).

Составьте таблицу из 5 наиболее вероятных угроз со следующими полями:

- Наименование угрозы (ID из БДУ при наличии).
- Источник угрозы (нарушитель).
- Объект воздействия.
- Возможные последствия.

Задание 3. АСУ ТП газоперерабатывающего предприятия. Сеть изолирована от интернета, имеет ограниченную связь с корпоративным сегментом. Физический доступ ограничен пропускной системой. Персонал: операторы, технологи, слесари, подрядчики.

Разработайте описание двух типов нарушителей (на выбор), включив:

- тип (внешний/внутренний), уровень возможностей (Н1–Н4) с обоснованием;
- мотивацию и цели воздействия;
- предполагаемый сценарий действий;
- признаки обнаружения.

Тест для самоконтроля

1. Какова основная цель построения модели угроз безопасности информации?

а) Выполнение формального требования регулятора для прохождения проверки.

б) Детальное описание всех возможных кибератак за последние пять лет.

в) Обоснованный выбор мер защиты путём идентификации актуальных угроз и объектов воздействия.

г) Составление перечня всего установленного программного обеспечения в организации.

2. Какой документ ФСТЭК России является методологической основой для определения актуальности угроз в государственных информационных системах?

а) ГОСТ Р 57580.1-2017 «Безопасность финансовых (банковских) операций».

б) Методика оценки угроз безопасности информации (утв. 05.02.2021).

в) Приказ Минцифры № 786 «Об утверждении требований к защите информации».

г) Доктрина информационной безопасности Российской Федерации.

3. К какой категории нарушителей по уровню возможностей (согласно классификации ФСТЭК) относятся организованные преступные группы, имеющие доступ к специализированным средствам проведения атак?

а) Н1 (базовый уровень).

б) Н2 (базовый повышенный уровень).

в) Н3 (средний уровень).

г) Н4 (высокий уровень).

4. Что из перечисленного является источником информации об угрозах, содержащим формализованные описания с уникальными идентификаторами (например, УБИ.015), обязательными для использования в отчётности перед регулятором?

а) MITRE ATT&CK.

б) OWASP Top 10.

в) Банк данных угроз безопасности информации ФСТЭК России.

г) NIST National Vulnerability Database.

5. Угроза безопасности информации признаётся актуальной, если:

а) Она описана в отчётах любой антивирусной компании за последний год.

б) Существует нарушитель, имеются уязвимости и возможны неприемлемые последствия.

в) Её реализация возможна теоретически, даже без наличия реальных уязвимостей.

г) Против не существуют сертифицированные средства защиты информации.

6. Какое последствие относится к нарушению целостности информации?

а) Несанкционированное копирование базы данных клиентов.

б) Блокировка веб-сайта компании в результате DDoS-атаки.

в) Несанкционированное изменение финансового отчёта перед его отправкой в налоговую.

г) Перехват трафика между сервером и рабочим местом администратора.

7. Какая современная тенденция заключается в атаке на менее защищённого подрядчика с целью последующего проникновения в инфраструктуру основной компании-цели?

а) Использование искусственного интеллекта для генерации фишинга.

б) Атака на цепочки поставок.

в) Маскировка вредоносного ПО под легитимные обновления.

г) Применение техник «Living off the Land».

8. В чём состоит основная сложность защиты от внутреннего нарушителя (инсайдера)?

а) Внутренний нарушитель всегда действует из финансовых побуждений.

б) Инсайдер уже имеет легитимный доступ к системе и знает особенности её защиты.

в) Внутреннего нарушителя невозможно обнаружить техническими средствами.

г) Действия инсайдера всегда попадают под действие уголовного кодекса.

9. Какой международный фреймворк систематизирует тактики (чего хочет достичь злоумышленник) и техники (как он это делает)?

а) ISO/IEC 27001;

б) COBIT 5;

в) MITRE ATT&CK;

г) ITIL.

10. К какому типу последствий относится ситуация, когда сотрудники организации не могут получить доступ к корпоративной CRM-системе в течение рабочего дня из-за сбоя в работе серверов?

а) Нарушение конфиденциальности.

б) Нарушение целостности.

в) Нарушение доступности.

г) Нарушение аутентичности.

11. Какой принцип является краеугольным камнем экономически обоснованного управления рисками и требует соразмерности средств защиты ценности активов

а) Принцип гарантированности.

б) Принцип непрерывности и эшелонированности защиты.

в) Принцип адекватности (соразмерности).

г) Принцип осведомленности пользователей.

12. Какая цель преследуется реализацией принципа минимизации привилегий (наименьших прав)?

а) Повышение производительности труда сотрудников.

б) Ограничение потенциального ущерба от ошибок или злонамеренных действий.

в) Сокращение времени на предоставление доступа новым сотрудникам.

г) Упрощение процедуры инвентаризации программного обеспечения.

13. Для чего предназначен этап приоритизации уязвимостей в жизненном цикле управления уязвимостями?

а) Для составления графика обновления антивирусных баз.

б) Для ранжирования перечня уязвимостей на основе контекста бизнеса, угроз и безопасности.

в) Для технической верификации результатов сканирования и отсеивания ложных срабатываний.

г) Для финального подтверждения успешной установки исправлений.

14. Какая современная методология, упомянутая в тексте, используется для прогнозирования вероятности эксплуатации уязвимости?

а) OWASP;

б) ISO 27034;

в) EPSS;

г) SAMM.

15. Какова цель микросетевой сегментации в парадигме нулевого доверия?

а) Замена всех физических межсетевых экранов на программные.

б) Объединение разнородных подсетей в единый отказоустойчивый кластер.

в) Обеспечение высокой скорости соединения между центрами обработки данных.

г) Явное определение и контроль коммуникаций между отдельными рабочими нагрузками.

16. Что является ключевым отличием моделей авторизации на основе атрибутов (ABAC) от моделей на основе ролей (RBAC)?

а) Использование только биометрических данных для идентификации.

б) Обеспечение более гранулированного и контекстно-зависимого контроля доступа.

в) Полный отказ от использования паролей.

г) Применение исключительно в государственных информационных системах.

17. Какой принцип лежит в основе архитектуры программно-определяемого периметра (SDP)?

а) Максимальная пропускная способность каналов связи.

б) Принцип «скрытия» инфраструктуры от неавторизованных пользователей.

в) Приоритет беспроводных технологий доступа над проводными.

г) Использование только открытых криптографических алгоритмов.

18. На каком этапе управления исправлениями происходит формальное утверждение плана развертывания на основе результатов тестирования и оценки рисков?

а) Оценка и приоритизация.

б) Утверждение.

в) Верификация и откат.

г) Документирование.

19. Какое требование предъявляет принцип непрерывности и эшелонированности защиты?

а) Применение средств защиты только от одного производителя.

б) Создание многоуровневой системы защиты с разнородными и взаимодополняющими мерами.

в) Обязательное использование биометрической аутентификации на всех рабочих местах.

г) Сокращение численности персонала, имеющего доступ к информационным системам.

20. Какую функцию выполняет учет и мониторинг в системе управления идентификацией и доступом?

а) Автоматическое создание и удаление учетных записей.

б) Обеспечение конфиденциальности передаваемых данных.

в) Неизменяемая регистрация событий доступа для анализа и расследования инцидентов.

г) Управление ролевой моделью и назначение привилегий.

Глава 7. ПОДХОД IT SERVICE MANAGEMENT (ITSM) И БИБЛИОТЕКИ ИТ-ИНФРАСТРУКТУРЫ (ITIL) В УПРАВЛЕНИИ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ

7.1. Сущность подхода IT Service Management (ITSM)

В современной научной и деловой литературе, посвященной вопросам цифровой трансформации, переход от технологически-ориентированного управления информационными технологиями к сервисно-ориентированной модели является одной из ключевых парадигм. Эта парадигма находит свое воплощение в подходе IT Service Management (ITSM) - управлении ИТ-услугами. Понимание сущности ITSM критически важно для анализа процессов управления цифровой инфраструктурой, поскольку данный подход определяет не просто набор правил эксплуатации оборудования, а философию взаимодействия ИТ-подразделения с бизнесом как поставщика ценности с потребителем.

Сущность ITSM раскрывается через фундаментальный сдвиг в восприятии роли информационных технологий внутри организации. В рамках классического (продуктового) подхода ИТ-отдел воспринимается как центр затрат, обслуживающий технические ресурсы: серверы, дисковые массивы, клиентские приложения. Критерием эффективности здесь выступает работоспособность техники как таковой. В свою очередь, ITSM же предлагает рассматривать ИТ не как набор технологий, а как совокупность услуг. Услуга в контексте ITSM — это средство передачи ценности потребителю, помогающее ему достичь определенных бизнес-целей без необходимости владеть сложной инфраструктурой и управлять связанными с ней рисками. Таким образом, фокус смещается с внутреннего устройства технологии на внешний результат для пользователя.

Данный подход базируется на процессной модели организации деятельности. В отличие от функционального управления, где сотрудники закреплены за узкими технологическими участками (администраторы сетей, баз данных и т.д.), процессная модель ITSM выстраивает сквозные цепочки действий, нацеленные на конкретный результат для потребителя. Это позволяет формализовать взаимодействие между различными техническими специалистами и подразделениями, делая работу ИТ предсказуемой и измеримой . Ключевым инструментом этой

измеримости выступают Соглашения об уровне услуг (SLA), которые фиксируют качественные и количественные параметры предоставления услуги, устанавливая тем самым прозрачную связь между ИТ и бизнесом (рис. 7.1).

Рис. 7.1. Структура методологии IT Service Management (ITSM)

Для более четкого понимания концептуальной разницы между традиционной эксплуатацией и сервисным подходом, их основные характеристики представлены в сравнительной табл. 7.1.

Ключевой особенностью современного этапа эволюции ITSM является расширение его методологической базы и интеграция с другими управленческими практиками. Если первоначально ITSM ассоциировался преимущественно с библиотекой ITIL (Information Technology Infrastructure Library), то сегодня он представляет собой эклектичную область знаний. Как отмечают эксперты, эффективное управление ИТ-услугами в условиях цифровой экономики требует синтеза подходов

из различных концепций, включая Lean IT (для устранения потерь и оптимизации потоков создания ценности), DevOps (для ускорения вывода изменений в продуктивную среду) и Agile Service Management (для повышения гибкости и адаптивности процессов). Это обогащает сущность ITSM, превращая его из статичного набора регламентов в динамичную систему, способную адаптироваться к быстрым изменениям бизнес-среды.

Таблица 7.1

Сравнительный анализ традиционного и сервисного подходов к управлению ИТ

Характеристика	Традиционный (технологический) подход	Сервисный подход (ITSM)
Объект управления	Технологии, оборудование, ПО	Услуги для бизнеса и пользователей
Цель	Обеспечить работоспособность техники	Обеспечить ценность для бизнеса, достижение бизнес-целей
Фокус внимания	Внутренний («что сломалось?»)	Внешний («как помочь пользователю?»)
Критерий успеха	Время простоя оборудования	Удовлетворенность пользователя, соблюдение SLA
Природа затрат	Непрозрачные, воспринимаются как «неизбежное зло»	Прозрачные, привязанные к объему и качеству потребленных услуг
Взаимодействие с бизнесом	Реактивное решение инцидентов (помощь при поломке)	Проактивное планирование и улучшение услуг под задачи бизнеса

Следствием такой эволюции стало появление концепции Enterprise Service Management (ESM), которая представляет собой естественное развитие идей ITSM за пределы ИТ-департамента. Сущность ESM заключается в применении проверенных сервисных практик (управление каталогом услуг, инцидентами, запросами, проблемами) к работе других функциональных подразделений компании: управление персоналом и кадровые службы, административно-хозяйственного отдела (АХО), юридической службы, финансового департамента. Логика этого расширения строится на том, что любой внутренний отдел, по сути, предоставляет услуги своим коллегам, и унификация подходов к управлению этими услугами позволяет создать единую прозрачную среду для всех сотрудников компании.

Практическая реализация подходов ITSM и ESM невозможна без соответствующего класса программного обеспечения - ITSM-систем. Современные платформы, такие как Naumen Service Desk, ITSM 365, Directum ESM, реализуют не только базовые процессы управления инцидентами и запросами, но и предоставляют инструменты для управления изменениями, конфигурациями (CMDB), уровнем услуг и портфелем проектов. Важной тенденцией 2024–2025 годов является консолидация функционала: заказчики стремятся уйти от «лоскутной автоматизации» с десятками разрозненных инструментов в пользу экосистемных решений, построенных на единой low-code платформе, что обеспечивает целостность данных и сквозную автоматизацию сквозь различные слои управления.[28]

Анализ сущности ITSM будет неполным без рассмотрения его целевых ориентиров. Внедрение сервисного подхода преследует несколько ключевых целей, которые можно структурировать в зависимости от уровня воздействия на организацию (табл. 7.2).[29]

Таким образом, сущность подхода IT Service Management выходит далеко за рамки простой автоматизации технической поддержки. Это комплексная управленческая дисциплина, которая переосмысливает роль ИТ в современной организации. Ее ядром является принцип создания ценности для бизнеса через формализацию и управление жизненным циклом услуг. Опираясь на лучшие практики (ITIL) и современные технологические платформы, ITSM трансформирует управление цифровой инфраструктурой, делая его прозрачным, измеримым и ориентированным на конечного потребителя.

[28] Корытко С.А. О новых подходах организации ИТ-инфраструктуры электросетевого комплекса в условиях цифровой трансформации / С.А. Корытко, Н.И. Лиманова. - Текст: непосредственный // Молодой ученый. - 2021. - № 5 (347). - С. 9-11. - URL: https://moluch.ru/archive/347/78059.

[29] ITSM тренды в 2025 году [Электронный ресурс]// Режим доступа: https://www.tadviser.ru/index.php (дата обращения: 02.02.2026).

Таблица 7.2

Целевые ориентиры внедрения ITSM в разрезе организационных уровней

Уровень управления	Ключевые цели внедрения ITSM	Ожидаемые результаты
Стратегический (бизнес-руководство)	Обеспечение соответствия ИТ-услуг стратегическим целям бизнеса; повышение прозрачности затрат на ИТ; управление ИТ-рисками	ИТ из центра затрат превращается в стратегического партнера; возможность обоснованного планирования ИТ-бюджета
Тактический (руководители ИТ)	Повышение эффективности использования ресурсов; стандартизация и автоматизация процессов; контроль качества услуг	Снижение операционных затрат; предсказуемость результатов; возможность объективной оценки работы подразделений
Операционный (пользователи/ сотрудники)	Повышение качества и доступности услуг; сокращение времени решения проблем; создание единого и удобного интерфейса для получения всех внутренних сервисов	Рост удовлетворенности сотрудников; снижение простоев в работе из-за технологических сбоев

В свою очередь, эволюция ITSM в сторону ESM свидетельствует об универсальности сервисных принципов и их применимости для управления всей совокупностью внутренних сервисов современного цифрового предприятия.

7.2. ITIL 4: ключевые практики и ценность для бизнеса

Выход версии ITIL 4 (Information Technology Infrastructure Library) ознаменовал собой концептуальный сдвиг в понимании управления ИТ-услугами (ITSM). Если предыдущие версии библиотеки фокусировались на процессах как на основных структурных элементах, то ITIL 4 вводит понятие «практики». Это изменение не является простой заменой терминологии, а отражает более комплексный и гибкий подход к управлению (рис. 7.2.).

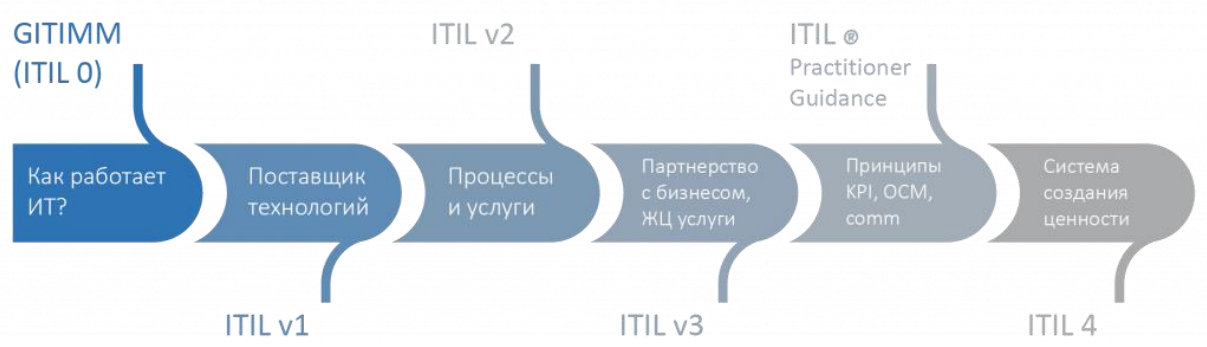

Рис. 7.2. Эволюция подхода ITIL 4 (Information Technology Infrastructure Library)

Практика в понимании ITIL 4 - это набор организационных ресурсов, предназначенных для выполнения работы или достижения цели. В отличие от процесса, который описывает последовательность действий, практика включает в себя не только процедуры, но и роли, компетенции, технологии, а также культуру и взаимодействие участников . Как отмечает архитектор ITIL 4 Роман Журавлев, прежняя жесткая привязка процессов к стадиям жизненного цикла (например, управление изменениями только к этапу «Преобразование услуг») создавала искусственные барьеры и провоцировала неверные организационные решения. В новой версии практики существуют вне временных рамок и могут быть задействованы в любой точке потока создания ценности .

Данный подход коррелирует с современными реалиями ведения бизнеса, где требуются гибкость и адаптивность. Общее количество практик в ITIL 4 составляет 34, и для удобства изучения они разделены на три группы:

- общие практики управления (general management practices);
- практики управления услугами (service management practices);
- практики технического управления (technical management practices).

Такая группировка носит дидактический характер и подчеркивает происхождение практик, однако в реальной деятельности организации они используются комплексно, в рамках сервисных потоков.

В этой связи с указанным выше, целесообразно рассмотреть наиболее значимые практики, составляющие ядро ITSM и претерпевшие существенные изменения в версии ITIL 4. Эти практики ориенти-

рованы на непосредственное взаимодействие с пользователями, обеспечение стабильности сервисов и их адаптацию к меняющимся потребностям бизнеса. К ним относятся управление инцидентами, управление проблемами, управление запросами на обслуживание и поддержка изменений (рис. 7.3).

Рис. 7.3. Основные направления ITIL 4 и взаимосвязи между ними

Управление инцидентами (Incident Management) сохраняет свою стратегическую цель: максимально быстрое восстановление нормального функционирования услуги и минимизация негативного влияния на бизнес. Однако ITIL 4 расширяет инструментарий данной практики. Традиционная модель эскалации дополняется подходом, известным как «роение». В отличие от жесткой иерархической передачи инцидента между линиями поддержки, «роение» предполагает, что к решению сложной проблемы привлекается кросс-функциональная группа специалистов, которые совместно работают над инцидентом до его полного разрешения. Такой подход признает, что в сложных распределенных системах корень проблемы может находиться на стыке различных технологий и компетенций. Кроме того, в ITIL 4 приоритизация инцидентов рассматривается не только с точки зрения бизнес-приоритетов, но и в контексте загрузки конкретных исполнителей и ограничения незавершенной работы (принципы Kanban).

Управление проблемами (Problem Management) смещает фокус с простого поиска корневых причин инцидентов на более глубокий анализ всех аспектов, которые могут стать источником сбоев. В ITIL 4 для анализа используется модель «4P»: проблемы могут крыться в продуктах (Products), процессах (Processes), людях (People) и поставщиках

(Providers/Partners). Такой подход позволяет выявлять системные недостатки, например, некорректно составленные договоры с поставщиками или устаревшие политики компании, которые провоцируют возникновение инцидентов. Практика также признает концепцию «технического долга», возникающую при длительном использовании обходных путей решения проблем.

Практика управления запросами на обслуживание (Service Request Management) в ITIL 4 делает акцент на максимальной стандартизации и автоматизации. Запросы на обслуживание, в отличие от инцидентов, являются стандартизированными действиями, заранее согласованными с бизнесом. Новая версия методологии подчеркивает, что эффективность этой практики напрямую влияет на общую удовлетворенность пользователей, поскольку именно здесь происходит основное взаимодействие потребителя с поставщиком услуг по поводу повседневных потребностей. Автоматизация исполнения запросов рассматривается как приоритетное направление для высвобождения ресурсов.

Наиболее существенные изменения затронули практику поддержки изменений (Change Enablement), которая в предыдущих версиях называлась управлением изменениями. ITIL 4 отходит от модели, где все изменения стремятся объединить в единый план, и признает сосуществование разных потоков изменений. Это связано с интеграцией методологии с подходами DevOps и Agile.

Высокоавтоматизированные изменения в программном обеспечении, проходящие через конвейер CI/CD (Continuous Integration/Continuous Delivery), обрабатываются иначе, чем изменения в инфраструктуре, требующие более традиционного подхода. Основная цель практики смещается с контроля за проведением изменений на поддержку их успешного внедрения при сохранении баланса между скоростью, надежностью и уровнем риска . Для наглядного сравнения эволюции ключевых практик данные представлены в табл. 7.3.

Ключевым понятием, пронизывающим все элементы ITIL 4, является «ценность» (value). Методология определяет ценность как «воспринимаемые преимущества, полезность и важность чего-либо». Ценность не может быть абстрактно назначена поставщиком услуги; она всегда определяется получателем - потребителем, пользователем или

заказчиком. Это смещает фокус с внутренних показателей эффективности ИТ-подразделения (таких как время решения инцидента или процент доступности сервиса) на внешние бизнес-результаты. В основе создания ценности в ITIL 4 лежит модель «совместного творчества» (value co-creation). Ценность не передается от поставщика к потребителю как готовый продукт, а создается совместно в процессе потребления услуги. Например, облачная платформа как таковая не имеет ценности без данных, которые загружает пользователь, и без тех бизнес-задач, которые он с ее помощью решает.

Таблица 7.3

Сравнительный анализ ключевых практик управления услугами в ITIL 4

Практика	Традиционный подход (ITIL v3/акцент)	Ключевые нововведения ITIL 4
Управление инцидентами	Восстановление услуги, иерархическая эскалация по линиям поддержки.	Внедрение «роения» (кросс-функциональных команд); учет ограничений незавершенного производства; гибкая приоритизация.
Управление проблемами	Поиск корневой причины инцидентов в технической части.	Анализ по модели «4P» (продукты, процессы, люди, поставщики); управление техническим долгом.
Управление запросами	Выполнение стандартных запросов пользователей.	Максимальная стандартизация и приоритет автоматизации исполнения; прямое влияние на пользовательский опыт.
Поддержка изменений	Контроль и согласование всех изменений через CAB (Change Advisory Board).	Разделение потоков (CI/CD и традиционные); поддержка внедрения; баланс скорости и риска; «безопасные эксперименты».

Таким образом, потребитель становится активным участником создания ценности. Это требует от организации-поставщика не только понимания технических характеристик услуги, но и глубокого погружения в контекст деятельности клиента.

Ценность для бизнеса от внедрения практик ITIL 4 может быть структурирована по нескольким уровням. На операционном уровне она выражается в повышении стабильности сервисов и предсказуемости ИТ-ландшафта, что снижает простои и связанные с ними финансовые потери. На тактическом уровне - в повышении эффективности использования ресурсов и оптимизации затрат за счет стандартизации и

автоматизации. Однако главным является стратегический уровень, где ценность проявляется в способности ИТ быстро адаптироваться к изменениям рынка и становиться драйвером цифровой трансформации бизнеса. Пять элементов ценности (видение, согласование, рычаги, уникальность, исполнение) должны работать в совокупности, чтобы организация могла не просто оказывать услуги, а создавать устойчивое конкурентное преимущество.

Достижение ощутимой бизнес-ценности невозможно без соблюдения ряда фундаментальных принципов, которые в ITIL 4 оформлены как руководящие принципы. Они являются универсальными рекомендациями, применимыми к любой организации и любой ситуации, направляющими действия при принятии решений.

Семь принципов ITIL 4:

1. Фокусируйтесь на ценности (Focus on value).
2. Начинайте с того, где вы находитесь (Start where you are).
3. Двигайтесь вперед итеративно, используя обратную связь (Progress iteratively with feedback).
4. Сотрудничайте и повышайте прозрачность (Collaborate and promote visibility).
5. Думайте и работайте целостно (Think and work holistically).
6. Делайте просто и практично (Keep it simple and practical).
7. Оптимизируйте и автоматизируйте (Optimize and automate).

Принцип «Фокусируйтесь на ценности» является основополагающим. Он требует, чтобы любое действие, любая инициатива или процесс оценивались с точки зрения того, какую ценность они создают для стейкхолдеров. На практике это означает необходимость ставить под сомнение устоявшиеся практики, например, формирование многостраничных отчетов, которые никто не читает. Ценность отчета определяется не его объемом, а тем, помогает ли он менеджменту принимать более взвешенные решения .

Принцип «Начинайте с того, где вы находитесь» призывает не игнорировать текущее состояние и накопленный опыт. Вместо того чтобы каждый раз начинать с «чистого листа», внедряя идеальные, но оторванные от реальности процессы, организации следует провести инвентаризацию имеющихся активов, компетенций и наработок. Часто в существующих практиках уже есть элементы, которые можно ис-

пользовать как основу для улучшений. Полный демонтаж старой системы и построение новой на пустом месте несет в себе высокие риски и не всегда оправдан.

Принцип «Двигайтесь вперед итеративно, используя обратную связь» напрямую перекликается с гибкими методологиями разработки. Крупные преобразования должны разбиваться на небольшие, управляемые этапы. После завершения каждого этапа необходимо собирать обратную связь от всех заинтересованных сторон и на ее основе корректировать дальнейшие действия. Это позволяет снизить неопределенность, быстрее реагировать на изменение требований и избежать ситуации, когда результат масштабного проекта никому не нужен.

Принципы «Сотрудничайте и повышайте прозрачность» и «Думайте и работайте целостно» направлены на преодоление разобщенности между подразделениями. Ценность создается не в изоляции, а на стыке компетенций. Разрозненная работа ИТ, маркетинга, отдела продаж и производства приводит к созданию сервисов, не отвечающих реальным потребностям клиентов. Прозрачность означает, что все участники процесса понимают цели друг друга и имеют доступ к информации, необходимой для эффективной работы. Целостный подход требует рассматривать услугу не как набор отдельных компонентов, а как единую систему, где изменение в одной части неизбежно влияет на все остальные .

Наконец, принципы «Делайте просто и практично» и «Оптимизируйте и автоматизируйте» завершают логическую цепочку создания ценности. Сложные, многоступенчатые процедуры и избыточная документация убивают ценность, замедляя выполнение задач и демотивируя сотрудников. Необходимо искать простые и понятные способы достижения целей. И только после того, как процесс отлажен и оптимизирован, его целесообразно автоматизировать. Автоматизация неэффективного процесса лишь позволяет быстрее получать неверный или бесполезный результат . Взаимосвязь этих принципов и их вклад в создание ценности для бизнеса обобщены в табл. 7.4.

Таблица 7.4

Вклад руководящих принципов ITIL 4 в создание бизнес-ценности

Принцип	Суть принципа	Проявление ценности для бизнеса
Фокусируйтесь на ценности	Все действия должны иметь измеримый вклад в результат для стейкхолдеров.	Исключение бесполезной работы; ориентация ИТ на реальные бизнес-потребности, а не на абстрактные метрики.
Начинайте с того, где вы находитесь	Анализ и использование текущего состояния, ресурсов и опыта.	Снижение затрат и рисков за счет отказа от повторной разработки; ускорение внедрения улучшений.
Двигайтесь итеративно	Разбиение крупных задач на мелкие этапы со сбором обратной связи.	Повышение адаптивности к изменениям; снижение риска крупных провалов; возможность быстрой коррекции курса.
Сотрудничайте и будьте прозрачны	Межфункциональное взаимодействие и открытость информации.	Преодоление разобщенности; сокращение времени решения проблем за счет коллективной экспертизы; повышение доверия между бизнесом и ИТ.
Думайте и работайте целостно	Рассмотрение сервисов и организации как единой взаимосвязанной системы.	Принятие более взвешенных решений; устранение узких мест и оптимизация системы в целом, а не её частей.

Таким образом, ITIL 4 представляет собой не просто обновленный сборник инструкций по управлению ИТ, а целостную философию управления, ориентированную на создание ценности. Переход от жесткой процессной модели к гибкой системе практик позволяет организациям выстраивать работу более эффективно, интегрируя современные подходы DevOps, Agile и Lean.

Ключевые практики управления услугами (инцидентами, проблемами, запросами, изменениями) получили новое развитие, вобрав в себя опыт, накопленный за десятилетия развития ITSM, и адаптировав его к реалиям цифровой экономики. Следование семи руководящим принципам гарантирует, что любые усилия по трансформации управления услугами будут не самоцелью, а инструментом достижения измеримых бизнес-результатов, создавая ценность как для организации-поставщика, так и для её клиентов.

7.3. Управление услугами: каталог услуг, портал самообслуживания

В парадигме ITSM, рассматривающей ИТ-подразделение как сервис-ориентированную организацию, ключевыми элементами взаимодействия с потребителями становятся каталог услуг и портал самообслуживания. Эти инструменты обеспечивают формализацию предложения, прозрачность обязательств и автоматизацию доступа к сервисам, что напрямую влияет на удовлетворенность пользователей и эффективность использования ресурсов. Методология ITIL определяет четкие границы и взаимосвязи данных понятий в рамках управления цифровой инфраструктурой.

Каталог услуг представляет собой структурированный и централизованный перечень всех активных ИТ-услуг, доступных пользователям. В терминологии ITIL каталог услуг является частью более широкого понятия - портфеля услуг (Service Portfolio), который включает в себя также концепции сервисов, находящихся в разработке (конвейер услуг), и выведенных из эксплуатации. Основная функция каталога - коммуникационная и информационная: он отвечает на вопросы пользователя о том, какие сервисы существуют и как их получить, избегая избыточной технической детализации.

Структура описания услуги в каталоге должна быть стандартизирована для однозначного понимания как потребителем, так и поставщиком. Анализ лучших практик позволяет выделить следующие обязательные атрибуты описания сервиса, представленные в табл. 7.5.

Эффективность каталога напрямую зависит от подхода к его наполнению. Вместо простого перечня ИТ-активностей (например, «настройка доступа к серверу») целесообразно использовать концепцию продуктового подхода, при котором услуга описывается как готовый к употреблению «сервисный пакет» (service package).[30] Такой пакет содержит не только описание, но и встроенные бизнес-процессы, шаблоны выполнения и критерии приемки.

[30] 2025: 6 трендов мирового рынка ИТ-инфраструктур [Электронный ресурс]// Режим доступа: https://www.tadviser.ru/index.php (дата обращения: 16.01.2026).

Таблица 7.5

Ключевые атрибуты описания услуги в каталоге (Service Catalog)

Атрибут	Описание	Назначение
Наименование и категория	Интуитивно понятное название, классификация (например, сетевые, прикладные услуги)	Идентификация и навигация
Описание и целевая аудитория	Четкое изложение состава услуги, ее бизнес-цели и группы пользователей, для которых она доступна	Формирование понимания ценности и применимости сервиса
Условия предоставления (SLA)	Согласованные показатели времени реакции и решения (регламенты), доступность сервиса (в процентах)	Установление измеримых обязательств поставщика
Процедура запроса	Пошаговая инструкция, необходимые формы ввода, информация о согласованиях	Обеспечение простоты и стандартизации заказа
Стоимость и порядок оплаты	Информация о цене услуги или модели внутренних расчетов (chargeback/showback)	Бюджетный контроль и прозрачность финансовых потоков
Ограничения и требования	Технические и организационные предпосылки для оказания услуги, а также зоны ответственности, не покрываемые сервисом	Управление ожиданиями и предотвращение конфликтов
Ответственный и поддержка	Контактные данные владельца услуги или службы поддержки	Обеспечение обратной связи и эскалации

Данное обстоятельство позволяет трансформировать абстрактную потребность пользователя в конкретный, измеримый и повторяемый результат.

Вторым ключевым элементом сервисной модели является портал самообслуживания (Self-Service Portal). Если каталог носит информационный характер, то портал - это интерактивная среда, предоставляющая пользователю интерфейс для непосредственного взаимодействия с поставщиком услуг . Портал самообслуживания выступает единой точкой входа (Single Point of Contact) для всех категорий потребителей - сотрудников, студентов или внешних клиентов - и обеспечивает доступ к функциональным возможностям ИТ-инфраструктуры в соответствии с их ролью и полномочиями. Архитектурно он относится к уровню пользовательского опыта (User Experience Layer) и может быть реализован в виде веб-приложения, обеспечивающего доступ к базе знаний, регистрацию запросов и отслеживание их статуса.

Ключевое различие между двумя рассматриваемыми понятиями заключается в их функциональной роли. Каталог услуг является структурированным источником данных о сервисах, в то время как портал самообслуживания - это инструмент для совершения действий на основе этих данных. Для конечного пользователя портал предоставляет «витрину» каталога, интегрированную с механизмами выполнения заказов. Как показано в табл. 7.6, эти понятия дополняют друг друга, образуя единую экосистему предоставления услуг.

Таблица 7.6

Сравнительная характеристика каталога услуг и портала самообслуживания

Характеристика	Каталог услуг (Service Catalog)	Портал самообслуживания (Self-Service Portal)
Основная функция	Информирование и структурирование предложения	Взаимодействие и выполнение транзакций
Содержимое	Детальное описание услуг, их атрибуты, условия предоставления	Интерфейсы для заказа услуг, поиска в базе знаний, отслеживания статусов заявок
Характер взаимодействия	Пассивный (ознакомление).	Активный (регистрация обращения, поиск решения)
Целевая аудитория	Пользователи и заказчики услуг, ИТ-персонал (технический каталог)	Конечные пользователи (сотрудники, клиенты)
Бизнес-ценность	Прозрачность, стандартизация описания, управление ожиданиями	Снижение нагрузки на сервисную службу, ускорение предоставления услуг, повышение удовлетворенности пользователей

Портал самообслуживания в современной цифровой инфраструктуре выходит за рамки простого инструмента регистрации инцидентов. Он становится платформой для автоматизации сквозных бизнес-процессов (Enterprise Service Management), объединяющей работу различных подразделений — от ИТ до HR и административно-хозяйственного отдела . Например, процесс оформления командировки может быть реализован на портале как комплексная услуга, инициирующая цепочку взаимосвязанных задач в ИТ-системах, отделах кадров и бухгалтерии, что минимизирует ручной труд и исключает потерю информации на стыках функциональных подразделений .

С методологической точки зрения, для успешной реализации концепции управления услугами необходима синергия этих двух элементов. Каталог услуг, разработанный без привязки к порталу, рискует стать статичным документом, не влияющим на операционную деятельность. В свою очередь, портал без структурированного и актуального каталога превращается в «черный ящик» для приема заявок, неспособный обеспечить стандартизацию и предсказуемость исполнения. Только их интеграция в рамках ITSM-системы, поддерживающей автоматизацию workflow и управление знаниями, позволяет реализовать главный принцип ITIL 4 - фокус на создании ценности для бизнеса путем эффективного управления услугами цифровой инфраструктуры

7.4. Процессы управления инцидентами, запросами на обслуживание, изменениями и релизами

В парадигме ITSM и библиотеки ITIL операционная деятельность ИТ-подразделения структурируется вокруг ряда взаимосвязанных процессов, обеспечивающих стабильность, контролируемость и адаптивность цифровой инфраструктуры.

Четыре ключевых процесса - управление инцидентами, управление запросами на обслуживание, управление изменениями и управление релизами - формируют каркас повседневной деятельности сервисного провайдера. Каждый из них решает уникальные задачи, имеет четко определенные границы и цели, однако их эффективная реализация возможна лишь при условии тесной интеграции и понимания различий между ними.

Управление инцидентами (Incident Management) является, пожалуй, наиболее заметным для пользователей процессом, поскольку его основная цель - максимально быстрое восстановление нормальной работы сервиса и минимизация негативного влияния на бизнес. Согласно определению ITIL, инцидент, представляет собой любое незапланированное событие, которое приводит к перерыву в предоставлении услуги или снижению ее качества. Это может быть как полная недоступность критичного приложения, так и некорректная работа отдельного компонента, например, принтера или сетевого порта. Ключевая характеристика инцидента - его экстренный характер, требующий немедленной реакции.

Жизненный цикл инцидента представляет собой строгую последовательность этапов. Он начинается с обнаружения и регистрации, которые могут инициироваться как самим пользователем (через портал самообслуживания, электронную почту или телефонный звонок), так и системами мониторинга, способными автоматически создавать заявку в Service Desk. Регистрация фиксирует не только факт сбоя, но и все сопутствующие данные, необходимые для дальнейшей работы.

Следующим этапом выступают категоризация и приоритизация, определяющие, к какой области инфраструктуры относится сбой и насколько срочно его необходимо устранить. Приоритет инцидента, как правило, вычисляется на основе его влияния на бизнес-процессы и срочности, зафиксированных в Соглашении об уровне услуг (SLA). Автоматизация этого этапа позволяет избежать ручных ошибок и гарантировать, что наиболее критичные сбои будут обработаны в первую очередь.

Далее следует диагностика и исследование, в ходе которых специалисты службы поддержки (первой, второй или третьей линии) пытаются определить причину сбоя и найти способ его устранения. Важную роль здесь играет интеграция с базой знаний (Known Error Database), где могут храниться описания ранее возникавших проблем и способы их обхода. Если найти быстрое решение не удается, запускается механизм эскалации - передачи инцидента более компетентным специалистам (функциональная эскалация) или информирования руководства о потенциальном нарушении сроков SLA (иерархическая эскалация).

Фаза решения и восстановления завершается успешным устранением сбоя, после чего инцидент переходит в стадию закрытия, требующую подтверждения со стороны пользователя, что проблема действительно решена. Методология ITIL подчеркивает, что тщательное документирование всех этих шагов позволяет не только обеспечить прозрачность работы, но и накапливать данные для последующего анализа и предотвращения подобных ситуаций в будущем .

В отличие от инцидентов, которые являются следствием сбоев, управление запросами на обслуживание (Service Request Management) ориентировано на обработку структурированных, предварительно авторизованных обращений пользователей. Запрос на об-

служивание (Service Request) — это формализованная просьба пользователя о предоставлении доступа к услуге, получении информации, консультации или выполнении стандартного изменения, которое не несет в себе высоких рисков и не требует экстренного согласования. Примерами таких запросов являются: сброс пароля, предоставление прав доступа к общей папке, установка типового программного обеспечения на новый компьютер или замена неисправной мыши.

Ключевая особенность этого процесса заключается в его предсказуемости и повторяемости. Поскольку большинство запросов являются рутинными, для них разрабатываются стандартные модели исполнения (Standard Fulfillment Models). Такая модель включает в себя четкую последовательность шагов, сроки выполнения, ответственных исполнителей и возможные пути эскалации. Пользователи, как правило, взаимодействуют с этим процессом через портал самообслуживания (Service Catalog), где они могут выбрать необходимую услугу из каталога и заполнить веб-форму. Это автоматизирует начальные этапы: регистрацию, категоризацию и даже назначение исполнителя. Основная ценность управления запросами на обслуживание заключается в разгрузке более дорогостоящих процессов управления инцидентами и изменениями за счет централизованного и быстрого предоставления стандартных услуг, что напрямую влияет на продуктивность работы пользователей и их удовлетворенность.

В то время как управление инцидентами направлено на устранение симптомов, а управление запросами - на предоставление доступа к услугам, процесс управления изменениями (Change Management) берет на себя функцию контроля над модификацией ИТ-инфраструктуры и сервисов. Изменением считается любое добавление, модификация или удаление компонента, которое может прямо или косвенно повлиять на предоставление ИТ-услуг . Цель процесса - не заблокировать изменения, а обеспечить их реализацию с минимальными рисками и наименьшим негативным воздействием на бизнес, используя стандартизированные методы и процедуры.

Для эффективного управления все изменения классифицируются по трем основным типам, каждый из которых имеет свой путь прохождения и уровень контроля (табл. 7.7).

Таблица 7.7

Типология изменений в процессе ITIL

Тип изменения	Характеристика	Пример	Процедура согласования
Стандартное изменение (Standard Change)	Предварительно авторизованное, низкорисковое, повторяющееся изменение с четко прописанной процедурой выполнения.	Выдача прав доступа, установка обновлений антивирусных баз, замена монитора.	Не требует созыва Консультативного совета по изменениям (CAB). Выполняется по запросу через Service Request.
Нестандартное изменение (Non-Standard Change)	Новое изменение с непредсказуемыми рисками, затратами и длительностью. Требует полной оценки и планирования.	Модернизация серверного оборудования, смена архитектуры сети, внедрение нового программного обеспечения.	Требует утверждения на Консультативном совете по изменениям (CAB) после детального анализа.
Экстренное изменение (Emergency Change)	Изменение, которое необходимо внедрить максимально быстро для устранения критического инцидента или серьезной угрозы безопасности.	Внесение изменений в конфигурацию межсетевого экрана для отражения атаки, установка критического «горячего» патча.	Согласовывается в ускоренном порядке, часто на экстренном заседании CAB (ECAB) или в устной форме, с последующей документальной фиксацией.

Жизненный цикл изменения начинается с создания Запроса на изменение (Request for Change, RFC). RFC регистрируется, после чего проходит этапы фильтрации и оценки. В ходе оценки менеджер изменений и эксперты должны ответить на ряд ключевых вопросов: какова причина изменения, какова ожидаемая выгода, какие риски с ним связаны, какие ресурсы потребуются, и каково его влияние на инфраструктуру и бизнес-процессы.

На основе оценки влияния и рисков принимается решение об утверждении (авторизации) изменения. Для нестандартных изменений эту функцию выполняет Консультативный совет по изменениям (Change Advisory Board, CAB) - группа людей, представляющих интересы как ИТ-подразделения, так и бизнеса . После утверждения осуществляется координация реализации, в ходе которой важно сверяться

с графиком изменений (Change Schedule), чтобы избежать конфликтов с другими работами и заранее оповестить заинтересованные стороны о возможных простоях (Projected Service Outage, PSO). Завершается процесс обзором и закрытием, где оценивается успешность проведенных работ и достижение поставленных целей.

Наконец, процесс управления релизами (Release Management) отвечает за стратегическое планирование, тестирование и ввод в эксплуатацию изменений, объединенных в одно целое - релиз. Под ним понимается совокупность одного или нескольких изменений, которые проходят через единый жизненный цикл как единое целое. В то время как управление изменениями фокусируется на контроле рисков и авторизации каждого отдельного изменения, управление релизами берет на себя «логистику» и техническую реализацию, особенно в части развертывания нового или обновленного программного и аппаратного обеспечения.

Этот процесс обеспечивает «мостик перехода» между разработкой и эксплуатацией, гарантируя, что новые возможности и исправления будут переданы пользователям упорядоченно и с надлежащим качеством. Жизненный цикл релиза включает в себя несколько фаз. На этапе планирования формируется «дорожная карта» релиза: собираются требования, оцениваются трудозатраты, определяются сроки и контрольные точки.

Далее следует этап сборки и тестирования, где изменения реализуются в контролируемой среде, после чего проходят всестороннюю проверку. Важно подчеркнуть, что тестирование должно проводиться в среде, максимально приближенной к продуктивной, чтобы выявить возможные ошибки до того, как они повлияют на пользователей. Фаза развертывания включает в себя непосредственно установку релиза в продуктивную среду, которая может проводиться поэтапно, чтобы минимизировать риски. Наконец, на этапе контроля качества производится оценка успешности релиза, сбор обратной связи от пользователей и анализ возникших инцидентов, что служит основой для планирования будущих улучшений.

Критически важным аспектом ITSM является понимание того, что описанные процессы не существуют изолированно. Напротив, они находятся в тесном взаимодействии, формируя единую систему управ-

ления услугами. Так, множество однотипных инцидентов (сбоев) могут стать основанием для создания проблемы, цель которой состоит в устранение первопричины. В свою очередь, решение проблемы практически всегда требует инициирования изменения. Это изменение, например замена неисправного сетевого коммутатора, будет зарегистрировано как RFC и пройдет процесс управления изменениями. После утверждения оно может быть включено в ближайший релиз по обновлению сетевой инфраструктуры (табл. 7.8).

Таблица 7.8

Сравнительный анализ процессов и их взаимосвязь

Процесс	Объект управления	Основная цель	Триггер для создания ...
Управление инцидентами	Инцидент (сбой)	Восстановить сервис как можно быстрее.	... проблемы (при повторении или высокой серьезности).
Управление запросами	Запрос на обслуживание	Предоставить стандартную услугу по запросу пользователя.	... стандартного изменения (если оно входит в модель обслуживания).
Управление проблемами	Проблема (причина)	Выявить и устранить корневую причину инцидентов.	... запроса на изменение (RFC) для устранения найденной причины.
Управление изменениями	Изменение	Реализовать изменение контролируемо.	... задания на реализацию в рамках релиза (или релизной единицы).
Управление релизами	Релиз	Внедрить набор изменений в эксплуатацию как единое целое.	... дальнейшего улучшения услуги на основе полученной обратной связи.

Таким образом, системное применение данных процессов позволяет превратить управление ИТ-инфраструктурой из набора хаотичных действий по «тушению пожаров» в предсказуемую и управляемую деятельность. Инциденты обрабатываются оперативно, стандартные запросы - эффективно, изменения - под строгим контролем рисков, а их внедрение происходит планово, в рамках целостных релизов. Такая архитектура процессов является фундаментом для построения зрелой и надежной цифровой инфраструктуры современного предприятия.

7.5. Интеграция DevOps-культуры и Agile-подходов в традиционный ITSM

Эволюция информационных технологий от вспомогательной функции к стратегическому драйверу бизнеса обусловила необходимость пересмотра устоявшихся подходов к управлению ИТ-услугами. Традиционные фреймворки, такие как ITIL (Information Technology Infrastructure Library), исторически ориентированные на стабильность, контроль и предсказуемость, столкнулись с вызовом со стороны Agile-методологий и DevOps-культуры, требующих скорости, гибкости и непрерывной поставки ценности. Долгое время в профессиональном сообществе существовала дихотомия: с одной стороны, необходимость обеспечения надежности и управления рисками (ITSM), с другой - потребность в быстрой разработке и внедрении изменений (Agile/DevOps). Однако, как показывает практика и развитие самого фреймворка ITIL, противопоставление этих подходов является неконструктивным. Современный вектор развития ИТ-менеджмента лежит в плоскости их глубокой интеграции, где DevOps выступает не как замена, а как эволюционное дополнение и инструмент реализации гибких процессов в структуре ITSM.

Ключевым событием, легитимизировавшим и концептуально оформившим этот синтез, стал выпуск версии ITIL 4 в 2019 году. Если предыдущие версии фреймворка (особенно ITIL v3) предлагали достаточно жесткую модель жизненного цикла услуги, то ITIL 4 совершил фундаментальный сдвиг в сторону интеграции с современными практиками. Он позиционирует себя как единая система, впитывающая и согласующая принципы Agile, Lean и DevOps, а не конкурирующая с ними . В основе этой философии лежит отказ от линейных процессов в пользу гибкой и адаптивной модели, способной функционировать в условиях высокой неопределенности и скорости изменений, характерных для цифровой экономики. Как отмечается в исследовании, посвященном ценности различных методологий, в сообществе сформировался консенсус, описываемый формулой (7.1):

$$DevOps = Agile\ (гибкая\ р\ азработка) + Lean\ (бережливое\ производство) + ITSM\ (управление\ услугами) \quad (7.1)$$

Центральным элементом этой интеграционной модели в ITIL 4 выступает Система ценности сервиса (Service Value System - SVS).

В отличие от линейного жизненного цикла, SVS представляет собой комплексную операционную модель, демонстрирующую, как различные компоненты и виды деятельности организации взаимодействуют для создания ценности. SVS включает в себя руководящие принципы, управление, цепочку создания ценности, практики и постоянное улучшение. Ключевым звеном здесь является Цепочка создания ценности (Service Value Chain), представляющая собой набор гибких видов деятельности, которые могут комбинироваться различными способами для создания, поставки и постоянного их улучшения. Данный аспект напрямую коррелирует с DevOps-подходом к организации потока создания ценности от возникновения идеи до ее эксплуатации.

Для наглядного понимания концептуальных изменений, вызванных интеграцией гибких методологий, целесообразно сравнить подходы традиционного ITSM (в версии ITIL v3) и современного ITSM (ITIL 4) (табл. 7.9).

Интеграция DevOps в ITSM оказывает наиболее ощутимое влияние на ключевые процессы управления инцидентами, проблемами и изменениями. Эмпирические исследования подтверждают, что практики, привнесенные DevOps-культурой, напрямую повышают эффективность этих процессов. Автоматизированный мониторинг приложений и инфраструктуры и короткие петли обратной связи позволяют обнаруживать сбои и проблемы на самых ранних этапах, а иногда и превентивно, до того, как они повлияют на пользователей. Практика непрерывной интеграции способствует ускорению устранения инцидентов за счет более частого и мелкого внесения изменений в код, что упрощает поиск источника ошибки.

Таблица 7.9

Сравнение традиционного (ITIL v3) и современного (ITIL 4) подходов к управлению ИТ-услугами

Характеристика	Традиционный ITSM (ITIL v3)	Современный ITSM (ITIL 4) с интеграцией Agile/DevOps
Базовый принцип	Управление жизненным циклом услуги	Создание ценности в системе (Service Value System)
Структура	Линейная: 5 этапов жизненного цикла	Модульная: 34 гибкие практики, объединенные ценностью
Фокус управления	Процессы и функции	Сквозные потоки создания ценности (value streams)
Отношение к изменениям	Контроль и минимизация рисков (Change Management как процесс)	Автоматизация, делегирование и ускорение (Change Control как практика)
Культура взаимодействия	Четкое разделение ролей (Dev, Ops, поддержка)	Кросс-функциональные команды и совместная ответственность

Особенно показательна трансформация управления изменениями. Традиционная модель с громоздкими заседаниями Консультативного совета по изменениям (Change Advisory Board - CAB) становится тормозом в среде, требующей высокой частоты релизов. Современный подход, закрепленный в ITIL 4, рекомендует отказываться от централизованного утверждения всех изменений в пользу автоматизированных и децентрализованных механизмов контроля. Автоматизированное развертывание (Automated Deployment) с использованием практик «инфраструктура как код» (Infrastructure as Code) и современных инструментов CI/CD позволяет ускорять авторизацию, координацию и внедрение стандартных и низкорисковых изменений, оставляя за CAB только управление наиболее сложными и высокорисковыми изменениями.

Успешная интеграция невозможна без изменения организационной культуры. Традиционные ITSM-команды и DevOps-команды долгое время существовали в парадигме «мы и они», что приводило к созданию организационных барьеров . Преодоление этих барьеров требует внедрения генеративной (ориентированной на высокую производительность) культуры по модели Уэструма, где приветствуется прозрачность, совместная ответственность за результат, а неудачи рассматриваются как возможности для обучения, а не повод для поиска

виновных . Такой подход реализуется через совместное использование инструментов, общие метрики (например, время восстановления сервиса и скорость изменения ценится одинаково высоко) и формирование кросс-функциональных команд, отвечающих за продукт на всех этапах его жизненного цикла. Важным инструментом здесь становится картографирование потока создания ценности, позволяющее всем участникам процесса - от разработки до эксплуатации и бизнеса - увидеть узкие места и совместно их устранять.

Таким образом, интеграция DevOps-культуры и Agile-подходов в ITSM представляет собой не механическое добавление новых инструментов к старым процессам, а глубокую трансформацию философии управления ИТ. Она знаменует переход от управления функциями и процессами к управлению сквозными потоками создания ценности для бизнеса. Ключевые эффекты от такой интеграции могут быть систематизированы по ряду направлений, представленных в табл.7.10.

Интеграция DevOps-культуры и Agile-подходов в традиционный ITSM, реализованная в ITIL 4, позволяет преодолеть фундаментальное противоречие между скоростью и стабильностью. Она создает основу для построения высокопроизводительных ИТ-организаций, способных одновременно обеспечивать надежность и устойчивость сервисов, а также быстро адаптироваться к меняющимся требованиям рынка, что является критическим фактором успеха в эпоху цифровой трансформации.

Подводя итог сказанному выше, следует отметить, что переход от технологически-ориентированной модели эксплуатации к сервисно-ориентированному управлению цифровой инфраструктурой является не просто сменой терминологии, а глубокой трансформацией философии взаимодействия ИТ-подразделения с бизнесом. В рамках этой парадигмы информационные технологии перестают восприниматься как центр затрат и приобретают статус стратегического партнера, деятельность которого оценивается через призму создаваемой для потребителя ценности и вклада в достижение бизнес-целей.

Таблица 7.10

Эффекты интеграции DevOps и Agile в ITSM по направлениям деятельности

Направление	Практики интеграции	Достигаемый эффект
Управление инцидентами и проблемами	Автоматизированный мониторинг, петли обратной связи, вместо эскалации	Ускорение обнаружения и разрешения инцидентов, снижение MTTR и переход к проактивному выявлению проблем
Управление изменениями и релизами	Непрерывная интеграция и доставка (CI/CD), автоматизация развертывания, «инфраструктура как код»	Увеличение частоты и надежности релизов, снижение риска внедрения за счет малых изменений
Организационная культура и взаимодействие	Кросс-функциональные команды, общие метрики и цели (OKR и KPI)	Разрушение барьеров между Dev и Ops, повышение доверия и прозрачности, рост вовлеченности и инновационного потенциала команд
Стратегия и управление ценностью	Фокус на потоки создания ценности, ориентация на бизнес-результаты, а не на активность внутри процесса	Повышение эффективности инвестиций в ИТ, улучшение качества обслуживания пользователей, обеспечение соответствия ИТ-стратегии целям бизнеса

Ключевым здесь является то, что выход версии ITIL 4 ознаменовал собой концептуальный отход от жесткой процессной модели к более гибкой и адаптивной системе практик. Данная эволюция отражает объективную потребность организаций в интеграции современных управленческих подходов, включая Agile, Lean и DevOps. Система ценности сервиса (Service Value System) и семь руководящих принципов ITIL 4 создают методологическую основу для преодоления традиционного противоречия между требованиями стабильности и надежности инфраструктуры, с одной стороны, и необходимостью скорости внедрения изменений и адаптивности к динамике рынка, с другой.

Неотъемлемым элементом реализации сервисной модели выступают инструменты взаимодействия с потребителем - каталог услуг и портал самообслуживания, которые в совокупности обеспечивают

формализацию предложения, прозрачность обязательств и автоматизацию доступа к сервисам. В свою очередь, операционная деятельность ИТ-подразделения выстраивается вокруг взаимосвязанных процессов управления инцидентами, запросами на обслуживание, изменениями и релизами. Их корректная имплементация и четкое понимание различий между ними формируют каркас, позволяющий преобразовать хаотичную деятельность по устранению сбоев в предсказуемый и управляемый механизм предоставления услуг.

Наконец, интеграция DevOps-культуры и Agile-подходов в методологию ITSM, закрепленная в ITIL 4, представляет собой закономерный этап развития дисциплины. Она ведет к трансформации организационной культуры в сторону кросс-функционального взаимодействия и совместной ответственности за конечный результат, разрушая традиционные барьеры между разработкой и эксплуатацией. Данное обстоятельство обеспечивает возможность одновременного повышения как скорости вывода новых функций, так и стабильности функционирования цифровой инфраструктуры. В свою очередь, в совокупности рассмотренные концепции и практики образуют целостную систему управления, необходимую для эффективного функционирования современной цифровой компании.

Подводя итог сказанному выше, необходимо заключить, что переход от технологически-ориентированной модели эксплуатации к сервисно-ориентированному управлению представляет собой фундаментальную трансформацию философии взаимодействия ИТ-подразделения с бизнесом. В рамках классического подхода ИТ воспринимаются как центр затрат, обслуживающий технические ресурсы, тогда как ITSM предлагает рассматривать информационные технологии как совокупность услуг, создающих ценность для потребителя. Критерий эффективности смещается с работоспособности техники на удовлетворенность пользователя и соблюдение соглашений об уровне услуг (SLA). Данная трансформация обеспечивает превращение ИТ из вспомогательной функции в стратегического партнера бизнеса.

Кроме того, анализ развития библиотеки ITIL демонстрирует концептуальный сдвиг от жесткой процессной модели (ITIL v3) к гибкой системе практик (ITIL 4). Введение понятия «практика» как набора организационных ресурсов, включающего не только процедуры, но и роли, компетенции, технологии и культуру взаимодействия, отражает

адаптацию методологии к современным реалиям цифровой экономики. Ключевые практики управления услугами — управление инцидентами, проблемами, запросами на обслуживание и поддержка изменений — получили существенное развитие: внедрение модели «роения» при обработке инцидентов, анализ по модели «4P» в управлении проблемами, разделение потоков изменений на стандартные, нестандартные и экстренные. В сою очередь, каталог услуг и портал самообслуживания являются взаимодополняющими элементами реализации сервисно-ориентированного подхода. Каталог услуг выполняет информационную функцию, обеспечивая формализацию предложения и прозрачность обязательств поставщика. Портал самообслуживания выступает интерактивной средой, предоставляющей пользователям единую точку доступа к сервисам и автоматизирующей выполнение стандартных запросов. Их интеграция в рамках ITSM-системы позволяет реализовать принцип фокуса на ценности за счет стандартизации взаимодействия и снижения нагрузки на сервисную службу.

В настоящее время установлено и доказано, что противопоставление ITSM и DevOps/Agile является неконструктивным. Современный вектор развития, закрепленный в ITIL 4, лежит в плоскости их глубокой интеграции, где DevOps выступает инструментом реализации гибких процессов в структуре ITSM. Ключевым элементом этой интеграционной модели выступает Система ценности сервиса (Service Value System), представляющая собой комплексную операционную модель, демонстрирующую взаимодействие компонентов организации для создания ценности. Практики непрерывной интеграции и доставки (CI/CD), автоматизированного развертывания и «инфраструктуры как код» трансформируют традиционное управление изменениями, позволяя ускорить внедрение низкорисковых изменений без потери контроля. Интеграция DevOps-культуры в ITSM требует изменения организационной культуры в сторону генеративной модели, характеризующейся прозрачностью, совместной ответственностью за результат и восприятием неудач как возможностей для обучения. Формирование кросс-функциональных команд, отвечающих за продукт на всех этапах жизненного цикла, совместное использование инструментов и единые метрики позволяют преодолеть традиционные барьеры между разработкой и эксплуатацией, обеспечивая одновременное повышение как

скорости вывода новых функций, так и стабильности функционирования цифровой инфраструктуры.

Таким образом, ITSM и ITIL 4 образуют целостную систему управления, необходимую для эффективного функционирования цифровой инфраструктуры современного предприятия в условиях цифровой трансформации.

Вопросы для обсуждения

1. Поясните особенности перехода от технологически-ориентированного управления к сервисно-ориентированному подходу ITSM.
2. Укажите, какие управленческие практики, помимо ITIL, интегрируются в современную методологию ITSM.
3. Охарактеризуйте концепцию Enterprise Service Management (ESM) с точки зрения эволюции сервисного подхода.
4. Сравните традиционный и сервисный подходы к управлению ИТ-инфраструктурой компании с позиции формирования природы затрат и характера взаимодействия с бизнесом.
5. Поясните, чем заключается ключевое различие между понятиями «процесс» в ITIL v3 и «практика» в ITIL 4.
6. Раскройте содержание понятия «ценность» в методологии ITIL4.
7. Перечислите семь руководящих принципов ITIL 4, пояснив сущность каждого из них.
8. Поясните содержание модели «4P» в практике управления проблемами ITIL 4.
9. Дайте определение «каталога услуг». Каково его место в общей структуре портфеля услуг?
10. Перечислите обязательные атрибуты описания услуги в каталоге и раскройте назначение каждого из них.
11. Проведите сравнительный анализ каталога услуг и портала самообслуживания с позиции создаваемой бизнес-ценности.
12. Объясните, каким образом следование руководящим принципам ITIL 4 способствует созданию бизнес-ценности.
13. Дайте определение инцидента. Охарактеризуйте его жизненный цикл.

14. Охарактеризуйте сущность запроса. Поясните, в чем заключаются различия между инцидентом и запросом на обслуживание.

15. Перечислите основные типы обращений пользователей, которые относятся к категории запросов на обслуживание.

16. Дайте определение релиза и перечислите его основные свойства.

17. Опишите взаимосвязь процессов управления инцидентами, проблемами, изменениями и релизами.

18. Охарактеризуйте основные эффекты интеграции DevOps и Agile в методологию ITSM.

19. Поясните, какие изменения в организационной культуре необходимы для успешной интеграции DevOps-подходов в ITSM.

20. Перечислите семь руководящих принципов ITIL 4, создающих методологическую основу для преодоления традиционного противоречия между требованиями стабильности и надежности инфраструктуры.

Практические задания

Задание 1. Проведите сравнительный анализ гипотетических ИТ-подразделений двух компаний:

- ИТ-отдел компании «А» функционирует в рамках традиционного (технологического) подхода.
- ИТ-отдел компании «Б» внедрил сервисно-ориентированную модель ITSM.

Для каждой компании опишите, как будет выглядеть:

- реакция на обращение пользователя о невозможности распечатать документ;
- процесс планирования закупки нового программного обеспечения;
- взаимодействие с руководством компании при обосновании ИТ-бюджета на следующий год;
- критерии оценки эффективности работы ИТ-специалистов.

Результаты оформите в виде сравнительной таблицы, дополнив ее аналитическим выводом о том, какой подход обеспечивает более высокую ценность для бизнеса.

Задание 2. В небольшой компании имеются ряд сложностей с управлением ее ИТ-инфраструктурой, которые нужно устранить, для этого решите ряд задач.

1. Классифицируйте приведенные ниже обращения пользователей, определив, к какому типу (инцидент, запрос на обслуживание, изменение) относится каждое из них, и обоснуйте свой выбор:

- Сотрудник сообщает, что не может войти в корпоративную учетную запись после трех неудачных попыток ввода пароля.

- Начальник отдела кадров запрашивает установку специализированного ПО для нового сотрудника, который выходит на работу через неделю.

- Пользователь жалуется, что принтер в отделе не печатает документы, хотя индикаторы горят зеленым.

- Система мониторинга зафиксировала аномальную нагрузку на сервер баз данных в ночное время.

2. Для каждого обращения опишите предполагаемый жизненный цикл обработки.

3. Выберите одно из обращений, классифицированное как инцидент, и смоделируйте гипотетическую ситуацию, при которой:

- данный инцидент становится основанием для создания проблемы;

- решение проблемы требует инициирования изменения;

- изменение включается в релиз.

Опишите эту цепочку взаимосвязей с указанием объектов управления на каждом этапе.

Задание 3. Разработайте фрагмент каталога услуг для ИТ-подразделения образовательной организации (ВУЗа), включив в него три услуги из разных категорий:

- Услуга 1. Предоставление доступа к корпоративной Wi-Fi сети (категория - сетевые услуги).

- Услуга 2. Создание учетной записи для абитуриентов в личном кабинете абитуриента (категория - учетные записи и доступ).

Для каждой услуги опишите следующие атрибуты:

- наименование, категория и краткое описание;

- целевая аудитория;

- условия предоставления (SLA) - время реакции, время решения, доступность;

- процедура запроса (пошагово);
- стоимость и порядок оплаты (для внутренних расчетов);
- ограничения и требования (технические и организационные);
- ответственный и поддержка (контактные данные).

Тест для самоконтроля

1. Что является в контексте ITSM средством передачи ценности потребителю, помогающее ему достичь определенных бизнес-целей?

а) Услуга.

б) Латентность.

в) Бизнес-процесс.

г) Время отклика.

2. Сущность ESM заключается …

а) В применении уникальных сервисных практик к работе других функциональных подразделений компании.

б) В применении стандартизированных сервисных практик к работе ограниченного числа подразделений компании.

в) В применении проверенных сервисных практик к работе функциональных подразделений компании.

г) В отсутствии применении сервисных практик в работе функциональных подразделений компании.

3. Целевые ориентиры внедрения ITSM не осуществляется…

а) На стратегическом уровне.

б) На тактическом уровне.

в) На операционном уровне.

г) На логическом уровне.

4. Каково общее число практик, составляющих ITIL 4?

а) 34;

б) 35;

в) 36;

г) 37.

5. На сколько групп разделены практики ITIL 4?

а) На шесть.

б) На пять.

в) На четыре.

г) На три.

6. Какой группы практик ITIL 4 не существует?

а) Общие практики управления.

б) Практики технического управления.

в) Практики технического регулирования.

г) Практики управления услугами.

7. В рамках управления проблемами в практике ITIL 4 используется модель…

а) «5Р»

б) «4Р»

в) «7Р»

г) «3Р»

8. Какая практика в предыдущих версиях до ITIL 4 называлась управлением изменениями?

а) Поддержка изменений.

б) Управление проблема.

в) Управление инцидентами.

г) Управление запросами.

9. Какое ключевое понятие пронизывает все элементы ITIL 4?

а) Доступность.

б) Ценность.

в) Реализуемость.

г) Ограниченность.

10. Сколько принципов, являющимися универсальными рекомендациями, реализуется в методологии ITIL 4?

а) Четыре принципа.

б) Пять принципов.

в) Шесть принципов.

г) Семь принципов.

11. Что представляет собой структурированный и централизованный перечень всех активных ИТ-услуг, доступных пользователям?

а) Портфель услуг.

б) Каталог услуг.

в) Эффективность услуг.

г) ИТ-активность.

12. Интерактивная среда, предоставляющая пользователю интерфейс для непосредственного взаимодействия с поставщиком услуг.

а) Набор услуг.

б) Каталог услуг.

в) Каталог решений.

г) Портал самообслуживания.

13. Как называется любое незапланированное событие, которое приводит к перерыву в предоставлении услуги или снижению ее качества?

а) Инцидент.

б) Проблема.

в) Сбой.

г) Изменение.

14. Что является формализованной просьбой пользователя о предоставлении доступа к услуге, получении информации, которое не несет в себе высоких рисков и не требует экстренного согласования?

а) Инцидент.

б) Уровень услуг.

в) Запрос на обслуживание.

г) Запрос на изменение.

15. Цель этого процесса - не заблокировать изменения, а обеспечить их реализацию с минимальными рисками, используя стандартизированные методы и процедуры.

а) Управление изменениями.

б) Управление релизами.

в) Управление проблемами.

г) Управление услугами.

16. Какой категории не существует в типологии изменений методологии ITIL 4.

а) Стандартное изменение.

б) Нестандартное изменение.

в) Экстренное изменение.

г) Пассивное изменение.

17. Совокупность одного или нескольких изменений, которые проходят через единый жизненный цикл как единое целое называется…

а) релизом;

б) проблемой;

в) инцидентом;

г) изменением.

18. Представляющая собой набор гибких видов деятельности, которые могут комбинироваться различными способами для создания, поставки и постоянного их улучшения.

а) Цепочка создания ценности.

б) Контроль и минимизация рисков.

в) Управление конфигурациями.

г) Управление сопровождением.

19. Какой формулой в общем случае можно описать подход DevOps?

а) DevOps =Agile + Lean + ITSM;

б) DevOps =Agile + Lean + ITIL;

в) DevOps =Agile + Lean + ITSM- ITIL;

г) DevOps =Agile + Lean + ITIL+SVS.

20. Интеграция DevOps в ITSM оказывает наиболее ощутимое влияние на ключевые процессы…

а) управления конфигурациями, релизами, услугами;

б) управления инцидентами, проблемами и изменениями;

в) управления эффективностью оказания ИТ-услуг;

г) управления жизненным циклом группы однородных и типовых ИТ-услуг.

Глава 8. АУДИТ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ

8.1. Понятие, компоненты и эволюция цифровой инфраструктуры

В условиях цифровой трансформации экономики информация и обеспечивающая её инфраструктура перестали быть исключительно технической подсистемой предприятия, превратившись в стратегический актив и источник рисков. Традиционный финансовый аудит, направленный на подтверждение достоверности бухгалтерской отчетности, уже не способен в полной мере удовлетворить потребности менеджмента в оценке надежности и эффективности информационных технологий. Это обусловило выделение аудита цифровой инфраструктуры в самостоятельное направление контрольной деятельности, которое может существовать как в рамках внутреннего аудита (операционный аудит), так и в форме специализированных внешних проверок.

Главная цель аудита цифровой инфраструктуры заключается в получении объективной и независимой оценки текущего состояния ИТ-среды организации, определении степени её соответствия установленным критериям и выработке обоснованных рекомендаций, направленных на повышение эффективности использования ИТ-активов, управление рисками и обеспечение непрерывности бизнес-процессов. Достижение этой цели предполагает решение комплекса взаимосвязанных задач, которые могут быть сгруппированы по нескольким ключевым направлениям.

В отличие от финансового аудита, где критерии оценки (ПБУ, МСФО) строго регламентированы, аудит цифровой инфраструктуры оперирует более широким спектром критериев. К ним относятся не только требования законодательства, но и лучшие отраслевые практики (методологии COBIT, ITIL), а также технические регламенты, внутренние политики компании. Анализ практики консалтинговых компаний позволяет выделить следующие основные задачи аудита цифровой инфраструктуры:

Инвентаризация и паспортизация активов: сбор и верификация данных о составе аппаратного и программного обеспечения, сетевой архитектуре, системах хранения данных и используемых внешних сервисах, создание актуальной документации.

- Оценка эффективности использования ресурсов: анализ загрузки серверов, систем хранения данных, сетевого оборудования с целью выявления неиспользуемых или неэффективно используемых мощностей («зомби-серверы», избыточное резервирование) и оптимизации затрат на владение инфраструктурой.

- Анализ управляемости и надежности: проверка наличия и эффективности регламентов резервного копирования, процедур управления изменениями, мониторинга инцидентов и обеспечения отказоустойчивости.

- Оценка соответствия нормативным требованиям: проверка выполнения предписаний регуляторов (ФСБ, Банк России и др.) в области обработки персональных данных, защиты критической информационной инфраструктуры и обеспечения кибербезопасности.

- Выявление уязвимостей и управление рисками: идентификация технических и организационных уязвимостей, построение модели угроз и оценка рисков, связанных с возможными инцидентами информационной безопасности. Основные задачи аудита цифровой инфраструктуры компании представлены ниже на рис.8.1.

Рис.8.1. Задачи аудита цифровой инфраструктуры

Многообразие перечисленных задач обуславливает необходимость дифференциации самого процесса проверки, что привело к формированию специализированных видов аудита, каждый из которых имеет собственную методическую базу и фокус исследования (табл. 8.1).

Таблица 8.1

Классификация видов аудита цифровой инфраструктуры по целям и задачам

Вид аудита	Основная цель	Ключевые объекты проверки	Типичные критерии оценки
Операционный аудит	Оценка эффективности, экономичности и результативности использования ИТ-ресурсов в бизнес-процессах	Бизнес-процессы, использующие ИТ; ИТ-бюджет; SLA с подразделениями; процессы управления ИТ	Соотношение «затраты-выгода», скорость выполнения операций, показатели SLA, уровень автоматизации
Аудит соответствия	Подтверждение соответствия ИТ-инфраструктуры требованиям законодательства, стандартов и внутренних регламентов	Документация (политики, приказы), настройки средств защиты, процедуры обработки данных	Требования 152-ФЗ, ГОСТ Р 57580, ISO/IEC 27001, PCI DSS, отраслевые стандарты ЦБ РФ
Аудит производительности	Выявление узких мест и причин деградации скорости работы ИТ-систем и сервисов	Серверное и сетевое оборудование, СУБД, код приложений, системы виртуализации	Время отклика, пропускная способность, загрузка процессора и памяти, количество транзакций в секунду
Аудит безопасности	Выявление уязвимостей и оценка уровня защищенности инфраструктуры от внешних и внутренних угроз	Сетевая периметрия, приложения, политики доступа, журналы событий (логи), физическая защита	Количество критических уязвимостей, уровень покрытия угроз, время детектирования инцидентов

Операционный аудит в контексте цифровой инфраструктуры выходит за рамки простой проверки технической исправности оборудования. Его предметом является эффективность использования информационных технологий для достижения бизнес-целей.

Как отмечается в научных исследованиях, аудит информационной инфраструктуры, являясь разновидностью внутреннего аудита, по своей сути тяготеет именно к операционному аудиту, поскольку призван оценивать не столько достоверность данных, сколько рациональность управления ресурсами.

В ходе операционного аудита оценивается, насколько текущая архитектура и практики управления ИТ соответствуют потребностям бизнеса, нет ли избыточности, дублирования функций, неоправданных затрат на поддержку устаревших решений.

Задача аудитора здесь - проанализировать взаимосвязь между затратами на ИТ и результатами деятельности компании, а также оценить, способствует ли ИТ-инфраструктура достижению стратегических целей или, напротив, тормозит развитие из-за низкой адаптивности и высокой сложности внесения изменений.

Аудит соответствия (compliance audit) является наиболее регламентированным видом проверки. Его целью является установление факта выполнения организацией обязательных требований, установленных законодательством, регулирующими органами или корпоративными стандартами. В российской практике наиболее востребованным является аудит соответствия требованиям Федерального закона № 152-ФЗ «О персональных данных»[31], а также нормативным актам Банка России (например, ГОСТ Р 57580.1-2017[32]) и Федеральной службы по техническому и экспортному контролю Российской Федерации (ФСТЭК) в области защиты критической информационной инфраструктуры.

Особенностью данного вида аудита является его документарный характер, тесно переплетающийся с технической проверкой. Аудитор не только проверяет наличие приказов и политик, но и должен эмпирически подтвердить, что декларируемые правила действительно выполняются. Например, при проверке соответствия стандарту ISO/IEC 27001 аудитор может запросить журналы системы контроля доступа за

---

[31] О защите персональных данных: Федеральный закон от 27.07.2006 № 152-ФЗ (ред. от 24.06.2025) // Консультант Плюс: справочно-правовая система. — Режим доступа: https://base.garant.ru/12148567/?ysclid=mkkv6fzu8i235523485

[32] ГОСТ Р 57580.1-2017. Безопасность финансовых (банковских) операций. Защита информации финансовых организаций. Базовый состав организационных и технических мер» // Консультант Плюс: справочно-правовая система. - Режим доступа: https://base.garant.ru/12148567/?ysclid=mkkv6fzu8i235523485

произвольную дату и потребовать предоставить обоснование (заявку на работы) для каждого зафиксированного входа в серверную, выявляя таким образом расхождения между регламентами и реальной практикой. Аудит соответствия также необходим для получения сертификатов PCI DSS (для компаний, работающих с платежными картами), которые являются обязательным условием для ведения деятельности во многих сферах.

Аудит безопасности (информационной безопасности) занимает особое место в системе проверок ввиду высокого уровня актуальных киберугроз. Его целью является получение объективной качественной и количественной оценки уровня защищенности информационной системы и выработка мер по снижению рисков реализации компьютерных атак. Данный вид аудита включает в себя несколько ключевых направлений деятельности, представленных в табл. 8.2.

Аудит производительности фокусируется на технических аспектах функционирования инфраструктуры и прикладного программного обеспечения. Его основная задача — диагностика «узких мест», приводящих к снижению скорости работы сервисов, увеличению времени отклика и, как следствие, к ухудшению пользовательского опыта и прямым финансовым потерям. Как показывает практика, необходимость в таком аудите часто возникает при росте нагрузки на систему, когда ранее работавшая конфигурация перестает справляться с объемом операций.

Инструментарий аудита производительности включает в себя нагрузочное тестирование, анализ технологических журналов серверов приложений и систем управления базами данных (СУБД), мониторинг использования вычислительных ресурсов. Результатом проверки является не просто констатация факта низкой производительности, а выявление ее первопричины — будь то неоптимальные запросы к базе данных, недостаточный объем оперативной памяти, ошибки конфигурации сетевого оборудования или архитектурные ограничения программного кода.

Таблица 8.2

Основные направления аудита информационной безопасности

Направление	Содержание работ	Ожидаемый результат
Анализ защищенности (Vulnerability Assessment)	Автоматизированное сканирование компонентов инфраструктуры для выявления известных уязвимостей (например, необновленное ПО, слабые пароли).	Перечень уязвимостей с указанием уровня критичности и рекомендации по их устранению.
Тестирование на проникновение (Pentest)	Моделирование действий реального злоумышленника для преодоления существующих мер защиты и получения доступа к целевым ресурсам (по методологии Black/Grey/White Box).	Отчет, содержащий описание вектора атаки, перечень успешно использованных уязвимостей и оценку возможности их эксплуатации .
Анализ соответствия технических мер защиты	Проверка корректности настроек межсетевых экранов, систем обнаружения вторжений, средств антивирусной защиты и политик управления доступом.	Заключение о соответствии конфигурации лучшим практикам (рекомендациям производителей, стандартам NIST, CIS).
Анализ физической безопасности	Оценка системы контроля доступа в серверные помещения и ЦОД, проверка эффективности видеонаблюдения и охраны.	Оценка рисков, связанных с несанкционированным физическим доступом к оборудованию.

Результатом комплексного аудита безопасности является формирование карты рисков, привязанной к конкретным бизнес-эффектам, и дорожной карты по устранению выявленных недостатков, что позволяет компании выстроить приоритеты в финансировании мероприятий по защите информации.

Таким образом, представленные виды аудита - операционный, соответствия, производительности и безопасности - не являются взаимоисключающими, а напротив, дополняют друг друга, формируя целостную картину состояния цифровой инфраструктуры организации. Выбор конкретного вида или их комбинации зависит от целей проверки, отраслевой специфики компании и текущей стадии её жизненного цикла.

8.2. Стандарты и фреймворки для аудита

Аудит цифровой инфраструктуры не может проводиться в вакууме; он требует опоры на формализованные стандарты и общепризнанные методики (фреймворки). Эти документы выполняют двоякую функцию. С одной стороны, они предоставляют аудитору конкретный перечень требований или контрольных целей, подлежащих проверке. С другой стороны, для аудируемой организации они служат руководством по выстраиванию эффективной системы управления и защиты информации. Выбор конкретного стандарта или фреймворка зависит от целей аудита: оценка общего управления IT (COBIT), проверка системы менеджмента информационной безопасности (ISO 27001) или соответствие строгим отраслевым требованиям платежной индустрии (PCI DSS). Понимание различий в их природе, масштабе и применимости является фундаментальной компетенцией современного аудитора.

В современной практике аудита цифровой инфраструктуры ключевое место занимает использование специализированных стандартов и фреймворков, среди которых доминирующее положение принадлежит семейству Control Objectives for Information and Related Technologies (COBIT). Разработанный Ассоциацией аудита и контроля информационных систем (ISACA), COBIT представляет собой всеобъемлющий набор руководящих принципов и лучших практик, предназначенных для эффективного управления информационными технологиями (ИТ) и осуществления их контроля. В условиях усиления регуляторных требований и усложнения киберугроз COBIT служит основным инструментом, обеспечивающим согласование целей ИТ с бизнес-стратегией, оптимизацию ресурсов и управление рисками.

С момента своего первого появления в 1996 году как набора контрольных задач, фреймворк претерпел значительную эволюцию.

Актуальная версия, COBIT 2019, выпущенная в 2018 году, базируется на фундаменте, заложенном версией COBIT 5, но при этом существенно расширяет его. Развитие направлено на адаптацию к динамичным изменениям цифрового ландшафта, поддержку цифровой трансформации и внедрение более гибких подходов к управлению. Новая архитектура фреймворка базируется на модели оценки производительности CMMI (Capability Maturity Model Integration), что позволяет

более точно оценивать зрелость процессов. Структурно COBIT 2019 основывается на ряде фундаментальных принципов и компонентов. [33]

Ключевым отличием от предшественника является расширение перечня принципов. Если COBIT 5 включал пять принципов, то COBIT 2019 оперирует шестью, которые обеспечивают целостность и адаптивность системы управления (табл. 8.3).

Таблица 8.3

Сравнение принципов COBIT 5 и COBIT 2019

№	Принципы COBIT 5	Принципы COBIT 2019	Ключевые изменения и смысл
1	Удовлетворение потребностей заинтересованных сторон	Обеспечение ценности для заинтересованных сторон	Акцент на создании ценности через баланс выгод, риска и ресурсов.
2	Комплексное (сквозное) покрытие предприятия	Целостный (холистический) подход	Сохранение принципа, но с усилением системного взгляда.
3	Применение единого интегрированного фреймворка	Единый интегрированный фреймворк	Интеграция с другими стандартами (ITIL, ISO 27001 и др.) .
4	Обеспечение целостного подхода	Динамичная система управления	Новый принцип, требующий адаптивности и гибкости управления к изменениям.
5	5. Разделение управления и менеджмента	5. Разделение управления и менеджмента	Четкое разграничение стратегического надзора и операционной деятельности.
6	—	6. Адаптация к потребностям предприятия	Новый принцип, признающий уникальность каждого предприятия и необходимость настройки системы под его контекст.

Помимо принципов, COBIT 2019 вводит концепцию компонентов системы управления, которые в совокупности обеспечивают ее работоспособность. Эти компоненты заменили понятие «энэйблеров» (факторов, способствующих успеху) из предыдущих версий и пред-

[33] Гришин Л. ТО САМОЕ. ВВЕДЕНИЕ И МЕТОДИКА СХЕМЫ COBIT 2019 [Электронный ресурс] / Лев Гришин, пер. с англ. // Режим доступа: https://blog.cortel.cloud/2023/02/28/zemlya-ili-oblako-ekonomika-vladeniya/?ysclid=mki95pgb9q242240226 (дата обращения: 03.03.2026).

ставляют собой факторы, которые в совокупности определяют, как организация управляет ИТ. К ним относятся: процессы, организационные структуры, принципы и политики, информация, культура и поведение, инфраструктура и приложения, а также люди и компетенции.

Такой подход позволяет рассматривать ИТ-аудит не как изолированную проверку, а как часть комплексной системы корпоративного управления.

Методология COBIT 2019 предоставляет аудитору детализированную модель процессов, сгруппированных по пяти основным доменам (областям). В отличие от COBIT 5, включавшего 37 процессов, новая версия расширила их перечень до 40 за счет добавления процессов управления данными (APO14 – Managed Data), управления проектами (BAI11 – Managed Projects) и управления гарантиями (MEA04 – Managed Assurance) .

Это расширение отражает растущую значимость данных как актива и проектного подхода в цифровой экономике. Для целей аудита это означает появление четких контрольных точек в тех сферах, которые ранее были описаны менее детально. Распределение процессов по доменам представлено в табл. 8.4.

Применение COBIT в аудите цифровой инфраструктуры базируется на риск-ориентированном подходе. Методология COBIT не предписывает жестких алгоритмов проверки, а позволяет аудитору построить модель анализа рисков, исходя из специфики бизнеса и его ИТ-ресурсов.

Аудит, основанный на COBIT, начинается с оценки ИТ-ресурсов, необходимых для достижения бизнес-целей, и идентификации потенциальных проблем и факторов, способных повлиять на процесс управления. Практическая реализация такого подхода предполагает, что критерии аудита формируются на основании предварительной оценки рисков, что позволяет объединить современные управленческие методики и профессиональное суждение аудитора.

Таблица 8.4

Домены и фокус процессов в COBIT 2019

Домен	Область	Примеры процессов (выборочно)
EDM (Evaluate, Direct and Monitor)	Оценка, направление и мониторинг	Определение и поддержка архитектуры предприятия, обеспечение оптимизации рисков, обеспечение прозрачности для заинтересованных сторон.
APO (Align, Plan and Organize)	Согласование, планирование и организация	Управление стратегией, инновациями, рисками (APO12), портфелями, данными (APO14), поставщиками, безопасностью (APO13) .
BAI (Build, Acquire and Implement)	Создание, приобретение и внедрение	Управление программами и проектами (BAI11), идентификация решений (BAI03), управление изменениями (BAI06), управление активами .
DSS (Deliver, Service and Support)	Поставка, обслуживание и поддержка	Управление операциями, инцидентами (DSS02), непрерывностью (DSS04), безопасностью услуг (DSS05) .
MEA (Monitor, Evaluate and Assess)	Мониторинг, оценка и анализ	Мониторинг соответствия внутренним и внешним требованиям, управление гарантиями (MEA04), система внутреннего контроля.

Важным преимуществом COBIT является его интеграционный потенциал. Фреймворк специально разработан для гармонизации с другими популярными стандартами и лучшими практиками. Он служит своего рода «зонтичной» структурой, которая позволяет связать стратегические цели бизнеса с операционными практиками. Например, COBIT может успешно сочетаться с библиотекой ITIL (Information Technology Infrastructure Library), фокусирующейся на управлении ИТ-услугами, и со стандартами NIST (National Institute of Standards and Technology), концентрирующимися на кибербезопасности. Для построения системы управления информационной безопасностью (СУИБ) COBIT предоставляет детализированные описания смежных процессов, таких как APO13 (Managed Security) и DSS05 (Managed Security Services), а также предлагает использовать свою модель качественных критериев для оценки информации.

Таким образом, COBIT 2019 представляет собой не просто перечень контрольных задач, а целостную методологию управления и аудита ИТ. Его использование в рамках аудита цифровой инфраструктуры позволяет перейти от фрагментарной проверки соответствия к комплексной оценке эффективности системы корпоративного управления ИТ. Благодаря встроенным принципам гибкости и адаптации, а также детализированной процессной модели, COBIT предоставляет аудитору инструментарий для анализа зрелости процессов, выявления узких мест и выработки рекомендаций, направленных на повышение общей эффективности бизнеса в условиях цифровой трансформации.

В современной практике аудита цифровой инфраструктуры ключевое место занимают международные стандарты серии ISO/IEC 27000, разработанные Международной организацией по стандартизации (ISO) и Международной электротехнической комиссией (IEC). Данные стандарты формируют методологическую базу для создания, внедрения, функционирования, мониторинга, анализа, поддержки и улучшения Системы менеджмента информационной безопасности (СМИБ, англ. ISMS) . Центральным документом в этой серии является стандарт ISO/IEC 27001, который содержит обязательные требования к СМИБ и определяет порядок проведения аудитов для оценки их выполнения. Понимание структуры и требований этого стандарта является фундаментальной компетенцией аудитора цифровой инфраструктуры.

Стандарт ISO/IEC 27001 задает процессно-ориентированный и риск-ориентированный подход к управлению информационной безопасностью. Аудит на соответствие этому стандарту направлен на проверку не только наличия формализованных политик и процедур, но и их практической эффективности. Ключевым документом, связующим требования стандарта с практическими мерами защиты, является Заявление о применимости (Statement of Applicability, SoA). SoA документирует необходимые меры (контроли) из приложения A (Annex A), которые выбраны организацией для обработки рисков, обосновывает их включение или исключение, а также подтверждает статус их внедрения . Именно SoA служит основой для планирования аудиторской программы, поскольку определяет границы и перечень контролей, подлежащих проверке.

Аудит в контексте ISO 27001 подразделяется на три типа, различающихся по целям, исполнителям и степени независимости. Для систематизации данных процессов используется классификация сторон, участвующих в аудите: первая, вторая и третья сторона.

В табл. 8.5 представлена сравнительная характеристика данных видов аудита.

Таблица 8.5

Типология аудитов информационной безопасности по ISO 19011 и ISO 27001

Тип аудита	Сторона проведения	Основная цель	Примеры инициаторов/ исполнителей	Влияние на сертификацию
Аудит первой стороны (First-party audit)	Внутренний аудит, проводимый самой организацией или сторонней организацией по её поручению	Оценка соответствия требованиям ISO 27001, выявление несоответствий для их устранения до внешнего аудита, проверка результативности СМИБ.	Собственная служба внутреннего аудита, привлеченная консалтинговая компания.	Обязательное требование п. 9.2 стандарта. Не ведет к выдаче сертификата, но является основой для подготовки к сертификации.
Аудит второй стороны (Second-party audit)	Внешний аудит, проводимый заинтересованной стороной (например, заказчиком или партнером)	Проверка СМИБ контрагента (поставщика услуг/ продуктов) на предмет соответствия договорным обязательствам и требованиям безопасности для минимизации рисков, связанных с третьими лицами.	Служба безопасности компании-заказчика в отношении своего поставщика облачных услуг.	Не влияет на формальную сертификацию по ISO 27001, но может быть условием для заключения или пролонгации контракта.
Аудит третьей стороны (Third-party audit)	Внешний аудит, проводимый независимым, аккредитованным органом по сертификации (Certification Body, CB)	Объективная и независимая проверка СМИБ на соответствие требованиям стандарта для выдачи и последующего подтверждения сертификата соответствия ISO 27001.	Аккредитованные органы (например, BSI, DNV), действующие под надзором национальных органов по аккредитации	Является основанием для выдачи сертификата. Включает первичную сертификацию, инспекционные (надзорные) и ресертификационные аудиты.

Процесс сертификационного аудита, проводимого третьей стороной, является наиболее строго регламентированной процедурой. Он состоит из двух обязательных этапов, которые детализированы в стандарте ISO 17021 (требования к органам по сертификации).[34] На первом этапе аудиторы проводят анализ документированной информации СМИБ, оценивают готовность организации к сертификации, проверяют полноту определения области применения и результаты первичной оценки рисков.

Основная задача этого этапа - подтвердить, что организация понимает требования стандарта и способна перейти к следующей стадии.

Второй этап является более глубоким и комплексным. Аудиторы проверяют практическое внедрение и эксплуатационную эффективность всех контролей, указанных в SoA, а также выполнение требований клаузул 4-10 стандарта. По итогам второго этапа формируется заключение о возможности выдачи сертификата.

Поддержание статуса действующего сертификата требует проведения регулярных последующих аудитов. График сертификационного цикла строго регламентирован и представлен в табл. 8.6.

Таблица 8.6

Структура и периодичность сертификационного цикла ISO 27001

Тип аудита в рамках цикла	Сроки проведения	Объект проверки	Возможные исходы
Первичная сертификация	Одноразовая процедура: Этап 1 и Этап 2.	Полное соответствие СМИБ требованиям ISO 27001:2022, включая все клаузулы и все выбранные контроли Annex A.	Выдача сертификата (сроком на 3 года) при устранении всех несоответствий.
Инспекционный контроль (Surveillance Audit)	Ежегодно (обычно через 12 месяцев после выдачи сертификата) в течение первого и второго годов цикла.	Выборочная проверка части требований стандарта (не менее 50% контролей за год) и анализ выполнения планов по улучшению, а также закрытие несоответствий, выявленных ранее .	Подтверждение действия сертификата на следующий год.

[34] ГОСТ Р ИСО/МЭК 17021-1-2025. Оценка соответствия. Требования к органам, проводящим аудит и сертификацию систем менеджмента. Часть 1. Требования (введ. .01.09.2025) [Электронный ресурс]// Режим доступа - https://docs.cntd.ru/document/1312251791?ysclid=mmc5kykis1768698346 (дата обращения 03.03.2036)

Тип аудита в рамках цикла	Сроки проведения	Объект проверки	Возможные исходы
Ресертификация (Recertification Audit)	Каждые 3 года, до истечения срока действия текущего сертификата.	Полномасштабная повторная проверка всей СМИБ, аналогичная по объему первичной сертификации, с учетом всех изменений в организации за прошедший период .	Перевыпуск сертификата на новый 3-летний срок.

Методология проведения внутреннего (первой стороны) и внешнего (третьей стороны) аудита базируется на общих принципах, зафиксированных в стандарте ISO 19011 «Руководящие указания по аудиту систем менеджмента». Аудит рассматривается как систематический, независимый и документированный процесс получения объективных свидетельств и их объективного оценивания с целью установления степени выполнения критериев аудита . Объективными свидетельствами могут выступать записи, документы, наблюдаемые факты или свидетельства персонала. Процесс сбора и верификации данных основан на выборке, что обуславливает определенный уровень неопределенности, который должен учитываться аудитором при формулировании выводов.

Важным аспектом является обеспечение компетентности и объективности аудиторов. Для проведения внутренних аудитов организация должна назначать аудиторов, которые не несут прямой ответственности за деятельность в проверяемой области и свободны от предвзятости и конфликта интересов . Результаты аудитов, включая выявленные несоответствия и возможности для улучшения, оформляются в виде отчета, который служит входным элементом для анализа СМИБ со стороны руководства и инициирует процессы корректирующих действий, требуемые в разделе 10 «Улучшение».

Таким образом, фреймворк ISO 27001 предоставляет не только набор требований к безопасности, но и встроенный механизм оценки их выполнения через регламентированную систему аудитов. Это превращает стандарт в замкнутый цикл управления (Plan-Do-Check-Act, PDCA), где аудит является ключевым инструментом функции «Проверка» (Check). Освоение данного фреймворка позволяет специалисту проводить аудит цифровой инфраструктуры не как разовую проверку

технических настроек, а как комплексную оценку эффективности всей системы управления информационной безопасностью организации.

В системе обеспечения безопасности платежных процессов центральное место занимает стандарт индустрии платежных карт (Payment Card Industry Data Security Standard, PCI DSS), разработанный Советом по стандартам безопасности PCI (PCI Security Standards Council, PCI SSC). Данный стандарт представляет собой глобально признанный фреймворк, обязательный к применению для всех организаций, которые хранят, обрабатывают или передают данные держателей платежных карт. К таким организациям относятся не только торгово-сервисные предприятия, но и процессоры платежей, поставщики услуг, а также сторонние вендоры, чья деятельность оказывает влияние на безопасность среды данных держателей карт (Cardholder Data Environment, CDE) . Основная цель PCI DSS заключается в минимизации рисков кражи данных и мошенничества путем внедрения единых и строгих правил безопасности.[35]

Структурно стандарт PCI DSS версии 4.0.1, как и его предшественники, базируется на шести ключевых целях, которые детализируются в двенадцати основных требованиях. Для наглядного представления логической взаимосвязи между целями, требованиями и объемом контрольных мероприятий, данные могут быть систематизированы так, как представлено в табл. 8.7.

Таблица 8.7

Структура и объем контрольных мероприятий стандарта PCI DSS 4.0.1

Цель стандарта	Требования PCI DSS	Количество базовых контролей
1. Построение и поддержание защищенной сети	Требование 1: Установка и поддержка конфигурации межсетевых экранов.	19
	Требование 2: Отказ от использования стандартных паролей и параметров, поставляемых вендором.	10
2. Защита данных держателя карты	Требование 3: Защита сохраненных данных.	19
	Требование 4: Шифрование передачи данных по открытым сетям.	3

[35] Что такое PCI DSS [Электронный ресурс]//Режим доступа - https://selectel.ru/blog/pci-dss (дата обращения 03.03.2026)

Цель стандарта	Требования PCI DSS	Количество базовых контролей
3. Поддержание программы управления уязвимостями	Требование 5: Защита от вредоносного ПО.	5
	Требование 6: Разработка и поддержка безопасных систем и приложений.	25
4. Внедрение строгих мер контроля доступа	Требование 7: Ограничение доступа к данным на основе служебной необходимости.	8
	Требование 8: Идентификация и аутентификация пользователей.	21
	Требование 9: Ограничение физического доступа.	20
5. Регулярный мониторинг и тестирование сетей	Требование 10: Логирование и мониторинг доступа к сетевым ресурсам.	28
	Требование 11: Регулярное тестирование систем безопасности.	12
6. Поддержание политики информационной безопасности	Требование 12: Поддержание политики, регламентирующей информационную безопасность.	34

Следует заметить, что структура PCI DSS 4.0.1 охватывает весь жизненный цикл защиты информации: от построения защищенной сети до регулярного мониторинга и поддержания политик информационной безопасности (рис. 8.2).

Рис. 8.2. Структура стандарта PCI DSS 4.0.1

Как демонстрирует таблица, требования не являются простым контрольным перечнем, а представляют собой комплексный подход, требующий от организации интеграции защитных механизмов в ежедневные операционные процессы. Например, Требование 10, содержащее 28 контролей, акцентирует внимание на необходимости не просто сбора логов, но и их регулярного анализа для выявления аномалий. В этом контексте современные решения, такие как платформы SIEM (Security Information and Event Management), например Splunk Enterprise Security, позволяют автоматизировать процесс мониторинга и коррелировать события безопасности, что напрямую способствует выполнению требований по постоянному мониторингу.

Процедура аудита на соответствие PCI DSS и необходимая отчетность напрямую зависят от объема обрабатываемых транзакций, что определяет уровень мерчанта или поставщика услуг. Данная классификация позволяет дифференцировать подход к валидации: от наиболее строгих ежегодных проверок с привлечением сертифицированного аудитора до упрощенных процедур самооценки (табл. 8.8).

Таблица 8.8

Классификация уровней соответствия PCI DSS для мерчантов и поставщиков услуг

Уровень	Категория мерчантов (годовой объем транзакций)	Категория поставщиков услуг	Требования к валидации
Уровень 1	Более 6 млн. транзакций по картам в год	Более 300 тыс. транзакций в год или хранение данных держателей карт третьих сторон	Ежегодный онсайт-аудит QSA (Qualified Security Assessor) и ежеквартальное сканирование ASV (Approved Scanning Vendor) .
Уровень 2	От 1 до 6 млн. транзакций в год	Менее 300 тыс. транзакций в год	Ежегодное заполнение Анкеты самооценки (SAQ) и ежеквартальное ASV-сканирование .
Уровень 3	От 20 тыс. до 1 млн. транзакций электронной коммерции в год	—	Ежегодное заполнение SAQ и ежеквартальное ASV-сканирование.
Уровень 4	Менее 20 тыс. транзакций электронной коммерции или менее 1 млн. всех транзакций в год	—	Ежегодное заполнение SAQ; ASV-сканирование при наличии внешнего IP-адреса.

Примечательно, что для поставщиков услуг уровня 1, а также для мерчантов уровня 1, процесс аудита (QSA-аудит) является обязательным и наиболее трудоемким. В ходе такой проверки аудитор оценивает соответствие более чем 250 требований, которые структурированы по шести упомянутым группам. Аудит затрагивает только ту инфраструктуру предприятия, которая взаимодействует с платежными системами, поэтому критически важным этапом подготовки является изоляция сегмента CDE от остальной сети предприятия. Результатом успешного прохождения аудита является получение Отчета о соответствии (Report on Compliance, ROC) и Свидетельства о соответствии (Attestation of Compliance, AOC), которые действительны в течение одного года и направляются в международные платежные системы или банки-эквайеры.

Следует особо подчеркнуть, что Совет PCI SSC не признает никакие иные формы сертификатов или документов, подтверждающих соответствие стандарту, кроме официальных шаблонов ROC, AOC и SAQ, размещенных на веб-сайте Совета. Любые сертификаты, выданные третьими сторонами в произвольной форме, не являются подтверждением соответствия и не должны приниматься организациями при проверке статуса комплаенса их партнеров. Использование неофициальных документов для валидации соответствия, в частности, в контексте требований 12.8 и 12.9, не допускается. Организации, получающие подобные сертификаты от своих поставщиков услуг, должны настаивать на предоставлении документации именно по утвержденным шаблонам PCI SSC .

Особое внимание в стандарте версии 4.0.1 уделяется управлению рисками, связанными с поставщиками услуг (Third-Party Service Providers, TPSP). Положения требования 12.8.4 обязывают организацию внедрить программу мониторинга статуса соответствия TPSP стандарту PCI DSS с периодичностью не реже одного раза в 12 месяцев. Целью данного требования является получение уверенности в том, что поставщик, выполняющий критичные функции (например, управление межсетевым экраном), соблюдает применимые к его услугам требования. Если TPSP не соответствует этим требованиям, они считаются невыполненными и для самой организации. В качестве подтверждения соответствия TPSP может предоставить свое Свидетельство о соответствии (AOC), а в случае отсутствия формальной оценки —

предоставить иные доказательства напрямую аудитору организации-клиента.

Более того, для поставщиков услуг стандарт вводит ряд дополнительных, более строгих обязательств. К ним относятся требования к проведению пентестов сегментации каждые шесть месяцев , внедрению процессов своевременного обнаружения и реагирования на отказы критически важных систем безопасности (межсетевых экранов, систем обнаружения вторжений, средств логирования) , а также использование уникальных учетных данных для удаленного доступа к каждому клиенту . Высшее руководство поставщика услуг (исполнительный менеджмент) должно нести персональную ответственность за программу соответствия PCI DSS, что подчеркивает важность комплаенса на стратегическом уровне управления. Таким образом, PCI DSS представляет собой не статичный набор правил, а эволюционирующий фреймворк, который задает высокую планку для аудита цифровой инфраструктуры.

Необходимо заметить, что несмотря на различия в целях и областях применения, между рассмотренными стандартами и фреймворками существует тесная взаимосвязь, которая активно используется на практике.

Сравнительный анализ стандартов и фреймворков аудита представлен ниже в табл. 8.9.

Таблица 8.9

Сравнительный анализ стандартов и фреймворков аудита

Характеристика	COBIT 2019	ISO/IEC 27001:2022	PCI DSS 4.0
Основная цель	Управление и контроль IT для достижения бизнес-целей, создание ценности через IT.	Создание, внедрение и улучшение системы менеджмента информационной безопасности (СМИБ).	Защита данных держателей платежных карт, предотвращение мошенничества.
Природа документа	Фреймворк (методология), набор руководств лучших практик.	Международный стандарт, устанавливающий требования к СМИБ.	Отраслевой стандарт, обязательный к исполнению для участников платежной системы.
Область применения (Scope)	Вся IT-инфраструктура и процессы организации, увязанные со стратегией бизнеса.	Организация в целом или ее отдельные подразделения, определенные в масштабе СМИБ.	Четко ограничена средой данных держателей карт (CDE).

Характеристика	COBIT 2019	ISO/IEC 27001:2022	PCI DSS 4.0
Характер требований	Рекомендательные, позволяющие гибко настраивать систему управления под специфику организации.	Формальные требования к процессам и контролем (Приложение А).	Императивные, детализированные требования (12 основных требований).
Процесс аудита	Оценка зрелости процессов, выявление пробелов для улучшения управления .	Двухступенчатый сертификационный аудит (Stage 1 и Stage 2) + ежегодные надзорные аудиты.	Регулярные проверки в зависимости от уровня организации.

Как видно из таблицы, стандарты решают разные задачи. Однако на практике они не исключают, а дополняют друг друга. Организация может использовать COBIT для выстраивания стратегического управления IT, внедрить ISO 27001 для системного управления информационными рисками и пройти сертификацию PCI DSS, чтобы иметь возможность обрабатывать платежи. Более того, существует значительная степень взаимного соответствия. Высокоуровневые цели контроля ISO 27001 покрывают многие детальные требования PCI DSS. Например, требования PCI DSS к шифрованию данных при передаче и хранении напрямую соотносятся с контрольными мерами из прил. А ISO 27001, касающимися криптографии. Это позволяет организациям выстраивать интегрированные системы менеджмента, оптимизируя усилия по прохождению различных аудитов. Для аудитора понимание этих взаимосвязей является ключом к проведению комплексной оценки, позволяющей избежать дублирования и выявить реальный уровень защищенности и управляемости цифровой инфраструктуры.

8.3. Методология проведения аудита

Процесс аудита цифровой инфраструктуры, в отличие от традиционного финансового аудита, характеризуется повышенной сложностью объектов проверки, динамичностью сред и высокой технической специализацией доказательств. Для обеспечения системности, полноты и соответствия профессиональным стандартам применяется структурированная методология, включающая четыре обязательных этапа: планирование, сбор доказательств, тестирование средств контроля и формирование итогового отчета. Каждый из этих этапов имеет

специфическое наполнение применительно к ИТ-среде и требует применения соответствующих инструментов и профессиональных суждений.

Планирование аудита цифровой инфраструктуры представляет собой фундаментальный этап, на котором определяются цели, объем и стратегия проверки. В контексте ИТ-аудита планирование начинается с понимания стратегических и операционных целей организации в области информационных технологий, а также с анализа ИТ-ландшафта в целом. Аудитору необходимо идентифицировать критичные для бизнеса системы, оценить степень их влияния на формирование финансовой и нефинансовой отчетности, а также выявить области, подверженные наибольшим рискам, таким как риски информационной безопасности, риски непрерывности деятельности и риски целостности данных. Существенной частью планирования является предварительная оценка системы внутреннего контроля, включая как общие средства контроля ИТ (ITGC — Information Technology General Controls), так и прикладные средства контроля, встроенные в бизнес-приложения. Для систематизации этой оценки используются специализированные вопросники и матрицы рисков.

Пример структуры вопросов для понимания ИТ-среды на этапе планирования представлен в табл. 8.10, основанной на подходах к структурированию аудиторских процедур .

Таблица 8.10

Фрагмент вопросника для понимания ИТ-среды на этапе планирования

Направление оценки	Ключевые вопросы для изучения	Источник информации
Организация ИТ-функции	Соответствует ли организационная структура ИТ-отдела масштабам деятельности? Разграничены ли обязанности разработчиков, администраторов и пользователей?	Штатное расписание, положения об отделах, должностные инструкции, интервью с руководителями.
Политики и процедуры	Утверждены ли политики информационной безопасности, управления изменениями, резервного копирования? Соответствуют ли они всем требованиям?	Внутренние регламенты, приказы об утверждении политик, журналы ознакомления сотрудников.

Направление оценки	Ключевые вопросы для изучения	Источник информации
ИТ-инфраструктура	Какова топология сети? Какое системное и прикладное программное обеспечение используется? Как организован доступ к критическим данным и серверам?	Схемы сети, конфигурационные файлы, данные инвентаризации, интервью с сетевыми администраторами.
Инциденты и проблемы	Были ли за последний период инциденты, связанные с нарушением безопасности или сбоями в работе систем? Как проводился их анализ и устранение?	Журналы регистрации инцидентов, отчеты по проблемам, акты расследований.

Результатом этапа планирования является разработка общего плана аудита и программы аудита, в которой детализируются характер, временные рамки и объем запланированных аудиторских процедур. На данном этапе также принимается важное решение о том, будет ли аудитор полагаться на средства контроля клиента для сокращения объема процедур по существу.

Сбор аудиторских доказательств в контексте цифровой инфраструктуры ориентирован на получение достаточных и надлежащих доказательств для формирования обоснованных выводов. Методы сбора доказательств в ИТ-аудите значительно шире традиционных и включают как ручные, так и автоматизированные процедуры. К числу основных методов относятся инспектирование (например, анализ настроек безопасности в операционной системе или базе данных), наблюдение (например, наблюдение за процедурой запуска системы в эксплуатацию), запросы и подтверждения (например, направление запросов администраторам баз данных), а также повторное выполнение (например, повторное проведение процедуры расчета заработной платы в тестовой среде). Специфическим для ИТ-аудита методом является использование специализированного программного обеспечения для анализа данных и конфигураций. Так, для инвентаризации активов и сбора информации о характеристиках компьютеров и периферии применяются инструменты автоматического сканирования сети, которые позволяют создать объективную базу данных о вычислительной технике без необходимости ручного обхода помещений. Структура собираемых данных при этом должна быть четко определена и

стандартизирована. Как показывает практика, даже для записей системного аудита (логов) критически важна строгая структура, включающая такие поля, как тип события, временная метка, источник, идентификатор и детализирующее сообщение. Эта структурированность позволяет в дальнейшем эффективно анализировать массивы данных.

Тестирование средств контроля является центральной процедурой при подтверждении надежности системы внутреннего контроля. В ИТ-аудите тестирование разделяется на две основные категории: тестирование общих средств контроля и тестирование прикладных средств контроля. Общие средства контроля (ITGC) охватывают среду, в которой функционируют прикладные системы, и включают контроль за процессами управления изменениями, доступа к данным и системам, а также за компьютерными операциями. Тестирование ITGC направлено на подтверждение того, что контрольная среда является стабильной и что прикладные средства контроля могут функционировать эффективно на протяжении всего проверяемого периода. В свою очередь, тестирование прикладных системах проверяет корректность обработки данных в конкретных программах. Пример такого подхода к тестированию требований безопасности и управления доступом приведен в табл. 8.11.

Таблица 8.11

Пример формата тестирования средств контроля доступа

Идентификатор контроля	Требование (Область контроля)	Описание тестовой процедуры	Ожидаемый результат
AC-01	Управление доступом (Access Control)	Запросить список пользователей, имеющих права администратора в критических системах. Сравнить его с кадровым составом ИТ-отдела и обоснованностью предоставления прав.	Список администраторов соответствует утвержденному перечню сотрудников, права предоставлены на основании служебной записки.
AU-02	Подотчетность и аудит (Audit and Accountability)	Выполнить попытку входа в систему с некорректными учетными данными и проверить, регистрируется ли это событие в журнале аудита с указанием времени, источника и результата попытки.	Событие неуспешной аутентификации зафиксировано в журнале безопасности с корректным заполнением всех обязательных полей.

Идентификатор контроля	Требование (Область контроля)	Описание тестовой процедуры	Ожидаемый результат
CM-03	Управление конфигурациями (Configuration Management)	Сравнить текущие настройки безопасности на эталонном сервере с утвержденным базовым стандартом конфигурации.	Настройки сервера соответствуют базовому стандарту конфигурации, отклонения отсутствуют или задокументированы и санкционированы.

В ходе выполнения тестов аудитор фиксирует выявленные отклонения, которые впоследствии классифицируются как недостатки (дефекты) системы контроля. Каждый недостаток оценивается с точки зрения его существенности и потенциального влияния на результаты деятельности организации.

Формирование отчета представляет собой заключительный и наиболее ответственный этап аудита, на котором результаты проведенных процедур и тестирования обобщаются и доводятся до сведения заинтересованных пользователей. Отчет по аудиту цифровой инфраструктуры должен быть не просто перечнем выявленных замечаний, а структурированным документом, содержащим описание цели и объема аудита, применявшейся методологии, а также выводы и рекомендации. Отчет, как правило, содержит введение с указанием оснований для проведения аудита и ограничений, связанных с использованием результатов; описание подхода и методов сбора доказательств; раздел, посвященный результатам аудита, где детально описываются выявленные сильные стороны и недостатки системы контроля с привязкой к конкретным областям (например, управление доступом, управление изменениями). Ключевым компонентом являются практические рекомендации по устранению выявленных недостатков, которые должны быть реалистичными и соответствовать специфике деятельности организации. В некоторых случаях в отчет включаются планы мероприятий по исправлению недостатков, разработанные руководством аудируемого лица. Качество итогового отчета является главным критерием эффективности проведенного аудита и основой для принятия управленческих решений по развитию и защите цифровой инфраструктуры предприятия.

8.4. Аудит облачной инфраструктуры: особенности и ключевые области проверки

Трансформация корпоративных архитектур, выражающаяся в миграции информационных систем в среду облачных вычислений, обусловливает необходимость пересмотра традиционных подходов к аудиту информационных технологий. Облачная инфраструктура, функционирующая на принципах виртуализации ресурсов, эластичности и модели коллективного использования, формирует принципиально иную среду контроля по сравнению с классическими центрами обработки данных. Как справедливо отмечается в актуальной редакции ISACA ITAF (5-е издание, 2026 г.)[36], объект аудита смещается от изолированных контролей к оценке целостных цифровых экосистем, охватывающих как внутренние сервисы, так и внешних поставщиков . Аудит облачной инфраструктуры представляет собой систематический процесс получения объективных свидетельств и оценки их независимым образом с целью определения соответствия конфигураций, политик безопасности и операционной эффективности предъявляемым требованиям, включая нормативные акты, стандарты и ожидания бенефициаров .

Ключевая особенность аудита облачной среды проистекает из разделяемой модели ответственности (shared responsibility model). В зависимости от модели обслуживания (IaaS, PaaS, SaaS) границы ответственности между поставщиком облачных услуг (CSP) и потребителем (CSC) смещаются, что напрямую влияет на глубину и методы аудиторских процедур. Объектом проверки выступают не столько физические активы, сколько сложные композитные сущности: виртуальные машины, сети, хранилища и контейнеры, существующие в виде программно-определяемых конфигураций. Методологическая основа для проведения таких проверок закладывается в международных стандартах, в частности, в ISO/IEC TR 3445:2022, который описывает роли и взаимодействие сторон, участвующих в аудите, а также подходы к формированию доверия к облачным услугам . Дальнейшее развитие этой области нашло отражение в проекте ISO/IEC PWI TS 3445, нацеленном на унификацию схем аудита и сертификации .

[36] ISACA Revamps IT Audit Framework [Электронный ресурс]// Режим доступа: https://www.fromtech.ru/blog/chto-znachit-on-premise/ (дата обращения: 06.03.2026).

Для структурирования процесса аудита и обеспечения полноты покрытия контролей целесообразно выделить пять ключевых областей проверки, каждая из которых характеризуется специфическими рисками и объектами оценки.

Данные области образуют иерархическую структуру, представленную в табл. 8.12.

Фундаментом для технической оценки выступает понимание модели реализации облачных услуг, в особенности для модели IaaS.

Таблица 8.12

Ключевые области аудита облачной инфраструктуры

Область аудита	Объекты проверки	Типовые риски
Управление доступом и идентификация (IAM)	Политики управления доступом (RBAC, ABAC), привилегированные учетные записи, сервисные аккаунты, механизмы MFA, федерация удостоверений.	Нарушение принципа наименьших привилегий, бесконтрольное использование сервисных аккаунтов, компрометация учетных данных с высокими привилегиями.
Безопасность конфигураций и уязвимости	Базовые образы, инфраструктура как код (IaC), правила сетевого экранирования (Security Groups, ACL), настройки сервисов хранения.	Небезопасные конфигурации по умолчанию, публичный доступ к хранилищам данных, избыточно открытые сетевые порты.
Защита данных и управление ключами	Классификация данных, шифрование в покое и при передаче (TLS), политики ротации ключей, механизмы DLP, соответствие требованиям резистентности.	Отсутствие шифрования критичных данных, неэффективное управление ключами шифрования, несоблюдение требований к локализации данных.
Мониторинг и управление инцидентами	Централизованный сбор логов (SIEM), целостность временных меток, планы реагирования (IR), уведомления о событиях безопасности.	Недостаточная глубина логирования, отсутствие корреляции событий, неактуальность или нефункциональность планов реагирования.
Соответствие нормативным требованиям	Соответствие отраслевым стандартам (PCI DSS, HIPAA) и GDPR, правомерность обработки персональных данных, юридическая значимость действий в облаке.	Нарушение регуляторных требований из-за неверной интерпретации модели ответственности, использование несертифицированных облачных сервисов.

Как детализировано в национальном стандарте ГОСТ Р 56045-2021, архитектура облачной среды включает четыре основных компонента: физические ресурсы (серверы, СХД, сетевое оборудование), механизмы виртуализации (гипервизоры, виртуальные коммутаторы), виртуальные ресурсы (виртуальные машины, сети) и систему управления услугами (портал самообслуживания, оркестратор).

Аудиторская проверка на данном уровне должна быть направлена на оценку корректности реализации изоляции между арендаторами, надежности гипервизора и защищенности плоскости управления. Аудитору надлежит оценить, применяются ли механизмы виртуализации таким образом, чтобы исключить несанкционированный доступ к данным «соседних» арендаторов, что является критическим требованием для публичных облаков.

Оценка управления доступом (IAM) выходит за рамки простой проверки наличия парольной политики. Современный аудит облачных IAM требует анализа матрицы прав доступа на предмет избыточных привилегий, выявления «мертвых» учетных записей и оценки безопасности сервисных аккаунтов, используемых для межсервисного взаимодействия. Верификация настроек многофакторной аутентификации (MFA) для привилегированных пользователей и консольного доступа является обязательной процедурой.

В контексте растущей динамики изменений облачной инфраструктуры традиционный периодический аудит уступает место модели непрерывной оценки соответствия. Применение инструментов класса CSPM и CNAPP позволяет автоматизировать сбор свидетельств и мониторинг отклонений от заданных базовых конфигураций в реальном времени. Интеграция аудиторских процедур в конвейеры CI/CD обеспечивает проверку декларативного кода инфраструктуры до его развертывания в продуктивной среде, что значительно снижает вероятность появления неустранимых несоответствий.

Наконец, существенное значение приобретает аудит внешних провайдеров и цепочек поставок, что коррелирует с развитием концепции «цифровых экосистем» в ITAF 5. Аудитор должен оценить не только самого поставщика облачных услуг, но и зависимости от субподрядчиков. Кроме того, аудит облачной инфраструктуры трансформируется в многомерную дисциплину, сочетающую глубокие технические знания архитектуры виртуализации, понимание регуляторного ландшафта и владение современными инструментами автоматизированной оценки.

Таким образом, аудит цифровой инфраструктуры представляет собой самостоятельное и динамично развивающееся направление контрольной деятельности, выделившееся в ответ на возрастающую роль информационных технологий как стратегического актива и источника рисков. В отличие от финансового аудита, ориентированного на ретроспективную оценку достоверности отчетности, аудит ИТ-среды нацелен на получение независимой оценки текущего состояния, эффективности использования ресурсов, управляемости, безопасности и соответствия нормативным требованиям.

Многообразие целей проверки объективно обусловливает дифференциацию аудита на специализированные виды: операционный аудит, аудит соответствия, аудит производительности и аудит безопасности. Каждый из этих видов характеризуется собственным объектом, методическим инструментарием и критериями оценки, что отражено в разработанной классификации. При этом данные виды не являются взаимоисключающими, а в комплексе формируют целостное представление о состоянии цифровой инфраструктуры организации.

В свою очередь, методологическую основу аудита цифровой инфраструктуры составляют специализированные стандарты и фреймворки, выполняющие функции как нормативной базы для проверки, так и руководства по построению эффективных систем управления. Установлено, что COBIT 2019 выступает в роли интегрирующей методологии стратегического управления ИТ, ISO/IEC 27001 задает требования к системе менеджмента информационной безопасности и встраивает аудит в замкнутый цикл управления PDCA, а PCI DSS 4.0.1 предоставляет детализированный и императивный набор правил для защиты среды платежных данных. Понимание областей применения и взаимосвязей между этими документами является фундаментальной компетенцией аудитора.

Сегодня процесс аудита цифровой инфраструктуры базируется на универсальной методологической последовательности, включающей этапы планирования, сбора доказательств, тестирования средств контроля и формирования отчета. Специфика ИТ-аудита проявляется в расширении методов сбора доказательств (инструментальное сканирование, анализ конфигураций, нагрузочное тестирование) и в разделении тестирования на общие средства контроля ИТ и прикладные

средства контроля. Итоговый отчет, содержащий не только выявленные недостатки, но и обоснованные рекомендации, служит основой для принятия управленческих решений по развитию и защите инфраструктуры.

Подводя итог сказанному выше, нужно заключить, что трансформация архитектур в сторону облачных вычислений обусловила необходимость адаптации традиционных аудиторских подходов. Ключевой особенностью аудита облачной инфраструктуры выступает модель разделяемой ответственности, определяющая границы проверки. Объектами оценки здесь становятся программно-конфигурируемые компоненты: виртуальные машины, политики IAM, сетевые экраны безопасности и IaC. Методология аудита в этой среде смещается от периодических проверок к модели непрерывной оценки соответствия с применением инструментов класса CSPM и CNAPP, а также требует анализа цепочек поставок и зависимостей от субподрядчиков облачных провайдеров.

Вопросы для обсуждения

1. Дайте определение аудита цифровой инфраструктуры. В чем состоит ключевое его отличие от финансового аудита?
2. Перечислите основные задачи и направления аудита цифровой инфраструктуры.
3. Охарактеризуйте особенности классификации видов аудита цифровой инфраструктуры по целям и задачам.
4. Укажите, какие документы составляют современную нормативно-правовую базу проведения аудита цифровой инфраструктуры.
5. Поясните основные направления аудита информационной безопасности.
6. Перечислите основные стандарты и фреймворки для аудита.
7. Укажите особенности применения стандарта COBIT при проведении аудита цифровой инфраструктуры.
8. Сравните основные принципы работы с COBIT 5 и COBIT 2019. В чем состоят их принципиальные различия?
9. Укажите типологию аудитов информационной безопасности по ISO 19011 и ISO 27001

10. Объясните особенности применения стандарта PCI DSS версии 4.0.1 при проведении аудита.

11. Поясните структуру и объем контрольных мероприятий стандарта PCI DSS 4.0.1

12. Поясните, каким образом строятся опросники и матрицы рисков при проведении аудита цифровой инфраструктуры.

13. Укажите специфику и направления сбора аудиторских доказательств в контексте аудита цифровой инфраструктуры.

14. Объясните специфику тестирования средств контроля в рамках проведения аудита цифровой инфраструктуры.

15. Дайте характеристику этапа «формирование отчета» проведения аудита инфраструктуры.

16. Поясните ключевые особенности аудита облачной инфраструктуры.

17. Перечислите основные нормативные документы, в которых содержатся положения о проведении аудита облачных инфраструктур.

18. Назовите основные причины, по которым понимание модели реализации облачных услуг выступает в роли фундамента технической оценки их качества.

19. Объясните, каким образом происходит оценка управления доступом при аудите облачной инфраструктуры.

20. Поясните основные направления аудита внешних провайдеров и цепочек поставок.

Практические задания

Задание 1. Определите, к какому виду аудита (операционный, соответствия, производительности или безопасности) относится каждая из описанных ситуаций. Кратко обоснуйте свой выбор, указав ключевой критерий оценки.

1. В ходе проверки аудитор установил, что фактическое время восстановления работоспособности критической информационной системы после сбоя превышает предельно допустимые значения, зафиксированные в соглашении об уровне услуг (SLA) с бизнес-подразделениями.

2. Организация планирует получить сертификат соответствия стандарту PCI DSS. Аудиторская группа проверяет, сегментирована ли

сеть таким образом, чтобы среда передачи данных держателей карт (CDE) была изолирована от остальной корпоративной сети.

3. Внедрение новой ERP-системы сопровождается жалобами пользователей на длительное ожидание при формировании сложных отчетов. Аудитор проводит анализ технологических журналов сервера баз данных для выявления наиболее затратных по времени запросов.

Задание 2. Определите, какой стандарт или фреймворк (COBIT 2019, ISO/IEC 27001 или PCI DSS) является наиболее релевантным для решения каждой из перечисленных ниже задач аудита. Поясните свой ответ, указав на конкретную характеристику выбранного документа.

1. Перед руководством компании стоит задача оценить зрелость процесса управления изменениями в ИТ и его вклад в достижение стратегических бизнес-целей. Аудитору необходимо выбрать методологию, которая позволит провести такую оценку на основе модели CMMI.

2. Поставщик облачных услуг (провайдер IaaS) уровня 1, обрабатывающий более 300 тыс. транзакций по картам в год, обязан пройти ежегодную проверку для подтверждения безопасности среды данных держателей карт своих клиентов.

3. Крупный банк проводит внутренний аудит системы менеджмента информационной безопасности (СМИБ). Аудиторам необходимо проверить выполнение требования п. 9.2 стандарта, а также оценить полноту и актуальность «Заявления о применимости».

Задание 3. Компания ООО «Контент Плюс» использует публичную облачную инфраструктуру (модель IaaS) для хранения и обработки персональных данных. Ранее аудит выявил риск «бесконтрольного использования сервисных аккаунтов с высокими привилегиями».

Необходимо:

1. Определить, к какой из ключевых областей аудита облачной инфраструктуры относится данный риск.

2. Разработать описание одной тестовой процедуры, направленной на проверку данного риска, включая:

- Требования (область контроля), т.е. формулировки того, что именно должно контролироваться.

- Описание тестовой процедуры, т.е. характеристика конкретные действия аудитора.

- Ожидаемый результат, т.е. необходимо указать, какой исход будет свидетельствовать об эффективности контроля.

Тест для самоконтроля

1. Главная цель аудита цифровой инфраструктуры заключается…?

а) В получении объективной и независимой оценки текущего состояния ИТ-среды организации.

б) В оценке пропускной способности сетевого оборудования

в) В оценке работы внедренного фреймворка в работу компании;

г) В оценке соответствия финансовым ресурсам компании.

2. В чем состоит основная цель операционный аудита инфраструктуры?

а) Выявление узких мест и причин деградации скорости работы ИТ-систем и сервисов.

б) Подтверждение соответствия ИТ-инфраструктуры требованиям законодательства, стандартов и внутренних регламентов.

в) Оценка эффективности, экономичности и результативности использования ИТ-ресурсов в бизнес-процессах.

г) Выявление уязвимостей и оценка уровня защищенности инфраструктуры от внешних и внутренних угроз.

3. Основная цель этого аудита - выявление узких мест и причин деградации скорости работы ИТ-систем и сервисов. О каком виде аудита идет речь?

а) Операционный аудит.

б) Аудит соответствия.

в) Аудит безопасности.

г) Аудит производительности.

4. Ключевыми объектами проверки этого аудита являются документация, настройки средств защиты, процедуры обработки данных. О каком виде аудита идет речь?

а) Аудит производительности.

б) Аудит безопасности.

в) Аудит соответствия.

г) Операционный аудит.

5. Что не входит в основные направления аудита информационной безопасности?

а) Анализ защищенности и физической безопасности

б) Тестирование на проникновение.

в) Анализ документации компании.

г) Анализ соответствия технических мер защиты.

6. Основные виды аудита (операционный, соответствия, производительности и безопасности) ….

а) Не являются взаимоисключающими, а напротив, дополняют друг друга.

б) Являются взаимоисключающими.

в) Работают совершенно независимо друг от друга.

г) Не могут производиться одновременно.

7. Какой из представленных документов не входит в нормативную базу проведения аудита цифровой инфраструктуры.

а) COBIT;

б) ISO 27001;

в) PCI DSS;

г) Федеральные стандарты бухгалтерского учета (ФСБУ).

8. Сколько процессов содержит COBIT 2019?

а) 36;

б) 38;

в) 40;

г) 42.

9. Процессы «определение и поддержка архитектуры предприятия», «обеспечение оптимизации рисков», «обеспечение прозрачности для заинтересованных сторон» COBIT 2019 относятся к доменам….

а) DSS (Deliver, Service and Support);

б) APO (Align, Plan and Organize);

в) BAI (Build, Acquire and Implement);

г) EDM (Evaluate, Direct and Monitor).

10. Аудит на соответствие этому стандарту направлен на проверку не только наличия формализованных политик и процедур, но и их практической эффективности.

а) PCI DSS;

б) COBIT;

в) ISO/IEC 27001;

г) ISO 19011.

11. Основная цель этого стандарта состоит в управление и контроль ИТ для достижения бизнес-целей, создание ценности через ИТ.

а) ISO 19011;

б) PCI DSS;

в) ISO/IEC 27001;

г) COBIT 2019.

12. Особое внимание в стандарте версии 4.0.1 уделяется …

а) управлению рисками, связанными с поставщиками услуг;

б) управлению рисками, связанными с обеспечиванием информационной безопасности;

в) управлению финансовыми рисками;

г) управлению рисками, связанными с сетевыми ресурсами.

13. Результатом этапа планирования является…

а) обоснование выбранного для аудита фреймворка;

б) разработка общего плана аудита и программы аудита;

в) назначение команды аудиторов;

г) обоснование сроков начала и окончания аудита..

14. Что является специфическим для ИТ-аудита методом проведения аудита цифровой инфраструктуры.

а) Использование типовых финансовых показателей.

б) Использование программного обеспечения общего пользования.

в) Использование специализированного программного обеспечения для анализа данных и конфигураций.

г) Использование сетевых ресурсов удаленного доступа.

15. Какой этап является заключительным при проведении аудита цифровой инфраструктуры?

а) Формирование отчета.

б) Оценка рисков.

в) Проверка работоспособности сетевых ресурсов

г) Проверка работоспособности системного программного обеспечения.

16. Что используется для формирования предварительной оценки системы внутреннего контроля?

а) Матрицы соответствия.

б) Графики рисков и соответствия.

в) Специализированные вопросники и матрицы рисков.

г) Справочные табличные формы.

17. В чем заключается ключевая особенность аудита облачной среды? разделяемой модели ответственности

а) В разделяемой модели ответственности.

б) В разделяемой модели соответствия.

в) В разделяемой модели рисков.

г) В разделяемой модели формы.

18. Какой области аудита облачной инфраструктуры соответствует политики управления доступом, привилегированные учетные записи, сервисные аккаунты, механизмы MFA, федерация удостоверений?

а) Мониторинг и управление инцидентами.

б) Защита данных и управление ключами.

в) Безопасность конфигураций и уязвимости.

г) Управление доступом и идентификация.

19. В рамках проведения аудита цифровой инфраструктуры выявлено нарушение регуляторных требований из-за неверной интерпретации модели ответственности, использование несертифицированных облачных сервисов. Какой области аудита это соответствует?

а) Управление доступом и идентификация.

б) Соответствие нормативным требованиям.

в) Мониторинг и управление инцидентами.

г) Безопасность конфигураций и уязвимости.

20. Что не требует аудит современный аудит облачных IAM?

а) Анализ матрицы прав доступа на предмет избыточных привилегий

б) Выявление «мертвых» учетных записей.

в) Время работы ПО общего назначения.

г) Оценки безопасности сервисных аккаунтов.

Глава 9. ПОСТРОЕНИЕ СИСТЕМЫ УПРАВЛЕНИЯ И АУДИТА ЦИФРОВОЙ ИНФРАСТРУКТУРЫ КОМПАНИИ

9.1. Разработка KPI и метрик для оценки эффективности управления инфраструктурой

При управлении цифровой инфраструктурой возникает вопрос: по каким признакам можно оценить эффективность ее менеджмента. Некоторые параметры системы измерить невозможно, и анализ качества ее работы определяется эмпирическим путем или интуитивно, приблизительно. Очевидно, что такие метрики не могут дать точный и конкретный ответ на поставленный перед аналитиком вопрос о качестве работы построенной инфраструктуры. Поэтому актуальным становится введение KPI, которые должны формализовать показатели и дать им четкое, математическое выражение. Но некоторые привычные ИТ-показатели – безотказной работы «uptime», загрузка CPU – для бизнеса бесполезны, а иногда вводят в заблуждение пользователя.

Долгое время эффективность ИТ оценивали по уровню доступности. 99,9% аптайма звучит как надежный показатель стабильности и устойчивой работы системы. Если система формально доступна, но фактически работает настолько медленно, что сотрудники испытывают стресс, а клиенты закрывают вкладку и уходят к конкурентам, то такие значения этого показателя ничего не значат. Аналитики IDC[37] неоднократно отмечали эту проблему: ИТ-руководители часто сосредотачиваются на метриках, которые не связаны с экономическими результатами и существуют отдельно от них. Количество инцидентов, стоимость владения и «uptime» - все эти показатели имеют значение, но они не дают ответа на основной вопрос о том, как технологии способствуют продвижению компании или они не взаимосвязаны с ней.

Например, из-за сбоя в системе маршрутизации служба доставки теряет один рабочий день. Серверы работают исправно, «uptime» составляет 100%. Очевидно, что точной информации о выручке в данном случае нет, следовательно, подсчет показателей ради самих показателей не имеет смысла и необходимо сосредоточиться на оценке их

[37] IDC - Data Analytics// [Электронный ресурс] Режим доступа: https://www.idc.com/data-analytics/ (дата обращения 03.03.2026)

вклада в бизнес и определить эффективность цифровой инфраструктуры с точки зрения продаж, производства и HR.

Для упорядочения показателей необходимо выстроить иерархию. Для этих целей важно иметь ввиду, что показатели можно разделить на три уровня: бизнес, сервис, технология.

1. Бизнес-уровень. На вершине - показатели, которые напрямую связаны с денежными средствами, скоростью изменений и репутацией.

Одним из них является «North Star Metric». У каждой компании есть своя «полярная звезда». В инфраструктуре это может быть скорость вывода продукта на рынок или MTTR для систем, приносящих выручку. Если интернет-магазин падает в «черную пятницу», важнее считать не минуты восстановления сервера, а потерянную выручку в минуту.

Еще одним показателем является «DXI» – индекс цифровой трансформации. По версии IDC, имеет смысл смотреть шире: насколько активно используются облака, ИИ, автоматизация – и как все это влияет на клиентскую ценность. Это уже не техаудит, а попытка измерить трансформацию бизнес-модели.

И, наконец, показатель «Time-to-Value (TTV)», который позволяет оценить, сколько проходит времени от запроса бизнеса до момента, когда команда реально начинает работать.

2. Качество сервиса и пользовательский опыт. Этот уровень — своего рода переводчик между стратегией и техникой. Показателем этого уровня является «доступность с поправкой на производительность». Мало понимать, что сервис работает, важно знать насколько он быстро выполняет свои функции. Например, 95-й перцентиль времени отклика: если 95% запросов укладываются в 500 мс, а оставшиеся 5% тянутся по 5 секунд — это тревожный сигнал. И пользователи его почувствуют.

Показатель удовлетворенности. NPS, короткие опросы после закрытия инцидента – все это помогает увидеть, как цифровая инфраструктура выглядит со стороны. Можно идеально закрыть тикет и при этом не решить проблему по сути. Иногда просроченные заявки напрямую влияют на премирование, в результате просрочек почти не будет, что демонстрирует повышение мотивация сотрудников.

3. Технологический фундамент. Это показатели так называемой «физики процессов», без которой невозможны все другие трансформации.

В рамках данной группы показателей могут быть рассмотрены четыре золотых сигнала SRE: задержка, трафик, ошибки и насыщенность. Подход, популяризированный практиками site reliability engineering, дает целостную картину состояния системы. Отслеживание насыщения ресурсов, например, позволяет заранее увидеть, что диск заполнится через две недели, а не внезапно ночью.

Уровень ошибок. Если платежный шлюз формально доступен, но 10% пользователей не могут завершить оплату из-за неудачного интерфейса, техническая «доступность API» будет 100%, а бизнес-конверсия — падать. Регулярный анализ помогает отличать случайные сбои от системных дефектов релиза.

Таким образом, хорошим вариантом KPI является минимализм в определении существенных показателей, вместо сбора множества малозначительных данных. Есть вариант измерять абсолютно все показатели, но в итоге появляется список из 50 показателей, при чем ни один из них не работает. Практика показывает: 2–3 ключевых KPI на команду обычно достаточно.

Принципы построения системы показателей оценки эффективности цифровой инфраструктуры могут быть сведены к следующим:

1. Один показатель – одна цель. Если в метрике смешаны рост, удержание и лояльность, получится абстрактное усредненное значение, которое не будет нести ценности для лица, принимающего решения.

2. Управляемость. Команда должна понимать, какие действия влияют на показатели. Иначе KPI превращается в фоновый шум.

3. Связь с ценностью. Если нельзя объяснить, как показатель отражается на клиенте или выручке, возможно, это просто технический индикатор для внутреннего пользования.

Еще одной практикой при определении эффективности системы является применение сравнительного анализа. Например, время реакции на инцидент – 15 минут. Интерпретировать этот показатель без сравнений не представляется возможным. Нужна либо динамика относительно него по сравнению с предыдущими значениями, либо его сопоставление с рыночными данными.

Исследования специалистов по KPI-аналитике показывают: сочетание метрик с элементами сравнительного анализа помогает задать базовую линию (baseline), внедрить улучшения и затем проверить эффект. Такой подход снижает дефекты и делает управление более прозрачным, ближе к data-driven модели.

Но и тут есть нюанс. В Axenix[38] предупреждают: механически переносить западные ориентиры, например от Gartner или IDC, на российскую практику рискованно. Контекст, масштаб, стек технологий — все имеет значение.

Зачастую цифры создают иллюзию контроля. Они достоверны, но их легко неправильно понять. Например, деградация RAID-массива. Система обслуживает запросы, доступность идеальна. Но риск потери данных вырос в разы. Автоматические датчики это не всегда подсветят.

Есть и так называемые «vanity metrics» - метрики тщеславия. Общее количество серверов или терабайты хранилища звучат внушительно, но сами по себе ничего не говорят об эффективности. Гораздо показательнее количество ресурсов, которые приносят пользу, а не простаивают.

Кроме того, KPI не является универсальной и статичной системой показателей. То, что критично для стартапа (скорость поиска product-market fit), для зрелой компании может отойти на второй план, уступив место стабильности и предсказуемости. Показатели стоит пересматривать хотя бы раз в квартал или после серьезного изменения стратегии.

Итак, разработка KPI для инфраструктуры — это не однократное определение системы показателей и метрик для данной компании с определенной архитектурой. Это постоянная и сложная работа, поддерживающая связь между структурой ИТ и бизнесом. В качестве ориентиров для разработки системы показателей можно принять следующие. Во-первых, отталкиваться необходимо от стратегии. Во-вторых, необходимо искать причинно-следственные связи между сбоями в работе инфраструктуры и бизнес-процессах. В-третьих, автоматизация

[38] AXENIX — консалтинг, технологии и цифровые решения для бизнеса//[Электронный ресурс] Режим доступа: https://axenix.pro/?ysclid=mmj34kea5z369827500 (дата обращения 09.03.2026)

сбора данных, дашборды в реальном времени, гораздо полезнее отчетов, которые читают раз в квартал. В-четвертых, использование KPI должно быть не для поиска виноватых в проблемах, а для выявления «узких мест».

Только при таком, многослойном и гибком подходе цифровая инфраструктура перестает быть «черным ящиком», который бесконечно потребляет бюджет, и превращается в понятный, управляемый механизм роста бизнеса, что является главным критерием успеха бизнеса.

9.2. Подготовка отчета для руководства: технические и бизнес-аспекты

Финалом любой проверки является составление отчета. От того, как он написан, зависит формирование бюджета на следующий год, решение о модернизации, реструктуризация компании и др. Отчет для руководства - это не просто документ, это инструмент продажи идей и обоснования решений.

Золотым правилом создания отчетов является то, что один отчет не может быть предназначен для двух разных аудиторий. Технический специалист хочет видеть конфигурационные файлы, дампы памяти и детальные логи. Директор хочет видеть деньги, риски и сроки. Поэтому современный подход к отчетности по ИТ-аудиту подразумевает создание документа, имеющего как минимум два уровня глубины.

Примерно такую философию закладывают в свои инструменты даже разработчики open-source решений для аудита: они генерируют единый отчет, но четко разделяют в нем исполнительное резюме (executive summary) для директора и детальную техническую часть для инженеров. Это не просто дань вежливости, это прагматичное решение: топ-менеджмент принимает решения на основе анализа, а не на основе первичных данных.

Исполнительное резюме – это самая важная часть отчета. Его задача – ответить на три вопроса, которые волнуют любого руководителя: возможные убытки из-за сбоев системы, безопасность хранения данных, перспективы развития бизнеса и их стоимость. При этом важно правильно и доступно для пользователей интерпретировать полученные данные: для технических специалистов сделать акцент на

ИТ-составляющей, а для руководителя и сотрудников финансовых служб – на коммерческой.

Структура исполнительного резюме обычно состоит из контекста и цели, ключевых выводов, бизнес-влияния, рекомендаций и дорожной карты.

Раздел «Контекст и цели» начинается, как правило, с одного-двух предложений о том, зачем проводился аудит. Например, «… В связи с участившимися жалобами клиентов на скорость работы личного кабинета и планами по выходу на маркетплейсы...».

В разделе «Ключевые выводы» содержится список из 3-5 самых важных проблем, которые реально угрожают бизнесу или открывают перед ним новые возможности. Необходимо сразу группировать проблемы по категориям: критические риски, операционные недостатки, точки роста.

Раздел «Бизнес-влияние» - самый важный блок. Здесь связывается каждый вывод с деньгами, репутацией или временем. Например, «Из-за устаревшей системы резервного копирования (глубина копирования — 1 неделя) при вирусной атаке мы гарантированно потеряем данные за последние 5-6 рабочих дней. Стоимость восстановления силами сторонних экспертов оценивается в X рублей, плюс репутационные потери от простоя…».

«Рекомендации и дорожная карта» - раздел резюме, в котором предлагаются не отдельные шаги по решению проблем, а направления стратегии компании. Например, «Этап 1 (1-2 месяца): внедрение облачной резервной копии критических данных для снижения рисков потери. Этап 2 (6 месяцев): миграция серверов базы данных на новое оборудование с отказоустойчивым кластером». Обязательно с визуализацией – диаграммами Ганта или простыми таблицами этапов.

Иногда в резюме выносят так называемую таблицу разрыва (Gap Analysis). Это сравнение того, где компания находится сейчас, и где она должна быть, чтобы соответствовать стратегии. Например, текущее состояние — «локальный дата-центр, загрузка 90%», целевое состояние — «гибридное облако, автоматическое масштабирование». Разрыв — «нет опыта работы с облаками, нет бюджета». Это очень наглядно и сразу показывает масштаб работ.

Если резюме пишется для пользователей, мало понимающих в ИТ, то основная часть — это территория профессионалов. Здесь уже

можно и нужно использовать специальную терминологию, потому что пользователи этой информации – это системные администраторы, DevOPS-инженеры и руководители технических отделов, которым предстоит эти проблемы исправлять.

Но и здесь есть свои правила. Технический отчет должен быть структурирован так, чтобы в нем можно было легко найти ответ на любой вопрос. Идеальный вариант – следовать стандартной академической или корпоративной структуре.

Типовая структура технической части: введение, характеристика объекта, анализ по областям, каталог, заключение, рекомендации и приложения. Приведем содержание каждой части.

1. Введение. Объект и предмет аудита, методы сбора данных (интервью, анализ логов, сканирование, пентест), сроки проведения.

2. Общая характеристика объекта. Краткое описание инфраструктуры. Лучше всего в цифрах и схемах. Например, количество физических/виртуальных серверов с разбивкой по ролям; сетевая архитектура (лучше представить актуальную схему, а не ту, что рисовали 5 лет назад); используемое ПО (с версиями и сведениями о лицензиях); количество пользователей и рабочих мест.

Детальный анализ по областям. Аппаратная часть: оценка физического состояния оборудования, сроки эксплуатации, загрузка ресурсов (CPU, RAM, дисковая подсистема). Хороший пример: «На сервере баз данных наблюдается высокая нагрузка на диск (95-99% времени отклика), что является узким местом и причиной замедления работы 1С». Программное обеспечение: анализ версий, лицензионная чистота, соответствие количеству рабочих мест. Отдельно должен быть проведен анализ конфигураций. Безопасность: обязательно должны быть не только выводы, но и доказательства. Например, распечатки логов с неудачными попытками входа, скриншоты уязвимостей, сведения о политике паролей. Резервное копирование - один из самых больных вопросов. В отчете нужно указывать не просто «бекапы есть», а глубину копирования, регулярность, тип носителей (на тех же дисках, что и исходные данные – это риск), и главное – результаты тестового восстановления. Потому что бекап, который нельзя восстановить, - это просто мусор на диске. Сеть и связь - диагностика СКС, анализ загрузки каналов связи, качество работы VoIP, если есть.

3. Каталог рисков. Перечень выявленных проблем, классифицированных по степени критичности (критические, высокие, средние, низкие). Для каждого риска – описание, потенциальные последствия и предлагаемое решение.

4. Заключение и рекомендации. В технической части рекомендации должны быть максимально конкретными. Не «усилить безопасность», а «настроить политику блокировки учетных записей после 5 неудачных попыток входа в AD». Не «модернизировать сервер», а «заменить дисковой массив на SSD NVMe для сервера 1С, добавить 32 ГБ ОЗУ».

5. Приложения. Сюда входит все, что перегружает основной текст: детальные логи, полные схемы сети, технические паспорта оборудования, скриншоты настроек, опросные листы сотрудников.

Отдельный уровень мастерства при подготовке отчета – это формулировка замечаний. Они должны быть максимально конкретными и констркивными, в них должна быть описана проблема, а не виновный в ее возникновении.

Особенно это важно при аудите по заказу внешних консультантов. Итоговый отчет должен давать объективную картину, но не создавать неблагоприятный психологический климат в коллективе. Руководству важно понять, что проблемы системные, а не личные, которые решить практически невозможно.

Еще одним важным компонентом составления качественного отчета является визуализация данных, предусматривающая наличие таблиц и графиков вместо длинного текста. Мозг человека устроен так, что картинку он воспринимает быстрее, чем текст. Поэтому хороший отчет (и executive summary, и техническая часть) должен быть богат визуальными элементами. К таким средствам визализации относятся:

– диаграммы Ганта – для отображения плана работ;

– круговые диаграммы – для распределения оборудования по срокам эксплуатации (например, сколько процентов старше 5 лет);

– графики – для демонстрации нагрузки на каналы и серверы во времени;

– таблицы сравнения – лучший способ показать разрыв между текущим и желаемым состоянием. В одной колонке — «Как есть», в другой — «Как должно быть», в третьей — «Что делать». В хороших

отчетах ИТ-блоков всегда есть слайды с цифрами — «IT-блок в цифрах», «Реализуемые проекты», «Статистика обращений в HelpDesk». Это создает ощущение прозрачности и управляемости.

Самая большая опасность при подготовке отчета – забыть, зачем он нужен. Отчет – это не архивный документ, который ляжет на полку. Это инструмент для принятия решений. Поэтому финалом любой презентации результатов аудита (особенно, если делается устная защита отчета) должен звучать так: «Если мы сделаем А, Б и В, мы получим рост производительности труда на 15%, снижение простоев на 20% и сэкономим Х миллионов на лицензиях в следующем году».

Только когда руководство видит прямую связь между техническими изменениями и бизнес-результатом, оно готово выделять бюджет. И задача аудитора – сделать эту связь очевидной, кристально чистой и подтвержденной цифрами из отчета.

Таким образом, подготовка отчета для руководства – это искусство компромисса. Хороший отчет не просто констатирует факты – он продает будущее. Будущее, в котором инфраструктура работает как часы, бизнес зарабатывает больше, а ИТ-директор спокойно выполняет свои функции.

9.3. Финансовый аудит ИТ и учет совокупной стоимости владения

В практике ИТ-аудита существует устойчивый миф: достаточно посчитать, сколько денег потрачено на «железо» и софт, и картина ясна. На самом деле, это лишь часть аудита. Финансовый аудит цифровой инфраструктуры начинается там, где заканчивается бухгалтерия. Бухгалтер имеет точную стоимость сервера, а финансовый аудит должен ответить на вопрос: во что реально обходится владение этим сервером на протяжении трех лет, включая зарплату администратора, электроэнергию, охлаждение, стоимость простоев и упущенную выгоду.

IT Financial Management (ITFM) позволяет принимать решения об инвестициях в ИТ на основе данных, а не интуиции. Центральное место здесь – совокупная стоимость владения (Total Cost of Ownership, TCO). Это не просто цифра в отчете, а инструмент стратегического планирования.

Когда компания покупает новый сервер или внедряет CRM, все помнят про цену этого имущества и стоимость лицензий. Однако, как показывают исследования, прямые затраты (капитальные расходы, CAPEX) составляют лишь 30-40% от реальной стоимости владения. Остальное – это операционные расходы (OPEX), которые часто находятся в разных бюджетах компании и их трудно отследить.

Классическая модель TCO делит затраты на несколько групп: прямые, косвенные, скрытые и условные.

1. Прямые затраты (капитальные и операционные):

– закупочная цена - стоимость оборудования, программного обеспечения, лицензий. Сюда же входят первоначальные расходы на внедрение и услуги интеграторов;

– обучение персонала - часто забываемая статья, в которой может быть отражено количество времени и денег, необходимых на обучение сотрудников для работы с новой системой. В случае с современными AI-решениями или сложными HCI-платформами это могут быть сотни человеко-часов.

2. Косвенные затраты (эксплуатационные):

– инфраструктура - электроэнергия, охлаждение, аренда стойкомест в ЦОДе. Для энергоемкого оборудования, такого как GPU-серверы для обучения нейросетей, эти расходы могут достигать 30% от стоимости самого железа ежегодно;

– заработная плата (и время): текущее администрирование, обновление версий, установка патчей безопасности. В расчете TCO крайне важно учитывать время сотрудников, потраченное на поддержку системы. Например, если система требует ручного вмешательства раз в неделю, это часы высокооплачиваемого специалиста, которые он не тратит на развитие;

– простои (downtime) – это самая коварная статья: аудит должен оценивать стоимость часа простоя критической системы для бизнеса. Если система генерации отчетов «виснет» на два часа каждый понедельник, это не просто нервы сотрудников – это прямые убытки, которые должны быть включены в TCO как риск или фактический ущерб.

3. Скрытые и условные затраты:

– shadow IT: отделы маркетинга или продаж, не дожидаясь ИТ, подключают свои SaaS-сервисы (ChatGPT, Midjourney). Эти расходы

часто не консолидируются, и их аудит может выявить переплату в 20-30% на дублирующих подписках;

– привязка к поставщику (Vendor Lock-in): если выбрано нишевое решение, которое плохо интегрируется с остальным ландшафтом, будущая миграция с него может стоить миллионов. Оценка TCO на горизонте 3-5 лет обязана учитывать эти риски.

Методология расчета: от бэк-офиса до стратегии.

Существует несколько подходов к расчетам. В академической среде, например, предлагается использовать многоуровневую классификацию затрат (по этапам жизненного цикла, по вероятности возникновения) с привлечением экспертных оценок и имитационного моделирования.

На практике для аудитора важно не просто получить цифру, а понять структуру. Один из эффективных методов – расщепление затрат по модели «от общего к частному» по трем уровням.

Уровень 1. Инфраструктура - считается все, что связано с железом, ЦОДом и сетями, сравнивается утилизация. Если серверы загружены на 10-15%, их TCO на единицу мощности (например, на виртуальную машину) взлетает до небес.

Уровень 2. Приложения - оценивается стоимость владения конкретными программными продуктами. Например, старая ERP-система, написанная на COBOL. Сама по себе лицензия может быть давно амортизирована, но стоимость ее поддержки (редкие специалисты, риски ошибок при модернизации) делает TCO такой системы непомерно высокой. Генеративный ИИ сегодня позволяет реверсивно инжинирить такой legacy-код, снижая стоимость владения за счет автоматизации анализа зависимостей.

Уровень 3. Бизнес-сервисы – определяется TCO функции «расчет зарплаты» или «обработка заказа» - складывается стоимость всех ИТ-компонентов, участвующих в процессе, и делится на количество операций.

Финансовый аудит – это не разовое упражнение по сбору чеков. Это системная проверка того, как в компании принимаются финансовые решения об ИТ. В рамках аудита необходимо ответить на ряд вопросов:

1. Прозрачность бюджетирования. Может ли CFO увидеть, сколько стоит поддержка каждого бизнес-подразделения в пересчете

на ИТ-ресурсы (showback/chargeback) или ИТ — это совокупные затраты?

2. Эффективность закупок. Практикуется ли сравнение нескольких вендоров по TCO, а не только по начальной цене? Как показывает практика, выбор решения на основе минимальной цены покупки часто оборачивается кратным ростом эксплуатационных расходов.

3. Управление жизненным циклом. Есть ли политика своевременного вывода устаревшего оборудования из эксплуатации или серверы работают по 10 лет, потребляя электроэнергию как новые, но выдавая производительность как 10-летние? Аудит часто вскрывает «кладбища слонов» - давно не нужное, но все еще работающее и потребляющее ресурсы оборудование.

В современных условиях финансовый аудит ИТ не может игнорировать облака и искусственный интеллект. По данным IBM, 73% компаний используют гибридные облака, но далеко не все понимают экономику этого использования[39]. Переход в облако часто маскирует старые проблемы: вместо покупки сервера просто начинают ежемесячно платить за виртуальную машину, которая может быть так же неэффективно использована.

Современная практика управления облачными расходами требует постоянного аудита использования ресурсов (FinOps). Например, выключен ли тестовый стенд на выходные, не закуплено ли зарезервированных инстансов (RI) больше, чем нужно и т.д. Это уже не инженерные, а финансовые вопросы.

Отдельная проблема в настоящий момент – инвестиции в искусственный интеллект (AI TCO). Исследования показывают, что лишь 25% CEO довольны возвратом инвестиций (ROI) от AI-проектов. Причинами этого является неучтенная TCO. Компании видят цену GPU или подписки на ChatGPT, но забывают про дата-инженеров, разметку данных, дообучение моделей и их поддержку. Финансовый аудит в этой области должен сравнивать сценарии: брать готовый SaaS, растить свою ML-команду или внедрять единую платформу. Для россий-

[39] Зачем бизнесу гибридное облако в 2026 году | РБК Компании [Электронный ресурс] // Режим доступа: https://companies.rbc.ru/news/8lDFRmOAWs/zachem-biznesu-gibridnoe-oblako-v-2026-godu/?ysclid=mmjdymstui740033641 (дата обращения 09.03.2026)

ского бизнеса добавляются факторы импортозамещения и санкционных рисков, что делает расчет ТСО еще более сложным, но и более критичным.

Итогом данного раздела аудита должен стать отчет, который ляжет в основу бюджета на следующие периоды. В нем обязательно должны быть:

– сравнение с бенчмарками – это то, как показатели компании (например, стоимость хранения 1 Гб данных или стоимость одного рабочего места) соотносятся со среднерыночными. Это сразу подсвечивает зоны неэффективности;

– скрытые резервы – это выявление неутилизируемых лицензий, «мертвого» оборудования, дублирующих подписок. Часто аудит окупает себя уже на этом этапе, просто «выключая» лишнее;

– дорожная карта оптимизации – это конкретные шаги с расчетом экономического эффекта. Например, «Миграция серверов X и Y на виртуализацию позволит сэкономить N рублей в год на электроэнергии и обслуживании»;

– финансовый аудит и ТСО – это не бухгалтерия. Это стратегическая разведка, которая показывает, где деньги реально работают на бизнес, а где просто тихо сгорают в стойке с серверами. Хороший аудитор всегда помнит: цифры – это язык, на котором готов говорить генеральный директор.

9.4. Обзор профессий будущего в области управления цифровой инфраструктурой (SRE, Cloud Architect, DevOps Engineer, IT Auditor)

Когда речь заходит об управлении цифровой инфраструктурой, сразу возникает проблема подбора персонала. Кто будет разрабатывать облачные архитектуры, кто следит, чтобы в пиковую пятничную нагрузку сайт продолжал отвечать пользователям, кто объединит процесс разработки и процесс эксплуатации, и последнее: кто даст беспристрастное заключение о степени безопасности и эффективности всей системы.

В современном мире рынок специальностей в сфере управления инфраструктурой переживает глобальную перестройку. Десять лет

назад DevOps-инженеры требовались лишь компаниям-новаторам, теперь эта профессия стала обычной. Похожее превращение ожидает и другие должности. Ключевые профессии, которые зададут облик отрасли в ближайшие годы, рассмотрены ниже.

Самая загадочная и быстрорастущая профессия – Site Reliability Engineer (SRE) – это специалист, который находится где-то на пересечении системного администратора, DevOps-инженера и разработчика, но при этом не закрывает ни одну из этих специальностей полностью.

Типичная ситуация в компании без SRE: в компании перестала работать корпоративная система, сотрудник пишет в поддержку — попадает на первую линию, которая перенаправляет заявку дальше. Вторая линия пытается разобраться, но не может, и передает вопрос третьей — разработчикам. Пока все эти специалисты договариваются между собой и выясняют приоритеты, проходят часы, а иногда и дни.

SRE работает иначе. В 99% случаев ему не нужно привлекать другие команды. Он сам анализирует инцидент, находит причину и устраняет ее. Его эффективность измеряется не количеством написанного кода, а временем восстановления системы (RTO) и объемом потерянных данных (RPO). Он тот самый человек, который поднимает сайт, когда все упало.

Работодателями SRE являются компании с высоконагруженными системами, где каждая минута простоя несет миллионные потери. Маркетплейсы вроде Wildberries, Ozon, Lamoda, финтех, банки, телекомоператоры. Рынок SRE, по прогнозам аналитиков, достигнет $5,53 млрд к 2033 году с ежегодным ростом 18,5%[40].

Но эволюция не стоит на месте. Сегодня экономика входит в эру AI SRE. Уже существуют десятки программных продуктов, предлагающих AI-агентов для решения задач надежности. Эксперты сходятся во мнении: искусственный интеллект может быстрее анализировать данные и предсказывать сбои, но в корпоративных масштабах его использование сталкивается с ростом затрат и снижением точности. Более того, возникает «парадокс объема»: AI-агенты генерируют код

[40] Новости / T1 ИИ [Электронный ресурс] // Режим доступа: https://t1-ai.ru/media/news/etoy_professii_ne_uchat_v_vuzah_no_vskore_ona_stanet_samoy_vostrebovannoy_sre-inzhener?ysclid=mmjfb04ymq719087692 (дата обращения 09.03.2026)

быстрее, но, оптимизируя локально, часто не видят глобального контекста сложной распределенной системы. Это создает разрыв в надежности, который может закрыть только человек.

Будущий SRE – это не тот, кто пишет скрипты для перезапуска серверов. Это архитектор безопасной среды для автономных систем, человек, определяющий «правила игры», за пределы которых AI-агент выйти не может.

Если SRE – это пожарный и спасатель, то Cloud Architect — это архитектор и генеральный проектировщик. Это специалист, который отвечает за облачную стратегию компании целиком.

По данным Gartner, мировые расходы на публичные облачные сервисы к 2025 году достигнут $723 млрд, увеличившись на 21,5%[41]. Этот взрывной рост создает беспрецедентный спрос на архитекторов, способных проектировать масштабируемые, безопасные и экономичные облачные решения.

Cloud Architect работает на стыке бизнеса и технологий. В его задачи входит:

– стратегическое планирование: оценка существующей инфраструктуры, выбор модели (публичное, частное, гибридное облако), разработка дорожной карты миграции;

– проектирование архитектуры: выбор сервисов (IaaS, PaaS, SaaS), проектирование сетей (VPC, CDN), обеспечение отказоустойчивости (мультирегиональность, автоскейлинг);

– безопасность и комплаенс: внедрение моделей zero-trust, управление доступом (IAM), обеспечение соответствия регуляторам (GDPR, 152-ФЗ, HIPAA);

– управление затратами (FinOps): выбор оптимальных инстансов, контроль неэффективно используемых ресурсов.

В 2026 году от архитектора ждут не просто знания кнопок AWS или Azure, а понимания фундаментальных принципов: как работают сети, как проектируются отказоустойчивые системы, как балансировать между производительностью и стоимостью. AI здесь выступает

[41] Gartner: расходы на облачные сервисы в 2025 году вырастут на 21,5% / Издательство «Открытые системы» //[Электронный ресурс] Режим доступа: https://www.osp.ru/articles/2024/1120/13058957?ysclid=mmjfjcnter386937851 (дата обращения 10.03.2026)

помощником, предлагая варианты, но окончательное решение, учитывающее бюджет, комплаенс и сроки, остается за человеком.

Ключевой навык Cloud Architect – умение говорить на двух языках: с бизнесом о деньгах и с инженерами о репликации и сетевых протоколах. Это редкое сочетание, за которое готовы платить. В разных ипостасях (Azure Architect, AWS Solutions Architect, инфраструктурный архитектор) эти специалисты стабильно входят в топ самых высокооплачиваемых ИТ-профессий.

DevOps-инженер – профессия, которая за последние десять лет прошла путь от нишевой диковинки до обязательного элемента любой современной ИТ-команды. И в 2026 году эта роль продолжает трансформироваться.

Раньше считалось, что DevOps – это про инструменты: Docker, Kubernetes, Jenkins, Ansible. Сегодня компании уже не спрашивают: «Знаешь ли ты Kubernetes?». Они спрашивают: «Можешь ли ты проектировать системы, отлаживать сбои в продакшене, адаптироваться, когда инструменты меняются, и работать с AI, а не бояться его?».

Настоящий DevOps-инженер 2026 года мыслит не в терминах конкретного синтаксиса YAML, а в терминах конвейеров и потоков. Он понимает, почему каждый этап пайплайна существует, где должны останавливаться сбои и как делать откаты, чтобы поставлять изменения быстро, но безопасно.

По данным IDC, компании стоят на пороге фундаментальных изменений. В прогнозе «FutureScape 2026» аналитики заявляют: Agentic AI (автономные AI-агенты) будут глубоко встроены в жизненный цикл разработки и эксплуатации[42].

К 2030 году 80% разработчиков будут работать в связке с автономными AI-агентами, а 65% предприятий встроят AI-агентов непосредственно в DevOps и DevSecOps-конвейеры. Это означает, что роль

[42] IDC FutureScape 2026 Predictions Reveal the Rise of Agentic AI and a Turning Point in Enterprise Transformation [Электронный ресурс]// Режим доступа: https://finance.yahoo.com/news/idc-futurescape-2026-predictions-reveal-123000266.html?guccounter=1&guce_referrer=aHR0cHM6Ly95YW5kZXgucnUv&guce_referrer_sig=AQAAAMqixDn1LwD8rhrCBoSz7B3TRAg_fkqt-spBkbFboye9wvYMdG5krIB-Wjd5dwsqwCGZOXISBNrumB-_WhVlyI9-H-rfpFWgU2JMJW-BRQqpacnk8bLckg49YsmxwxOqUCWts5zqHYqsEkt-aixX_LYtQbo7mrNwQ7ASXPFOXw9FKb (дата обращения: 06.03.2026).

человека смещается: он больше не пишет каждый шаг в коде, а проектирует правила, управляет AI-агентами и проверяет их работу на предмет галлюцинаций и ошибок, невидимых для автоматизированных тестов.

Тот, кто сегодня учится просто копировать ответы AI в продакшен, завтра умножит количество катастроф. Тот, кто понимает основы, использует AI как ассистента, ускоряя решение рутинных задач – от написания скриптов до анализа логов.

И, наконец, четвертая профессия, которая часто остается в тени, но становится критически важной по мере усложнения систем – IT-аудитор. Но не тот аудитор, который проверяет, заполнены ли журналы, а аудитор нового поколения.

Цифровая инфраструктура усложняется. В ней появляются «черные ящики» - AI-модели, логику которых сложно объяснить. Атаки теперь совершаются машинами со скоростью, недоступной человеку. В этих условиях роль внутреннего ИТ-аудитора меняется кардинально.

Совет директоров больше не спрашивает о количестве проведенных проверок. Теперь вопросы звучат иначе:

1. Об операционной устойчивости, т.е. насколько компания способна выдержать тройной удар: отказ облака, сбой у провайдера и кибератаку одновременно. Ответ должен базироваться не на абстрактных уровнях зрелости, а на результатах реальных учений.
2. Об AI-рисках, т.е. возможно ли аудитору дать независимое заключение о рисках AI-систем. Ему не нужно быть data scientist’ом, но он обязан иметь карту: где применяется AI, на какие решения влияет, кто отвечает за результат и как отслеживаются отклонения.
3. О качестве данных, т.е. насколько достоверны данные, на которых основаны выводы аудитора. В мире, где решения принимаются на основе логов и метрик, аудитор должен понимать, откуда берутся данные, кто их обслуживает и о чем они умалчивают.

Современному ИТ-аудитору нужно разбираться в облачных платформах, понимать основы работы IoT и, конечно, ориентироваться в стандартах по AI: NIST AI RMF, ISO 42001 или CSA AI Controls Matrix. Он должен уметь проверять не только конфигурации файрволов, но и то, как AI-модели обрабатывают персональные данные и не дискриминируют ли они клиентов.

Парадокс в том, что если инженеры создают систему, а бизнес ей пользуется, то именно аудитор становится тем «адвокатом дьявола», который задает неудобные вопросы. И в 2026 году без такого специалиста компания рискует двигаться на ощупь в тумане.

В целом, если посмотреть на SRE, Cloud Architect, DevOps и IT Auditor сквозь призму ближайших лет, становится очевидным: выживут не те, кто знает конкретный инструмент, а те, кто понимает фундаментальные принципы.

Современный специалист – это человек, который:

– понимает, как работают сети, даже если пользуется AI для их настройки;

– думает об архитектуре в категориях отказоустойчивости и безопасности, а не в терминах «кнопок»;

– воспринимает AI как партнера, который ускоряет рутину, но не доверяет ему ключи от продакшена без присмотра;

– умеет коммуницировать: SRE – с разработчиками, архитектор – с финансистами, аудитор – с советом директоров.

Границы между этими профессиями будут размываться. DevOps будет все больше думать об архитектуре, SRE – автоматизировать себя с помощью AI, переходя на уровень мета-управления, архитектор – погружаться в экономику, аудитор – осваивать дата-сайенс.

Но в одном можно быть уверенным: спрос на людей, способных управлять сложностью цифровой инфраструктуры, не исчезнет. Потому что даже самый умный AI пока не готов брать на себя ответственность за бизнес-решения и отвечать за простои.

Таким образом, эффективность управления цифровой инфраструктурой не может быть оценена исключительно на основе технических метрик (доступность, загрузка вычислительных ресурсов). Ключевым требованием к системе показателей является их многоуровневый характер, обеспечивающий декомпозицию от стратегических бизнес-показателей (North Star Metric, индекс цифровой трансформации) через показатели качества сервиса к технологическому фундаменту («четыре золотых сигнала» SRE). Принципами построения такой системы выступают целевая ориентированность, управляемость и связь с ценностью для бизнеса.

В свою очередь, коммуникация результатов аудита цифровой инфраструктуры требует дифференцированного подхода к структурированию отчетности. Исполнительное резюме, ориентированное на лиц, принимающих стратегические решения, должно концентрироваться на бизнес-влиянии выявленных проблем (финансовые потери, репутационные риски) и содержать дорожную карту их решения. Техническая часть отчета, адресованная профильным специалистам, должна включать детализированный анализ по областям (аппаратное обеспечение, программное обеспечение, безопасность, резервное копирование) и конкретные рекомендации по устранению недостатков.

Финансовый аудит информационных технологий базируется на концепции совокупной стоимости владения (TCO), которая в отличие от бухгалтерского учета, учитывает не только прямые капитальные затраты, но и косвенные эксплуатационные расходы (электроэнергия, охлаждение, заработная плата персонала), а также скрытые и условные издержки (простои, привязка к поставщику, неэффективное использование ресурсов). Методология расчета совокупной стоимости владения позволяет выявить резервы оптимизации и обосновать стратегические решения по развитию инфраструктуры.

Подводя итог, сказанному выше, необходимо заключить, что профессиональная структура кадрового обеспечения управления цифровой инфраструктурой претерпевает существенную трансформацию под влиянием усложнения технологического ландшафта и внедрения искусственного интеллекта.

Ключевыми профессиями здесь выступают Site Reliability Engineer (обеспечение надежности высоконагруженных систем), Cloud Architect (проектирование облачной стратегии), DevOps Engineer (автоматизация конвейеров разработки и эксплуатации) и IT-аудитор нового поколения (независимая оценка операционной устойчивости, AI-рисков и качества данных). Именно указанные выше роли требуют не только владения инструментарием, но и понимания фундаментальных принципов, а также способности к межфункциональной коммуникации.

Вопросы для обсуждения

1. В тексте говорится о «ловушке 99,9% аптайма». Приведите примеры, когда система формально работала, но бизнес нес убытки. Охарактеризуйте метрику, которую нужно было отслеживать вместо аптайма?
2. Поясните, по каким причинам для упорядочения показателей KPI необходимо выстроить иерархию.
3. Объясните, почему KPI не должны быть статичными. Как часто, их нужно пересматривать и что может служить триггером для такого пересмотра?
4. Поясните ситуацию: исполнительное резюме отчета для директора часто называют «самой важной частью». Согласны ли вы с этим?
5. Подробно объясните, что произойдет, если написать исполнительное резюме техническим языком, перегрузив его деталями?
6. Поясните, в чем состоит разница между «тщеславными метриками» (vanity metrics) и действительно полезными KPI.
7. Приведите примеры метрик, которые «красиво» выглядят в отчете, но не помогает управлять инфраструктурой.
8. Представьте, что вы проводите аудит и нашли критическую уязвимость, за которую отвечает конкретный сотрудник. Как можно сформулировать этот пункт в отчете, чтобы не демотивировать команду, но добиться исправления проблемы?
9. Охарактеризуйте концепцию «скрытых затрат» (Shadow IT). Объясните, почему бизнес-подразделения предпочитают покупать ИТ-сервисы самостоятельно, минуя ИТ-департамент.
10. Поясните, каким образом аудит может помочь легализовать сервисы и взять их под контроль?
11. Объясните ситуацию: расчет TCO часто требует учета «стоимости простоев». Каким образом можно оценить стоимость одного часа простоя для интернет-магазина, для внутренней CRM, которой пользуются только сотрудники? От чего зависят эти цифры?
12. Объясните разницу между Site Reliability Engineer (SRE) и классическим системным администратором. В каких компаниях появление SRE критически необходимо, а в каких можно обойтись без него?

13. Охарактеризуйте «парадокс объема» при использовании AI-агентов в SRE. Поясните, в чем он заключается и почему человек пока не может полностью доверить надежность систем искусственному интеллекту?

14. Объясните, по каким причинам DevOps-инженер 2026 года — это не просто «специалист по Docker и Kubernetes», а также какие его навыки выходят на первый план.

15. Поясните, как меняется роль IT-аудитора в эпоху «черных ящиков» (AI-моделей). Какие новые риски он должен оценивать?

16. Можно ли утверждать, что профессии Cloud Architect и IT Auditor в какой-то момент сольются? Объясните, должен ли архитектор разбираться в комплаенсе, а аудитор — в архитектуре?

17. Если у компании ограниченный бюджет, а нанять нужно только одного специалиста из четырех (SRE, Cloud Architect, DevOps, IT Auditor), кого бы вы выбрали в первую очередь и почему? Ответ аргументируйте.

18. В тексте подчеркивается важность «умения говорить на двух языках» — с бизнесом и с инженерами. Поясните, каким образом проверить это умение у кандидата на собеседовании. Приведите пример неудачной и удачной формулировки одной и той же технической проблемы для директора.

19. Существует мнение, что через 5-10 лет нейросети полностью заменят DevOps- и SRE-инженеров. Опираясь на текст, приведите аргументы «за» и «против» этого утверждения.

20. Поясните, по каким причинам границы между профессиями будущего в области управления цифровой инфраструктурой (SRE, Cloud-архитектор, DevOps-инженер, ИТ-аудитор) не смогут полностью исчезнуть.

Практические задания

Задание 1. Разработка системы KPI для стоматологической клиники.

Сеть частных стоматологических клиник (10 филиалов) использует облачную CRM для записи пациентов, внутреннюю медицинскую систему (хранение снимков, историй болезней) и общую бухгалтерию. В последний месяц участились жалобы от регистраторов: «система тормозит», пациенты нервничают в очереди. Главный врач считает, что ИТ работает отвратительно, но ИТ-директор утверждает, что «все серверы зеленые и аптайм 99,9%».

Задание:

1. Предложите трехуровневую систему метрик (бизнес-метрики → метрики качества сервиса → технические метрики), которая свяжет работу инфраструктуры с удовлетворенностью пациентов.
2. Для каждого уровня приведите 1-2 конкретных показателя (например, уровень отказов от записи, время отклика интерфейса CRM, загрузка дисковой подсистемы).
3. Объясните, как изменение технической метрики (например, рост времени отклика диска) повлияет на бизнес-показатель (например, на лояльность пациентов).

Задание 2. Финансовый аудит и выбор стратегии: «Покупаем сервер или уходим в облако?»

Компания – средний производственный холдинг. Бухгалтерия требует обновить сервер для 1С.

Есть два варианта:

- вариант А (On-premise) - купить сервер (стоимость 1,5 млн руб.), лицензии (500 тыс. руб.), нанять подрядчика для настройки (200 тыс. руб.). Сервер служит 5 лет;
- вариант Б (Cloud) - арендовать виртуальную машину в облаке с предоплатой на год – 1,2 млн руб./год.

Задание:

1. Используя концепцию совокупной стоимости владения (TCO), перечислите как минимум 5 скрытых затрат, которые не учтены в варианте А (On-premise) (например, электроэнергия, зарплата администратора, стоимость простоев, охлаждение, замена комплектующих).

2. Перечислите 2-3 скрытых риска и затраты, которые могут возникнуть в варианте Б (Cloud), помимо явной абонентской платы.

3. На основе вашего приблизительного анализа сделайте вывод о том, при каких условиях (срок жизни сервера, рост нагрузки, требования безопасности) вариант А выгоднее варианта Б и наоборот.

Задание 3. Подготовка Executive Summary по результатам «пожара»

Вчера в компании произошел серьезный инцидент. В 14:00 из-за ошибки при обновлении конфигурации на маршрутизаторе (инженер забыл сохранить ее перед перезагрузкой) на 2 часа «упал» интернет-магазин. Потери выручки оцениваются в 3 млн руб.

Сейчас вам нужно подготовить небольшой отчет (0,5 страницы) для генерального директора.

Вам предоставлены факты:

– причина: human error (ошибка инженера);

– техническая деталь: отсутствие автоматического бэкапа конфигураций сетевого оборудования перед внесением изменений;

– время простоя: 2 часа 10 минут;

– реакция: инцидент обнаружили по звонку из отдела продаж, система мониторинга не среагировала должным образом.

Задание:

Напишите текст исполнительного резюме, который должен включать:

1. Описание инцидента простым языком (без технического жаргона).

2. Оценку бизнес-влияния (сумма потерь, репутационные риски).

3. Корневую причину (не просто «ошибка Петрова», а системная проблема в процессах).

4. Конкретные рекомендации (2-3 пункта) с примерными сроками, которые предотвратят повторение ситуации.

Тест для самоконтроля

1. Какой показатель, по мнению аналитиков IDC, относится к «устаревшему подходу» к оценке эффективности ИТ?

а) Индекс цифровой трансформации (DXI).

б) Время безотказной работы (аптайм).

в) Удовлетворенность пользователей (NPS).

г) Время вывода продукта на рынок (Time-to-Market).

2. Что такое «тихая поломка»?

а) Инцидент, о котором никто не сообщил в техподдержку.

б) Ситуация, когда система работает, но риск потери данных или сбоя критически вырос (например, деградация RAID-массива).

в) Ошибка в логах, которая не влияет на производительность.

г) Плановая замена оборудования.

3. Какая структура отчета по аудиту считается оптимальной для восприятия разными аудиториями?

а) Единый технический отчет для всех.

б) Отчет, состоящий из краткого резюме для руководства и детальной технической части.

в) Устная презентация без письменного отчета.

г) Два совершенно разных отчета, не связанных друг с другом.

4. Что должно быть отражено в исполнительном резюме отчета для генерального директора в первую очередь?

а) Полные логи сервера за период аудита.

б) Подробная схема сети после исправления ошибок.

в) Влияние выявленных проблем на бизнес (деньги, риски, репутация).

г) Пофамильный список сотрудников, допустивших ошибки.

5. Что из перечисленного относится к «скрытым затратам» (косвенным) при расчете ТСО?

а) Стоимость закупки сервера.

б) Цена лицензий на ПО.

в) Расходы на электроэнергию и зарплату администраторов.

г) Гонорар юриста за проверку договора поставки.

6. Какой метод управления облачными расходами набирает популярность и требует постоянного аудита использования ресурсов?

а) GDPR;

б) FinOps;

в) DevOps;

г) PMBoK.

7. Почему, согласно данным IBM, 73% компаний, использующих гибридные облака, часто не понимают экономику такого использования?

а) Облака всегда дороже собственных серверов.

б) Расходы маскируются: вместо покупки сервера компании просто начинают ежемесячно платить за неэффективно используемые виртуальные машины.

в) Слишком сложная отчетность от облачных провайдеров.

г) Из-за санкций невозможно оплачивать облачные сервисы.

8. Какова главная задача Site Reliability Engineer (SRE)?

а) Писать код новых функций для сайта.

б) Продавать ИТ-услуги клиентам.

в) Минимизировать время восстановления (RTO) и потерю данных (RPO) при сбоях.

г) Разрабатывать дизайн-макеты интерфейсов.

9. В чем заключается «парадокс объема» при использовании AI в SRE?

а) AI требует слишком много места на диске.

б) AI быстро пишет код, но часто не видит глобального контекста сложной системы, что создает новые риски для надежности.

в) AI не умеет анализировать логи.

г) AI работает только в маленьких компаниях.

10. Что из перечисленного НЕ входит в типичные обязанности Cloud Architect?

а) Стратегическое планирование миграции в облако.

б) Управление затратами (FinOps).

в) Написание кода бизнес-логики приложения.

г) Проектирование отказоустойчивой архитектуры.

11. Как изменилась роль DevOps-инженера к 2026 году?

а) От него требуется только знание синтаксиса YAML.

б) Его роль сместилась от знания конкретных инструментов к проектированию конвейеров и управлению AI-агентами.

в) DevOps-инженеры больше не нужны, их заменили AI.

г) Они занимаются только тестированием.

12. Что должен уметь современный IT-аудитор в эпоху искусственного интеллекта?

а) Писать нейросети с нуля.

б) Оценивать риски AI-систем, проверять отсутствие дискриминации и обработку персональных данных моделями.

в) Заменять AI-агентов на рабочих местах.

г) Только проверять журналы доступа к серверам.

13. Какой процент разработчиков, по прогнозу IDC, будут работать в связке с автономными AI-агентами к 2030 году?

а) 20%;

б) 50%;

в) 80%;

г) 100%.

14. «Shadow IT» — это:

а) Программное обеспечение, работающее только в темное время суток.

б) ИТ-сервисы и решения, которые бизнес-подразделения покупают и используют самостоятельно, без ведома ИТ-департамента.

в) Техника, украденная с предприятия.

г) Облачные сервисы, запрещенные в стране.

15. В тексте упоминается, что хороший отчет по аудиту «продает будущее». Что это означает?

а) Отчет должен продавать конкретное оборудование.

б) Отчет должен рекламировать услуги аудиторской компании.

в) Отчет должен показывать, как инвестиции в ИТ сегодня приведут к росту бизнеса и снижению рисков завтра.

г) Отчет должен быть написан настолько красиво, чтобы его можно было продать как книгу.

16. Какая метрика из «золотых сигналов» SRE отвечает за предсказание скорого исчерпания ресурсов (например, места на диске)?

а) Задержка (Latency).

б) Трафик (Traffic).

в) Насыщенность (Saturation).

г) Ошибки (Errors).

17. Почему при выборе ИТ-решения опасно ориентироваться только на минимальную цену покупки?

а) Дешевое оборудование всегда ломается.

б) Это может обернуться кратным ростом эксплуатационных расходов (OPEX) в будущем, что покажет расчет TCO.

в) Дешевое ПО не имеет лицензии.

d) Это запрещено законом о госзакупках.

18. Что такое «таблица разрыва» (Gap Analysis) в отчете?

а) Таблица, показывающая прогноз прибыли.

б) Сравнение текущего состояния инфраструктуры («как есть») с целевым («как должно быть») для наглядной демонстрации объема работ.

в) Список уволенных сотрудников.

г) Расписание перерывов на кофе для ИТ-отдела.

19. Какая профессия будущего, по мнению авторов текста, должна уметь задавать «неудобные вопросы» и быть «адвокатом дьявола» для сложных систем?

а) Cloud Architect.

б) SRE.

в) DevOps Engineer.

г) IT Auditor.

20. Как, согласно тексту, следует формулировать замечания в отчете, чтобы не демотивировать команду, но решить проблему?

а) Указывать конкретную фамилию виновного для принятия кадровых решений.

б) Описывать системную проблему, а не искать виноватого (например, «система бекапов не обеспечивает надежность», а не «Петя не настроил бекапы»).

в) Не указывать замечания вообще, чтобы никого не обидеть.

г) Выносить замечания только в устной форме.

ЗАКЛЮЧЕНИЕ

В учебном пособии представлено систематическое изложение теоретических, методологических и прикладных аспектов управления и аудита цифровой инфраструктуры современного предприятия. Содержание пособия структурировано таким образом, чтобы обеспечить последовательное формирование у обучающихся целостного представления о цифровой инфраструктуре как сложном, многоуровневом и динамично развивающемся объекте, требующем комплексного подхода к управлению, контролю и оценке эффективности.

В пособии получили теоретическое развитие вопросы выравнивания ИТ-стратегии с бизнес-целями, представленные через четырехуровневую модель стратегического, тактического, операционного и измерительного контуров управления. Систематизированы архитектурные принципы проектирования современной инфраструктуры: масштабируемость, отказоустойчивость, безопасность и экономическая эффективность, выявлены взаимосвязи и конфликтные зоны между данными принципами, требующие сбалансированных архитектурных компромиссов. Проведен сравнительный анализ архитектурных моделей (монолитная, сервис-ориентированная, микросервисная), позволивший выявить закономерности их применения в зависимости от зрелости организации и характера решаемых бизнес-задач.

Значительное внимание уделено методологии управления производительностью и доступностью цифровой инфраструктуры. Представлена иерархическая концепция метрик, связывающая фундаментальные технические показатели (латентность, пропускная способность) с индикаторами уровня обслуживания (SLI), внутренними целями (SLO) и формальными соглашениями (SLA). Теоретически обоснована роль систем наблюдаемости (Prometheus, Grafana, ELK-стек) как технологической основы для проактивного управления и оперативной диагностики состояния инфраструктуры. Разработана классификация уровней стратегий аварийного восстановления с дифференциацией по целевым показателям RTO и RPO.

Существенное место в пособии занимает анализ подходов к управлению цифровой инфраструктурой на основе методологии IT Service Management (ITSM) и библиотеки ITIL 4. Выявлен концепту-

альный сдвиг от жесткой процессной модели к гибкой системе практик, включающих не только процедуры, но и роли, компетенции, технологии и культуру взаимодействия. Теоретически обоснована необходимость интеграции DevOps-культуры и Agile-подходов в традиционный ITSM, представленная через модель «совместного творчества ценности» (value co-creation) и концепцию Системы ценности сервиса (Service Value System).

Практическая значимость пособия определяется его направленностью на формирование у обучающихся компетенций, необходимых для их будущей профессиональной деятельности, связанной не только с областью управления и аудита цифровой инфраструктуры, но и аспектами экономического обоснования инвестиций в ее развитие, расширение и модернизацию.

Каждая глава издания содержит методически проработанный комплекс практических заданий, вопросов для обсуждения и тестов для самоконтроля, что обеспечивает возможность применения полученных теоретических знаний для решения конкретных задач, возникающих в деятельности современных организаций.

Материалы пособия могут быть использованы не только в учебном процессе при подготовке бакалавров и магистров, но и в системе дополнительного профессионального образования, при подготовке к сертификации в области управления ИТ и внутреннего аудита, а также в практической деятельности специалистов, занятых проектированием, эксплуатацией и аудитом цифровой инфраструктуры организаций различных отраслей и масштабов. Оно будет также полезно при подготовке обоснования инвестиций в реализацию различных проектов, связанных с разработкой и развитием цифровой инфраструктуры любого уровня сложности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Баланов, А.Н. Управление IT-проектами: учебное пособие для вузов/ А.Н. Баланов. - Санкт-Петербург: Лань, 2024. - ISBN 978-5-507-49698-3. - Текст: электронный. - URL: https://e.lanbook.com/book/428081 (дата обращения: 18.01.2026).

2. Бирюков А. ИТ-стратегии: какие бывают и как их использовать [Электронный ресурс]/ А. Бирюков // Режим доступа: https://habr.com/ru/companies/otus/articles/943210/ (дата обращения: 06.01.2026).

3. Бурцев Д.С. Инфраструктура и ресурсное обеспечение цифровой экономики: учеб. Пособие/ Д. С. Бурцев [и др.]. - СПб: Университет ИТМО, 2021. – 190 с. - Режим доступа: https://books.ifmo.ru/file/pdf/3020.pdf (дата обращения: 16.01.2026).

4. Гагаринская Г.П. Повышение эффективности управления производительностью труда организации на основе безопасных цифровых технологий/ П.В. Гагаринская [и др.] // Вестник Евразийской науки, 2021 №1, https://esj.today/PDF/28ECVN121.pdf (доступ свободный).

5. Грибанов Ю. И., Руденко М. Н., Аленина К. А. Современные подходы к формированию цифровой инфраструктуры // Управленческое консультирование. 2020. № 8. С. 88–98. DOI 10.22394/1726-1139-2020-8-88-98

6. Глушков В.А. Применение систем DLP и SIEM для предотвращения и выявления преступлений, направленных на критическую информационную инфраструктуру/ В.А. Глушков, О.А. Беларева // Образование и право. 2024. №3. URL: https://cyberleninka.ru/article/n/primenenie-sistem-dlp-i-siem-dlya-predotvrascheniya-i-vyyavleniya-prestupleniy-napravlennyh-na-kriticheskuyu-informatsionnuyu (дата обращения: 18.01.2026).

7. ГОСТ Р 57580.1-2017. Безопасность финансовых (банковских) операций. Защита информации финансовых организаций. Базовый состав организационных и технических мер» // Консультант Плюс: справочно-правовая система. - Режим доступа: https://base.garant.ru/12148567/?ysclid=mkkv6fzu8i235523485

8. ГОСТ Р 53114-2008. Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения. – Введ. 2009-10-01. – Москва: Стандартинформ, 2010. – 20 с.

9. ГОСТ Р 56215-2014/ISO/TS 8000-150:2011 Качество данных. Ч. 150. Основные данные. Структура управления качеством = Data quality.

Part150. Master data: Quality management framework. Введ. 2015-07-01. – Москва: Стандартинформ, 2020.– 21 с.

10. ГОСТ Р ИСО/МЭК 17021-1-2025. Оценка соответствия. Требования к органам, проводящим аудит и сертификацию систем менеджмента. Часть 1. Требования (введ. .01.09.2025) [Электронный ресурс]// Режим доступа - https://docs.cntd.ru/document/1312251791?ysclid=mmc5kykis1768698346 (дата обращения 03.03.2036)

11. ГОСТ Р 56045-2021/ISO/IEC TS 27008:2019. Информационные технологии. Методы и средства обеспечения безопасности. Рекомендации по оценке мер обеспечения информационной безопасности– Введ. 30.11.2021. – Москва: Стандартинформ, 2021. – 34 с.

12. Гришин Л. ТО САМОЕ. ВВЕДЕНИЕ И МЕТОДИКА СХЕМЫ COBIT 2019 [Электронный ресурс] / Лев Гришин, пер. с англ. // Режим доступа: https://blog.cortel.cloud/2023/02/28/zemlya-ili-oblako-ekonomika-vladeniya/?ysclid=mki95pgb9q242240226 (дата обращения: 03.03.2026).

13. Демина А.К. Управление инцидентами информационной безопасности/ А.К. Демина // Международный журнал гуманитарных и естественных наук. 2024. №5-1 (92). URL: https://cyberleninka.ru/article/n/upravlenie-intsidentami-informatsionnoy-bezopasnosti-1 (дата обращения: 18.01.2026).

14. Емельянов В.А., ИТ-инфраструктура организации: учебное пособие/ В.А. Емельянов. - Москва : КноРус, 2022. - 144 с. - ISBN 978-5-406-09892-9. - URL: https://book.ru/book/943918 (дата обращения: 18.01.2026).

15. Ермакова, А.Н. Цифровой след: формирование и управляемость в экономике данных: монография / А. Н. Ермакова. – Ставрополь : Ставропольский государственный аграрный университет, 2024. - 256 с. - Текст: электронный. - URL: https://znanium.ru/catalog/product/2234184 (дата обращения: 18.01.2026)

16. Заводцев И.В. Программные и программно-аппаратные средства защиты информации в объектах информационной инфраструктуры: учебное пособие/ И.В. Заводцев, А.В. Крупенин, С.В. Скрыль. – М: ACADEMIA, 2023. – 288 с. - 978-5-0054-0439-8

17. Зачем бизнесу гибридное облако в 2026 году | РБК Компании [Электронный ресурс] // Режим доступа: https://companies.rbc.ru/news/8lDFRmOAWs/zachem-biznesu-gibridnoe-oblako-v-2026-godu/?ysclid=mmjdymstui740033641 (дата обращения 09.03.2026)

18. Затраты на IT инфраструктуру. Сравнение облака и on premise [Электронный ресурс] // Режим доступа: https://blog.cortel.cloud/2023/02/28/zemlya-ili-oblako-ekonomika-vladeniya/?ysclid=mki95pgb9q242240226 (дата обращения: 17.01.2026).

19. Какие технологии помогают бизнесу построить единую ИТ-инфраструктуру [Электронный ресурс] // Режим доступа: https://digtlab.ru/tpost/rzfhfyfrr1-kakie-tehnologii-pomogayut-biznesu-postr?ysclid=mk2z328jiu499207780 (дата обращения: 06.01.2026).

20. Корытко С.А. О новых подходах организации ИТ-инфраструктуры электросетевого комплекса в условиях цифровой трансформации / С.А. Корытко, Н.И. Лиманова. - Текст: непосредственный // Молодой ученый. - 2021. - № 5 (347). - С. 9-11. - URL: https://moluch.ru/archive/347/78059.

21. Макаренко С. И. Аудит безопасности критической информационной инфраструктуры. Учебное пособие. – СПб.: Наукоемкие технологии, 2023. – 122 с. - ISBN 978-5-907618-78-7

22. Мамедли Р.Э. Системы управления базами данных: Учебное пособие. – Нижневартовск: Изд-во Нижневартовского государственного университета, 2021. – 214 с. ISBN 978-5-00047-585-0

23. Милославская Н.Г. Управление инцидентами информационной безопасности: учеб. пособие/ Н. Г. Милославская, А.И. Толстой. - 2-е изд., испр. и доп., 2025. - ISBN: 978-5-9912-1041-6.

24. Моррис Киф. Программирование инфраструктуры: практическое пособие/ Киф Моррис. - 2-е издание. М., 2023. – 416 с. - 978-5-9775-1901-4

25. Морозов И. М. Информационная безопасность в цифровой инфраструктуре ТЭК: технологии мониторинга и оценка эффективности// Вестник науки. 2025. №5 (86). URL: https://cyberleninka.ru/article/n/informatsionnaya-bezopasnost-v-tsifrovoy-infrastrukture-tek-tehnologii-monitoringa-i-otsenka-effektivnosti (дата обращения: 18.01.2026).

26. Мультиклауд: как строить распределенную инфраструктуру в 2025 году [Электронный ресурс] // Режим доступа: https://companies.rbc.ru/news/FEb0OQ7BLn/multiklaud-kak-stroit-raspredelennuyu-infrastrukturu-v-2025-godu/ (дата обращения: 19.01.2026).

27. НАЦИОНАЛЬНАЯ ЦИФРОВАЯ ИНФРАСТРУКТУРА ФИНАНСОВОГО РЫНКА (Доклад для общественных консультаций) [Электронный ресурс]// Банк России. – Режим доступа:

https://www.wipo.int/edocs/pubdocs/ru/wipo-pub-2000-2023-exec-ru-global-innovation-index-2023.pdf (дата обращения: 08.01.2026).

28. Национальная цифровая инфраструктура. Выбираем модель управления [Электронный ресурс]// Режим доступа: https://plusworld.ru/journal/2022/plus-3-2022/natsionalnaya-tsifrovaya-infrastruktura-vybiraem-model-upravleniya/nalichnoe-denezhnoe-obrashchenie/?ysclid=mk1eoly9v78168091 (дата обращения: 05.01.2026).

29. Новости / Т1 ИИ [Электронный ресурс] // Режим доступа: https://t1-ai.ru/media/news/etoy_professii_ne_uchat_v_vuzah_no_vskore_ona_stanet_samoy_vostrebovannoy_sre-inzhener?ysclid=mmjfb04ymq719087692 (дата обращения 09.03.2026)

30. О защите персональных данных: Федеральный закон от 27.07.2006 № 152-ФЗ (ред. от 24.06.2025) // Консультант Плюс: справочно-правовая система. — Режим доступа: https://base.garant.ru/12148567/?ysclid=mkkv6fzu8i235523485

31. Пронин А.Ю. Основные тенденции развития цифровой инфраструктуры в интересах национальной экономики [Электронный ресурс]/ А.Ю. Пронин// Экономические исследования и разработки. – Режим доступа: http://edrj.ru/article/09-08-24 (дата обращения: 04.01.2026).

32. Рынок облачных технологий в России: импортозамещение, безопасность данных и перспективы роста [Электронный ресурс]// Режим доступа: https://delprof.ru/press-center/open-analytics/rynok-oblachnykh-tekhnologiy-v-rossii-importozameshchenie-bezopasnost-dannykh-i-perspektivy-rosta/?ysclid=mki93a78ef979797749 (дата обращения: 17.01.2026).

33. Сетевые технологии в России – 2025: разбор трендов [Электронный ресурс]/ А. Федоров // Режим доступа: https://dzen.ru/a/aDbUmk3nFC8zwWey?ysclid=mkgxclbq18962166339 (дата обращения: 16.01.2026).

34. Смелянский Р. Технологии реализации программно конфигурируемых сетей: Overlay vs OpenFlow [Электронный ресурс]/ Руслан Смелянский // Режим доступа: https://www.osp.ru/lan/2014/04/13040709 (дата обращения: 18.01.2026).

35. Сырцева В.Ю. Белая книга цифровой экономики [Электронный ресурс]/ В.Ю. Сырцева [и др.]// Режим доступа: https://files.data-economy.ru/Docs/White_paper_2023_.pdf (дата обращения: 16.01.2026).

36. Топ-17 СУБД разных видов: обзор [Электронный ресурс]// Режим доступа: https://www.ihc.ru/articles/top-17-subd-raznykh-vidov-obzor.html?ysclid=mkgzig8svq196517541 (дата обращения: 16.01.2026).

37. О защите персональных данных: Федеральный закон от 27.07.2006 № 152-ФЗ (ред. от 24.06.2025) // Консультант Плюс: справочно-правовая система. — Режим доступа: https://base.garant.ru/12148567/?ysclid=mkkv6fzu8i235523485

38. Федоров А. Что такое ИТ-инфраструктура и из каких компонентов она состоит [Электронный ресурс]/ А. Федоров // Режим доступа: https://cyberprotect.ru/blog/it-infrastructure-intro?ysclid=mk1kblc7jg194849748 (дата обращения: 05.01.2026).

39. Хайруллина, А. Р. Цифровая инфраструктура как среда принятия управленческих решений в малом и среднем предпринимательстве / А. Р. Хайруллина // Экономика, предпринимательство и право. – 2021. – Т. 11, № 5. – С. 1151-1166. – DOI 10.18334/epp.11.5.112066

40. Худяков Д.С. Разработка ИТ-стратегии на основе оценки эффективности ИТ-процессов / Д.С. Худяков// Вестник Академии знаний. 2024. №4 (63). URL: https://cyberleninka.ru/article/n/razrabotka-it-strategii-na-osnove-otsenki-effektivnosti-it-protsessov (дата обращения: 06.01.2026).

41. Цифровая экономика: 2025 : краткий статистический сборник / В.Л. Абашкин, Г.И. Абдрахманова, К.О. Вишневский, Л.М. Гохберг и др.; Нац. исслед. ун-т «Высшая школа экономики». – М. : ИСИЭЗ ВШЭ, 2025. – 120 с. – 300 экз. – ISBN 978-5-7598-3025-2 (в обл.

42. ЦИФРОВОЙ БИЗНЕС: ИТ-СТРАТЕГИЯ БЛИЖАЙШЕГО БУДУЩЕГО [Электронный ресурс]// Режим доступа: https://www.comindware.ru/assets/dl2/pdf/comindware-digital-business-report.pdf?ysclid=mk2ytyl3fx646089164 (дата обращения: 06.01.2026).

43. Что такое CAPEX и OPEX и зачем их считать [Электронный ресурс]// Режим доступа: https://sales-generator.ru/blog/capex-i-opex/ (дата обращения: 08.01.2026).

44. Что такое PCI DSS [Электронный ресурс]//Режим доступа: https://selectel.ru/blog/pci-dss (дата обращения 03.03.2026)

45. AXENIX — консалтинг, технологии и цифровые решения для бизнеса[Электронный ресурс]//Режим доступа: https://axenix.pro/?ysclid=mmj34kea5z369827500 (дата обращения 09.03.2026)

46. CAPEX и OPEX: что это, разница, как рассчитать и анализировать [Электронный ресурс]// Режим доступа: https://vladimir.1cbit.ru/blog/capex-i-opex-chto-eto-raznitsa-kak-rasschitat-i-

analizirovat/?utm_referrer=https%3A%2F%2Fya.ru%2F (дата обращения: 08.01.2026).

47. IDC FutureScape 2026 Predictions Reveal the Rise of Agentic AI and a Turning Point in Enterprise Transformation[Электронный ресурс]// Режим доступа: https://finance.yahoo.com/news/idc-futurescape-2026-predictions-reveal-123000266.html?guccounter=1&guce_referrer=aHR0cHM6Ly95YW5kZXgucnUv&guce_referrer_sig=AQAAAMqixDn1LwD8rhrCBoSz7B3TRAg_fkqt-spBkbF-boye9wvYMdG5krIBWjd5dwsqwCGZOXISBNrumB-_WhVlyI9-H-rfpFWgU2JMJWBRQqpacnk8bLckg49YsmxwxOqUCWts5zqHYqsEkt-aixX_LYtQbo7mrNwQ7ASXPFOXw9FKb (дата обращения: 06.03.2026).

48. ISACA Revamps IT Audit Framework [Электронный ресурс]// Режим доступа: https://www.fromtech.ru/blog/chto-znachit-on-premise/ (дата обращения: 06.03.2026).

49. Gartner: расходы на облачные сервисы в 2025 году вырастут на 21,5% / Издательство «Открытые системы» // [Электронный ресурс] Режим доступа: https://www.osp.ru/articles/2024/1120/13058957?ysclid=mmjfjcnter386937851 (дата обращения 10.03.2026)

50. Make defensible decisions in shifting markets [Электронный ресурс]// Режим доступа: https:// https://www.idc.com/data-analytics/ (дата обращения: 08.03.2026).

51. On-Premise: что это, чем отличается от облака и кому подходит [Электронный ресурс]// Режим доступа: https://www.fromtech.ru/blog/chto-znachit-on-premise/ (дата обращения: 19.01.2026).

52. SaaS или On-Premise: сравниваем архитектуру сервисов видеосвязи[Электронный ресурс]// Режим доступа: https://kontur.ru/talk/spravka/48332-saas_ili_on_premise?ysclid=mki916olne777724040 (дата обращения: 17.01.2026).

53. ITSM тренды в 2025 году [Электронный ресурс]// Режим доступа: https://www.tadviser.ru/index.php (дата обращения: 02.02.2026).

54. 2025: 6 трендов мирового рынка ИТ-инфраструктур [Электронный ресурс]// Режим доступа: https://www.tadviser.ru/index.php (дата обращения: 16.01.2026).

РЕКОМЕНАТЕЛЬНЫЙ БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Баранова, И. В. Аудит информационных систем и цифровой инфраструктуры : учебник для вузов / И. В. Баранова, Д. А. Чистов. – Москва : Юрайт, 2024. – 312 с. – (Высшее образование). – ISBN 978-5-534-16782-3.

2. Бурцев Д.С. Инфраструктура и ресурсное обеспечение цифровой экономики: учеб. пособие/ Д. С. Бурцев [и др.]. - СПб: Университет ИТМО, 2021. – 190 с. - Режим доступа: https://books.ifmo.ru/file/pdf/3020.pdf (дата обращения: 16.01.2026).

3. Гайдамакин, Н. А. Аудит информационной безопасности и цифровой инфраструктуры : учебное пособие / Н. А. Гайдамакин, С. В. Петров. – Екатеринбург: Издательство Уральского университета, 2023. – 248 с. – ISBN 978-5-7996-3741-9.

4. Емельянов В.А., ИТ-инфраструктура организации: учебное пособие/ В.А. Емельянов. - Москва: КноРус, 2022. - 144 с. - ISBN 978-5-406-09892-9. - URL: https://book.ru/book/943918 (дата обращения: 18.01.2026).

5. Косачев, Ю. В. Аудит цифровых экосистем : учебник для магистратуры / Ю. В. Косачев, Е. А. Федорова ; под науч. ред. Ю. В. Косачева. – Санкт-Петербург : Издательство СПбГЭУ, 2024. – 286 с. – ISBN 978-5-7310-6123-8.

6. Сироткин, С. А. Цифровой аудит: методология и инструментарий : учебник / С. А. Сироткин. – Москва : ИНФРА-М, 2025. – 294 с. – (Высшее образование: Магистратура). – ISBN 978-5-16-019875-6.

7. Тарарин, А. М. Инфраструктура пространственных данных: учебное пособие/ А. М. Тарарин. - Н. Новгород: ННГАСУ, 2023. – 279 с. - ISBN 978-5-528-00558-4. - Текст: электронный. - URL: https://znanium.ru/catalog/product/2151372 (дата обращения: 16.03.2026)

8. Шестаков, А. П. Технологический аудит цифровой инфраструктуры промышленных предприятий / А. П. Шестаков, М. И. Лебедев. – Екатеринбург : Издательство УрФУ, 2023. – 196 с. – ISBN 978-5-7996-3864-5.

Учебное электронное издание

АБДУЛЛАЕВ Низами Видади оглы
КУЛИКОВА Ирина Юрьевна
МУРАВЬЕВА Надежда Викторовна

АУДИТ ЦИФРОВОЙ ИНФРАСТРУКТУРЫ КОМПАНИИ

Учебное пособие

Издается в авторской редакции

Системные требования: Intel от 1,3 ГГц; Windows XP/7/8/10;
Adobe Reader; дисковод CD-ROM.

Тираж 8 экз.

Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых.
600000, Владимир, ул. Горького, 87.