

**А. Б. ГОЛЬДШТЕЙН, С. В. КИСЛЯКОВ,
М. А. ФЕНОМЕНОВ**

ОТКРЫТАЯ ЦИФРОВАЯ АРХИТЕКТУРА ДЛЯ РАЗРАБОТКИ СИСТЕМ УПРАВЛЕНИЯ ИНФОКОММУНИКАЦИЯМИ

УЧЕБНОЕ ПОСОБИЕ

**Учебное пособие разработано
при поддержке Благотворительного Фонда Владимира Потанина
в рамках реализации проекта-победителя
«Разработка учебной дисциплины
“Системы автоматизации управления в сетях 5G/6G”»**

**САНКТ-ПЕТЕРБУРГ
2024**

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ,
СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ
им. проф. М. А. БОНЧ-БРУЕВИЧА»
(СПбГУТ)**

А. Б. Гольдштейн, С. В. Кисляков, М. А. Феноменов

ОТКРЫТАЯ ЦИФРОВАЯ АРХИТЕКТУРА ДЛЯ РАЗРАБОТКИ СИСТЕМ УПРАВЛЕНИЯ ИНФОКОММУНИКАЦИЯМИ

УЧЕБНОЕ ПОСОБИЕ

**Учебное пособие разработано
при поддержке Благотворительного Фонда Владимира Потанина
в рамках реализации проекта-победителя
«Разработка учебной дисциплины
“Системы автоматизации управления в сетях 5G/6G”»**

СПб ГУТ)))

**САНКТ-ПЕТЕРБУРГ
2024**

УДК 681.5(075.8)

ББК 32.966я73

Г63

Рецензенты:

доктор технических наук, старший научный сотрудник,
директор по науке ООО «Протей СпецТехника»

Н. А. Соколов,

кандидат технических наук, доцент кафедры ССиПД

Мутханна Аммар Салех Али

Утверждено редакционно-издательским советом СПбГУТ

в качестве учебного пособия

Протокол № от

Гольдштейн, А. Б.

Г63 Открытая цифровая архитектура для разработки систем управления инфокоммуникациями : учебное пособие / А. Б. Гольдштейн, С. В. Кисляков, М. А. Феноменов ; СПбГУТ. – Санкт-Петербург, 2024. – 77 с. – Текст : электронный.

ISBN 978-5-89160-338-7

Содержит теоретический материал по разработке систем эксплуатационной поддержки и бизнеса операторов связи OSS/BSS на основе новейшей концепции TM Forum Open Digital Architecture. Данный материал необходим для изучения дисциплин: «Стратегии управления инфокоммуникационными сетями», «Бизнес-процессы в инфокоммуникациях», «Общая информационная модель управления», «Системы эксплуатационного управления OSS/BSS». В качестве практических примеров применения компонентного подхода рассматриваются системы «АРГУС NRI» и «АРГУС WFM».

Предназначено для студентов вузов направлений подготовки: 11.03.02 «Инфокоммуникационные технологии и системы связи» (профиль «Сети связи и системы коммутации»); 11.04.02 «Инфокоммуникационные технологии и системы связи» (профиль «Системы управления инфокоммуникациями»).

УДК 681.5(075.8)

ББК 32.966я73

ISBN 978-5-89160-338-7 © Гольдштейн А. Б., Кисляков С. В., Феноменов М. А., 2024

© Федеральное государственное бюджетное образовательное учреждение высшего образования «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича», 2024

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ	4
ВВЕДЕНИЕ	5
1. ОСНОВНЫЕ ИНСТРУМЕНТЫ NGOSS/FRAMEWORX	8
2. ОТКРЫТАЯ ЦИФРОВАЯ АРХИТЕКТУРА	12
2.1. Переход от Framework к открытой цифровой архитектуре (ODA)	12
2.2. Фреймворки TOGAF и ITIL в архитектуре ODA	13
2.3. Основные составляющие открытой цифровой архитектуры (ODA)	25
2.4. Функциональная архитектура ODA	26
2.5. Информационная модель SID	29
2.6. Структура программных приложений ODA	29
2.7. ODA-компоненты	32
2.8. Межкомпонентное взаимодействие и аспекты безопасности	34
2.9. Различие подходов Framework и открытой цифровой архитектуры (ODA)	38
3. ПОСТРОЕНИЕ СИСТЕМ НА ОСНОВЕ ODA	40
3.1 Система технического учета «АРГУС NRI»	40
3.2. Система управления рабочими ресурсами «АРГУС-WFM»	57
ЗАКЛЮЧЕНИЕ	74
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	75

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

AMS	– Asset Management System
API	– Application Programming Interface
BSS	– Business Support System
CFS	– Customer Facing Service
eTOM	– Enhanced Telecom Operations Map
FA	– Functional Architecture
FF	– Functional Framework
HTTP	– HyperText Transfer Protocol
IIoT	– Industrial Internet of Things
IMS	– IP Multimedia Subsystem
IoT	– Internet of Things
NGN	– Next Generation Network
NGOSS	– Next Generation Operating Support System
NRI	– Network Resource Inventory
ODA	– Open Digital Architecture
OSS	– Operation Support System
REST	– Representational State Transfer
RFS	– Resource Facing Service
SID	– Shared Information and Data Model
SLA	– Service Level Agreement
SOA	– Service Oriented Architecture
TAM	– Telecom Application Map
TMF	– Tele Management Forum
ZOOM	– Zero-touch Orchestration, Operations and Management (проект некоммерческой организации TM Forum)
ВОЛС	– волоконно-оптические линии связи
ИТ	– информационные технологии
НТЦ	– научно-технический центр
МСЭ	– Международный союз электросвязи
ТУ	– технический учет

ВВЕДЕНИЕ

В 2018 г. TM Forum объявил о начале разработки новой концепции построения систем автоматизации – открытой цифровой архитектуры (ODA, Open Digital Architecture) [1]. ODA, по утверждению ее авторов, перестраивает архитектуру OSS/BSS так, чтобы она могла соответствовать быстро меняющейся коммерческой, технической и нормативной среде для поставщиков услуг связи и услуг на базе 5G, IoT и соответствующим требованиям. Во главу угла ставятся сквозная автоматизация, скоростной вывод новых услуг оператора на рынок и реальное время реакции на запрос пользователя. Одно из главных нововведений в ODA – это внедрение компонентного подхода к построению B/OSS решений в отличие от традиционного, когда решения являются «тяжеловесными монолитами». Бизнес-идеей в ODA выступает возможность разработки компонентов разными вендорами, т. е. любое сложное решение в ландшафте может быть собрано из ODA-компонентов разных производителей ПО. Эта возможность достигается стандартизацией функциональности компонентов (в рамках ODA уже создан набор компонентов и описана их функциональность), а также фиксированным набором прикладных программных интерфейсов (API), которыми наделяются компоненты для связи между собой и с внешними системами.

В некоторых маркетинговых документах TM Forum говорится об упразднении OSS/BSS как ключевых элементов IT-ландшафтов операторов связи, а также eTOM и TAM как основных инструментов. При этом в других документах, касающихся ODA, говорится, что продолжается использование ранее разработанных активов и артефактов для развития ODA: NGOSS/Framework, eTOM, TAM, SID и т. д.

В [2] утверждается, что «...поставщики услуг видят решение своих проблем через трансформацию своих OSS/BSS... Им требуется новое поколение систем, охватывающих современные IT-технологии, такие как сервис-ориентированные архитектуры (SOA) и микросервисные подходы, открытый исходный код, виртуализация и облачные архитектуры».

Важными причинами развития новой концепции являются также и технологии, требующие другой, более высокой динамики управления ресурсами и услугами – это программно-определяемые сети (SDN) и виртуализация сетевых функций (NFV) в сочетании с переходом к облачным вычислениям, 5G/6G и IoT. SDN и NFV, облачные вычисления меняют фундаментальную инфраструктуру поставщиков услуг связи (CSP). Требования и стремления клиентов также меняются. Сегодня клиенты B2B и B2C ожидают, что услуги будут предоставляться по требованию в течение нескольких минут, а не дней и недель.

Управление онлайн-услугами, устранение неполадок и выставление счетов в режиме реального времени являются основными требованиями и становятся все более сложными, особенно по мере того, как CSP используют все более сложные партнерские услуги и бизнес-модели [4, 5]. Все это требует значительно более гибкой архитектуры OSS/BSS [6]. Например, в традиционной сети процессы технического учета (NRI) относительно медленные, но в инфраструктуре SDN и NFV виртуальные машины, предоставляющие услуги связи, должны быть подготовлены за считанные секунды в любом месте инфраструктуры, а это означает, что конкретный экземпляр клиента может быть идентифицирован только с помощью NRI-решения реального времени, которое постоянно обновляется. Без более гибких ИТ-систем ценность SDN и NFV не может быть реализована.

Бизнес-менеджеры и архитекторы выразили обеспокоенность по поводу отсутствия гибкости в традиционных системах поддержки бизнеса и в системах поддержки операционной деятельности, которые рассматриваются как камень преткновения для создания новых бизнес-возможностей в облаке, 5G и IoT. Предполагаемое будущее гибкое предприятие должно быть цифровым по своей сути и обладать возможностями, охватывающими как предоставление, так и эксплуатационную поддержку цифровых услуг.

ТМФ видит будущее в обеспечении гибкости бизнеса за счет:

- сквозной автоматизации;
- «zero touch» – скоростного вывода новых услуг оператора на рынок;
- использования «model driven» – подхода к управлению жизненным циклом услуг в OSS/BSS (model driven agile service lifecycle).

ODA перестраивает архитектуру OSS/BSS так, чтобы она могла соответствовать быстро меняющейся коммерческой, технической и нормативной среде для поставщиков услуг связи и услуг на базе 5G, IoT и соответствующим требованиям. При этом в своих разработках ТМФ продолжает использовать предыдущие активы и артефакты для развития ODA: NGOSS/Framework, SOA, ZOOM.

Предложение касательно ODA заключается в создании ограниченного набора архитектурных принципов в качестве общих правил и рекомендаций. Цель состоит в том, чтобы обеспечить уровень консенсуса в отрасли путем разработки высокоуровневой архитектуры, согласованной всеми заинтересованными сторонами.

Поскольку принципы архитектуры отражают уровень консенсуса в масштабах всего предприятия и отражают дух и мышление предприятия, они сильно зависят от базовых концепций предприятия и архитектуры, в данном случае концепций ODA. Поэтому в рамках ODA члены TM Forum разрабатывают общее понимание этих требований для будущих систем.

Этот документ представляет собой один из ключевых артефактов, которые будут лежать в основе архитектуры информационных систем.

ODA предлагает согласованный в отрасли проект, язык и набор ключевых принципов проектирования, которым необходимо следовать. Он обеспечивает прагматические пути для перехода от поддержки монолитных устаревших программных решений к управлению гибкими облачными возможностями, которые можно организовать с помощью искусственного интеллекта (ИИ). Это эталонная архитектура, которая сопоставляет открытые API TM Forum с функциями технической и бизнес-платформы.

Сотрудничество TM Forum под руководством ведущих архитекторов, ведущих CSP, включая AT&T, Bharti Airtel, BT, Orange, Telefonica и Vodafone, выявило общие требования к переосмысленной OSS/BSS, что, в свою очередь, привело к разработке ODA.

Сфера функциональной архитектуры ODA заключается в следующем:

- определить таксономию функций бизнеса и инфраструктуры;
- сгруппировать эти функции и указать их отображение во Framework;
- предложить сценарии использования функций и реализовывать их с помощью нейтральных в технологическом отношении.

На конец 2023 г. более 50 крупных международных компаний – операторов связи и разработчиков ПО – подписали манифест, в котором они поддерживают переход на ODA с традиционных OSS/BSS.

Напомним, для удобства читателя, какие инструменты предложены организацией TM Forum в рамках концепций NGOSS/Framework и сравним их с предлагаемыми инструментами ODA.

1. ОСНОВНЫЕ ИНСТРУМЕНТЫ NGOSS/FRAMEWORX

Первой концепцией разработки OSS/BSS была NGOSS (Next Generation Operations Systems and Software – системы следующего поколения для поддержки операционной деятельности компании связи), разработанная организацией TeleManagement Forum. Разработка стартовала в 1995 г. Важнейшим инструментом, предложенным тогда разработчиками концепции, была карта бизнес-функций оператора связи – Telecom Operations Map (ТОМ), которая позже была стандартизирована на уровне Международного союза электросвязи (МСЭ).

Позже ТОМ развилась в расширенную карту бизнес-функций оператора связи eТОМ (enhanced Telecom Operations Map). Она до сих пор является базой для анализа и проектирования бизнес-процессов поставщика инфокоммуникационных услуг.

Еще два важнейших инструмента концепции NGOSS – это единая информационная модель SID (Shared Information and Data Model) и карта программных приложений ТАМ (Telecom Applications Map). SID, частично стандартизированная МСЭ, содержит определение и описание элементов и структур данных, задействованных в бизнес-процессах инфокоммуникационной компании и совместно используемых различными компонентами ее информационных сущностей. Карта ТАМ содержит структурированный набор функций программных приложений оператора.

Концепция Frameworkx пришла на смену NGOSS в 2010 г. В качестве основных инструментов в ней сохранились и развились три описанных выше инструмента – eТОМ, ТАМ, SID, только теперь eТОМ приобрела название Business Process Framework, ТАМ – Applications Framework, а модель SID – Information Framework. На сегодняшний день подавляющее большинство операторов связи, разработчиков, интеграторов применяют в своей деятельности эти три основных инструмента.

Для того, чтобы пояснить основные идеи перехода к ODA, кратко раскроем суть и наполнение инструментов NGOSS/Frameworkx.

Business Process Framework (eТОМ) – это эталонная модель, или среда бизнес-операций (бизнес-функций), предназначенная для поставщиков услуг связи, ИТ-компаний и предприятий, деятельность которых связана с предоставлением различных инфокоммуникационных услуг. Модель систематизирует основные бизнес-операции поставщика услуг. Модель является стандартом и определяет бизнес-операции поставщика услуг.

Концептуальный уровень карты бизнес-операций Business Process Framework (eТОМ) (рис. 1) отражает высокоуровневую классификацию бизнес-операций оператора и наличие всех субъектов взаимодействия в процессах поставки и потребления услуг – клиентов, поставщиков, партнеров, акционеров, собственников.

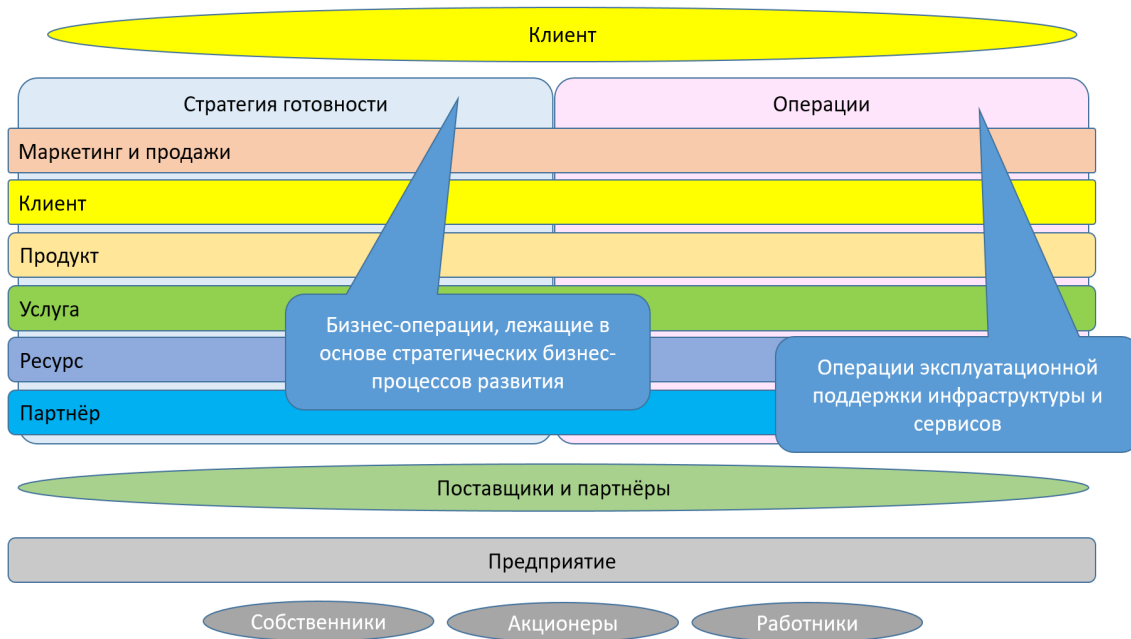


Рис. 1. Концептуальный вид Business Process Framework (eTOM)

Карта построена по принципу «пирамиды», где на вершине оказываются наиболее обобщенные названия групп операций (бизнес-функций). Стало быть, в самом ее основании – конкретные неделимые (атомарные) операции. Показанные для примера некоторые группы операций (рис. 2) декомпозируются на несколько подгрупп и далее до самого «низкого» уровня неделимых операций. Вертикали карты содержат операции, которые могут быть задействованы в сквозных бизнес-процессах поставщика услуг или оператора связи.

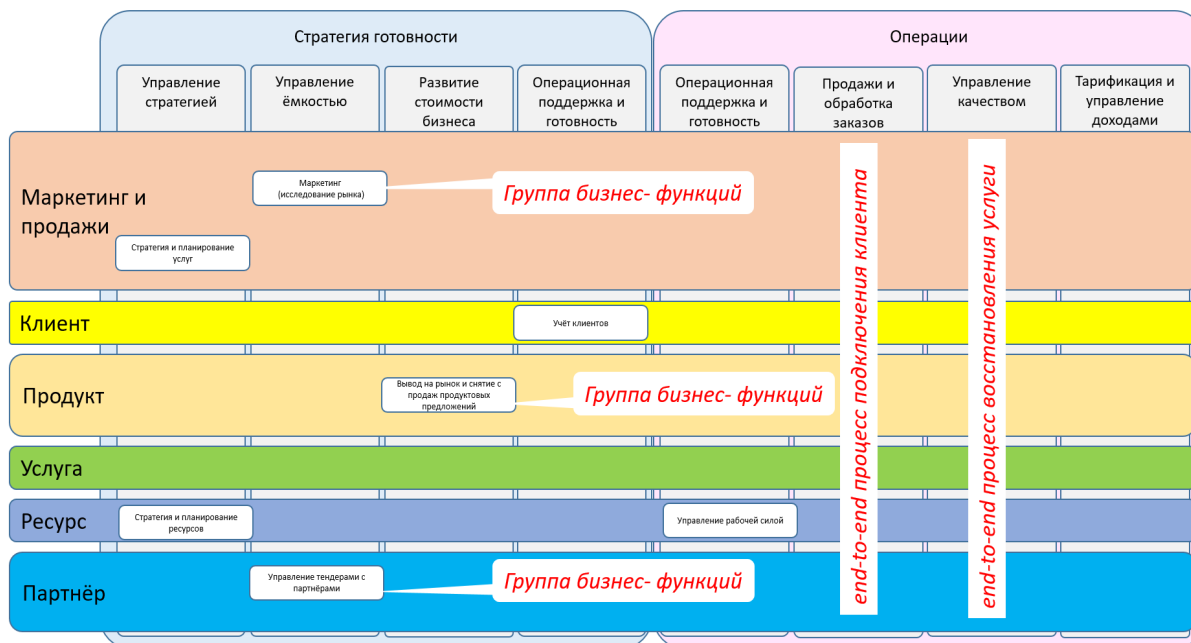


Рис. 2. Пример групп второго уровня и сквозных бизнес-процессов

В качестве эталонной информационной модели для предприятий отрасли связи выступает единая информационная модель (Shared Information and Data model – SID, она же Information Framework), разработанная TM Forum и частично принятая МСЭ в качестве стандарта. SID (рис. 3) содержит набор информационных сущностей и структур данных, задействованных в бизнес-процессах инфокоммуникационной компании и совместно используемых различными компонентами ее информационных систем.

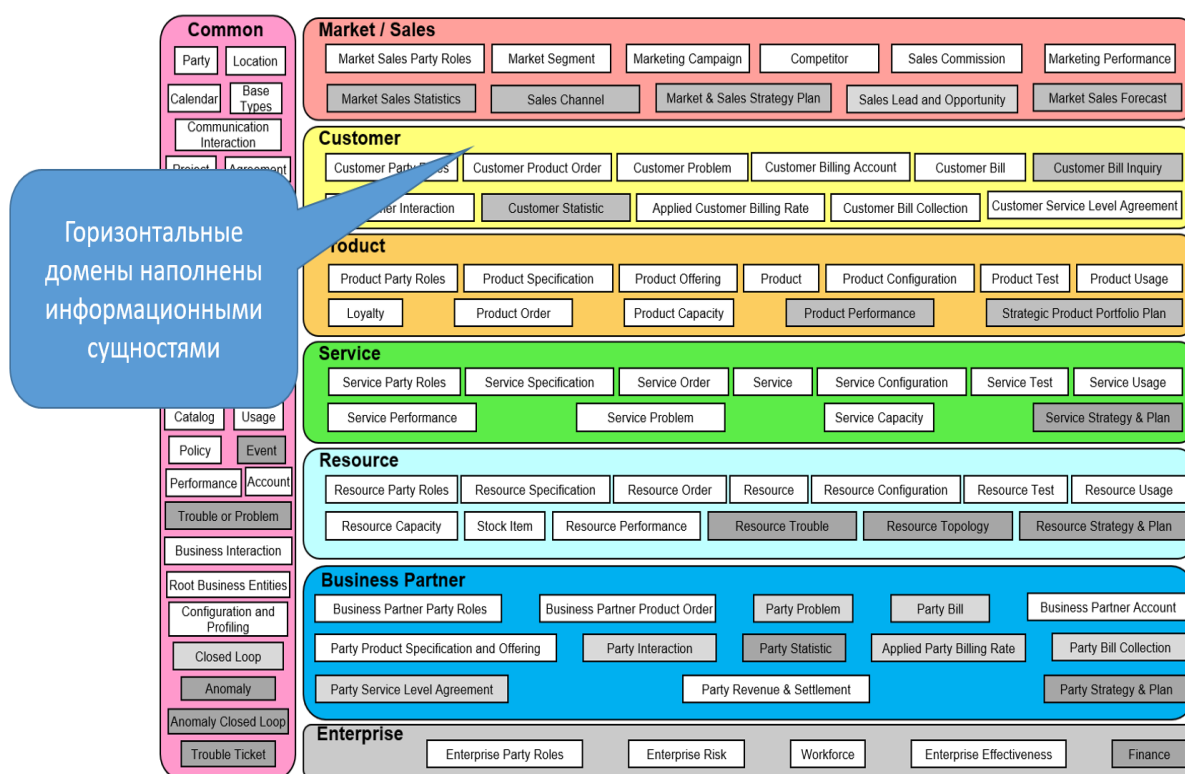


Рис. 3. Общий вид модели SID

SID является объектно-ориентированной моделью. При ее разработке использованы все основные инструменты и принципы объектно-ориентированного подхода: наследование, модульность, абстрагирование и т. д., а значит, модель обладает и всеми его преимуществами – она является модульной, легко расширяемой и масштабируемой. Все это позволяет строить на базе SID информационную модель для конкретной, имеющей свою специфику компании или программного продукта.

Карта программных приложений (ТАМ) (рис. 4), которую также называют средой программных приложений (Application Framework) содержит структурированный набор программных функций, обеспечивающий 100-процентную автоматизацию оператора. Являясь одним из компонентов Framework, карта приложений ТАМ связывает процессы eТОМ и элементы информационной модели SID, с одной стороны, с функциями программных приложений и модулей системы OSS/BSS – с другой.

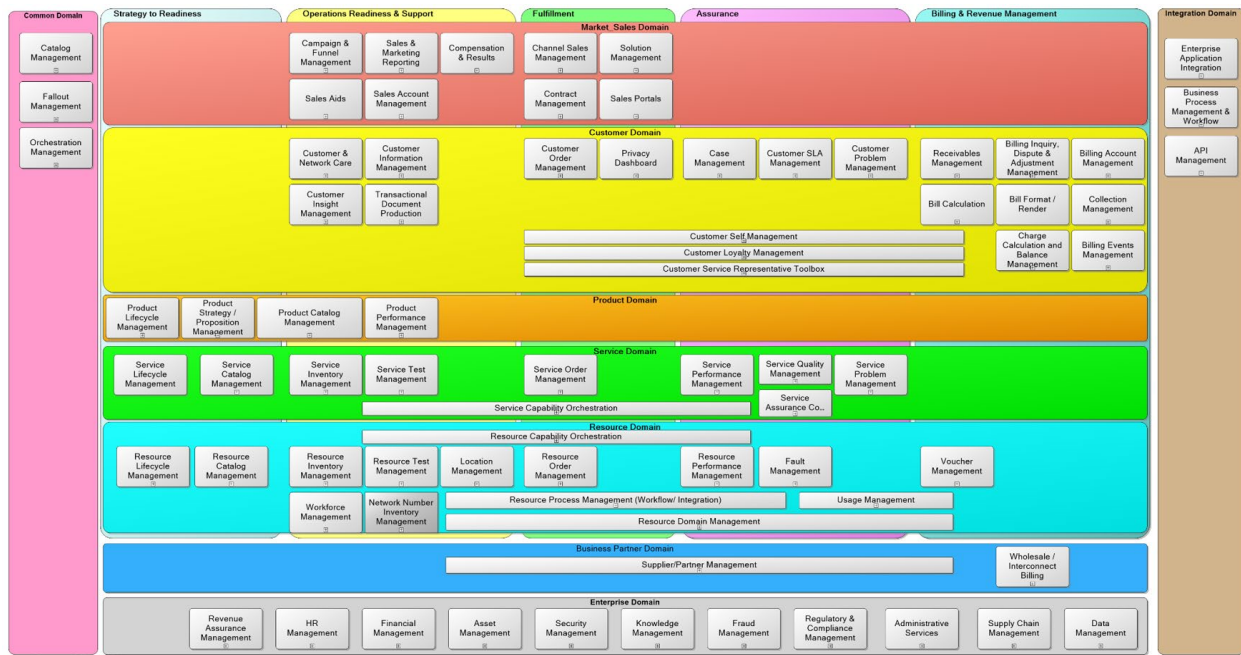


Рис. 4. Карта программных приложений Applications Framework (TAM)

Нетрудно увидеть схожесть структур всех трех инструментов: горизонтальные и вертикальные домены совпадают.

Все три инструмента – Business Process Framework, Shared Information and Data model и Applications Framework применяются на разных стадиях разработки OSS/BSS-решений (рис. 5).

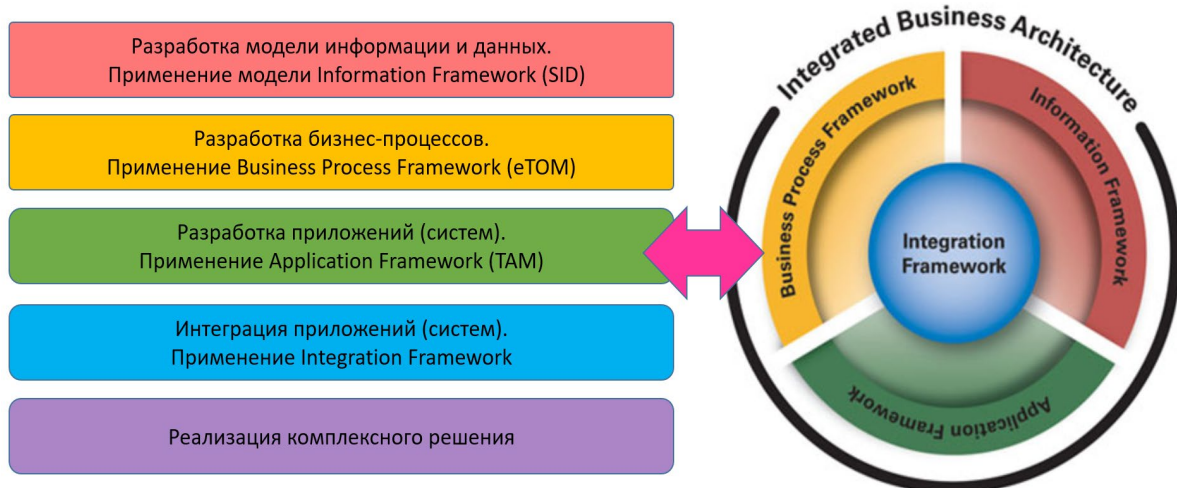


Рис. 5. Место инструментов TM Forum Frameworkx в процессе разработки OSS/BSS-решений.

2. ОТКРЫТАЯ ЦИФРОВАЯ АРХИТЕКТУРА

2.1. Переход от Framework к открытой цифровой архитектуре (ODA)

Традиционная конкуренция в сфере телекоммуникаций строилась на эксклюзивном доступе к сетевым ресурсам, например, к мобильным, фиксированным, междугородним кабельным сетям и т. д. Отрасль сегодня выходит за рамки прежней парадигмы «трубного бизнеса». Поставщикам услуг связи нужно беспокоиться не только о конкуренции с Apple или Google, но также участвовать в экосистемах других независимых поставщиков программного обеспечения – автомобильных (транспортных) экосистемах или экосистемах поставщиков сетевого оборудования.

Практика показала, что побеждают платформы с наиболее активными экосистемами и способностью собирать данные об их взаимодействии.

Поставщики услуг связи должны выйти за рамки моделей управления и операционной деятельности, в которых они имеют абсолютный контроль, и стать посредниками в цифровой экономике. Это означает новую архитектуру, которая позволит CSP стать платформенно-ориентированными.

Бизнес-модели

Цифровые экосистемы используют принципы платформенных бизнес-моделей, которые облегчают обмен между потребителями и производителями. Поставщики услуг связи могут извлечь выгоду из бизнес-моделей платформы двумя способами:

- 1) монетизация своей сети и ИТ-активов через стороннюю платформу с использованием открытых API;
- 2) управление собственной платформой и выполнение функций владельца платформы.

У CSP есть несколько преимуществ в качестве владельцев платформы. Например, они контролируют коммуникационную инфраструктуру, которая позволяет сторонам соединяться. Они также имеют большую клиентскую базу и владеют высокоразвитыми и безопасными системами поддержки бизнеса. Наконец, они управляют операциями по обслуживанию клиентов и логистике для поддержки своего существующего бизнеса, который может быть расширен для поддержки новых направлений бизнеса.

Еще одним важным драйвером перехода на ODA является то, что инфокоммуникации, как технологии, проникли во всевозможные сферы деятельности – банковскую, торговую, медицинскую и т. д. и, следовательно,

фреймворки NGOSS/Framework, разработанные строго под телекоммуникационного оператора, перестали быть актуальными: их просто неудобно применять для автоматизации других бизнесов, кроме телекоммуникационного.

2.2. Фреймворки TOGAF и ITIL в архитектуре ODA

Взаимодействие с TOGAF

Разработчики ODA приняли в качестве одного из главных ориентиров в разработке архитектуры ODA принципы и наработки архитектурного фреймворка TOGAF (The Open Group Architecture Framework) (рис. 6).

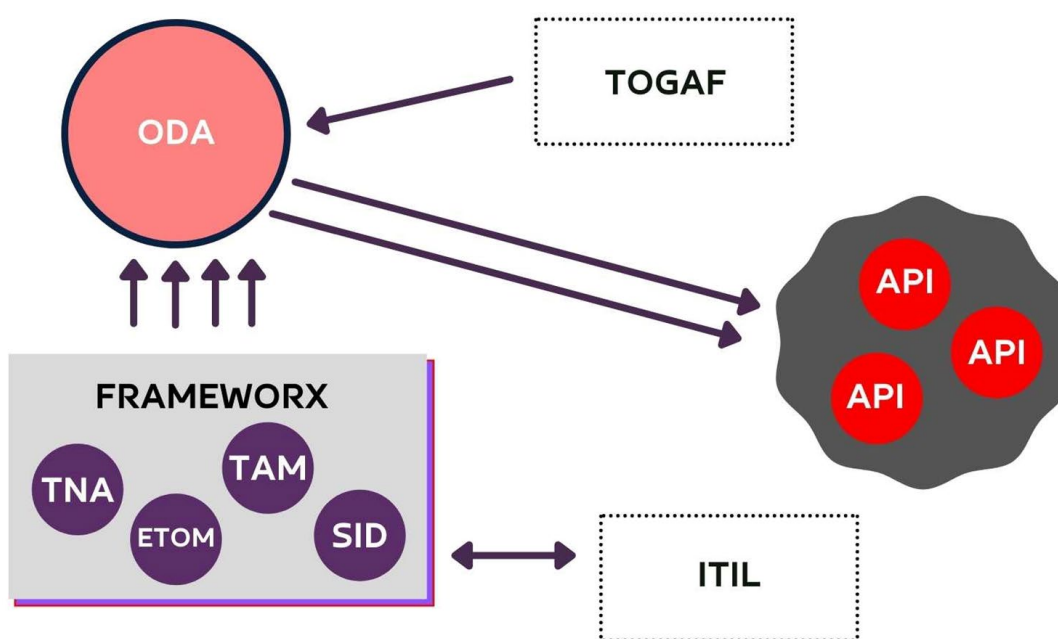


Рис. 6. Взаимосвязь фреймворков TOGAF – высокоуровневый подход к проектированию

TOGAF – методология/подход (framework) для описания архитектуры предприятия, который предлагает подход для проектирования, планирования, внедрения ИТ-архитектуры предприятия и управления ей. TOGAF разрабатывается с 1995 г. группой The Open Group на основе фреймворка Министерства обороны США TAFIM.

В соответствии с TOGAF архитектуру предприятия можно представить в виде четырех основных доменов.

Бизнес архитектура (Business) – определяет стратегию предприятия, структуру управления и ключевые бизнес процессы.

Архитектура данных (Data) – описывает логическую и физическую структуру данных организации, а также структуру корпоративных ресурсов для управления данными.

Архитектура приложений (Application) – служит своеобразной картой всех используемых корпоративных приложений и определяет следующие аспекты:

- участие каждого из приложений в бизнес-процессах компании;
- взаимодействие приложений друг с другом и внешними сервисами.

Технологическая архитектура (Technology) – определяет структуру и логику программного обеспечения и аппаратной среды, необходимых для работы бизнес-приложений и доступа к нужным данным. Этот уровень включает всю поддерживающую инфраструктуру: сети, сервера, процессинг и т. п.

В процессе работы над Видением архитектуры ODA, были выделены следующие ключевые позиции, основанные на анализе современных задач и драйверов бизнеса и подходов к автоматизации бизнеса:

– переход от монолитных «тяжеловесных» OSS/BSS к более гибким решениям на основе микросервисов, взаимодействующим по открытым или стандартным API;

– учет концепции и артефактов архитектурного фреймворка TOGAF;

– применение методов Agile Model Driven Management для достижения гибкого управления жизненным циклом услуг;

– предложения о том, как необходимо преобразовать OSS и BSS так, чтобы учесть изменения в бизнесе, отрасли и технологиях;

– восемь ключевых групп артефактов, необходимых для достижения гибкости бизнеса и автоматизации операций, основанных на концепциях TOGAF;

– описания концепции, принципов и видения, которые отражают путь развития архитектуры ODA;

– Методы Agile Model Driven Management для гибкого управления жизненным циклом услуг.

Артефакты архитектурного фреймворка ODA, включают в себя:

а) спецификации стандартов;

б) описания возможностей, основанных на стандартах, лучших практиках, методологиях и инструментах, которые обеспечивают поддержку управления, необходимую для гибкого жизненного цикла услуг;

в) описания требований и инструментов на уровнях: бизнеса, информационных систем, технической архитектуры, архитектуры внедрения и развертывания.

Итак, архитектура ODA представлена в восьми основных группах артефактов (рис. 7), рожденных на основе передового опыта архитектуры предприятия TOGAF. Вместе эти группы образуют комплексную архитектуру для полной автоматизации жизненного цикла услуг внутри предприятия, например, у поставщика услуг связи.

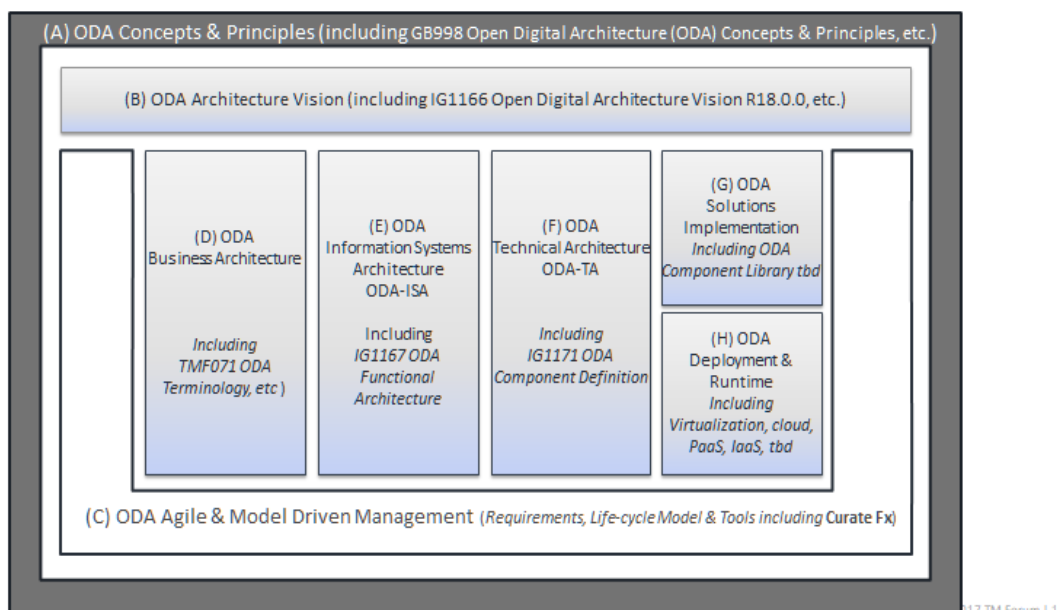


Рис. 7. Группы артефактов открытой цифровой архитектуры TM Forum

Каждая из основных групп артефактов ODA Framework предоставляет отдельный набор задач для различных групп разработчиков.

Архитектурный подход к реализации ODA достигается за счет управления требованиями, создавая следующие группы артефактов архитектуры ODA:

- (A) ODA Concepts and Principles – Концепции и принципы ODA – документирует принципы, концепции и требования архитектуры, которым должна соответствовать каждая группировка Architecture Framework;
- (B) ODA Architecture Vision Document – Видение архитектуры ODA – этот документ определяет 8 групп структуры архитектуры ODA, дает определение артефактов, необходимых в каждой из них, таких как: спецификации, возможности, лучшие практики, включая архитектурные методы и метрики, инструменты.

Возможности ODA по обеспечению гибкости жизненного цикла услуг отражены в следующих группах артефактов:

- (C) Agile and model driven Management – Гибкое и основанное на моделях управление определяет требования, моделирование и подход к инструментарию для жизненного цикла ODA, что является ключом к достижению гибкости бизнеса. Описанные здесь методы/методологии, метрики и инструменты будут использоваться для управления требованиями ко всем этим артефактам для удовлетворения потребностей различных заинтересованных сторон;

- (D) ODA Business Architecture – Бизнес-архитектура ODA, которая будет описывать стратегию продуктов и/или услуг, организационные, функциональные, процессные, информационные и географические аспекты бизнес-среды на основе бизнес-принципов, бизнес-целей и стратегических движущих сил (включая общие словарь, терминологию ODA и т. д. – включая терминологию TMF071 ODA R18.0.0 и т. д.);

- (E) ODA Information Systems Architecture (ISA) – Архитектура информационных систем ODA (ISA), которая должна обеспечивать функциональную архитектуру (IG1167 Функциональная архитектура ODA R18.0.0) и концепции архитектуры данных и приложений;

- (F) ODA Technology Architecture – Технологическая архитектура ODA – архитектура, которая станет основой работы по реализации. Основная концепция заключается в определении компонента ODA для повторного использования и простой интеграции. В рамках этих артефактов ODA рассмотрит, какие соответствующие ресурсы технологической архитектуры доступны (например, Linux Foundation ONAP, Linux Foundation Acumos, ETSI ENI, ETSI ZSM и т. д. Технические стандарты 3GPP 5G и т. д.);

- (G) ODA Solution Implementation – Артефакты реализации решения ODA, которые предоставят механизмы для оценки и выбора сценариев реализации, лучших практик (например, варианты сборки, покупки или повторного использования и подварианты в рамках этих основных вариантов), а также описание стратегических параметров изменений (оценка окружающей среды, управление рисками и т. д.). Это представление также должно включать передовой опыт внедрения и миграции, а также модели управления, позволяющие планировать внедрение (IG1171 ODA Component Definition R18.0.0);

- (H) ODA Deployment and Run-time Models – Модели развертывания и выполнения ODA – установят процесс управления изменениями архитектуры для развертывания архитектуры ODA и базовых показателей времени выполнения. Данный артефакт будет включать в себя основу для постоянного мониторинга развития технологий и изменений в деловой среде. В нем будут изложены передовые методы развертывания и выполнения (deployment and run-time operations) операций.

Каждая из этих групп артефактов описана в следующих разделах и основана на развитии текущих артефактов TM Forum, включая Framework.

А. Концепции и принципы ODA.

Принципы архитектуры управляют архитектурным процессом, влияя на разработку, обслуживание и использование архитектуры предприятия. Принятый здесь подход соответствует стандартам TOGAF. В соответствии

с TOGAF рекомендуемый шаблон приведен в табл. 1. Принципы соответствуют методологии принципов TOGAF и разделены на сектора.

Таблица 1

Сектора методологии TOGAF

Сектор	Описание
1. Business Principles	Всеобъемлющие, но общие принципы, определяющие согласованность с организационными подразделениями. Эти принципы воплощают здоровье и успех организации в целом
2. Information & Data Principles	Предоставляет рекомендации по использованию информации и данных, а также управлению как структурированными, так и неструктурированными данными, опираясь на местные и региональные стратегии, передовой опыт и режимы регулирования
3. Application Principles	Нетехнологические принципы, позволяющие освободить организацию от ограничений, специфичных для конкретного поставщика, или запатентованного подхода к решениям
4. Technology Principles	Гарантирует, что компоненты, разработанные в разных функциональных блоках или средах или для разных исполнений, совместимы между функциональными блоками

Бизнес-архитектура представляет собой целостное, многомерное представление бизнеса о возможностях, сквозном обеспечении ценности, информации и организационной структуре, а также взаимосвязь между этими бизнес-взглядами и стратегиями, продуктами, политиками, инициативами и заинтересованными сторонами (источник: Руководство по своду знаний по бизнес-архитектуре® (BIZBOK® Guide, A Guide to the Business Architecture Body of Knowledge® (BIZBOK® Guide))).

Информационная архитектура обеспечивает описание структуры и взаимодействия основных типов и источников данных предприятия, логических активов (данных), физических активов и ресурсов управления данными. Источник: TOGAF 9.2.

Документ GB998 Open Digital Architecture (ODA) Concepts & Principles определяет следующие основные требования к информации и данным: гибкость бизнеса, более низкая стоимость эксплуатации, поддержка гибких бизнес-моделей, поддержка нескольких поставщиков, гибкое управление, возможность использовать гибкость облака.

Это позволит придерживаться в дальнейшем следующей логики:

- заказчики определяют свои бизнес-требования, используя однозначный набор функций, связанных с Framework, для согласованности в выражении требований путем повторного использования спецификаций функций;

- разработчики же смогут разрабатывать повторно используемые спецификации компонентов ODA из групп этих функций.

Ниже приводятся несколько ключевых требований, которыми руководствуется работа команды ODA.

1. Возможность использовать все преимущества облачных технологий.

Для реализации системы, использующей преимущества облачной технологии, требуется правильная архитектура и подход к проектированию. Например, ключевым преимуществом облака является эластичное масштабирование, но для этого необходим правильный подход к разработке, чтобы он имел ценность. Устаревшие приложения, перенесенные в облачную среду, редко поддерживают независимое масштабирование.

2. Более низкая стоимость автоматизации операций.

Более низкая стоимость операций, в конечном счете, обеспечиваемая AI, станет ключом к успеху в сложных компаниях будущего, потому что это снизит стоимость эксплуатации сети, а также ИТ-систем и процессов.

3. Поддержка гибких бизнес-моделей.

4. Поддержка нескольких поставщиков.

Несмотря на то, что в течение многих лет CSP создавали ИТ-системы с компонентами от разных поставщиков, обычно это были поставщики телекоммуникационных услуг. Теперь, когда CSP конкурируют не только в рамках телекоммуникационных услуг, возможность интеграции компонентов, разработанных вне телекоммуникаций, становится критической.

С появлением таких проектов, как ONAP, Open Source MANO (OSM) и Open Baton, любое определение мультивендора должно также включать поддержку программного обеспечения с открытым исходным кодом. Многие ПЦУ используют открытый исходный код в своих будущих ИТ-системах. TM Forum продолжает тесно сотрудничать с ключевыми группами для обмена соответствующими активами и обеспечения взаимодействия.

5. Гибкость бизнеса.

Команда ODA определила несколько подробных требований к гибкости и активно использует многолетнюю работу в рамках проекта TM Forum «Zero-touch Orchestration, Operations and Management» (ZOOM) по разработке подробных спецификаций требований. Цель состоит в том, чтобы свести к минимуму время, затраты и риски, связанные с запуском нового продукта или услуги, или введением в эксплуатацию новой вспомогательной услуги.

6. Использование открытых API.

Ключевой концепцией является «горизонтальная интеграция», которая относится к созданию предложений, предоставляемых в партнерстве несколькими поставщиками услуг.

Команда ODA разработала основной подход к будущей архитектуре, который устраняет разделение функциональности OSS и BSS. Хотя OSS и BSS могут оставаться в отдельных рабочих доменах, они разработаны как часть единой архитектуры [2].

В. Видение архитектуры ODA.

Документ IG1166 ODA Architecture Vision описывает видение развития инициативы ODA, фиксируя текущие базовые артефакты, соответствующие предыдущие публикации и предлагая будущие артефакты для реализации требований ODA.

Этот документ о видении архитектуры ODA фиксирует, что:

- эволюция цикла разработки ODA получила должное признание и одобрение со стороны членов, и поддержку и приверженность Руководящего комитета по сотрудничеству TM Forum;
- принципы бизнеса, бизнес-цели и стратегические бизнес-движущие силы инициативы ODA были отражены в разрабатываемых артефактах;
- определяет масштаб и приоритеты усилий по созданию базовой архитектуры ODA;
- определяет артефакты архитектуры.

С. Гибкое и основанное на моделях управление.

Гибкий и основанный на моделях (model driven) подход к требованиям, инструментарию и реализации жизненного цикла сервисов ODA является ключом к достижению гибкости бизнеса. Эта группировка ODA Framework обеспечивает гибкость бизнеса за счет индустриализации и автоматизации изменений, особенно тех, которые необходимы для создания и модификации сервисов. Она расширяет концепции управления требованиями на протяжении всего жизненного цикла услуг в рамках методологии разработки архитектуры TOGAF (ADM).

В нем представлены модели жизненного цикла, методологии и инструменты. Этот документ:

- определяет общие и уникальные требования – бизнес, операционные и технические путем формального фиксирования единых целевых требований из данного документа IG1166 ODA Architecture Vision и документов IG1167 ODA Concepts and Principles;
- управляет жизненным циклом требований во всех группах ODA Architecture Framework и обеспечивает соответствие принципам (с расстановкой приоритетов и оценкой влияния);
- определяет гибкий подход к реализации на основе моделей (model driven), который поддерживает модели для каждой из точек зрения жизненного

цикла, описанных позже, методологии их трансформации и инструменты для перехода от одной точки зрения к другой;

– описанный здесь инструментарий будет применяться для управления требованиями ко всем артефактам ODA, используемым для работы с различными заинтересованными сторонами и внедрениями с целью автоматизации и ускорения жизненного цикла услуг.

D. Бизнес-архитектура ODA.

Данный документ:

- описывает стратегию продукта и/или услуги, организационные, функциональные, процессные, информационные и географические аспекты бизнес-среды, основанные на принципах ведения бизнеса, бизнес-целях и стратегических движущих силах (включая общую лексику, документированную в TMF071 ODA Terminology);

- определяет организацию предприятия для поставщика цифровых услуг/помощника цифровых услуг;

- разрабатывает функции предприятия и модели управления;

- определяет и устанавливает общую терминологию (TMF 071 ODA Terminology);

- предоставляет карты возможностей, модели процессов и карты ценностей заинтересованных сторон экосистемы цифрового бизнеса.

E. Архитектура информационных систем ODA.

Архитектура информационных систем (ISA) будет включать функциональную архитектуру (IG1167 ODA Functional Architecture), а также концепции и артефакты для архитектур данных и приложений. Она включает документы, диаграммы информационных систем, модели, руководство(-а), охватывающие:

- функциональную архитектуру;

- архитектуру данных (модели, потоки и т. д., безопасность, миграция, жизненный цикл);

- модели данных и приложений.

Было выявлено несколько требований ODA Vision, которые приведены ниже, при этом следует отметить, что в TOGAF архитектура данных и приложений рассматривается как часть архитектуры информационных систем.

Требования:

- гибкость бизнеса;

- снижение стоимости эксплуатации;

- поддержка гибких бизнес-моделей;

- возможность работы с экосистемой;

- поддержка нескольких поставщиков;

- гибкое управление;
- возможность использовать «гибкость облака».

Основные документы, учитывающие указанные требования, следующие: IG1167 ODA Functional Architecture, GB921 Business Process Framework (eTOM) Suite, GB922 Information Framework, GB929 Application Framework (TAM).

Также, TM Forum разработала еще ряд документов, необходимых для построения систем автоматизации современных поставщиков услуг. IG1171 ODA Component Definition определяет состав компонентов, описывает их обобщенную функциональность. IG1117 OSS / BSS Futures Overview и IG1118 OSS / BSS Futures: Preparing the Future Mode of Operation описывают принципы трансформации OSS/BSS.

Место ITIL

В 1980-х гг. британское правительство поручило Центральному агентству по вычислительной технике и коммуникациям (ССТА) разработать общие принципы эффективного использования ИТ-сервисов в Великобритании. Так появились первые документы и были разработаны методики сбора и обработки «лучших практик» (best practices) в управлении ИТ-услугами.

В конце 1980-х – начале 1990 гг. вышла серия книг о том, как управлять ИТ-услугами и о взаимодействии ИТ-области с пользователями этих услуг. Эта библиотека книг и была названа Библиотекой инфраструктуры информационных технологий или ITIL (the IT Infrastructure Library). В дальнейшем ССТА было объединено с Государственной торговой палатой или OGC, которая и стала владельцем библиотеки ITIL.

ITIL, по сути, представляет собой набор публикаций, содержащих рекомендации по предоставлению качественных услуг, а также процессов и компонентов, необходимых для их поддержки. Основная цель ITIL – продвижение современных знаний и обмен опытом в предметной области. Основная особенность ITIL – организация управления услугами в виде совокупности процессов.

В настоящее время ITIL символизирует собой целую индустрию, которая включает в себя:

- 1) фундаментальную информационную базу знаний в области Управления ИТ услугами во всем мире и отражает фундаментальные основы ведущих мировых практик в ИТ-области;
- 2) систему обучения ИТ специалистов «по ITIL»;
- 3) систему сертификации ИТ специалистов;
- 4) ИТ Консалтинг – продвижением процессов ITIL и обучением будущих специалистов отрасли занимается целый ряд компаний-консультантов;

5) программное обеспечение для автоматизации работы ИТ служб «ITIL inside»;

6) всемирное ITIL-содружество на базе множества форумов и сообществ.

Ключевой принцип ITIL это – планомерная и поэтапная трансформация (изменения и улучшения) уже существующих процессов, а ни в коем случае не рекомендации или «стандарты» для «глобальных перестроек».

Центральным и ключевым термином ITIL является Сервис (Service), который в русскоязычной литературе часто называют услугой. Приведем определение из Глоссария ITIL v3:

«ИТ-услуга (Сервис) – способ предоставления ценности заказчику. Услуга обладает следующими характеристиками:

- удовлетворяет одну и более из четко определенных потребностей заказчика;
- направлена исключительно на достижение бизнес-целей заказчика;
- воспринимается заказчиком как единое целое, как результат удовлетворения его ожиданий».

ITIL и eTOM (ODA) не стремятся решить один и тот же набор проблем. Вместо этого они дополняют друг друга. Таким образом, возможно и практически предоставлять услуги ITIL «надлежащей практики» с использованием технологических потоков, построенных с использованием eTOM (т. е. структуры бизнес-процессов). Практики ITIL сфокусированы на предоставлении ИТ-услуг внутри организации и на процессе непрерывного совершенствования услуг и процессов их обеспечивающих. ITIL содержит передовые подходы и лучшие практики для организации модели предоставления ИТ-услуг. В ней описаны некоторые процессы и шаблоны, но не детальная методология реализации процессного подхода. Компания, которая принимает решение использовать ITIL, получает общие принципы, но конкретные процессы должна разработать под свою инфраструктуру самостоятельно.

Подход «ITIL+eTOM» заключается в позиционировании платформы eTOM как средства «внедрения» ITIL, т. е. поддержки разработки решений, ориентированных на ITIL.

ITIL позиционирует себя как набор передовых практик, и это оказывает значительное влияние при применении ITIL на предприятии. Конкретные передовые практики применяются в различных областях бизнеса в соответствии с выбором соответствующего пользователя. Но эти передовые практики требуют включения их в соответствующую предметную область деятельности предприятия, в соответствующий контекст бизнеса, если хотите, в конкретные сквозные бизнес-процессы. А это как раз предлагает eTOM.

То есть eTOM позволяет сформировать «канву» (= бизнес-процесс), в которую вписывается лучшая практика ITIL, применяемая в рамках этой канвы (рис. 8).

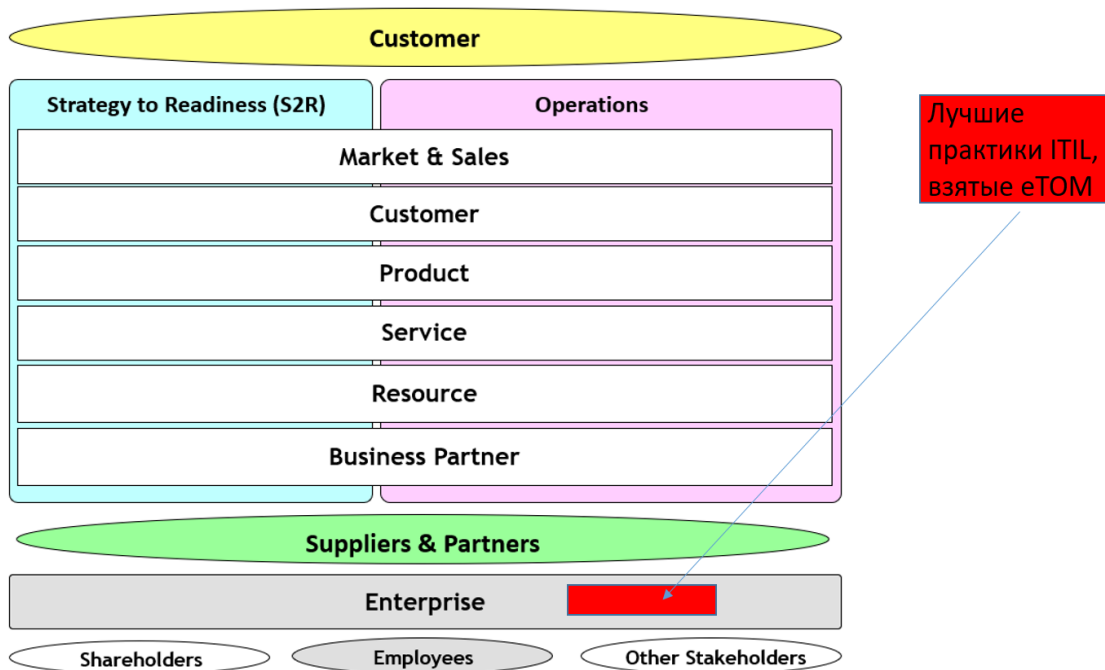


Рис. 8. Место ITIL на карте eTOM

Обе платформы имеют сильные и слабые стороны и их объединение принесло бы значительные выгоды всем отраслевым секторам, связанным с предоставлением конвергентных услуг на рынке.

Принципиальная разница eTOM и ITIL состоит в том, что первая представляет собой конкретные бизнес-процессы, которые отображаются непосредственно в организации. Вторая – представляет собой лучшие практики, которые могут быть внедрены в конкретные процессы в организации (рис. 9). eTOM фокусируется на предоставлении услуг внешним клиентам. ITIL в первую очередь фокусируется на обслуживании внутренних ИТ-заказчиков.

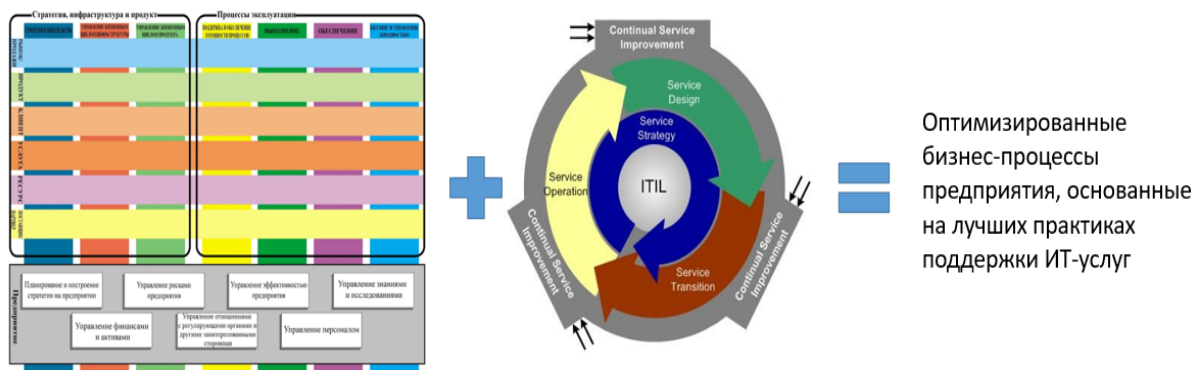


Рис. 9. Иллюстрация результата совместной работы eTOM и ITIL

ITIL:

- ориентирована на развитие и реализацию механизмов поддержки внутренних для предприятия ИТ-услуг;
- представляет практики, которые внедряются в бизнес-процессы предприятия;
- наполняет лучшими практиками поддержки ИТ-сервисов бизнес-процессы предприятия.

eТОМ:

- ориентирована на поддержку бизнеса предприятия;
- содержит полный перечень бизнес-функций предприятия;
- требует учета лучших практик поддержки собственной инфраструктуры и собственных сервисов.

Области ITIL представлены непосредственно в структуре бизнес-операций eТОМ (от релиза к релизу формулировки могут отличаться) (табл. 2).

Таблица 2

Группы функций в eТОМ и ITIL

Группы операций по ITIL	Группы операций по eТОМ
Управление каталогом услуг	Управление каталогом услуг
Управление уровнем обслуживания	Управление качеством обслуживания
Управление емкостью	Управление емкостью услуг
Управление доступностью	Управление готовностью услуг
Управление конфигурациями	Управление конфигурацией и активацией
Управление инцидентами	Управление инцидентами
Управление проблемами	Управление проблемами

Стоит обратить внимание, что процессы передовой практики ITIL совместимы с подходом к проектированию в eТОМ, поэтому мы можем использовать определения отдельных областей ITIL в качестве описаний/определений процессов, которые выглядят аналогично существующим элементам процессов eТОМ. Однако важно отметить, что у них другой акцент, и их следует интерпретировать как определяющие рекомендации или ограничения на способы применения процессов eТОМ. Таким образом, процессы ITIL можно рассматривать как дополнение к существующим элементам процессов eТОМ. Это различие жизненно важно, если мы хотим избежать того, что может рассматриваться как дублирование между областью

процессов ITIL и элементами процессов в других частях eTOM. ITIL и eTOM не перекрываются по группам функций, а дополняют друг друга.

В последних релизах Business Process Framework, начиная с 22-го, упоминание об ITIL отсутствует. Это может быть обусловлено тем, что за многие годы взаимодействия ITIL и eTOM первое как бы растворилось во втором: по названиям группировок в eTOM можно проследить наличие формулировок из ITIL.

Что касается взаимодействия ODA и ITIL, то его есть смысл рассматривать через призму наследственности. То есть, если eTOM включает лучшие практики ITIL на протяжении всех релизов, а ODA, в свою очередь основывается на eTOM, то это можно расценивать как традиционное взаимодействие ODA и ITIL. А дальше нужно отталкиваться от решаемой вами задачи.

2.3. Основные составляющие открытой цифровой архитектуры (ODA)

Реализация проекта ODA основывается на использовании концепции NGOSS/Framework TM Forum, однако предлагает свой набор инструментов:

- функциональная архитектура бизнес-функций FA (Functional Architecture), приходит на замену известной нам расширенной карте бизнес-процессов eTOM (enhanced Telecom Operations Map), необходимой для проектирования и разработки решений OSS/BSS;

- функциональная структура программных приложений поставщика цифровых услуг FF (Functional Framework) появляется вместо карты приложений телекоммуникационной компании TAM (Telecom Applications Map), содержащей в себе классификацию функций, выполняемых используемыми в компании программными приложениями, в том числе системами OSS/BSS;

- модель SID (Shared Information and Data model), которая выступает в качестве эталонной информационной модели для предприятия отрасли связи;

- набор типовых ODA-компонентов;

- Open APIs – набор стандартных API, обеспечивающих быструю, повторяемую и гибкую интеграцию между операциями и системами управления.

Если немного заглянуть вперед, то изменения произошли следующие (рис. 10): Business Process Framework трансформируется в ODA Functional Architecture, Application Framework трансформируется в ODA Functional Framework, модель SID продолжает оставаться одним из инструментов ODA.

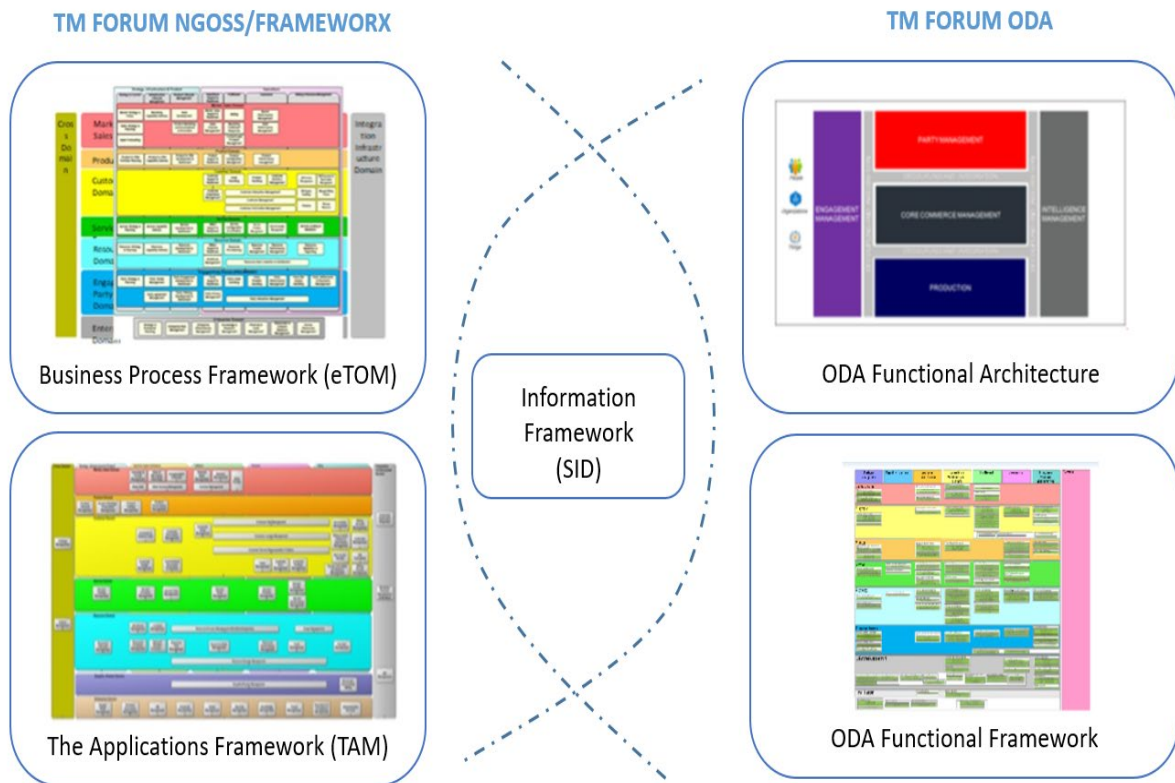


Рис. 10. Артефакты Framework и ODA

Рассмотрим подробнее инструменты, предлагаемые ODA.

2.4. Функциональная архитектура ODA

В основу функциональной архитектуры был заложен ряд принципов методологии TOGAF, которая является более высоким уровнем абстракции. Этот аспект предлагается к самостоятельному изучению чтобы не перегрузить информацией данное пособие.

Каждый из блоков ODA FA (рис. 11) представляет собой группировки бизнес-операций по аналогии с eTOM, только реорганизованных так, чтобы быть универсальным фреймворком и подходить для любого поставщика цифровых услуг (ПЦУ). Другими словами, наполнение блоков ODA отличается от наполнения карты eTOM. Рассмотрим назначение каждого блока.

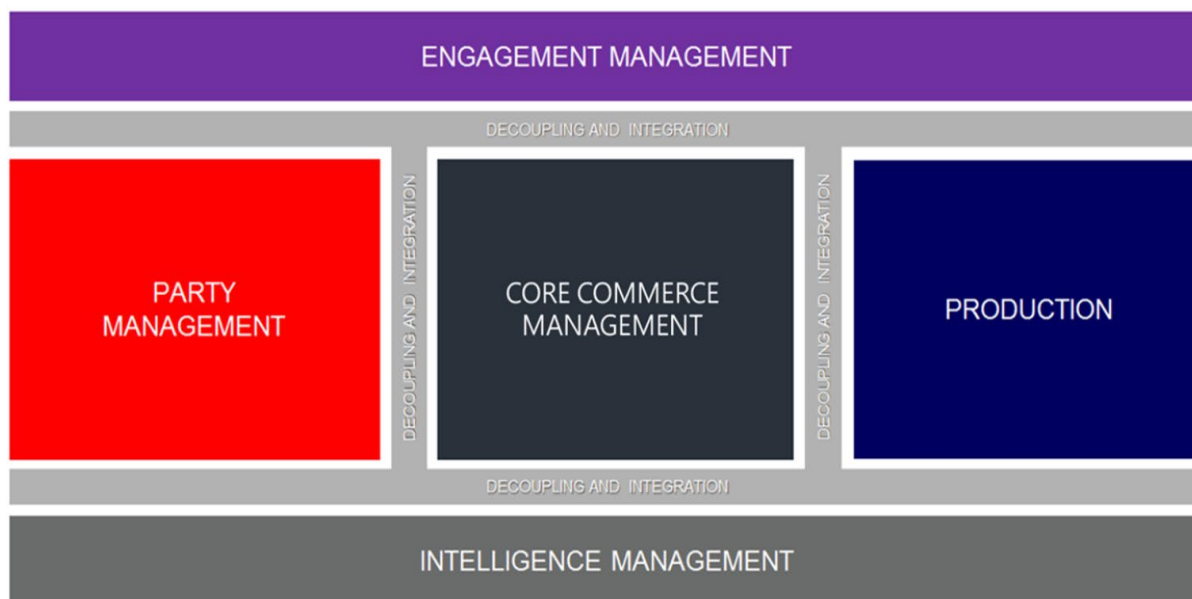


Рис. 11. Концептуальный вид ODA Functional Architecture

Блок управления взаимодействием (Engagement Management) содержит бизнес-функции (операции), отвечающие за взаимодействие с клиентами (пользователями), заказчиками, партнерами, поставщиками, а также за взаимодействие внутри предприятия между работниками и процессами, например, управление личным кабинетом пользователя, управление пользовательским опытом, организацию доступа к контенту и управление им, управление пользовательским интерфейсом, авторизация и аутентификация пользователей и др.

Блок управления взаимодействиями с партнерами (PM, Party Management) содержит операции, реализующие взаимодействие с поставщиками или партнерами – со всеми, кто вовлечен в бизнес-взаимодействие с предприятием. Операции этого блока частично совпадают с операциями горизонтального домена «Поставщик/Партнер» карты eTOM. При этом эти операции собраны из разных вертикалей карты.

Блок управления бизнесом (Core Commerce Management) содержит операции, приводящие именно к получению прибыли. Многие операции перенесены сюда непосредственно из горизонтальных доменов «Рынок/Продажи», «Продукт», «Клиент», «Партнер» карты eTOM, например, управление предложением и продуктовым портфелем (Product Offerings and Products Catalogue Management), управление заказами (Order Handling) и др.

Производственный блок (Production) отвечает за подготовку и управление жизненным циклом служб клиентского и ресурсного обслуживания независимо от типа технологии (физической, виртуальной, проверки технической возможности подключения и т. д.). В рамки производственного

блока входят операции создания, доставки, использования, поддержки, гарантирования работоспособности продуктов у заказчика. Он обеспечивает комплексное управление эксплуатационными функциями для служб, сервисов и ресурсов. Если определить коротко, то этот блок содержит перечень операций создания инфраструктуры, поддержки готовности инфраструктуры и продуктов компании. Аналогично предыдущим блокам операции этого блока частично «перекочевали» из доменов «Услуга» и «Ресурс» (на пересечении с вертикалью «Эксплуатация ресурсов» и «Поддержка готовности» (RM&O Support & Readiness) карты eTOM.

Блок интеллектуального управления (Intelligence Management) содержит операции, отобранные из разных доменов карты eTOM, которые отвечают за аналитику по отношению к бизнес-процессам, оборудованию или поддержке принятия решений. В частности, в этот блок собраны такие группировки операции, как анализ результатов продаж (Sales Performance Management), оценка производительности продукта (Product Performance Management), управление качеством услуг (Service Quality Management), управление производительностью (Resource Performance Management) и др.

Наполнение блоков реализовывалось перераспределением операций из доменов карты eTOM по блокам ODA. Карта eTOM стала как бы донором для ODA (рис. 12).

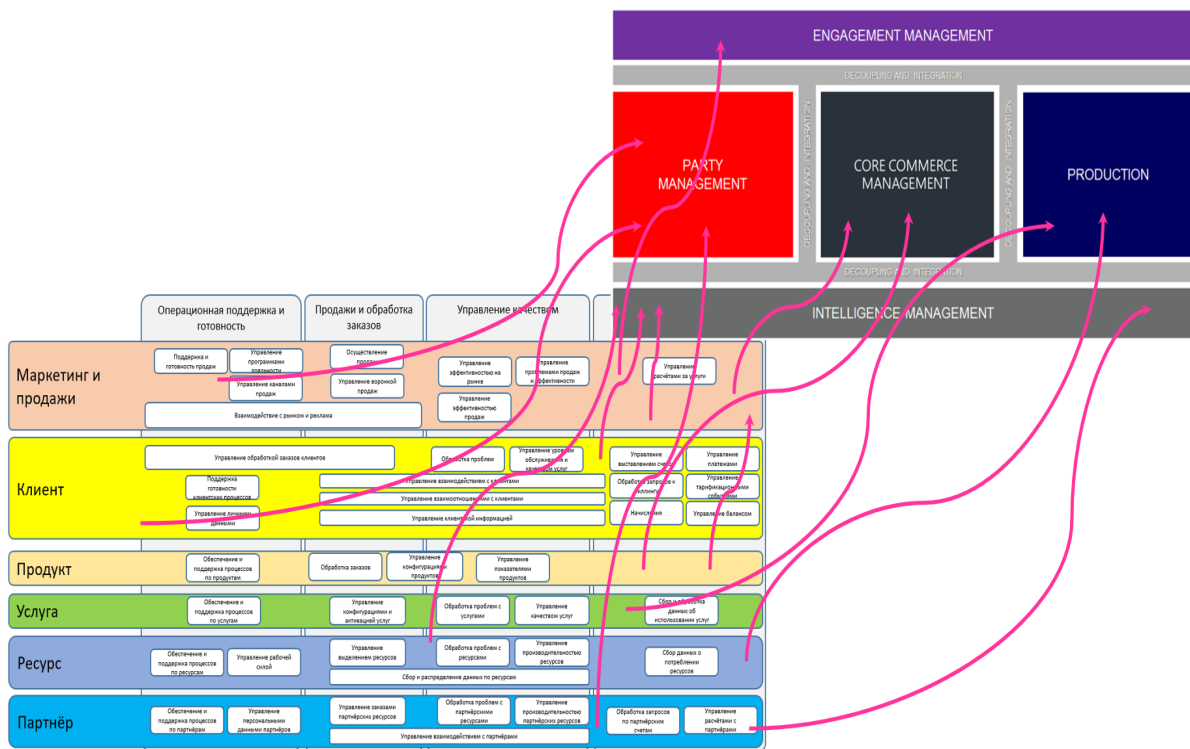


Рис. 12. Иллюстрация перераспределения операций из eTOM в ODA

2.5. Информационная модель SID

Структура SID является инструментом не только в концепции Framework, но теперь и в ODA. Бизнес-сущности SID, точно так же, как и в Framework, объединяются по функционально-смысловому признаку в так называемые ABE (Aggregate Business Entity) – информационные блоки (например, «Клиент», «Счет клиента», «Заказ»), которые в свою очередь группируются в домены. Формально домен можно определить, как совокупность данных и функций, связанных с определенным аспектом деятельности компании. Каждый домен является относительно замкнутым – элементы внутри него тесно связаны между собой, тогда как связь между доменами выражена гораздо слабее.

2.6. Структура программных приложений ODA

Структура приложений (Functional Framework, FF) представляет собой карту функций программного обеспечения, поддерживающего бизнес-процессы поставщика цифровых услуг. Она содержит группы функций, структурированные на основе горизонтальных и вертикальных доменов по принципу карты TAM. Другими словами, FF – это функционально ориентированная система библиотек. Функция из FF является базовым строительным блоком программного ODA-компонента, т. е. определяет часть его функциональности. Разработка приложений для автоматизации на основе ODA-компонент принципиально отличает подход ODA от подхода Framework. Последний, как известно, описывал принципы и подходы к разработке OSS/BSS, которые, по сути, упраздняются в новой концепции. FF сильно перекликается с картой TAM и практически имеет тот же смысл с точки зрения жизненного цикла разработки IT-решения. На рис. 13 *а* и *б* показаны функции поддоменов группы ресурсов уровня 2 в соответствии с их локализацией в горизонтальных и вертикальных доменах.

FF помогает разделить функциональность крупного приложения на более мелкие и управляемые программные сервисы – ODA-компоненты. Каждый домен FF отвечает за определенный аспект функциональности информационной системы. Видно, что все функции приложений имеют свой индивидуальный цифровой идентификатор, а описание каждой функции приложения можно найти в документе GB929.

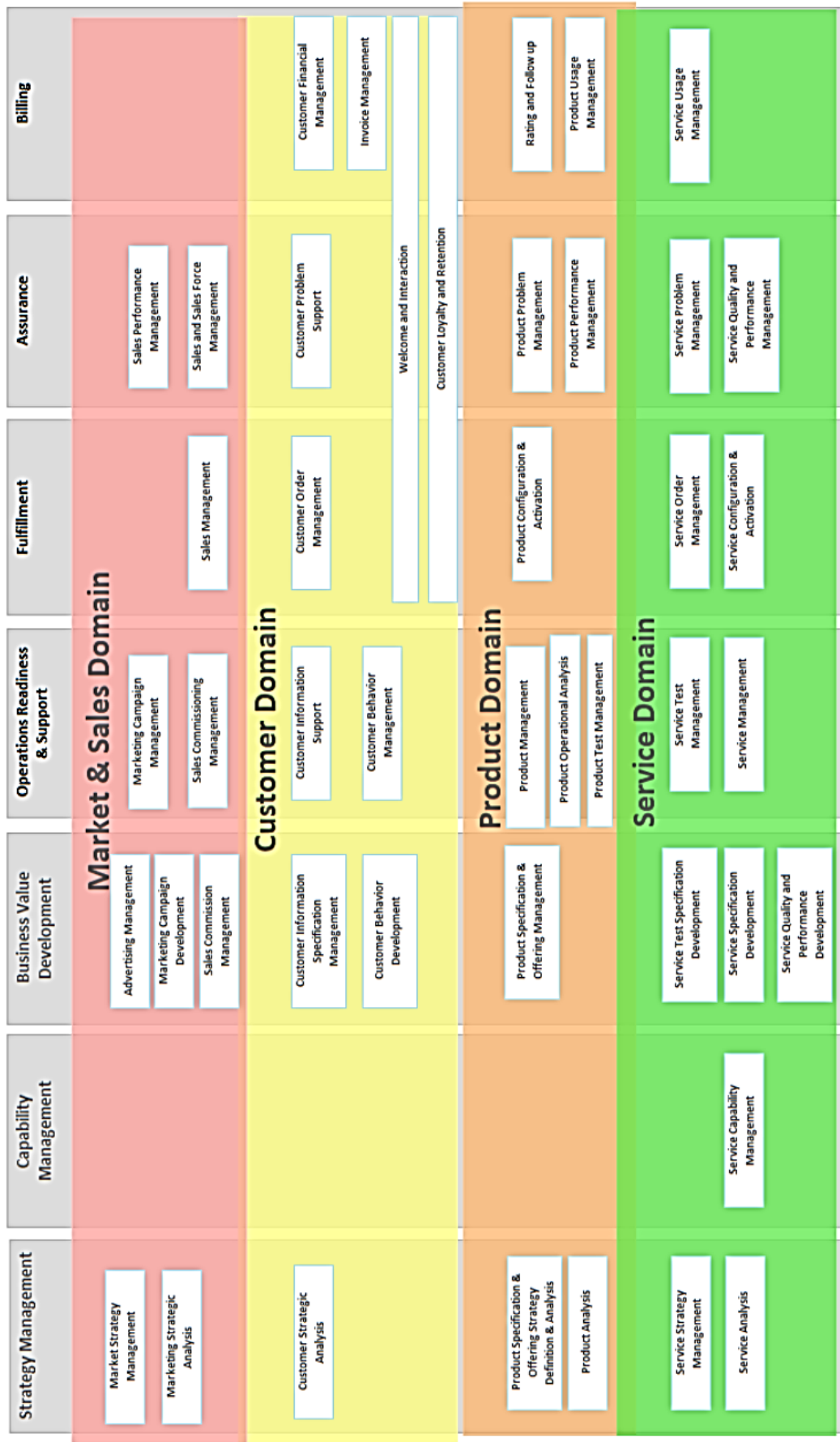


Рис. 13. (Начало) Функциональность блока ресурсов в FF

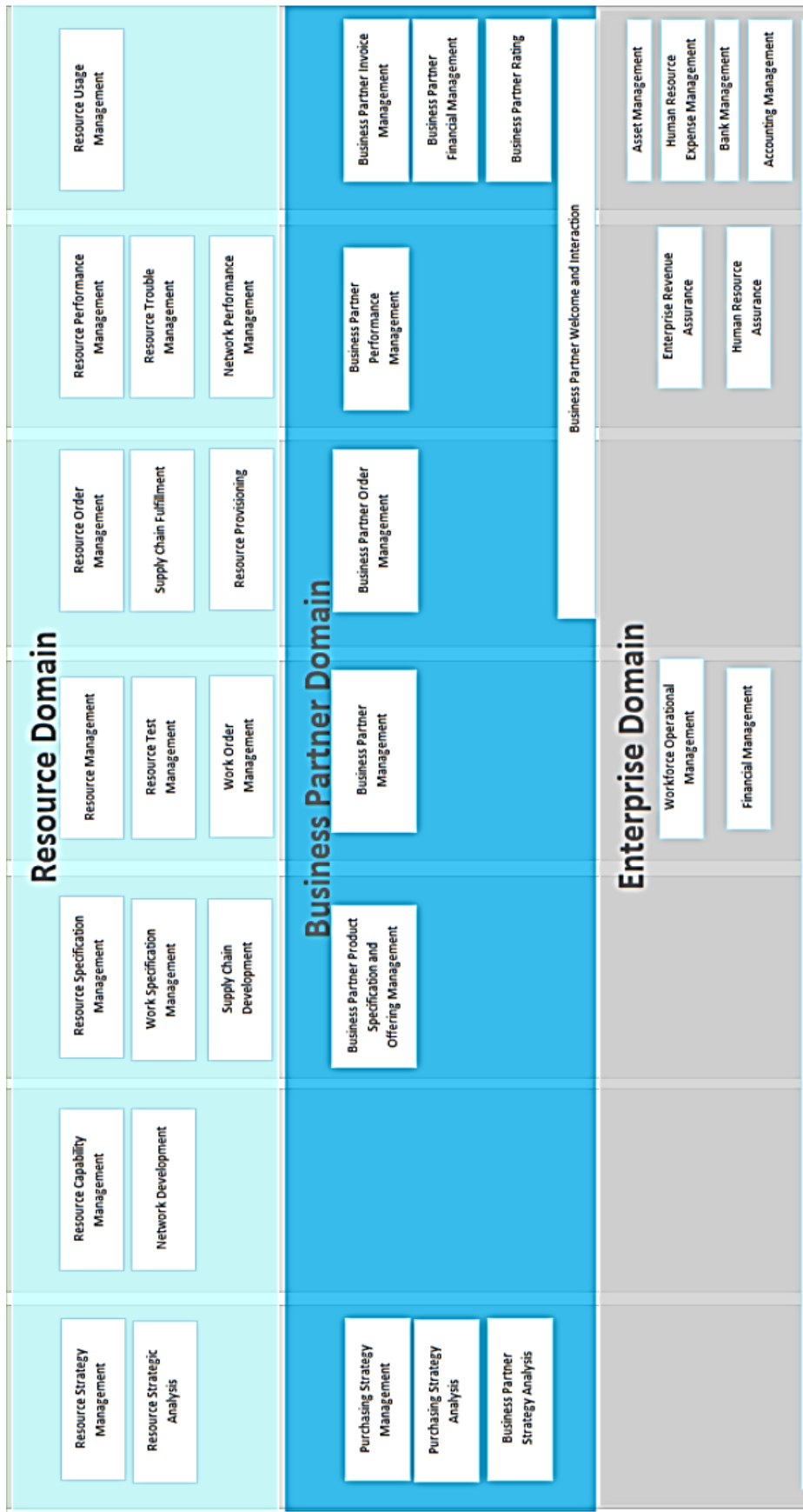


Рис. 13. (Окончание) Функциональность блока ресурсов в FF

2.7. ODA-компоненты

Разработка приложений для автоматизации на основе ODA-компонентов принципиально отличает концепцию ODA от Framework. ODA-компоненты – это автономные программные неделимые модули, предназначенные для реализации конкретных бизнес-целей. Они обладают минимальной, но при этом достаточной функциональностью для описания системы. За счет использования Open API компоненты легко интегрируются и взаимозаменяются.

Модули обычно представляют собой логически связанные группы функций или компонентов, которые выполняют определенные задачи или обрабатывают определенные типы данных. Например, в системе технического учета может быть модуль местоположения. Модуль местоположения отвечает за учет и хранение информации о местоположении различных объектов или ресурсов внутри сетевой инфраструктуры. Этот модуль обеспечивает возможность точно определить, где находятся различные элементы сети или оборудования, что является важным для эффективного управления и контроля.

Каждый модуль обычно имеет четкие интерфейсы и внутреннюю логику работы. Компоненты представляют собой независимые функциональные единицы, которые могут быть переиспользованы в различных модулях или системах. Они обычно представляют собой абстрактные блоки, реализующие определенные функции или сервисы. Границы модулей и компонентов могут быть определены на основе функциональных требований, архитектурных принципов и организационных потребностей системы. Хорошо определенные границы позволяют обеспечить логическую разделенность, независимость и повторное использование компонентов, что способствует гибкости и эффективности разработки и поддержке системы технического учета.

ODA-компонент графически представляется в виде шестиугольника (рис. 14). В нем описываются определенные функции, которые ему необходимы, и которые могут быть предоставлены через API.

ODA-компоненты используют ту же функциональную структуру, что и TAM, но с более детальной группировкой функций в программные блоки на основе микросервисов.

Компоненты развертываются на ODA Canvas, который не только предоставляет среду выполнения, но и также предоставляет службы, которые используются всеми компонентами для таких вещей как идентификация безопасности и управление доступом.

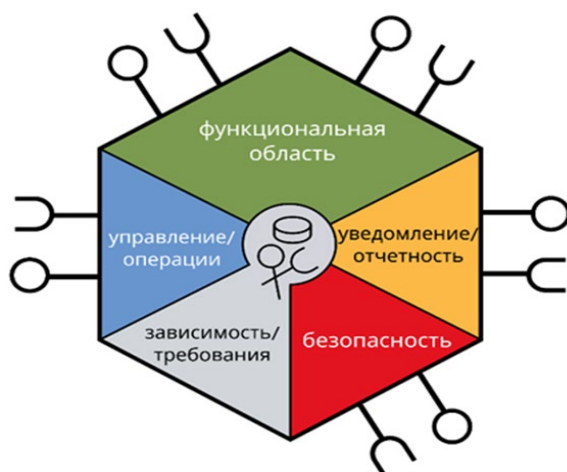


Рис. 14. Структура ODA-компонента

Функциональная область отвечает за то, что компонент выполняет свою основную задачу, исходя от функциональности. Область безопасности говорит о том, что компонент должен соответствовать требованиям конфиденциальности. Область, отвечающая за уведомления и отчетность информирует о том, что компонент должен сообщать данные и обмениваться информацией с другими компонентами системы. Управление и операции необходимы для того, чтобы компонент был управляемым и работоспособным.

Чтобы выделить необходимые компоненты, следует обратиться к карте ODA-компонентов. Карта ODA-компонентов представляет собой структурированное описание блоков, содержащих компоненты (рис. 15). Она позволяет визуализировать и организовать компоненты ODA, их взаимосвязи и функциональные возможности.

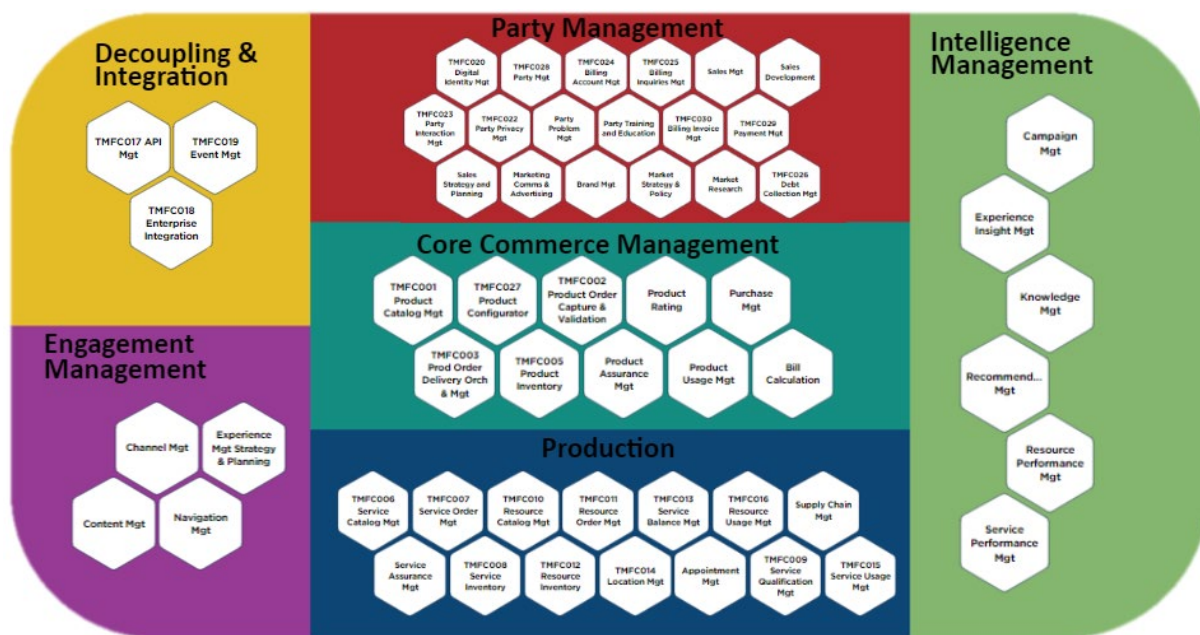


Рис. 15. Карта ODA-компонентов

Карта ODA-компонентов включает следующую информацию:

- название компонента: каждый компонент ODA имеет свое уникальное название, которое отражает его роль и функциональность;
- описание компонента: краткое описание того, что делает компонент и какую функциональность он предоставляет в рамках ODA;
- взаимосвязи: указываются связи и зависимости между компонентами ODA. Это позволяет понять, как компоненты взаимодействуют друг с другом и как они взаимодействуют с внешними системами;
- функциональные возможности: описываются основные функции и возможности, которые предоставляет каждый компонент. Это помогает понять, какие задачи компонент выполняет и какие сервисы он предоставляет для других компонентов или пользователей;
- интерфейсы: указываются типы интерфейсов, через которые компоненты взаимодействуют друг с другом или с внешними системами.

Карта ODA-компонентов помогает «визуализировать» архитектуру системы и логически организовать компоненты. Она облегчает понимание взаимосвязей между компонентами, их ролей и вклада в общую функциональность системы. Карта ODA-компонентов является важным инструментом при проектировании, разработке и управлении системами на основе открытой цифровой архитектуры.

2.8. Межкомпонентное взаимодействие и аспекты безопасности

Общие положения

Один из основных составляющих элементов ODA – прикладные программные интерфейсы API (Application Programming Interface). API – это набор определенных правил, позволяющих различным приложениям взаимодействовать друг с другом. Он действует как промежуточный уровень, который обрабатывает передачу данных между системами, позволяя компаниям открывать данные и функции своих приложений для внешних сторонних разработчиков, деловых партнеров и внутренних отделов своих компаний.

Open API – это интерфейсы прикладного программирования с открытым исходным кодом, доступ к которым можно получить по протоколу HTTP. Также известные как общедоступные API, они имеют определенные конечные точки API и форматы запросов и ответов [19].

Определения и протоколы в API помогают компаниям соединять множество различных приложений, которые они используют в повседневных операциях, что экономит время сотрудников и устраняет разрозненность,

мешающую совместной работе и инновациям. Для разработчиков документация API предоставляет интерфейс для связи между приложениями, упрощая интеграцию приложений.

При использовании API в рамках ODA компоненты могут взаимодействовать следующим образом:

- компонент *A* может использовать API компонента *B* для вызова его функций или получения данных. Например, компонент *A* может вызывать API компонента *B* для получения информации о ресурсах сети или выполнения определенных операций;
- компонент *A* может отправлять события или уведомления через API компонента *B*. Например, компонент *A* может использовать API компонента *B* для отправки событий об авариях или предупреждений, которые требуют внимания или дальнейшей обработки;
- компоненты могут обмениваться данными через API для обеспечения согласованности информации. Например, компонент *A* может передавать данные о клиентах или услугах компоненту *B*, чтобы обновить или расширить информацию, необходимую для выполнения своих функций.

Все API в ODA строятся в соответствии с определенными стандартами и спецификациями, которые обеспечивают совместимость и согласованность между компонентами. Такие стандарты, как TM Forum API (рис. 16), определяют набор общих интерфейсов и протоколов, которые позволяют компонентам ODA взаимодействовать между собой независимо от их реализации или поставщика. API обеспечивает удобный способ интеграции компонентов и расширения функциональности системы без необходимости полной замены или изменения всех компонентов.

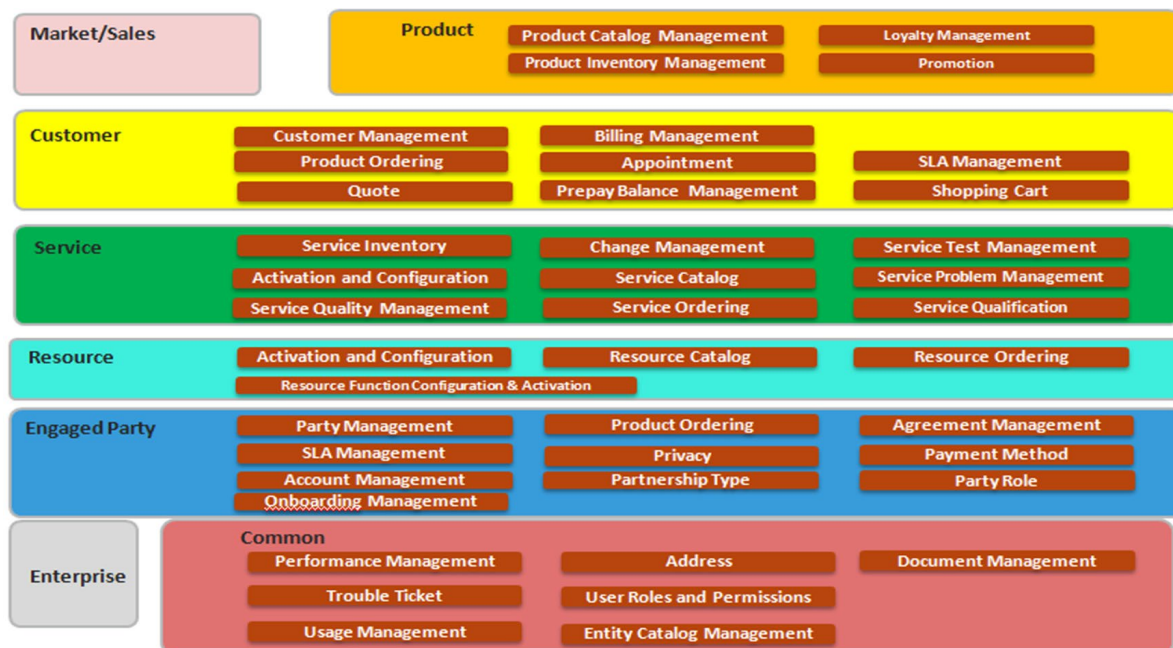


Рис. 16. Структура API для реализации подхода ODA

Открытые API позволяют различным программным системам взаимодействовать без необходимости дорогостоящей и трудоемкой интеграции. Они также необходимы для того, чтобы компании могли взаимодействовать с внешними партнерскими платформами. Открытые API-интерфейсы не зависят от технологий и могут использоваться в любом сценарии цифровых услуг – от разработки ИТ-систем следующего поколения, до предоставления услуг Интернета, умных городов или внедрения функций виртуальной сети. Интерфейсы основаны на REST и могут использоваться практически любым языком программирования.

Каждый ODA-компонент имеет свой список Exposed API и Dependent API:

1) **Exposed APIs (экспонируемые API)** – это список API, доступных для использования в компоненте. Они определяют, как компонент может взаимодействовать с внешним миром и какие операции он может выполнить;

2) **Dependent APIs (зависимые API)** – это список API, которые компонент использует для удовлетворения своих требований. Они определяют, какие другие компоненты должны быть доступны для использования, чтобы этот компонент работал должным образом.

Взаимодействие между Exposed APIs и Dependent APIs заключается в том, что компонент может использовать Dependent APIs для выполнения своих функций, а также предоставлять Exposed APIs для использования другими компонентами. При проектировании компонента необходимо учитывать как Exposed APIs, так и Dependent APIs, чтобы обеспечить корректную работу всей системы.

Кроме списков Exposed и Dependent API, каждый компонент обладает перечнем Event. Список Event – это раздел, в котором перечислены события, которые компонент может генерировать, а также список событий, которые он может использовать. Каждое определенное событие может иметь несколько источников.

Аспекты безопасности

Нельзя не обратить внимание на задачи безопасности, особенно когда решение строится из компонентов, разработанных разными вендорами.

Управление безопасностью в ODA рассматривается как неотъемлемая часть гибкого управления изменениями.

Основные концепции информационной безопасности – это конфиденциальность (Confidentiality), целостность (Integrity), доступность (Availability). Конфиденциальность подразумевает предотвращение несанкционированного раскрытия данных/информации. Целостность гарантирует,

что данные/информация не могут быть изменены таким образом, чтобы сделать их неточными или бесполезными для своей цели. Доступность предполагает, что данные/информация должны быть легко доступны для санкционированного использования. В целом это означает, что системы должны быть спроектированы таким образом, чтобы обеспечить их отказоустойчивость.

При разработке компонентов ODA следует руководствоваться современными концепциями безопасности с нулевым уровнем доверия (zero-trust). За исключением случаев, когда это необходимо для интеграции устаревших подсистем, компромиссы недопустимы. Для того чтобы устранить необходимость в доверии к системе и/или сети, необходимо заложить доверие в устройства и сервисы. То есть многофакторная аутентификация является обязательным условием архитектуры с нулевым доверием.

«Нулевое доверие» означает, что данные должны быть защищены:

- в состоянии покоя;
- при передаче;
- во время обработки внутри системы и за ее границами (насколько это возможно).

Защита в состоянии покоя

Защита в состоянии покоя требует, чтобы все формы сохраняемых данных хранились в зашифрованном виде. Это касается данных, хранящихся в базах данных SQL и NoSQL, файлов, съемных носителей и любых других хранилищ данных. К числу данных, подлежащих шифрованию, относятся:

- данные клиентов/арендаторов;
- данные конфигурации;
- ключи и учетные данные;
- хранилища конфигураций;
- хранилища компонентов (здесь основное внимание уделяется защите от несанкционированного доступа);
- системные журналы;
- журналы аудита.

Защита при передаче

Защита данных при передаче через внешний, публичный Интернет в центры обработки данных или пограничные узлы является стандартом уже много лет. Подход ODA Zero-Trust требует распространения этой концепции на все коммуникации, включая данные, которыми обмениваются компоненты ODA, и данные, которыми обмениваются внутри компонентов ODA (например, между микросервисами).

В дополнение к требованию шифрования при транспортировке данных необходимо также контролировать пути передачи данных. Для этого необходимо внедрить так называемые сетки сервисов (Service Meshes), обеспечивающие функции коммуникационной политики и точки реализации политики.

Чтобы проиллюстрировать их назначение, представьте себе простой компонент ODA, в котором каждый арендатор имеет экземпляры микросервиса *A*, микросервиса *B* и хранилища данных *C*. Функция политики будет гарантировать, что эти сервисы могут использоваться только по назначению:

- арендатор *X* может получить доступ только к своим экземплярам микросервисов *A_x*, *B_x* и к своему фрагменту хранилища данных *C_x*;
- арендатор *Y* может получить доступ только к экземплярам *A*, *B* и *C* соответственно.

Подробно аспекты безопасности рассматриваются в документе «GB1008 ODA Governance, Security & Privacy Architecture Principles». Учитывая ограниченный объем учебного пособия, предлагаем самостоятельно изучить документ.

2.9. Различие подходов Frameworkx и открытой цифровой архитектуры (ODA)

ODA делает акцент на открытость и гибкость, позволяя использовать открытые стандарты, API и интеграцию с различными системами. Она поддерживает концепцию модульности и микросервисов, что позволяет компонентам и модулям работать независимо и быть переиспользованными в разных контекстах.

Концепция Frameworkx предоставляет более детализированные модели, структуры и процессы для управления операторами связи. Она определяет конкретные этапы и шаги, а также стандарты и рекомендации для выполнения этих задач. Она также предоставляет специализированные модели данных для учета, биллинга, управления услугами и других аспектов операторов связи.

Со своей стороны, ODA оставляет больше свободы в организации и определении бизнес-процессов. Он предоставляет общую архитектурную основу и компоненты, но не определяет детали каждого процесса. Это позволяет организациям более гибко адаптировать и настраивать бизнес-процессы в соответствии со своими уникальными потребностями и стратегиями.

Кроме того, ODA акцентирует внимание на использовании современных технологий, таких как облачные вычисления, микросервисная архитектура, искусственный интеллект. ODA поддерживает интеграцию с различными внешними системами и платформами, что способствует созданию открытых экосистем и сотрудничеству с другими организациями.

В целом, хотя Framework и ODA имеют общие цели по улучшению управления бизнесом и процессами, они различаются по своей специфической области применения, подходам и уровню детализации. Framework сфокусирован на операторах связи и предоставляет конкретные модели и процессы для этой отрасли, в то время как ODA охватывает более широкий спектр отраслей и акцентирует внимание на гибкости, интеграции и использовании современных технологий для всех поставщиков цифровых услуг.

3. ПОСТРОЕНИЕ СИСТЕМ НА ОСНОВЕ ODA

В данном разделе рассматривается применение новых разработок TM Forum на практике. В качестве примеров рассмотрим две системы, «требующие перевода» на ODA. Одна из систем – это «АРГУС NRI» (Network Resource Inventory, или Технический Учет) – система класса Inventory, предназначенная для автоматизации процессов учета, обработки и анализа информации по линейно-техническим объектам, сооружениям сети и услугам с помощью современных информационных технологий. Вторая – система управления рабочими ресурсами «АРГУС WFM» (Workforce Management, WFM). WFM необходима операторам связи (и не только) для автоматизации бизнес-процессов управления выездными специалистами – инсталляторами и инженерами, выезжающими на адреса к клиентам для установки, настройки или восстановления услуг.

Параграфы, посвященные каждой из упомянутых систем, имеют примерно одинаковую структуру для удобства восприятия. Сначала идет описание задач и функциональности систем, затем предлагается один из бизнес-процессов (БП), поддерживаемый системой, или фрагмент архитектуры системы. И, наконец, предлагается фрагмент решения, основанного на ODA-компонентах. Предложенные фрагменты – это лишь примеры, охватывающие часть функциональности систем, но никак не полноценные описания решений.

3.1. Система технического учета «АРГУС NRI»

Задачи учета сетевых ресурсов, линий связи, инженерной инфраструктуры, являются ключевыми задачами поддержки эксплуатации сетей. Максимально точная и достоверная информация о наличии и доступности ресурсов, их загруженности, месторасположении является необходимым кирпичиком в задании качества оказания услуг клиентам, внедрения новых услуг, быстрой адаптации к меняющимся требованиям клиентов [20].

Структура системы АРГУС NRI включает в себя укрупненно три блока «Учет и паспортизация», «Управление услугами» и «Функциональные модули» (рис. 17). Ниже подробно рассмотрено, что включено в каждый блок системы.

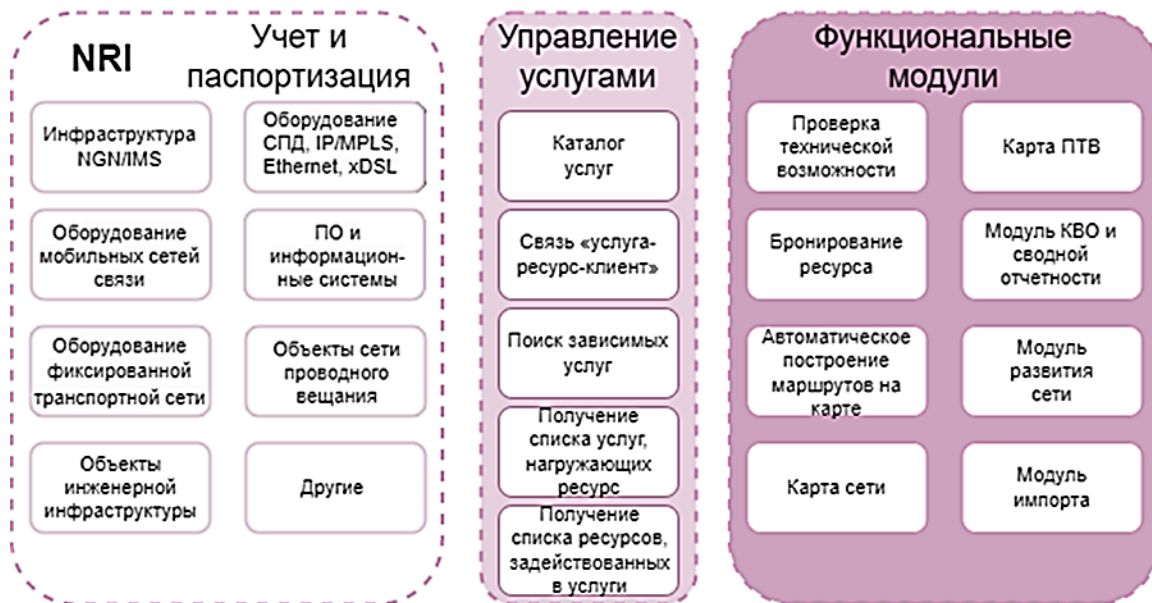


Рис. 17. Структура системы «АРГУС NRI»

Блок «Учет и паспортизация» позволяет осуществлять учет и паспортизацию следующих технологических доменов: оборудование сетей NGN/IMS, сооружения, оборудования транспортной сети и ВОЛС, объекты кабельной инфраструктуры и многое другое (рис. 18).



Рис. 18. Составляющие блока «Учет и паспортизация».

Сервисный блок (услуги) включает в себя учет услуг, управление услугами. В «АРГУС NRI» реализована возможность «занимать услугами ресурсы сети», а для автоматизации этих процессов существует специальный набор инструментов «линейной бухгалтерии», осуществляющий подбор ресурсов под услуги и их резервирование.

Функциональный блок включает в себя набор функций и инструментов, реализующих бизнес-цели оператора связи в рамках задач эксплуатационного управления ресурсами. К функциям относятся, например, автоматическая проверка технической возможности, бронирование ресурса, автоматическое построение маршрутов на карте, отображение охвата технологиями на карте и т. д.

Функциональность системы «АРГУС NRI»

Функциональность системы отражена группировками TM Forum Business Process Framework (рис. 19):

- Resource Process Management (Управление процессами уровня ресурсов) предоставляет взаимодействие с системами NMS;
- Resource Inventory Management (Учет ресурсов) дает возможность получать консолидированные отчетности по всей инфраструктуре и оборудованию на сети Оператора;
- Location Management (Управление местоположением ресурса) позволяет проводить поиск местоположения ресурса на карте;
- Capability Specification Management (Управление возможностями ресурсов) дает контроль и упрощение внесения данных за счет использования шаблонов;
- Service Inventory Management (Учет услуг) подразумевает учет услуг, связанных с соответствующим ресурсом;
- Network number Inventory Management (Учет номерной емкости сети) дает автоматизированный подбор номера и предоставление пула номеров для выбора номера клиентом;
- Resource Order Management (Управление заказами на ресурсы) позволяет проводить автоматическую проверку технической возможности предоставления услуги.



Рис. 19. Возможности NRI по TM Forum Business Process Framework

Бизнес-процесс ввода оборудования в эксплуатацию

На рис. 20 представлен бизнес-процесс ввода оборудования в эксплуатацию. Рассмотрим его подробнее. Этот процесс занимает основное место в рамках NRI, так как он отвечает за выполнение основной функции системы – учета сетевых ресурсов.

Бизнес-процесс включает в себя формирование заявки на выдачу оборудования, обработки заявки и контроля выполнения задачи. Ввод в эксплуатацию оборудования напрямую влияет на удовлетворенность клиентов и, соответственно, на репутацию поставщика услуг.

Первый шаг данного бизнес-процесса включает в себя формирование заявки клиентом на необходимое обслуживание (в рамках NRI это будет ввод оборудования в эксплуатацию). Затем сторонняя система (Naumen) проверяет запрос и формирует заявку на выдачу оборудования. После чего главное управление материально техническим обеспечением формирует заказ и переводит в статус подготовки, далее формирует требования в заказе и подбирает оборудование на складах, если оборудование есть, переводит заказ в статус «обработан».

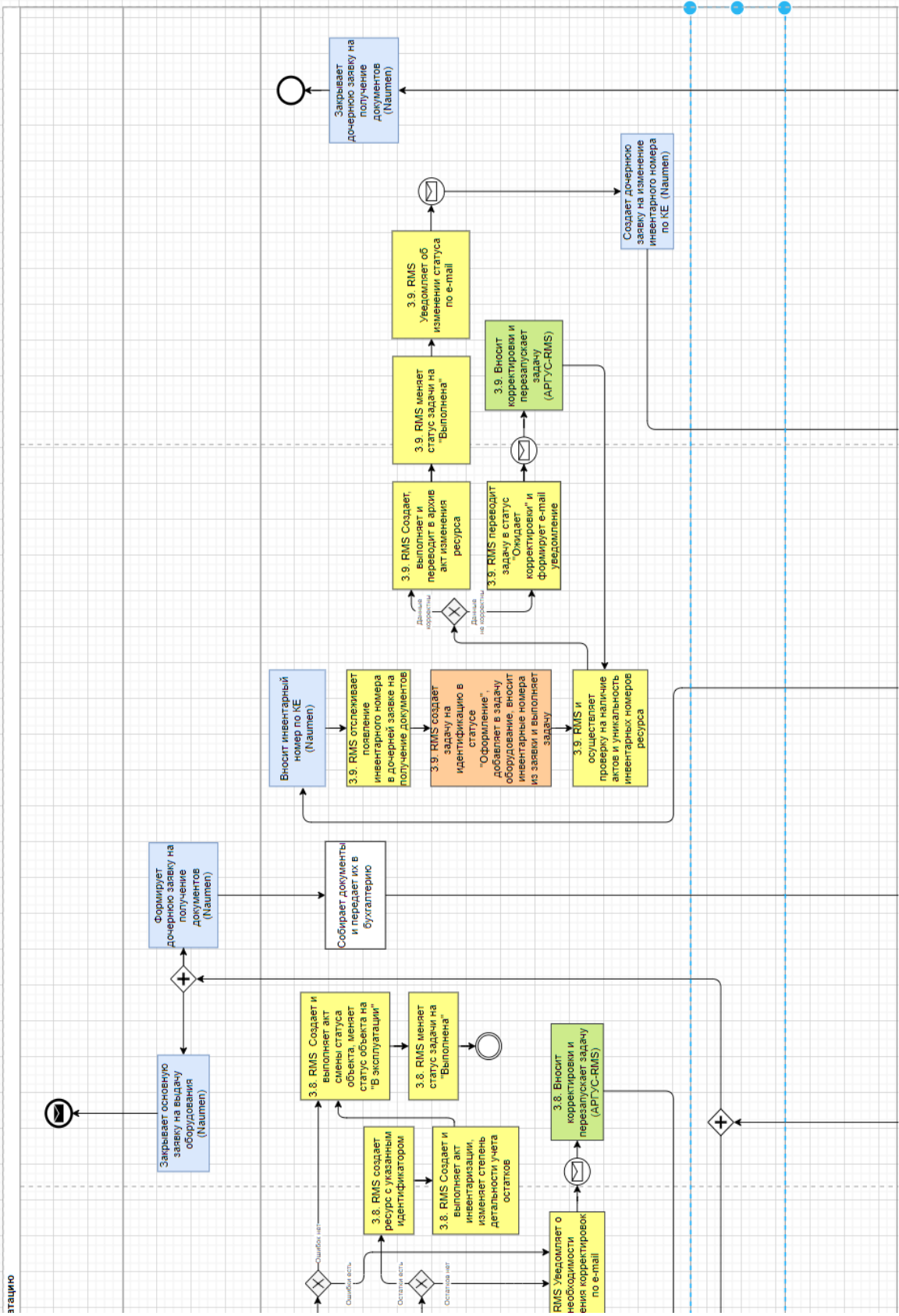


Рис. 20. (Окончание) Бизнес-процесс ввода оборудования в эксплуатацию

Работник склада собирает оборудование и заносит в систему учета. После этого подписываются документы. Система создает задачу, использует данные местоположения. Далее начинается поиск ресурсов по указанным номерам, создается и выполняется акт смены статуса объекта из «в эксплуатации» на «выполнено».

После того, как акт ввода в эксплуатацию и инвентарный номер переданы, система добавляет в задачу оборудование, вносит номера из заявки и выполняет задачу. Далее происходит изменение и наклейка инвентарного номера, подпись акта ввода в эксплуатацию и полное закрытие заявки.

Модули системы «АРГУС NRI»

Основные модули в системе могут варьироваться в зависимости от конкретной предметной области и требований к системе. Конкретные модули выбираются на основе функциональных требований и целей системы.

На рис. 21 представлен вариант границ модулей/компонентов системы технического учета.

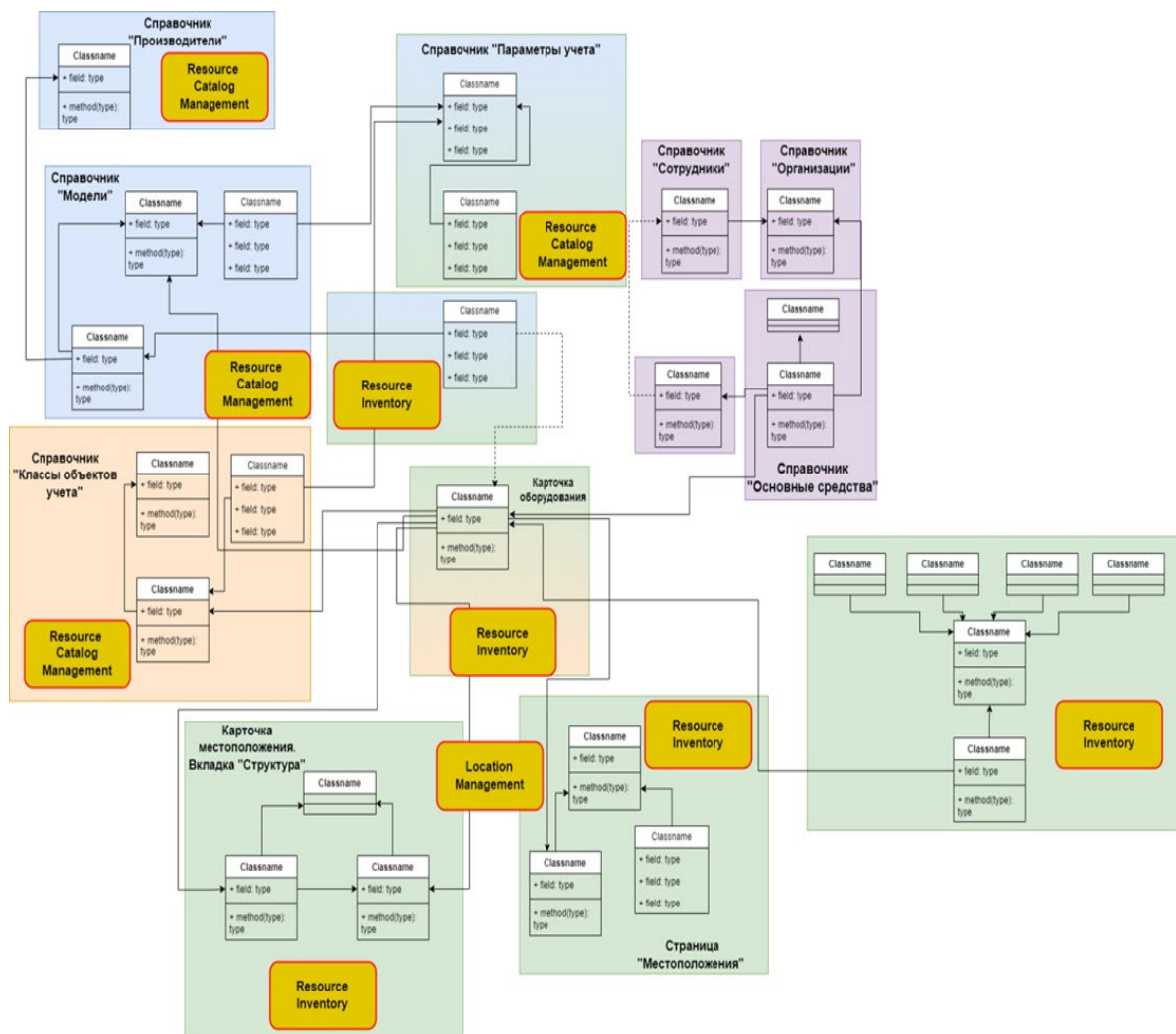


Рис. 21. Выделенные компоненты для системы NRI

Для системы NRI необходимы компоненты, которые будут реализовывать следующие функции.

ОДА-компонент, отвечающий за справочные данные сетевых элементов. Он должен хранить информацию о справочнике «Производители», справочнике «Модели», справочнике «Классы объектов учета», справочнике «Параметры учета», т. е. следующую информацию об элементах сети:

- идентификация элементов – уникальные идентификаторы для каждого элемента сети, которые позволяют однозначно распознать элементы в системе;
- тип элемента – классификация элементов по типу устройств, например, маршрутизаторы, коммутаторы, серверы и т. д.;
- производитель – информация о производителе элемента, включая название компании или бренда оборудования;
- модель и версию – указание модели и версии каждого элемента, что помогает идентифицировать конкретные характеристики и возможности элемента;
- технические характеристики – дополнительные технические параметры сетевого элемента, такие как пропускная способность, поддерживаемые протоколы, интерфейсы, количество портов и другие технические характеристики;
- серийный номер – уникальный номер, присвоенный оборудованию производителем, который помогает идентифицировать конкретное устройство;
- состояние компонента – информация о текущем состоянии компонента, такая как работоспособность, доступность или статус на обслуживании;
- зависимости и связи – связи и зависимости компонента с другими компонентами или ресурсами в сети, что позволяет визуализировать и понять взаимосвязи между различными элементами сети;
- история изменений – историческая информация о внесенных изменениях в компоненты, такая как дата создания, модификации или удаление компонента.

Следующим понадобится компонент, хранящий в себе информацию о местоположении, а также управляющий ею:

- местоположение – местоположение оборудования, включая информацию о здании, комнате или точном расположении в сети;
- история и изменения – исторические данные о изменениях в состоянии, конфигурации или местоположении оборудования;

- конфигурация и настройки – информация о конфигурации и настройках оборудования, такая как – IP-адреса, маски подсети, VLAN-настройки, доступы и другие параметры, которые определяют его работу в сети;
- состояние оборудования: информация о текущем состоянии оборудования, такая как – активное, отключенное, в ремонте и т. д.

Также необходим компонент NRI, учитывающий физические и логические связи сетевого оборудования – по сути, основной компонент учета сетевого оборудования, ИТ-оборудования, лицензий, номерной емкости и т. д.

Выделение ODA-компонентов для системы NRI

Для выделения конкретных ODA-компонентов из набора «стандартных» необходимо изучить описание компонентов на сайте tmforum.org, и, отталкиваясь от требуемой функциональности модулей системы, подобрать компоненты.

На рис. 22 изображена карта ODA-компонентов, включающая в себя шесть функциональных блоков ODA FA. На ней выделены компоненты, которые выбраны в нашем примере для системы технического учета.

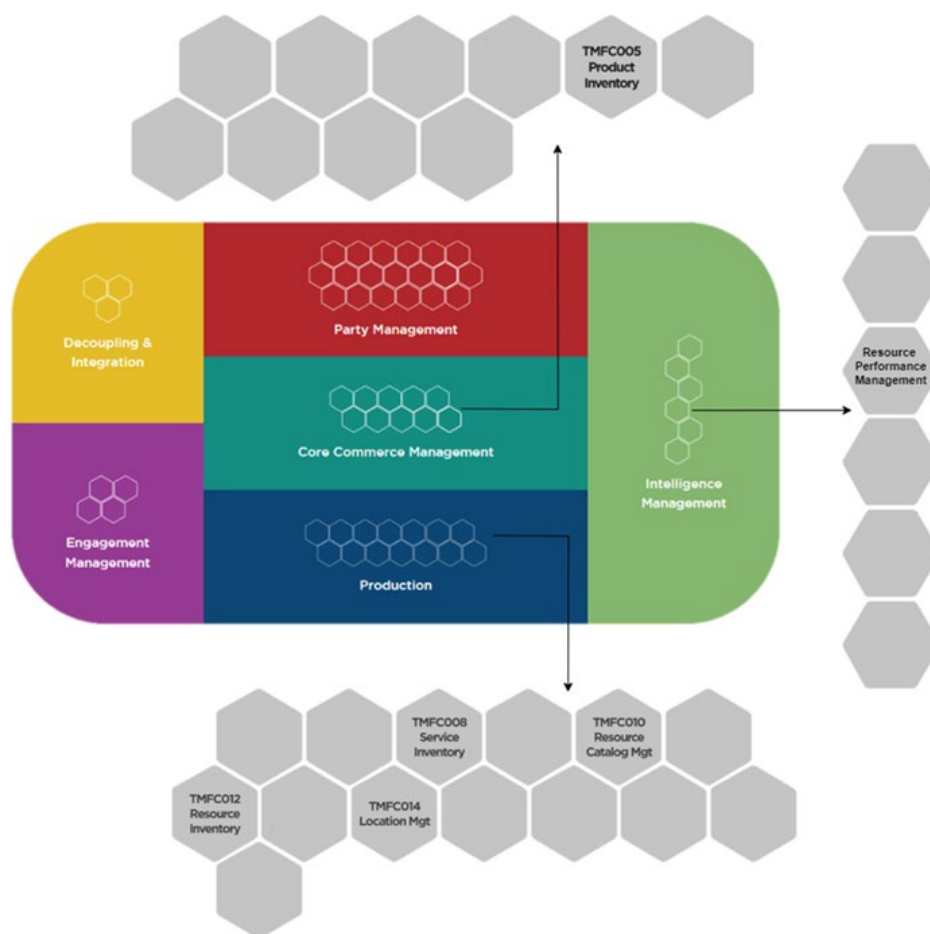


Рис. 22. Карта ODA-компонентов с подходящими компонентами для системы технического учета

Из блока Core Commerce Management был выделен один компонент TMFC005 Product Inventory. Он отвечает за учет продуктов. Компонент Product Inventory имеет функциональные возможности, которые позволяют создавать элементы учета, организовывать учет, искать или фильтровать базу данных, отслеживать запасы, контролировать запасы и проводить аудит запасов [16].

Из блока Production было выделено четыре компонента: TMFC008 Service Inventory, TMFC010 Resource Catalog Management, TMFC012 Resource Inventory, TMFC014 Location Management.

TMFC008 Service Inventory отвечает за учет услуг, которые связаны с элементами Product Inventory. Компонент Service Inventory хранит и поддерживает информацию о различных типах услуг и отслеживает жизненный цикл услуги от момента создания до ее вывода [6].

TMFC010 Resource Catalog Management отвечает за организацию сбора спецификаций ресурсов, которые идентифицируют и определяют все требования к ресурсу. Компонент управления каталогом ресурсов позволяет управлять перечнем ресурсов и настраивать их отображение [4].

TMFC012 Resource Inventory отвечает за учет ресурсов (всех вводимых ресурсов, включая запасы, детали, активы и производственные компоненты), которые принадлежат или удерживаются для распределения или использования организацией. Компонент Resource Inventory отвечает за управление физическими ресурсами, такими как сетевое оборудование, линии связи, серверы и другие аппаратные компоненты, необходимые для предоставления услуг связи. Он также отслеживает и регистрирует все ресурсы, доступные в сети, содержит информацию о физических характеристиках ресурсов. Resource Inventory предоставляет функциональность для резервирования и выделения ресурсов под конкретные услуги или клиентские запросы. Он обеспечивает контроль доступа к ресурсам и управление их использованием [5].

TMFC014 Location Management – компонент, который позволяет легко ссылаться на географические места, важные для других объектов. Его также можно использовать для проверки географических данных, чтобы убедиться, что они соответствуют реальному адресу. Наконец, его можно использовать для поиска адреса. Он обеспечивает централизованное хранение и доступ к данным о расположении сетевого оборудования, линий связи, центров обработки данных и других элементов инфраструктуры оператора связи. Компонент позволяет регистрировать и идентифицировать местоположения физических ресурсов и услуг [3].

После выбора ODA-компонентов для системы необходимо также реализовать интеграцию компонентов в систему – выбрать соответствующие API, определить, как компоненты будут взаимодействовать между собой.

Взаимодействие ODA-компонентов в системе NRI

Если представить описание взаимодействия компонентов на диаграмме последовательности (рис. 23), то:

- Resource Catalog Management (Управление каталогом ресурсов) отвечает за управление информацией о доступных имеющихся ресурсах в системе технического учета, содержит информацию о различных типах ресурсов, их характеристиках, свойствах и доступности. Данный компонент предоставляет API для запроса и обновления информации о ресурсах;

- Resource Inventory (Учет ресурсов) отвечает за учет имеющихся ресурсов, включая запасы и управление физическими ресурсами, такими как сетевое оборудование, серверы, кабели и другие элементы инфраструктуры. Содержит информацию о расположении ресурсов, их статусе, состоянии и связях с другими ресурсами. Компонент Resource Inventory взаимодействует с Resource Catalog Management для получения актуальной информации о ресурсах и их свойствах;

- Location Management (Управление местоположением) отвечает за управление информацией о географическом положении ресурсов, содержит информацию о физических местоположениях, например, здания, помещения, комнаты. Компонент Location Management связывается с Resource Inventory для связывания ресурсов с соответствующими местоположениями и обновления информации о их местоположении при необходимости.

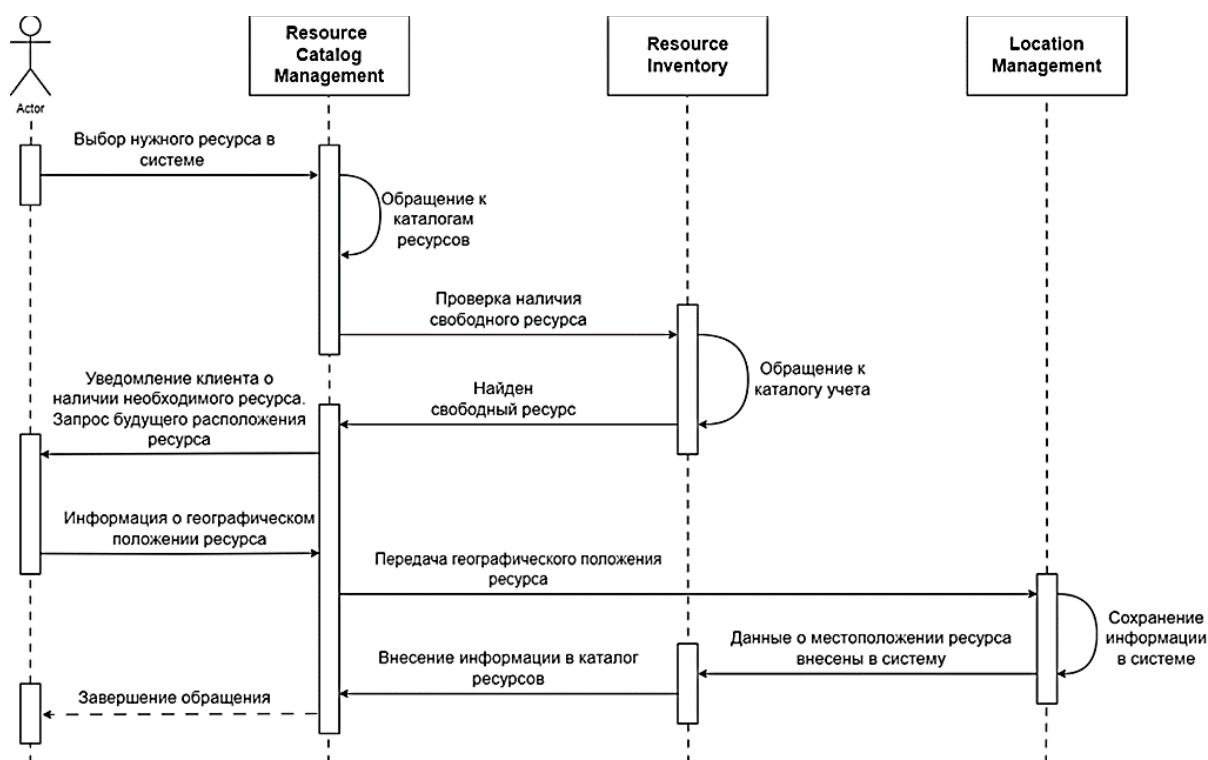


Рис. 23. Диаграмма последовательности взаимодействия компонентов для системы технического учета

Таким образом, компоненты Resource Catalog Management, Resource Inventory и Location Management взаимодействуют для обеспечения учета и управления ресурсами в системе технического учета. Resource Catalog Management предоставляет информацию о доступных ресурсах, Resource Inventory отслеживает физические ресурсы и их свойства, а Location Management управляет информацией о местоположении ресурсов.

Отобранные для системы ODA-компоненты необходимо интегрировать. Для этого используем предлагаемые TM Forum прикладные программные интерфейсы (API), регламентируемые выделенными компонентами. Это значит, что каждый компонент уже имеет набор стандартно предлагаемых API. Каких именно – описано в документации к каждому компоненту.

Пример зависимых и экспортируемых существующих API для компонента Resource Inventory представлен на рис. 24 и в табл. 3 и 4.

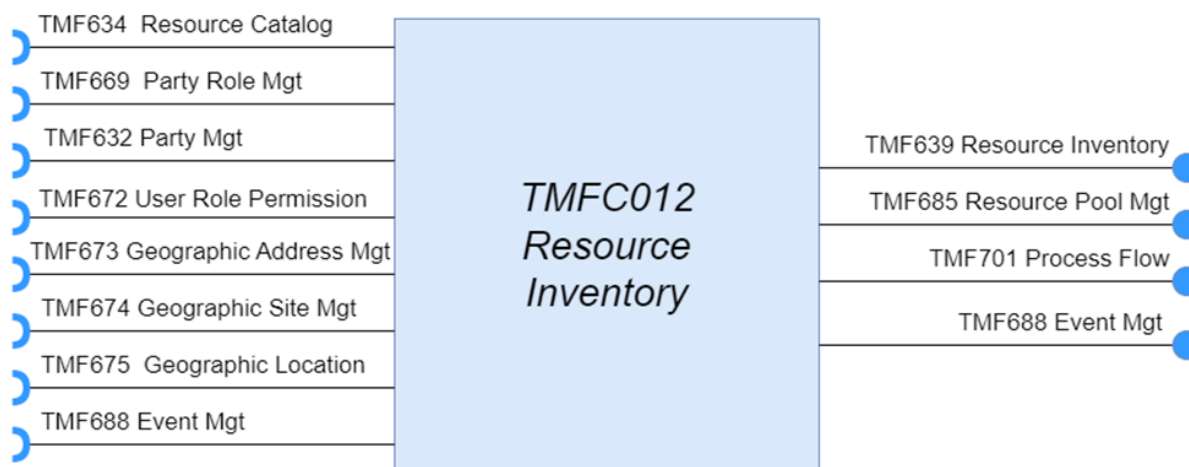


Рис. 24. Компонент Resource Inventory с API

В табл. 3 и 4 представлены API к ODA-компоненту Resource Inventory.

Таблица 3

Зависимые API для компонента Resource Inventory

TM Forum Open APIs	Краткое описание
TMF634 Resource Catalog Mgt	Спецификация REST API управления каталогом ресурсов позволяет управлять всем жизненным циклом элементов каталога ресурсов и консультироваться с элементами каталога ресурсов во время процессов
TMF669 Party Role Mgt	Механизм для общих ролей, включающий такие операции, как создание, обновление, поиск, удаление
TMF632 Party Mgt	Предоставляет механизм для управления участниками, такой как создание, обновление, извлечение, удаление

TM Forum Open APIs	Краткое описание
TMF672 User Role Permission	Роль пользователя определяется как объект, который определяет набор привилегий, охватывающих различные функции или управляемые активы. Когда пользователю назначается определенная роль, ему фактически присваиваются все привилегии, определенные для этого типа роли, для этого пользователя создаются соответствующие разрешения
TMF 673 Geographic Address Mgt	Предоставляет стандартизированный клиентский интерфейс для системы управления адресами. Это позволяет искать адреса по всему миру. Его также можно использовать для проверки данных адреса, чтобы убедиться, что они соответствуют реальному географическому положению
TMF 674 Geographic Site Mgt	Отвечает за управление географическими сайтами. Сайт может представлять собой физическое место, такое как здание, помещение или зона, связанное с поставщиком цифровых услуг
TMF 675 Geographic Location	Позволяет организациям управлять информацией о географических местоположениях, таких как географические координаты, границы территорий, региональные зоны и другие атрибуты, связанные с местоположением
TMF688 Event Mgt	API управления событиями предоставляет стандартизированный клиентский интерфейс к системе управления корпоративными событиями для создания, управления и получения событий, связанных с обслуживанием, для управления рабочими процессами автоматизации, уведомления других поставщиков услуг о перебоях в обслуживании и нарушениях SLA, запуска создания заявок на устранение неполадок и включения более сложных сценариев взаимодействия между системами управления. API управления событиями также можно использовать для передачи событий бизнес-уровня в поддержку других процессов

Таблица 4

Экпортируемые API для компонента Resource Inventory

TM Forum Open APIs	Краткое описание
TMF639 Resource Inventory	Целью этого API является предоставление стандартизированного механизма для запросов и управления учета ресурсов
TMF685 Resource Pool Mgt	Предоставляет функциональность для эффективного использования и распределения ресурсов, таких как люди, оборудование, инфраструктура и другие активы
TMF701 Process Flow	Позволяет управлять бизнес-процессами. Он предоставляет всю необходимую информацию для решения бизнес-задачи, требующей ручного действия
TMF688 Event Mgt	Аналогично

В табл. 5 и 6 представлены API к ODA-компоненту Resource Catalog Management.

Таблица 5

Зависимые API для компонента Resource Catalog Management

TM Forum Open APIs	Краткое описание
TMF634 Resource Catalog Mgt	Спецификация REST API управления каталогом ресурсов позволяет управлять всем жизненным циклом элементов каталога ресурсов и консультироваться с элементами каталога ресурсов во время нескольких процессов
TMF669 Party Role Mgt	Стандартизированный механизм для общих партийных ролей, включающий такие операции, как создание, обновление, поиск, удаление и уведомление о событиях
TMF632 Party Mgt	Стандартизированный механизм для управления участниками, такой как создание, обновление, извлечение, удаление и уведомление о событиях
TMF662 Entity Catalog Mgt	Каталог сущностей предназначен для предоставления потребителям любого объекта SID через каталог с его спецификацией и политикой, обеспечивающими управление его содержимым
TMF672 User Role Permission	Роль пользователя определяется как объект, который определяет набор привилегий, охватывающих различные функции или управляемые активы. Когда пользователю назначается определенная роль, ему фактически присваиваются все привилегии, определенные для этого типа роли, и для этого пользователя создаются соответствующие разрешения
TMF688 Event Mgt	API управления событиями предоставляет стандартизированный клиентский интерфейс к системе управления корпоративными событиями для создания, управления и получения событий, связанных с обслуживанием, для (индикативного) управления рабочими процессами автоматизации, уведомления других поставщиков услуг о перебоях в обслуживании и нарушениях SLA, запуска создания заявок на устранение неполадок и включения более сложных сценариев взаимодействия между системами управления. API управления событиями также можно использовать для передачи событий бизнес-уровня в поддержку других процессов

Таблица 6

Экспортируемые API для компонента Resource Catalog Management

TM Forum Open APIs	Краткое описание
TMF634 Resource Catalog Mgt	Спецификация REST API управления каталогом ресурсов позволяет управлять всем жизненным циклом элементов каталога ресурсов и консультироваться с элементами каталога ресурсов во время нескольких процессов, таких как процесс заказа

TM Forum Open APIs	Краткое описание
TMF701 Process Flow	Позволяет управлять бизнес-процессами. Он предоставляет всю необходимую информацию для решения бизнес-задачи, требующей ручного действия
TMF688 Event Mgt	Аналогично

В табл. 7 представлены API к ODA-компоненту Location Management.

Таблица 7

Зависимые API для компонента Location Management

TM Forum Open APIs	Краткое описание
TMF632 Party Mgt	Предоставляет стандартизированный механизм для управления участниками, такой как создание, обновление, извлечение, удаление и уведомление о событиях
TMF 674 Geographic Site Mgt	Отвечает за управление географическими сайтами. Сайт может представлять собой физическое место, такое как здание, помещение или зона, связанное с поставщиком цифровых услуг
TMF688 Event Mgt	API управления событиями предоставляет стандартизированный клиентский интерфейс к системе управления корпоративными событиями для создания, управления и получения событий, связанных с обслуживанием, для (индикативного) управления рабочими процессами автоматизации, уведомления других поставщиков услуг о перебоях в обслуживании и нарушениях SLA, запуска создания заявок на устранение неполадок и включения более сложных сценариев взаимодействия между системами управления. API управления событиями также можно использовать для передачи событий бизнес-уровня в поддержку других процессов
TMF 673 Geographic Address Mgt	Предоставляет стандартизированный клиентский интерфейс для системы управления адресами. Это позволяет искать адреса по всему миру. Его также можно использовать для проверки данных географического адреса, чтобы убедиться, что они соответствуют реальному географическому адресу
TMF 674 Geographic Site Mgt	Аналогично
TMF 675 Geographic Location	Позволяет организациям управлять информацией о географических местоположениях, таких как географические координаты, границы территорий, региональные зоны и другие атрибуты, связанные с местоположением
TMF701 Process Flow	Позволяет управлять бизнес-процессами. Он предоставляет всю необходимую информацию для решения бизнес-задачи, требующей ручного действия
TMF688 Event Mgt	Аналогично

На рис. 25 представлена архитектура фрагмента новой системы на основе ODA-компонентов.

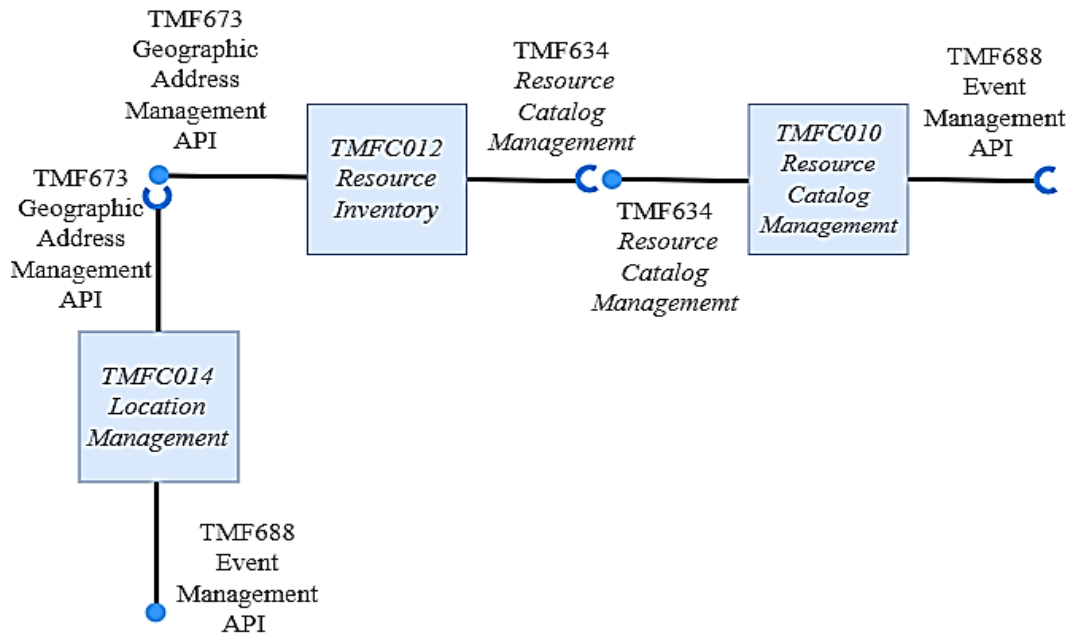


Рис. 25. Архитектура фрагмента системы на основе ODA-компонентов

Чтобы программные блоки могли взаимодействовать друг с другом, необходимо сопоставить зависимые API одного компонента с экспортируемыми API другого (рис. 26).

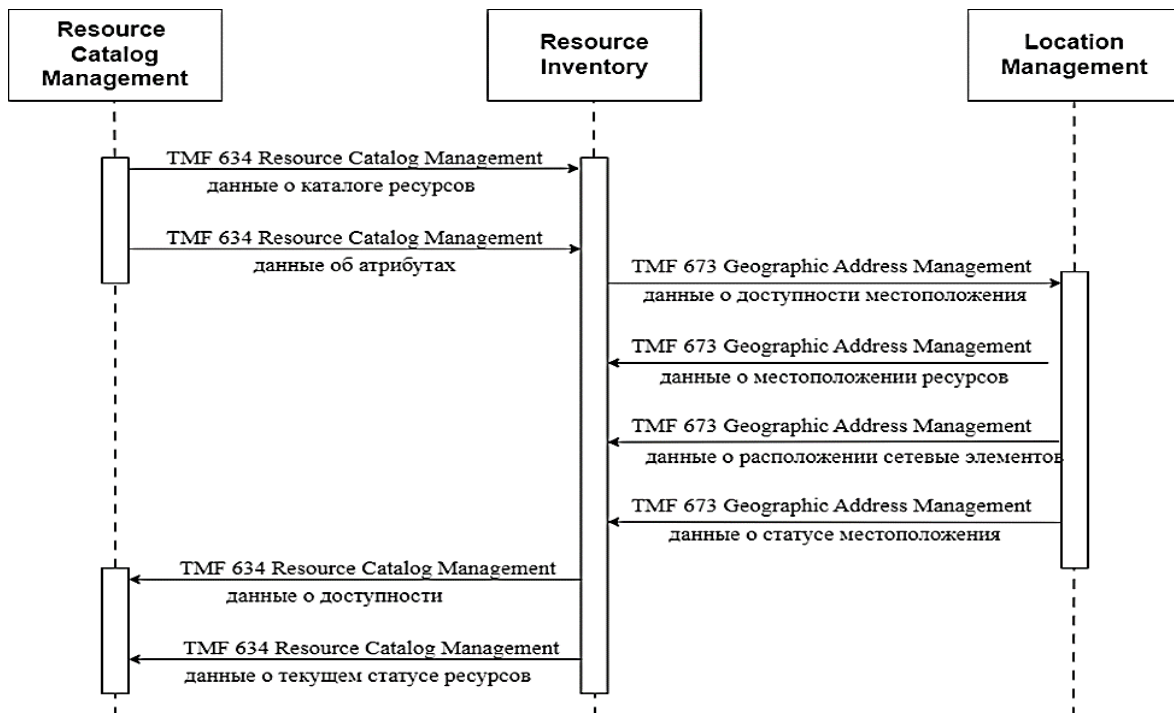


Рис. 26. Диаграмма последовательности передачи данных по API между компонентами

Компонент Resource Catalog Management взаимодействует с компонентом Resource Inventory для получения информации о доступных ресурсах с помощью API 634. При взаимодействии между этими компонентами передается следующая информация:

- о каталоге ресурсов, который содержит описание всех доступных ресурсов в системе. Это включает различные типы сетевых узлов, устройств, услуг и другие ресурсы, которые могут быть предоставлены поставщиком цифровых услуг;

- атрибутах. Атрибутами могут быть технические характеристики и параметры настроек;

- доступности. Это позволяет определить, какие ресурсы доступны и могут быть предоставлены клиентам в определенное время;

- текущем статусе ресурсов. Это включает информацию о занятости ресурсов, их состоянии и доступности для использования.

Взаимодействие между компонентами Resource Catalog Management и Resource Inventory позволяет динамически получать и обновлять информацию о ресурсах, чтобы предоставлять актуальные данные клиентам и управлять ресурсами эффективно.

Компонент Resource Inventory взаимодействует с компонентом Location Management для получения информации о местоположении ресурсов с помощью API 673. При взаимодействии между этими компонентами передается следующая информация:

- о местоположении ресурсов. Это включает координаты, адреса и другие данные, определяющие физическое расположение ресурсов;

- расположении сетевых элементов, таких как узлы связи, оборудование и инфраструктура сети. Это помогает определить, где находятся сетевые ресурсы и как они связаны между собой;

- доступности местоположений. Это позволяет определить, какие местоположения доступны для размещения ресурсов и предоставления услуг;

- статусе местоположений. Это включает информацию о доступности, загруженности и других параметрах, определяющих текущее состояние местоположений.

Взаимодействие между компонентами Resource Inventory и Location Management позволяет Resource Inventory точно знать, где расположены ресурсы и какие местоположения доступны для использования. Это помогает оптимизировать управление ресурсами, планировать размещение и обеспечивать эффективное использование инфраструктуры связи.

3.2. Система управления рабочими ресурсами «АРГУС-WFM»

Управление рабочей силой (Workforce Management, WFM) является одним из неотъемлемых направлений деятельности почти любого оператора связи. Бизнес-процессы этого направления нацелены на реализацию поддержки операционной деятельности как выездных сотрудников (курьеры, специалисты по установке оборудования, инженеры аварийных бригад), так и специалистов на местах (внутренние подразделения ИТ, например) [7] (рис. 27).

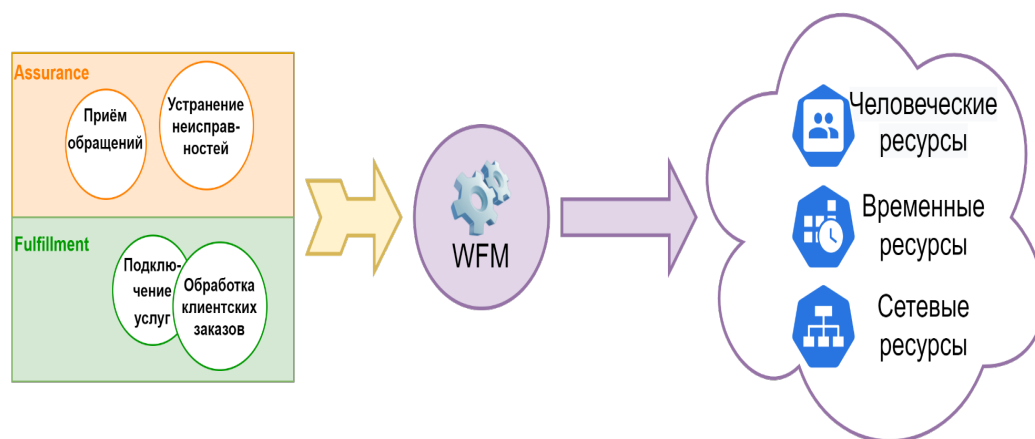


Рис. 27. Типовые бизнес-процессы, автоматизируемые системой WFM

Системы класса WFM фокусируется в первую очередь на автоматизации управления выездными службами, которые занимаются инсталляциями, устранением повреждений, активными продажами и т. п. WFM позволяет оператору автоматизировать процессы распределения и выполнения выездных задач между работниками, а также упрощает управление этими процессами.

Функциональные возможности WFM

Workforce Management начала разрабатываться на основе TM Forum Framework, когда еще не существовало ODA. Функциональность системы WFM может быть традиционно представлена группировками бизнес-функций по Business Process Framework (eTOM) (рис. 28):

– Workforce Management Reporting (Отчетность об управлении персоналом) – позволяет получать детальные отчеты по различным аспектам управления персоналом, таким как использование ресурсов, продуктивность и оценка эффективности работы;

– Workforce Schedule Management (Управление рабочими расписаниями) – помогает создавать гибкие графики работы, учитывая не только требования компании, но и пожелания сотрудников. Также включает возможности для оптимизации расписаний и управления отсутствиями сотрудников;

– Work Order Analysis (Анализ заказов на работы) – позволяет отслеживать и анализировать выполнение работ, а также эффективность рабочих процессов и затраты на выполнение задач;

– Work Order Assignments & Dispatch (Назначение и распределение заказов на работы) – обеспечивает автоматическую отправку заказов на работы нужным сотрудникам, учитывая их навыки, опыт и доступность;

– Work Order Tracking Management (Управление отслеживанием заказов на работы) – помогает контролировать процесс выполнения заказов на работы, включая отслеживание статусов и своевременное информирование о ходе выполнения работ;

– Workforce Configuration and Setup (Настройка и конфигурирование системы управления персоналом) – обеспечивает настройку системы управления персоналом в соответствии с требованиями компании, включая настройку прав доступа, настройку параметров графиков работы и других настроек.

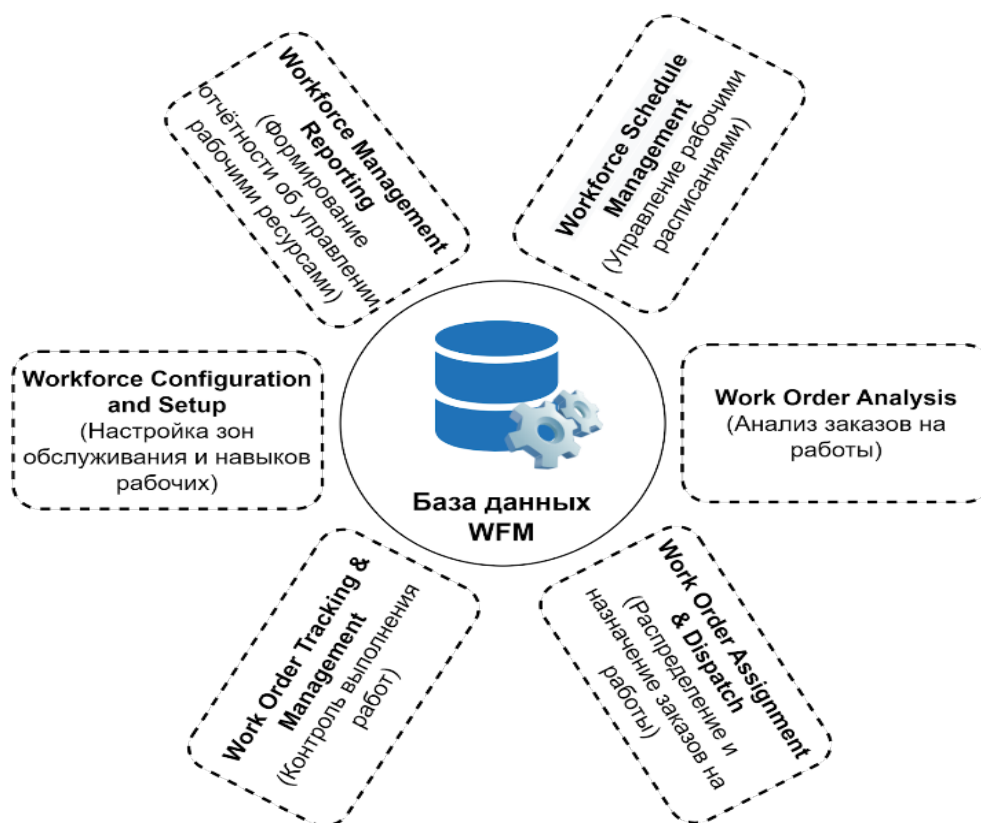


Рис. 28. Возможности системы Workforce Management

Система WFM в автоматическом режиме создает расписания выездных специалистов, оптимизирует его на основе различных критериев.

Система позволяет учитывать специализацию участка и специфику работ по наряду, а также настраивать длительность времени выполнения одного наряда (временную сетку) на участке.

Система включает «Справочник работ», где фиксируется длительность каждой работы по нарядам. При формировании наряда система автоматически учитывает суммарное время выполнения всех работ у клиента, что помогает руководителю участка планировать расписание и время работы у клиента более эффективно.

При адаптации системы Workforce Management к открытой цифровой архитектуре необходимо обеспечить реализацию этих функциональных возможностей.

На самом деле, в данном примере функциональность системы будет несколько расширена за счет добавления возможностей взаимодействия с клиентом напрямую. В «стандартном» же варианте – система WFM не должна заниматься фиксацией обращений клиентов и автоматизацией сопровождения заказов. Здесь это сделано намеренно для того, чтобы более подробно показать особенности работы с инструментами ODA.

Порядок использования инструментов ODA

Первоначально была выделена функциональность системы WFM, которая послужит основой для дальнейшей разработки.

Для определения границ системы на функциональной архитектуре, следует выбрать группировки бизнес-функций из блоков Functional Architecture. Эти группировки позволят определить бизнес-задачи, выполняемые внутри предприятия, и определить роль ODA-компонента в бизнес-процессе.

Затем, на основе Functional Framework, следует определить конкретные функциональные возможности системы. Functional Framework предоставляет каталог функций программного обеспечения, которые могут быть использованы для реализации системы WFM на основе ODA.

Таким образом, процесс построения системы WFM на основе ODA включает определение бизнес-требований, определение границ системы бизнес-функций по Functional Architecture, определение функциональности на основе Functional Framework и, наконец, разбиение системы на ODA-компоненты. В результате будет сформирована система, состоящая из множества взаимосвязанных по API программных компонентов, способных эффективно поддерживать необходимые бизнес-процессы предприятия.

Выделение группировок бизнес-функций для системы WFM no Functional Architecture

Все операционные бизнес-функции процессы сконцентрированы в блоках Party Management, Core Commerce Management и Production.

Бизнес-функции, которые могут быть использованы в формировании бизнес-процессов WFM, сконцентрированы в двух блоках: Production и Party Management.

Из блока Production подходят следующие группировки бизнес-функций:

– SM&O Support & Readiness – отвечает за управление инфраструктурой сервисов, обеспечивая наличие необходимой сервисной емкости и готовности поддерживать процессы SM&O (управление, выполнение, обеспечение качества и выставление счетов);

– RM&O Support & Readiness – управление инфраструктурой ресурсов, чтобы гарантировать наличие и готовность соответствующих ресурсов приложений, вычислительных и сетевых ресурсов, необходимых для поддержки процессов Fulfillment, Assurance и Billing, а также для инстанцирования и управления экземплярами ресурсов, и для мониторинга и отчетности по возможностям и затратам отдельных процессов FAB.

В результате выбраны группировки бизнес-функций ODA FA, которые соответствуют бизнес-целям системы WFM. Например, группировка функций SM&O Support & Readiness может быть связана с бизнес-функциями Workforce Schedule Management, Work Order Assignments & Dispatch, Work Order Tracking Management и Workforce Configuration and Setup, так как она относится к управлению задачами, персоналом и обеспечению качества. Группировка функций RM&O Support & Readiness соответствует Workforce Management Reporting и Work Order Analysis, так как она отвечает за мониторинг и отчетность по затратам и возможностям, связанным с процессами WFM. Группировки бизнес-функций из блока Party Management могут быть использованы системой управления рабочей силой в процессах, связанных с клиентскими взаимодействиями, например, согласованием места и времени встречи или формированием рейтинга специалиста на основе обратной связи клиента по выполненным работам.

Бизнес-процесс обработки заявки на обслуживание

В качестве основного примера на рис. 29 представлен бизнес-процесс обработки заявки на обслуживание. Этот процесс занимает основное место в рамках WFM, так как он отвечает за выполнение основной функции системы – обеспечение обслуживания клиентов. Бизнес-процесс включает в себя прием заявок от клиентов, их обработку, назначение исполнителей

и контроль выполнения задачи. Качество обработки заявок напрямую влияет на удовлетворенность клиентов и, соответственно, на репутацию поставщика услуг. Поэтому в рамках WFM уделяется особое внимание оптимизации процесса обработки заявок, чтобы минимизировать время выполнения задач и максимизировать качество обслуживания.

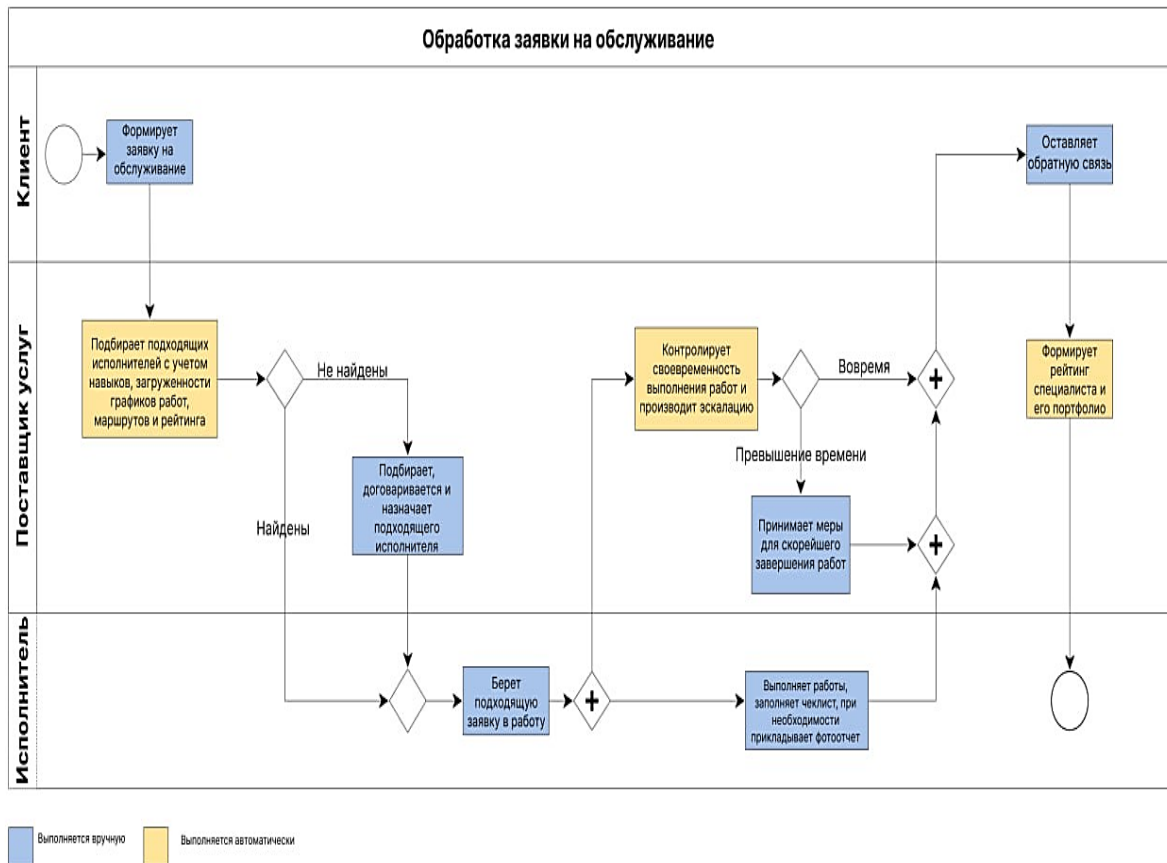


Рис. 29. Бизнес-процесс обработки заявки на обслуживание

Первый шаг данного БП включает в себя формирование заявки клиентом на необходимое обслуживание (в рамках WFM это будет подключение услуги по новому договору, дополнительной услуги или отключение услуги). Затем поставщик услуг автоматически подбирает подходящих исполнителей, учитывая их навыки, загруженность, графики работы, маршруты и рейтинг. Если подходящий исполнитель найден, он принимает заявку. Если нет, поставщик услуг вручную подбирает исполнителя для работы.

Исполнитель выполняет работы и заполняет «чеклист», при необходимости прикладывая фотоотчет. Поставщик услуг контролирует своевременность выполнения работ и производит автоматическую эскалацию в случае необходимости. Если время превышено, поставщик услуг принимает меры для скорейшего завершения работ.

После завершения работ клиент оставляет обратную связь. На основе этой обратной связи поставщик услуг формирует рейтинг специалиста и его портфолио автоматически. В целом, эти сценарии позволяют оптимизировать процесс обработки заявок на обслуживание, сократить время на поиск исполнителей, повысить качество работы и улучшить взаимодействие между клиентами и поставщиками услуг.

Выделение ODA-компонентов для системы WFM

Для определения необходимых компонентов требуется изучить функциональную структуру (Functional Framework Functions) в спецификации компонента, где описаны поддерживаемые функции компонента.

В качестве примера будет продемонстрирована выборка функций из спецификации компонента Party Interaction Management (табл. 8). Для остальных компонентов системы подобную выборку функций предлагается проделать самостоятельно.

Таблица 8

Функции компонента Party Interaction Management

ID функции	Наименование функции	Описание функции	Функции поддомена уровень-1	Функции поддомена уровень-2
163	Contact Queuing (Очередь контактов)	Обеспечивает возможность постановки контакта в очередь до тех пор, пока подходящий сотрудник не станет доступным для работы с контактом	Welcome and Interaction	Customer Interaction Management
168	Voice Channel Contact Routing (Маршрутизация голосовых контактов)	Обеспечивает возможность клиенту общаться с представителем службы поддержки по телефону, включая запрос информации у клиента (например, с помощью IVR) о характере его запроса и маршрутизацию контакта к наилучшему доступному сотруднику с необходимыми навыками для обработки контакта	Welcome and Interaction	Customer Interaction Management

ID функции	Наименование функции	Описание функции	Функции поддомена уровень-1	Функции поддомена уровень-2
196	Customer Interaction Logging (Журналирование взаимодействия с клиентом)	Предоставляет сбор и хранение всех контактов с клиентом через все каналы, включая историю взаимодействия, историю заказов, историю обращений по проблемам, историю взаимодействия по вопросам оплаты, управления делами и т. д.	Welcome and Interaction	Customer Interaction Management
239	Recommendation to Customer Notification (Уведомление о рекомендации клиенту)	Обеспечивает необходимые инструменты для связи с клиентом через предпочитаемый канал, такой как SMS, электронная почта или социальные медиа. Для контактов, поступающих от клиента, самообслуживания или контактного центра, можно получить рекомендацию и гибкий поток действий для выполнения соответствующего обслуживания клиента (например, решение вопросов по кредиту, инициирование спора по счету через самообслуживание)	Welcome and Interaction	Customer Interaction Management

Так как одна из основных функций системы управления человеческим ресурсом – это назначение и распределение заказов на работы, то в системе должны быть реализованы возможности для установления связи с клиентом, записи в историю его обращения для передачи в соответствующие участки или подразделения. Помимо этого, должны быть реализованы функции, которые бы смогли автоматизировать работу контакт-центра в части маршрутизации звонков клиентов к сотрудникам, обладающими необходимыми навыками.

Функции, которые закрывают данные требования, описаны в спецификации компонента Party Interaction Management. Так, например, функция Customer Interaction Logging из табл. 8 предоставляет сбор и хранение контактов клиента и истории заказов и обращений, закрывая собой одно из требований к функциональности системы.

На рис. 30 приведен ODA-компонент Party Interaction Management с набором необходимыми для взаимодействия с другими компонентами API.

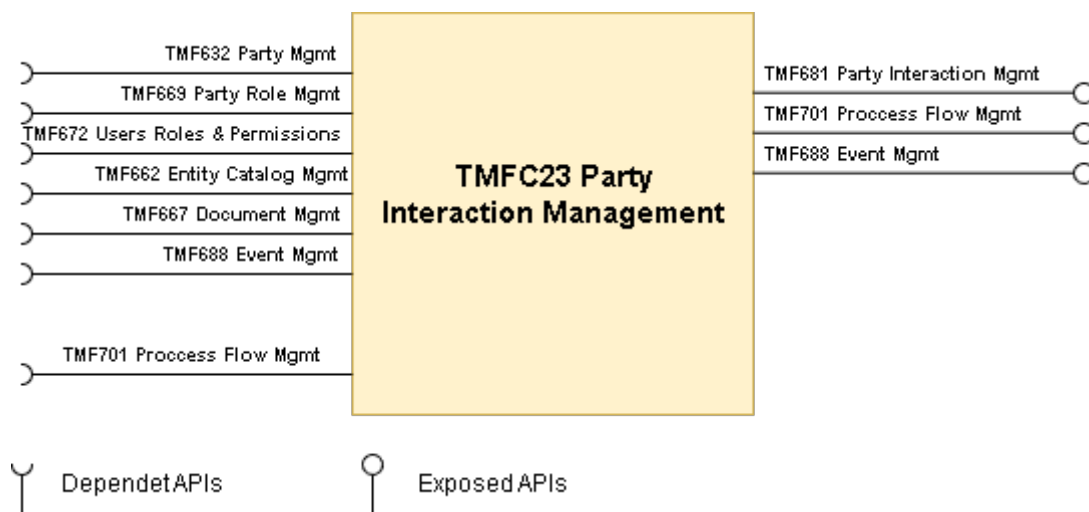


Рис. 30. ODA-компонент Party Interaction Management и его API

Список и описания этих API приведены в табл. 9.

Таблица 9

Список API компонента Party Interaction Management

TM Forum Open APIs	Краткое описание
Dependent APIs	
TMF 632 Party Mgmt	Предоставляет стандартизированный механизм для управления участниками, такой как создание, обновление, извлечение, удаление и уведомление о событиях. Участником может быть физическое лицо или организация, имеющая какие-либо отношения с предприятием
TMF 699 Party Role Mgt	API управления продажами предоставляет интерфейсы для потенциальных клиентов, возможностей продаж, предложений по продажам и других возможностей управления для поддержки деятельности по продажам для построения отношений с потенциальным клиентом, которым может быть лицо или организация, заинтересованные в товарах и/или услугах, и, возможно, ставшие фактическими клиентами с одним или несколькими подписками

TM Forum Open APIs	Краткое описание
Dependent APIs	
TMF 672 Users Roles & Permission	Роль пользователя определяется как объект, который определяет набор привилегий, охватывающих различные функции и/или управляемые активы. Когда пользователю назначается определенная роль, ему фактически присваиваются все привилегии, определенные для этого типа роли, и для этого пользователя создаются соответствующие разрешения
TMF 662 Entity Catalog Mgmt	Каталог сущностей предназначен для предоставления потребителям любого объекта SID через каталог с его спецификацией и политикой, обеспечивающими управление его содержимым. API обеспечивает управление спецификациями объектов и их ассоциациями посредством операций CRUD
TMF 667 Document Mgmt	Предоставляет операции для синхронизации документов и версий документов в разных системах. Он также предоставляет операции для загрузки документов пользователями, а также для просмотра документов онлайн
Exposed APIs	
TMF 701 Process Flow Management	API ProcessFlow позволяет управлять бизнес-процессами. Он предоставляет всю необходимую информацию для решения бизнес-задачи, требующей ручного действия
TMF 683 Party Interaction Mgmt	Взаимодействие с пользователем собирает информацию о прошлых взаимодействиях, чтобы повторно использовать ее в будущем. Это позволяет агентам лучше обслуживать пользователей, зная этапы, которые они прошли
TMF 688 Event	API управления событиями предоставляет стандартизированный клиентский интерфейс к системе управления корпоративными событиями для создания, управления и получения событий, связанных с обслуживанием, для (индикативного) управления рабочими процессами автоматизации, уведомления других поставщиков услуг о перебоях в обслуживании и нарушениях SLA, запуска создания заявок на устранение неполадок и включения более сложных сценариев взаимодействия между системами управления. API управления событиями также можно использовать для передачи событий бизнес-уровня в поддержку других процессов

В табл. 10 представлены компоненты, выбранные для системы WFM на основе выделенных функций, и их описание.

Таблица 10

ODA-компоненты для системы WFM

Название компонента	Краткое описание функциональности
Intelligence Management	
Recommendation Management	Управление рекомендациями обеспечивает функциональность для принятия решений, например, обработку клиентов и действий для каждой конкретной ситуации на основе наблюдения за тем, что клиент испытывает в данный момент
Party Management Block	
TMFC028 Party Management	Отвечает за управление данными о клиентах, их идентификацию и аутентификацию. Обеспечивает создание и редактирование профилей клиентов
Party Problem Management	Управление проблемами, сообщенными клиентами, разрешение этих проблем, предоставление информации о статусе проблемы
Production Block	
Appointment management	Обеспечивает стандартные механизмы организации встреч с учетом свободных временных окон (слотов). В качестве параметров используется дата, время, место встречи
TMFC014 Location Management	Отвечает за управление информацией о местоположениях, например адреса, принадлежность к различным зонам и т. д.
Decoupling And Integration	
TMFC019 Event Management	Компонент управления сообщениями, который отвечает за обработку событий, происходящих из компонентов. Обработка записей о деятельности включает моделирование сообщений о событиях, проверку синтаксиса и форматирование сообщений о событиях и отправку/получение сообщений между компонентами

В результате получена система, разбитая на программные компоненты. Необходимые ODA-компоненты (табл. 9), которые соответствуют требованиям системы, были найдены в блоках Production и Party Management. Для обеспечения взаимодействия между программными блоками были включены все компоненты из блоков Engagement Management и Decoupling And Integration.

Взаимодействие ODA-компонентов системы WFM

Теперь необходимо встроить данные ODA-компоненты в бизнес-процесс. Последовательность действий компонентов в бизнес-процессе представлена следующим образом.

Когда клиент обращается к оператору связи для подключения услуги, первым компонентом, с которым он взаимодействует, является Party Interaction Management. Компонент определяет, является ли клиент новым или уже зарегистрированным в системе, чтобы предложить соответствующие действия. Если клиент уже зарегистрирован, информация передается в компонент Digital Identity Management для аутентификации и определения роли клиента.

Digital Identity Management авторизует клиента и передает информацию об идентификации в компонент Party Management. Основная задача компонента Party Management заключается в управлении профилем клиента. Если клиент сообщает о проблеме, запрос автоматически связывается с его профилем в Party Management. Затем компонент Party Problem Management отслеживает этот запрос, обрабатывает его и предоставляет информацию о проблемах и их причинах для подробного описания заявки. Он также помогает в оценке приоритетности обращения и выборе соответствующего исполнителя.

Компоненты Appointment Management и Location Management также участвуют в выборе подходящего исполнителя. Appointment Management использует информацию о доступности исполнителей и временных слотах, а Location Management обрабатывает географические данные клиента для определения маршрута и удаленности адреса. При передаче параметров (дата, время, адрес клиента) от Party Management, Appointment Management назначает исполнителя для выезда к клиенту.

После того, как заявка принята в работу исполнителем, Service Assurance Management контролирует своевременность выполнения работ и предпринимает упреждающие и реактивные действия для своевременного завершения работ, если это необходимо. Результат работы передается в компонент для формирования рейтинга исполнителя. Recommendation Management на основе обратной связи от клиента о качестве выполненных работ формирует рейтинг и портфолио специалиста. Графически (помимо

описания) межкомпонентное взаимодействие может быть представлено в виде диаграммы последовательности (здесь не приводится).

После того, как определена последовательность действий компонентов, необходимо определить интерфейсы взаимодействия для каждого компонента. Выбираются API из предлагаемых на основе описаний (документаций) компонентов.

На рис. 31 изображено взаимодействие ODA-компонентов посредством API. Отображены только компоненты, отвечающие за операционные процессы внутри обсуждаемого бизнес-процесса. Чтобы программные блоки могли взаимодействовать друг с другом, необходимо сопоставить dependent API одного компонента с exposed API другого.

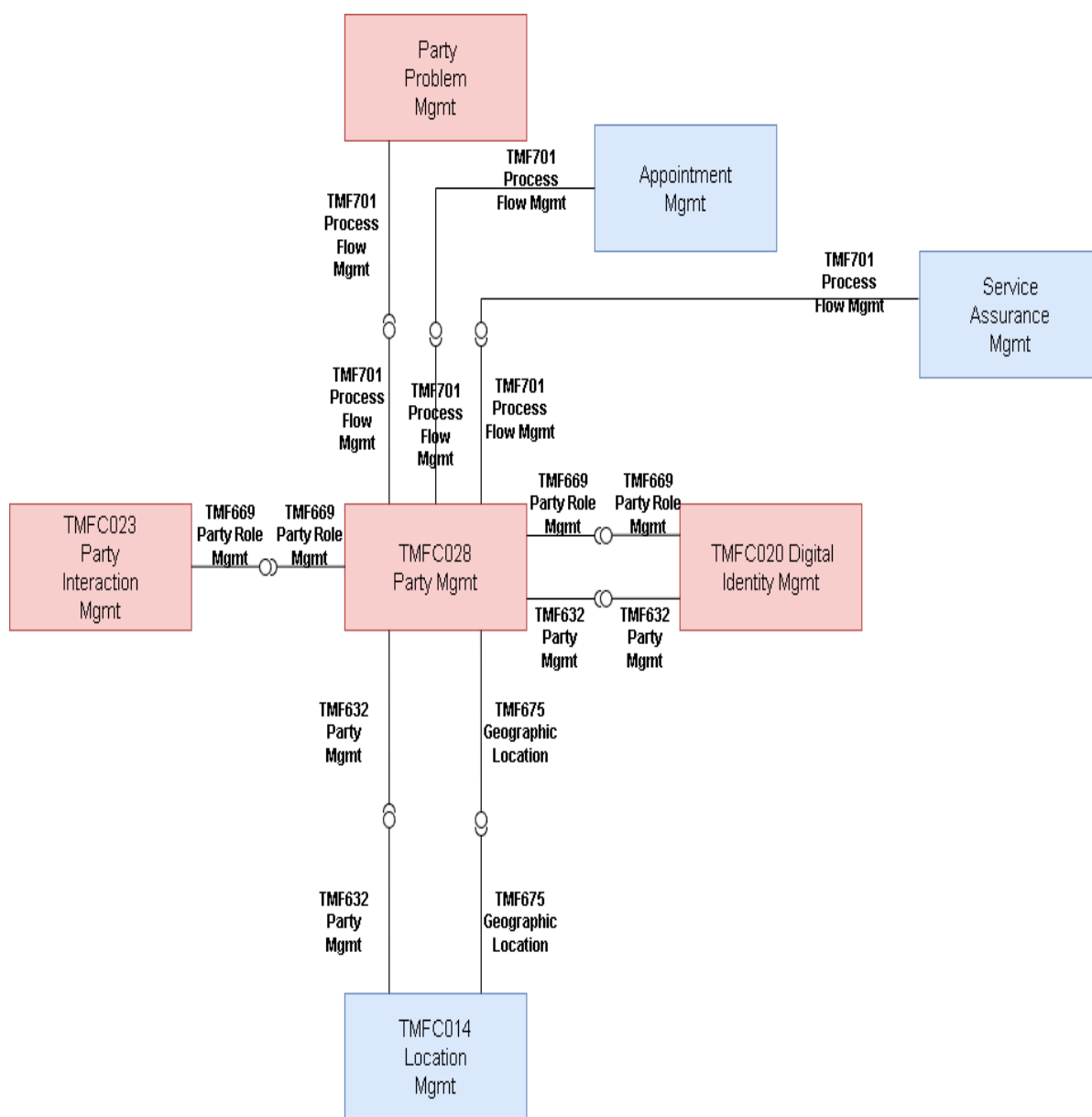


Рис. 31. Взаимодействие выделенных для системы WFM ODA-компонентов

Чтобы получить сведения о партнерской роли, Party Interaction с помощью API TMF669 связывается с Party Mgmt. На основе этих данных определяются доступные действия для клиента. Также Party Mgmt при помощи TMF632 передает сведения с данными клиента, чтобы в ответ получить его географический адрес (по TMF 675).

Если программному блоку требуется передать не какую-либо информацию, а запустить процесс, то это осуществляется при помощи TMF701 Process Flow Mgmt. Например, Party Mgmt сформировал профиль клиента, и требуется привязать обращение. Тогда он при помощи TMF701 запускает процесс, передающийся на дальнейшую обработку Party Problem Mgmt.

Если компонент генерирует какое-либо событие, то оно сначала обрабатывается в Event Mgmt, и затем передается на обработку соответствующему компоненту. Так, например, после завершения наряда исполнителем Service Assurance Mgmt создает событие, уведомляющее о сроках завершения работ сотрудником. Event Mgmt это обрабатывает и через API TMF688 передает компоненту Recommendation Mgmt, который уже составляет рейтинг на сотрудника.

API работает через HTTP-запросы, где GET используется для получения информации, POST – для создания новых ресурсов, PUT – для обновления существующих ресурсов и DELETE – для удаления ресурсов. Все созданные события обрабатываются в компоненте Event Management, после чего передаются необходимым компонентам для дальнейших действий.

Решения для университетов

ODA позволяет разрабатывать системы, автоматизирующие любой бизнес, так как именно компонентный подход вкуче с открытыми или стандартными API дает максимальную гибкость в архитектуре и функциональности. Применяя результаты ODA, покажем практическую сторону разработки системы управленческого учета (Asset Management System, AMS).

Системы AMS предназначены для поддержки бизнес-процессов (БП) управления жизненным циклом (ЖЦ) разного рода материальных ценностей (МЦ) университетов (и не только): объектов сетей связи и инфраструктуры, элементов энергетических сетей, объектов недвижимости, расходных материалов, ИТ-оборудования, ПО и других материальных ценностей (рис. 32).

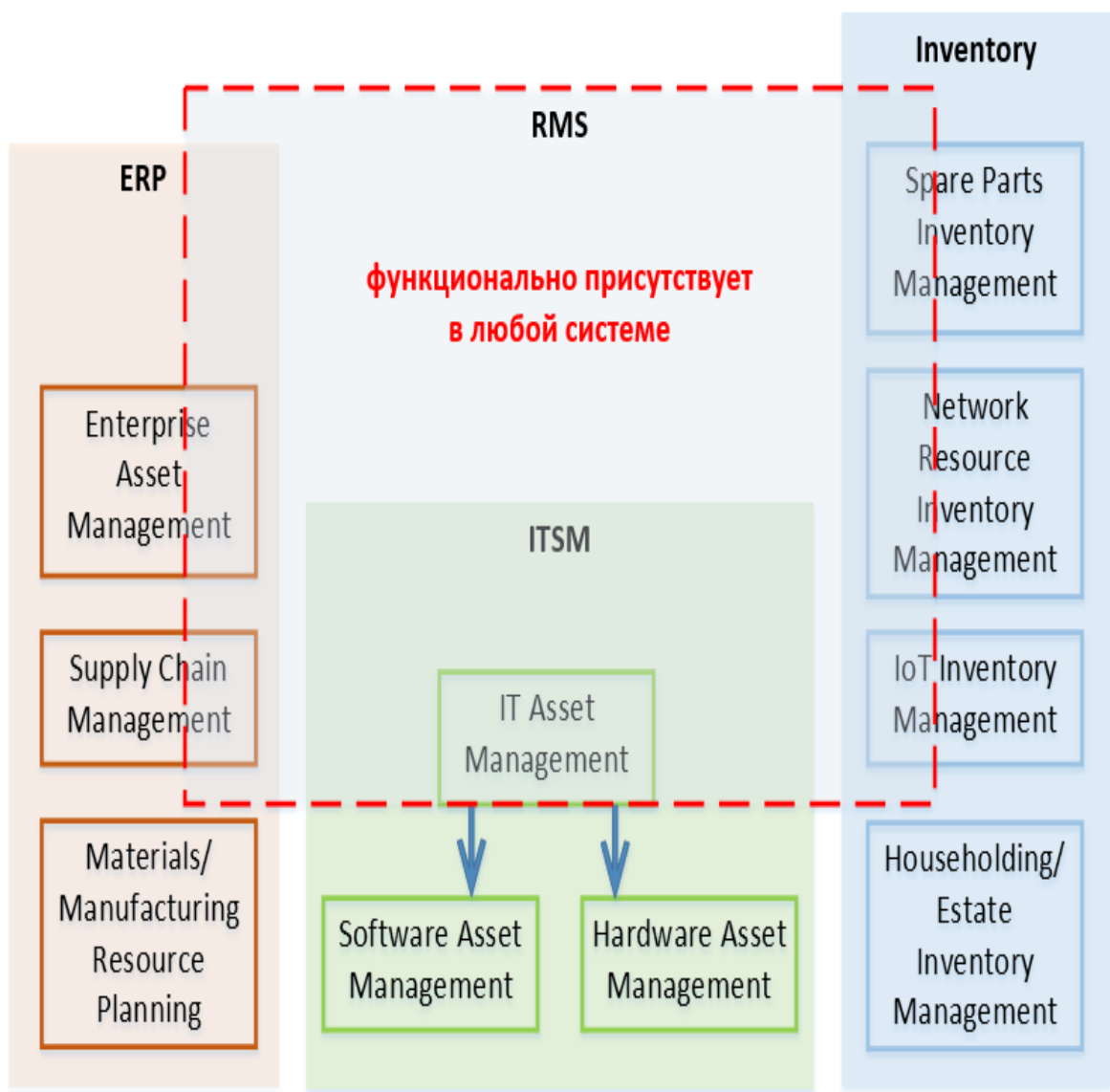


Рис. 32. Иллюстрация функционального присутствия RMS в отраслевых системах управленческого учета

На основе анализа видов материальных ценностей в университетах потребностей выделено функциональное универсальное ядро системы AMS (рис. 33).

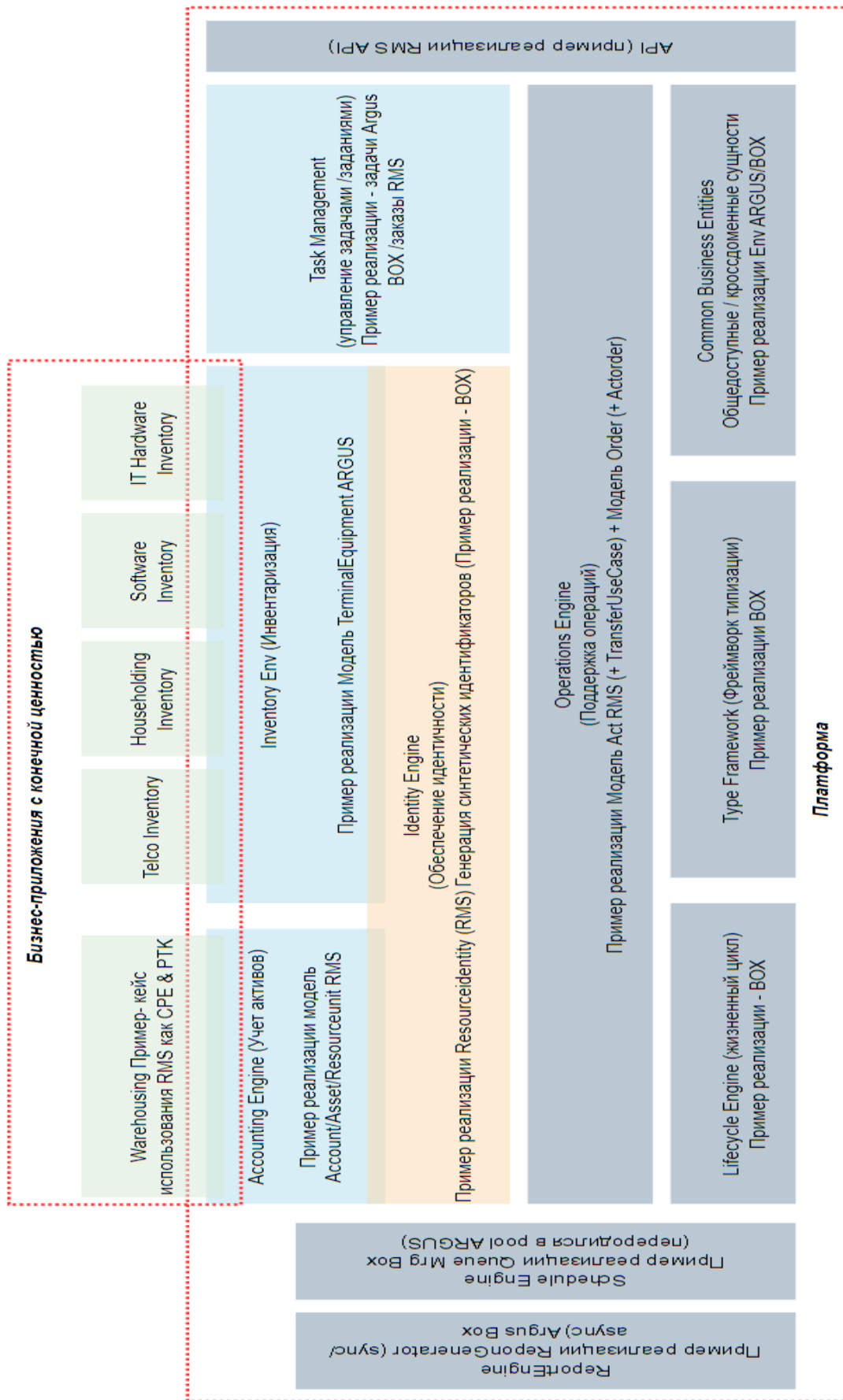


Рис. 33. Универсальное ядро системы управленческого учета

НТЦ АРГУС разработана платформа AMS (рис. 34), позволяющая создавать гибкие проектные решения под самые разные архитектуры предприятий и их бизнес-процессов. За основу взята платформа JMIХ. На ее основе развернута специфическая для решений AMS инфраструктура. В базе инфраструктуры (нижний слой):

- Reporting tools – редактор отчетов;
- UI\UX tools – инструменты для сборки пользовательского интерфейса;
- BPMs – программный движок бизнес-процессов;
- Roles/Permissions – инструменты управления ролями и правами доступа;
- Notifications – управление уведомлениями через различные каналы – электронную почту, sms, push и т. д.;
- Attrib-Value Engine – инструменты для расширения моделей инфраструктуры дополнительными параметрами;
- Integration tools – инструменты интеграции.

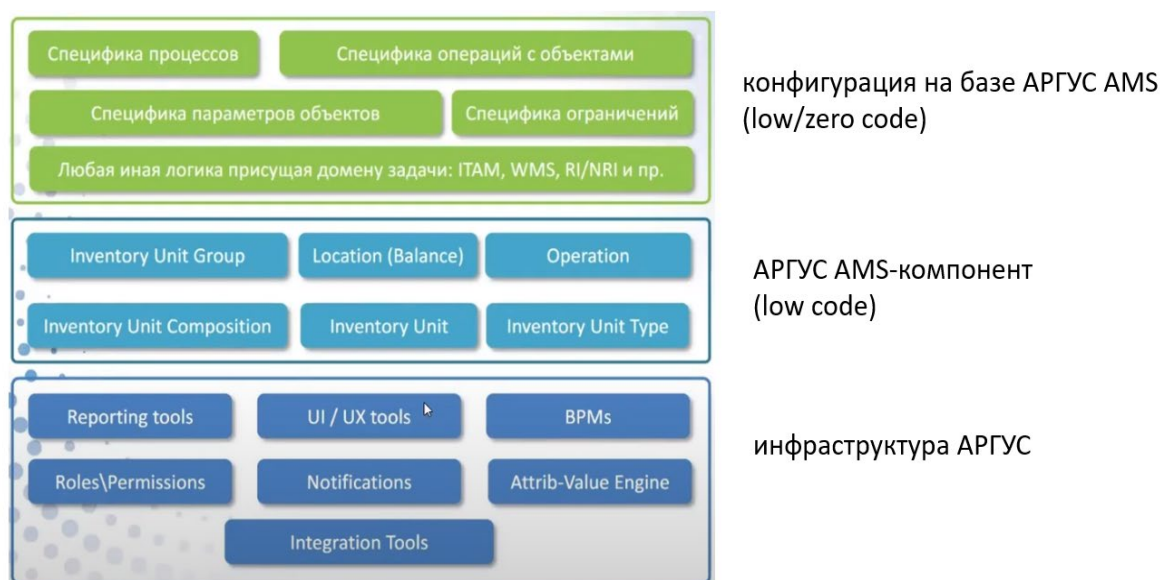


Рис. 34. Платформа «АРГУС AMS»

Система AMS как продукт теперь включает 6 базовых сущностей (средний слой AMS-компонент):

- 1) Inventory Unit – это абстрактное представление любого объекта, который подлежит учету (колеса, оборудование, недвижимость...);
- 2) Location – абстрактное понятие, позволяющее задать систему координат, в рамках которых будет вестись учет Inventory Unit;
- 3) Operation – описывает поведение объектов. Это элемент бизнес-процесса;
- 4) Inventory Unit Group – группа объектов учета;
- 5) Inventory Unit Composition – композиция объектов учета;
- 6) Inventory Unit Type – тип объекта учета.

Выделение таких сущностей позволит реализовывать различные сценарии управленческого учета, придавая этим понятиям различный бизнес-смысл. Эти объекты являются самостоятельными, их можно настраивать независимо.

В слое AMS-компонент реализовано 6 сущностей в режиме low-code¹, что не потребовало и не потребует писать большого количества строк кода. Это позволит в дальнейшем не нести больших затрат на программирование и программистов. Необходимые доработки под конкретный проект и сценарии можно будет реализовывать в режиме low-code.

Верхний слой платформы AMS (рис. 34) содержит сущности, которые настраиваются и взаимосвязываются в режиме zero-code², когда не требуется знание языков программирования.

Итого, разработана платформа для создания приложений, в которой уже заложено поведение, основные свойства и т. д. На базе этой low-code – AMS-платформы можно создавать проектные конфигурации под конкретные бизнес-процессы.

Как раз для создания конкретного приложения и можно использовать студентов и аспирантов СПбГУТ для конфигурирования верхнего слоя платформы (под конкретную задачу), в частности под задачу создания системы управленческого учета для университетов. Предполагается, что под задачи управленческого учета ресурсов университетов РФ будет конкретная конфигурация на базе платформы «АРГУС AMS».

¹ Low-code – это метод разработки в визуальном интерфейсе по принципу drag-and-drop, но с элементами ручного кода. В платформе для Low-code разработки пользователь перемещает блоки с уже готовым кодом и получает продукт с нужным функционалом. Готовые модули в Low-code ускоряют работу с типовыми задачами и избавляют от повторяющихся действий, но для индивидуальных решений, настроек и персонализации придется писать код. Разработка в платформе проходит по готовым шаблонам или свободно. Также поддерживаются интеграции, есть встроенные сервисы. Low-code не исключает классическую разработку и программистов, а ускоряет их работу. Даже ручной коддинг – не всегда проект с нуля. Опытные программисты в потоке типовых заказов иногда берут шаблоны своего же кода для экономии времени.

² Zero-code или No-code – метод создания сайтов и приложений без кода. Nocode разработка подходит для непрофессиональных разработчиков (Citizen Developers) – обычных пользователей, которые не умеют писать код. Такой zerocoder создает продукт по принципу drag-and-drop в визуальном интерфейсе, где кода нет совсем. Zero coding максимально простой, научиться работе с платформой легко – не нужны глубинные знания работы цифровых продуктов. Пользователю достаточно понимать будущий функционал и суметь собрать под него решение из готовых блоков. Zeroкоддинг – минимум затрат на разработку решения для определенной бизнес-задачи: создания лендинга, приложения, автоматизации процессов. Компании не придется нанимать дополнительных разработчиков [21].

ЗАКЛЮЧЕНИЕ

ODA – это новейшая эталонная архитектура, которая предполагает ряд принципиальных изменений в разработке систем автоматизации бизнеса поставщиков цифровых услуг. Основные изменения – это переход на облачные решения, смена монолитных программных продуктов на платформы и взаимодействующие через открытые стандартные API программные компоненты.

Преыдушие разработки – концепции NGOSS/Framework – были ориентированы исключительно на автоматизацию операторов связи. В силу того, что инфокоммуникации зашли глубоко во все сферы человеческой деятельности, теперь потребовалось переработать архитектуру с тем, чтобы она удовлетворяла любой другой сфере, где предоставляются цифровые услуги.

В ODA функции карты eTOM берет на себя функциональная архитектура (FA), а функции карты TAM – структура приложений (FF) [8]. FA – это сопоставление и группировка бизнес-функций предприятия; FF содержит функции программных приложений предприятия. FA и FF связаны, но отличаются друг от друга по назначению аналогично eTOM и TAM.

ODA стремится сделать оператора связи более похожим на поставщика IaaS или PaaS. Единые методы и согласованные API упрощают работу по подключению новых сервисов и служб. Вместо «монолита» OSS/BSS каждый сервис будет представлять собой четко определенный функционально самостоятельный компонент, взаимодействующий с другими компонентами.

TM Forum предлагает перейти от разрозненной OSS/BSS среды к открытому рынку программного обеспечения. ODA показывает пути для перехода от монолитных устаревших программных решений к управлению гибкими облачными решениями, основанными на стандартных программных блоках, которые можно покупать точно так же, как и стандартное оборудование у различных производителей.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. IG1167 ODA Functional Architecture Exploratory Report v6.0.0. : [Электронный ресурс]. – URL: <https://www.tmforum.org/resources/standard/ig1167-oda-functional-architecture-exploratory-report-v6-0-0/>. (Дата обращения: 01.07.2023). – Режим доступа: требуется авторизация.
2. GB1033 Functional Framework : [Электронный ресурс]. – URL: <https://www.tmforum.org/resources/> (Дата обращения: 11.10.2023). – Режим доступа: требуется авторизация.
3. TM Forum Technical Specification «Location Management» : [Электронный ресурс]. – URL: <https://www.tmforum.org/resources/> (Дата обращения: 10.11.2023). – Режим доступа: требуется авторизация.
4. TM Forum Technical Specification «Resource Catalog Management» : [Электронный ресурс]. – URL: <https://www.tmforum.org/resources/> (Дата обращения: 11.10.2023). – Режим доступа: требуется авторизация.
5. TM Forum Technical Specification «Resource Inventory» : [Электронный ресурс]. – URL: <https://www.tmforum.org/resources/> (Дата обращения: 19.12.2023). – Режим доступа: требуется авторизация.
6. TM Forum Technical Specification «Service Inventory» : [Электронный ресурс]. – URL: <https://www.tmforum.org/resources/> (Дата обращения: 11.12.2023). – Режим доступа: требуется авторизация.
7. *Атцик, А. А.* Теория и практика автоматизации бизнес-процессов современного оператора связи : учебное пособие / А. А. Атцик, А. Б. Гольдштейн, А. В. Никитин ; СПбГУТ. – Санкт-Петербург, 2016. – 91 с.
8. *Атцик А. А.* Эксплуатационное управление инфокоммуникациями : учебное пособие / А. А. Атцик, А. Б. Гольдштейн, М. А. Феноменов ; СПбГУТ. – Санкт-Петербург, 2013. – 67 с.
9. *Гольдштейн, А. Б.* Современные подходы к автоматизации бизнес-процессов операторов связи : учебное пособие / А. Б. Гольдштейн, С. В. Кисляков ; СПбГУТ. – Санкт-Петербург, 2020. – 84 с.
10. *Гольдштейн, А. Б.* Методы разработки систем управления сетями пятого поколения : учебное пособие / А. Б. Гольдштейн, С. В. Кисляков, М. А. Феноменов ; СПбГУТ. – Санкт-Петербург, 2021. – 69 с.
11. *Гольдштейн, А. Б.* Программирование систем управления инфокоммуникационными сетями : учебное пособие / А. Б. Гольдштейн, С. В. Кисляков, М. В. Усков ; СПбГУТ. – Санкт-Петербург, 2017. – 74 с.
12. *Самуйлов, К. Е.* Чукарин А. В., Яркина Н. В.. Введение в управление инфокоммуникациями : учебное пособие / К. Е. Самуйлов, Н. В. Серебренникова, А. В. Чукарин, Н. В. Яркина. – Москва : РУДН, 2008. – 87 с.
13. *Самуйлов, К. Е.* Бизнес-процессы и информационные технологии в управлении телекоммуникационными компаниями / К. Е. Самуйлов, А. В. Чукарин, Н. В. Яркина. – Москва : Альпина Паблишер, 2016. – 512 с.
14. *Гольдштейн, А. Б.* Концепция открытой цифровой архитектуры: эволюция или революция? / А. Б. Гольдштейн, С. В. Кисляков // Вестник связи, 2022. – № 6. – С. 22–28.
15. *Кисляков, С. В.* Адаптация систем NRI и WFM к открытой цифровой архитектуре TM FORUM / С. В. Кисляков, Е. А. Лочкарев, Д. И. Сухомлинов ; СПбГУТ // Подготовка профессиональных кадров в магистратуре для цифровой экономики (ПКМ-2022). – Санкт-Петербург, 2022. – С. 755–759.

16. *Кисляков, С. В.* Разработка бизнес-процесса подключения услуги на основе TM FORUM OPEN DIGITAL ARCHITECTURE / С. В. Кисляков, В. В. Майоров // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2022). – Санкт-Петербург: 2022. – С. 150–156.

17. Расширенная карта процессов оператора связи (eТОМ®) Структура бизнес-процессов для отрасли телекоммуникационных услуг // Jet Info. – 2007. – № 7. – С. 3–21.

18. *Райли, Д.* NGOSS: Построение эффективных систем поддержки и эксплуатации сетей для оператора связи / Д. Райли, М. Кринер. – Альпина Бизнес Букс, 2007. – 192 с.

19. *Митра, Р.* Непрерывное развитие API. Правильные решения в изменчивом технологическом ландшафте / Р. Митра, М. Амундсен, М. Меджуи, Э. Уайлд. – 2 изд. – O'Reilly, 2021. – 369 с.

20. Единый комплекс продуктов НТЦ АРГУС : [Электронный ресурс]. – URL: <https://argustelecom.ru/produktyi/platforma-argus.html> . (Дата обращения: 14.04.2023).

21. Что такое Low Code и Zerocode и когда применяются : [Электронный ресурс]. – URL: <https://vc.ru/marketing/374861-что-такое-low-code-i-zero-code-i-kogda-primenyayutsya> (дата обращения: 10.01.2024).

**Гольдштейн Александр Борисович
Кисляков Сергей Викторович
Феноменов Михаил Александрович**

**ОТКРЫТАЯ ЦИФРОВАЯ АРХИТЕКТУРА
ДЛЯ РАЗРАБОТКИ СИСТЕМ УПРАВЛЕНИЯ
ИНФОКОММУНИКАЦИЯМИ**

Учебное пособие

Электронное издание

Редактор *Л. К. Паршина*

Компьютерная верстка *С. Н. Скляровой*

План издания 2024 г., доп. п. 1

Подписано к изданию 14.10.2024

Объем 4,75 уч.-изд. л.

Редакционно-издательский отдел СПбГУТ
193232 СПб., пр. Большевиков, 22

