



***it*SMF**

СООБЩЕСТВО ПРОФЕССИОНАЛОВ ITSM

2018
АЛЬМАНАХ
itSMF России
Избранные статьи

Оглавление

Подходы и практика ITSM

Новая версия ITIL, что будет?	2
<i>Роман Журавлев</i>	
ITSM 101: начало работы с ITIL.....	5
<i>Стьюарт Рэнс</i>	
Стандартизация в цифровую эпоху.	8
<i>Антон Боганов</i>	
Нарушая границы, или Как сервисная модель управления может повысить эффективность бизнеса в целом.....	13
<i>Руслан Смаков</i>	
Лучше один раз увидеть	20
<i>Александр Шпер</i>	
Новые подходы к развитию ИТ-сервисов в «Газпром нефти»	30
<i>Алексей Поперлюков</i>	
ITIL и риски быстрого изменения приложений. Как организовать поддержку в новых условиях?	38
<i>Владимир Аношин</i>	

Подходы Agile и DevOps

«Путешествие дилетантов» на остров DevOps	46
<i>Константин Зимин, Владимир Павлов</i>	
Девять человеческих проблем, отравляющих DevOps.....	56
<i>Синтия Харви</i>	
Мы должны убить DevOps.	59
<i>Майк Даффи</i>	

Вокруг ITSM

Есть С.Л.О.Н.а по частям. Внедрить ITAM и не подавиться	62
<i>Всеволод Шадрин</i>	
Как правильно выбрать метрики управления знаниями	69
<i>Рик Джослин</i>	
Совместное использование фреймворка Synefin и подхода Swarming для лучшего управления инцидентами	74
<i>Джон Холл</i>	

Административный директор:
 Юлия Ромачева

Редактор:
 Константин Зимин

Дизайн и верстка:
 Наталия Долгая, Ольга Кладовикова

Литредактура и корректура:
 Елена Моржакова

Альманах содержит статьи, опубликованные на портале Real ITSM и специально переработанные для альманаха.

© itSMF Russia 2018
 Все права защищены.

Ни одна часть настоящего издания ни в каких целях не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами, если на это нет письменного разрешения itSMF Russia.



Уже почти год, точнее с 1 ноября 2017 года, компания AXELOS ведет проект по обновлению ITIL и написанию новой версии библиотеки. О том, каковы цели и ограничения, о направлениях изменений и организации этих работ, мы беседуем с Романом Журавлевым, Product Development Manager компании AXELOS¹.

Новая версия ITIL, что будет?

Расскажите, пожалуйста, о проекте обновления ITIL, который сейчас идет в AXELOS. Какие задачи и ограничения поставлены?

Роман Журавлев: Во-первых, это очень масштабное изменение ITIL, сравнимое с тем, что произошло в 2007 году, когда ITIL v2 заменили ITIL v3. И гораздо менее похожее на то обновление ITIL v3, которое было проведено в 2011 году. Идет очень серьезная и большая работа.

Во-вторых, мы планируем, что это будет последнее большое обновление ITIL. Конечно, развитие ITIL не остановится, но оно будет более плавным и гибким, без больших релизов. И для этого мы изменили

структуру библиотеки таким образом, чтобы можно было обеспечить постепенное обновление контента.

В-третьих, несмотря на масштабность и структурность изменений, обновленный ITIL должен быть совместим с текущей версией. Нам необходимо обеспечить плавный переход на новую версию как для организаций, которые используют ITIL, так и для людей, которые его изучают и сертифицируются. Есть масштабная практика, работающая инфраструктура и система продуктов, есть рынки, работающие на ключевых компонентах текущей версии ITIL. Очень важный вопрос – сертификация, и мы понимаем, что очень важно сохранить ценность того, что уже достигнуто. Чтобы уже полученные знания

¹Интервью частично написано на основе вебинара Романа Журавлева, организованного компанией Cleverics.

и сертификаты не нужно было выбрасывать, чтобы они дополняли то, что станет необходимо завтра. И при разработке новой системы сертификации одна из задач – обеспечить максимально плавный и учитывающий текущие достижения переход на новые экзамены и сертификаты.

– Насколько серьезно изменится новый ITIL по сравнению с текущей версией? Что можно сказать конкретно про новшества?

– С одной стороны, выделена очень существенная часть, которая остается неизменной. Это базовые принципы, которые для большинства компаний по-прежнему актуальны. Например, инциденты нужно разрешать как можно быстрее. Мы верим в то, что значительная часть практик, описанных в ITIL v3, как и раньше, эффективна для многих компаний. Но, с другой стороны, несомненно, условия изменились и практики работы тоже.

Изменения пройдут по нескольким ключевым направлениям.

Во-первых, расширяется охват деятельности, которую описывает библиотека. Примерно так же, как при переходе от ITIL v2 к ITIL v3, библиотека охватила вопросы стратегии и проектирования услуг, в новой версии будут описаны все аспекты полного жизненного цикла услуг. Полный жизненный цикл включает в себя этап создания систем, на базе которых эти услуги работают, то есть разработку систем и/или их закупку. В текущей версии ITIL в данном месте пробел, но в новой версии эта деятельность будет описана – добавляется большая часть, которую условно можно назвать проектной деятельностью.

Однако есть и другие направления изменений. Во-вторых, и это, наверное, главное с точки зрения практики, меняется структура описания того, что мы привыкли называть процессами. ITIL всегда был процессно-ориентированной методологией, основным источником практических знаний было описание процесса. Хотя далеко не всегда такое описание соответствовало общепринятой процессной терминологии, но в целом акцент был сделан на то, какие шаги и в какой последовательности необходимо сделать, чтобы, например, от состояния «обнаружен инцидент» прийти к состоянию «инцидент решен».

Однако, для того чтобы прийти к такому состоянию, помимо знания шагов важно еще понимать, как устроены команды, которые это делают, какие и как привлекаются ресурсы, как меняется информация, какие используются инструменты и как вовлекаются третьи стороны. Эти компоненты вместе работают на результат.

При написании новой версии ITIL описание каждой практики мы стараемся сделать по возможности сбалансированным. Команды и люди, органи-

зационная структура, инструменты, потоки информации и информационные объекты, шаги и процедуры, точки вовлечения партнеров и возможные схемы взаимодействия с ними, – для каждой практики мы описываем все эти аспекты.

В новой версии ITIL расширяется охват деятельности, которую описывает библиотека, добавляется большая часть, которую условно можно назвать проектной деятельностью.

– Это очень серьезное изменение, ITSM всегда основывался на двух ключевых подходах – услуги и процессы.

– Мы приняли решение добавить полезную практическую информацию, которую многие ждут и которой в текущей версии ITIL недостаточно. В связи с чем изменится и название элементов, которые ранее были описаны как процессы. Теперь для этого мы используем термин «практика». В эту практику входит много аспектов и элементов. Более того, она необязательно должна представляться в виде процесса, практика может включать в себя более чем один процесс.

– Будет ли включено что-то, связанное с гибкими практиками Agile и DevOps?

– Да, будет. Методы, техники, практики и принципы, которые за прошедшие с момента последнего обновления 10 лет показали свою эффективность, как минимум будут упомянуты. Agile старше ITIL, многие его практики доказали свою эффективность, и это можно считать успешной практикой. Широко применяемые элементы найдут свое место в новой версии ITIL. Но, конечно, отдельного процесса DevOps или Agile не будет, их элементы встроятся в структуру деятельности.

– Как организована работа по обновлению ITIL?

– Учитывая сложность и масштаб задач, мы потратили много усилий на выработку общих подходов и решений. С одной стороны, необходимо было создать автономные команды, работающие над каждой книгой. С другой стороны, изменения, которые они делают, должны коррелировать друг с другом, чтобы не получилось «коня налево повернул,

а сам направо поскакал». И мы нашли необходимую организационную структуру. На верхнем уровне создан архитектурный комитет, в который входят представители от каждой обновляемой книги. Над каждой книгой работают автономные команды, в каждой из них есть три редактора-архитектора, которые принимают ключевые решения по содержанию.

Мы определили состав книг, а также подобрали авторов. На старте проекта мы приглашали всех, кто хотел поучаствовать в нем. Затем все потенциаль-

– Каким образом будет обеспечиваться плавное обновление ITIL?

– Во-первых, разработанная структура библиотеки открыта. Она предполагает, что если через год мы поймем, что нужна еще одна книга по какому-то вопросу, что в этих знаниях есть потребность и по этой теме уже сложились хорошие практики и мы знаем, что писать, то мы сможем в эту структуру встроить еще одну книгу. Мы не фиксируем окончательный состав библиотеки, определяем только начальное ядро, дальше она может развиваться.

Во-вторых, на время написания книги команда авторов постоянна, однако общий состав авторов, обновляющих ITIL, меняется, по мере необходимости подключаются различные эксперты. Дело в том, что кроме непосредственно книг пишутся еще и так называемые дополнительные материалы. Различные публикации к основным книгам были и раньше, однако их объем был меньше ожиданий. В рамках этого обновления ITIL мы считаем очень важным разработать существенный объем дополнительных материалов, которые впоследствии будут постоянно обновляться.

Для книг, которые сейчас в работе, для каждой главы определены темы, которые не войдут в основную книгу, но по которым нужно написать 10–15 страниц дополнительных материалов. Эти материалы не попадают в основную книгу по разным причинам – либо объем книги велик, либо они описывают возможную практику, может быть не единственно возможную, необязательно отражающую «лучшие практики». Приведу пример: есть много способов приоритизации инцидентов. Их не стоит описывать в основной книге, но про каждый можно написать небольшой дополнительный материал и предложить читателям выбирать между этими способами (и мы не будем об этом спрашивать на экзамене). А если через три года жизнь покажет, что какие-то из этих способов более популярны, когда мы увидим на практике, что определенный метод хорош и про него все хорошо отзываются, тогда мы добавим его в основную книгу (и, возможно, в экзамен). Дополнительные материалы дают нам возможности получения обратной связи и обновления, которые ранее были недоступны.

Таким образом, потенциально бесконечное количество дополнительных материалов предполагает потенциально бесконечное число авторов, которые будут их писать. Обновление ITIL 2007 года было довольно закрытым проектом, но сейчас мы стараемся широко вовлекать профессиональное сообщество. И общий состав авторов, обновляющих ITIL, меняется. Поэтому поиск экспертов, готовых стать авторами, что-то добавлять или комментировать, будет постоянным. ITIL – это постоянная работа по систематическому обновлению материалов. ♦

Команды и люди, организационная структура, инструменты, потоки информации и информационные объекты, шаги и процедуры, точки вовлечения партнеров и возможные схемы взаимодействия с ними – для каждой практики мы описываем все эти аспекты.

ные авторы были структурированы по компетенциям и направлениям. Были сделаны кросс-проверки, кто кого знает и кто может подтвердить, что эксперт действительно опытный. Очень важно, что таким образом вовлечены эксперты, которым есть что сказать на основе своей практики проектов. Кроме того, у каждой главы книги есть несколько независимых рецензентов. В результате над обновлением ITIL работает довольно большой коллектив.

Ключевую роль играют архитекторы книг – это 12 человек, и только 4 из них работают в компании AXELOS. Важно отметить, что команда интернациональная, в нее входят эксперты из Великобритании, Нидерландов, Австралии, Японии, США и Канады.

Axelos Membership

Для распространения дополнительных материалов предусмотрена система AXELOS Membership, которая по сути является системой подписки на дополнительные и основные материалы библиотеки. Эта подписка стоит недорого, более того, начиная с января 2018 года все, кто сдал экзамен ITIL Foundation, получают год подписки бесплатно.

Там нет всех книг ITIL, но много дополнительной информации. Это большой объем дополнительной информации, который со временем станет все ценнее и ценнее. К сожалению, сейчас в России эта подписка не очень распространена, возможно, вследствие языкового барьера. Возможно, в дальнейшем в ней появятся материалы на русском.



Стюарт Рэнс (Stuart Rance)

Консультант в области ITSM и безопасности, преподаватель. Автор книги ITIL Service Transition 2011 и ведущий автор свода лучших практик RESILIA Cyber Resilience Best Practice Solutions, опубликованного в июне 2015 года.

Подозреваю, что большинство читателей слышали об ITIL, самом популярном в мире своде практик для управления ИТ-услугами. И я уверен, многие из вас даже пытались использовать ITIL. Но, к сожалению, большинство убедилось в том, что на деле все не так просто, как казалось. В этой статье я расскажу, насколько легко начать работу с ITIL и как быстро она может дать ощутимые результаты вам, вашим коллегам и клиентам¹.

ITSM 101: начало работы с ITIL

Вероятно, вы уже читали другие статьи о том, как начать работу с ITIL, и думаете, что знаете, о чем будет рассказано в моей статье. О том, как внедрять процессы и инструменты для управления инцидентами, управления проблемами, управления изменениями и так далее для трансформации вашего ИТ-отдела. Но я не буду этого делать. Независимо от того, что вы, возможно, читали ранее, изначально ITIL не о реализации процессов и инструментов. Нет. ITIL учит нас, что управление ИТ-услугами – это создание ценности для наших клиентов. И первый шаг к этому – определить, кто наши клиенты.

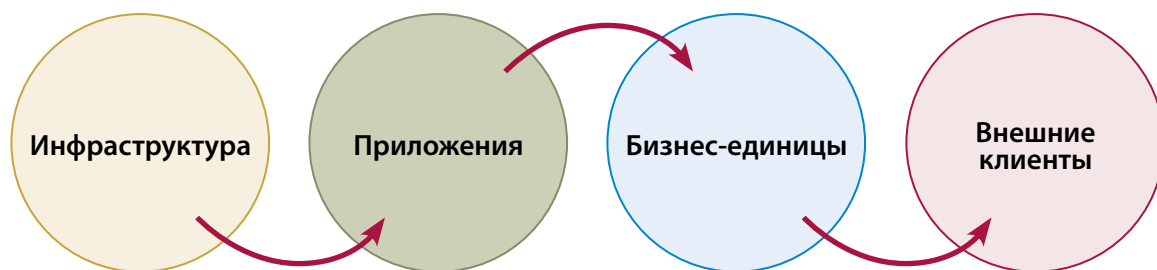
Наши клиенты – кто они?

Кажется, это очевидный вопрос, однако я постоянно поражаюсь, сколько людей, работающих в ИТ, не в состоянии дать четкий ответ. Если вы думаете о связях в типичном ИТ-отделе, то у вас может сложиться образ, показанный на рисунке 1.

Инфраструктуру формируют команды, называемые, например, «команда систем хранения» или Windows-команда. Инфраструктурные команды имеют клиентов в командах приложений. У команд приложений есть клиенты в бизнес-подразделениях, например

¹Stuart Rance [ITSM 101: Getting Started with ITIL](#). Статья предоставлена порталом RealITSM и компанией Cleverics. Публикуется с некоторыми изменениями.

Рис. 1.
Клиентские
связи и поток
создания ценности
в типичном
ИТ-отделе.



«Производство» или «Продажи». А эти бизнес-подразделения, в свою очередь, создают ценность для внешних, платящих компаний, клиентов. Я говорю о клиентах, которые приносят деньги, но очевидно, что некоторые услуги предоставляются для общественного блага, а не для платящего заказчика.

Если все в ИТ понимают, как они вписываются в поток создания ценности и как способствуют созданию ценности для платящих клиентов, то это отличное начало. Однако если люди в команде систем хранения думают лишь о том, что они должны делать, чтобы предоставить системы хранения, а команда

И если вы понимаете, каким должен быть результат и какую он создает ценность для клиента, это позволит вам планировать и проектировать всю вашу деятельность таким образом, чтобы помогать создавать ценности для клиентов.

Но ITIL также говорит, что вы должны думать о затратах и рисках, связанных с каждой услугой, и что они также вносят вклад в создаваемую ценность. Затраты включают в себя не только стоимость создания и запуска услуги, но и любые затраты клиентов на ее использование. Для ИТ-услуги поддержки ипотеки они могут включать обучение ипотечных агентов, каналы до филиалов, предоставление ноутбуков или планшетов, рабочего места, питание и многие другие расходы.

Если все в ИТ понимают, как они вписываются в поток создания ценности и как способствуют созданию ценности для платящих клиентов, то это отличное начало.

приложений думает только, что поставить бизнес-единицам, то все становится гораздо более сложным. Подход, когда сотрудники делают то, что им сказали, но клиенты остаются недовольны результатами, малоэффективен. Самое время изменить его.

Результаты, стоимость, затраты и риски

Слово «результаты», упомянутое в последнем абзаце, – очень важный термин в ITIL. Результаты – это то, чего ИТ-услуги помогают достичь нашим клиентам. Именно результаты создают ценность для клиентов. Например, в финансовой компании может быть ИТ-услуга «поддержка ипотеки», помогающая обрабатывать заявки на ипотеку. Результатом для клиентов является кредит, и это ценность, поскольку позволяет им купить дом или квартиру.

Каждая ИТ-услуга должна создавать ценность, помогая клиентам достигать желаемых результатов.

Риски, которые следует учитывать, могут включать время, когда ИТ-услуга не работает, или сеть недоступна, или планшет агента неисправен. Существует также риск, связанный с предоставлением ипотеки тому, кто не в состоянии погасить кредит, а также риск перевода денег на неправильный банковский счет. Все эти риски, как и многие другие, должны управляться, чтобы ИТ-услуга предоставления ипотеки приносила ценность клиентам. И требуется много усилий, чтобы постоянно улучшать уровень управления такими рисками.

Постоянное улучшение

Постоянное улучшение – это еще один критически важный термин ITIL, который вам необходимо понять. Менеджеру может показаться, что лучший способ начать работать с ITIL – найти книгу, которая расскажет, как управлять ИТ-услугами, внимательно прочитать, а затем следовать инструкциям. Однако такой подход не сработает. Какими бы умными вы ни были, сколь тщательно ни старались сделать все по книге, вы очень быстро обнаружите, что тонете в деталях и, что еще хуже, сталкиваетесь со значительным сопротивлением, потому что практики в книге не были адаптированы к работе в конкретном окружении.

Нужно начать с чего-то очень простого, а затем использовать мониторинг и обратную связь для улучшения. Таким образом, вы очень быстро сможете предоставить клиентам некоторую ценность и со временем, по мере улучшения, создавать все больше и больше ценности. Постоянно улучшайте ИТ-услуги, командные навыки или процессы, которые используете для управления инцидентами или чего-либо еще. Есть книга ITIL о постоянном улучшении, его основы очень просты и могут быть применены на всех уровнях.

1. Убедитесь, что все понимают, чего вы пытаетесь достичь. Как правило, это видение, четко переданное представление о том, как выглядит успех.
2. Измерьте, насколько хорошо вы работаете, но помните, что измерения – это не цель, это просто инструмент, который можно использовать, чтобы обозначить тренды и пороговые значения.
3. Получите обратную связь в каждой точке цепочки создания ценности. Это подразумевает общение с людьми, которые являются прямыми получателями вашей работы, и понимание вашего влияния на общую цепочку создания ценности и платящих компании клиентов.
4. Подумайте о том, каким образом можно делать вещи лучше и разработайте безопасный эксперимент для проверки своих идей.
5. Запустите свой эксперимент и посмотрите, были ли вы правы. Если да, утвердите свою идею и начните так работать в будущем; если вы ошиблись, вернитесь к предыдущему состоянию и попробуйте что-то другое.

Если вы работаете, предполагая, что постоянное улучшение – это что-то, что делает менеджер по регулярному совершенствованию, в то время как вы работаете по-прежнему, вы не сможете развиваться. Чтобы добиться постоянного улучшения, все в организации должны думать о том, как повысить качество работы. Иногда это влечет за собой некоторые изменения, не влияющие на других людей. В иных случаях могут произойти серьезные структурные изменения, требующие участия большей части сотрудников компании. Большие изменения или небольшие, но постоянное улучшение должно продолжаться все время и на всех уровнях.

Как начать

Если вы дочитали до этих строк, вероятно, вы уже знаете, что нужно сделать, чтобы начать работу. Это так же просто, как разговаривать с окружающими людьми, чтобы убедиться, что вы понимаете,

как делаете вклад в создание ценности для клиентов. Если вы управляете командой, поговорите с ней о клиентах, стоимости, результатах, затратах и рисках. Если все в команде будут думать о том, что они делают в этих терминах, следующий шаг будет намного проще.

Далее следует подумать о циклах обратной связи. Кто может попросить вас оставить отзыв о проделанной им работе, чтобы убедиться, что это то, что нужно? И кому вы можете предложить обратную связь, чтобы помочь им улучшить вашу поддержку? Подумайте о том, какие измерения вы можете сделать, чтобы увидеть тенденции и предсказать проблемы, прежде чем они произойдут.

Чтобы добиться постоянного улучшения, все в организации должны думать о том, как повысить качество работы.

Наконец, проведите улучшения. Начните с малого, с простых вещей, которые можно реализовать с низким риском, и убедитесь, что вы измеряете влияние, которое они оказывают на окружающих вас людей, а также на стоимость, результаты, затраты и риски ИТ-услуг. Это тот самый момент, когда детали в публикациях ITIL становятся полезными. Когда вы обнаружите необходимость совершенствования управления изменениями, инцидентами или проблемами и так далее, можно посмотреть в руководство ITIL, чтобы почерпнуть идеи. Не просто скопировать написанное в книге, а использовать ее в качестве отправной точки для вашего собственного улучшения.

Все это может показаться слишком простым, чтобы быть эффективным, но лишь потому, что это действительно просто! Сложнее всего изменить организационную культуру, чтобы сосредоточиться на улучшении, влияющем на ценность, которую вы создаете для клиентов. Проводить улучшения уже намного проще!

Узнать больше о некоторых идеях из этой статьи можно прочитав книгу ITIL Practitioner Guidance. Если вы сделаете это, вы также откроете для себя девять руководящих принципов, которые помогут сохранить ваши улучшения управления ИТ-услугами. ♦



Антон Боганов

Председатель Ассоциации организаций и специалистов в сфере управления информационными технологиями «ИТ сервис-менеджмент форум» (itsSMF).
E-mail для связи: a.bogdanov@itsmforum.ru

Проблему предконтактного оппортунизма недобросовестного продавца и падкого на дешевизну потребителя можно решить, введя правило гарантии качества!

Модель Акерлофа «Рынок лимонов»: неопределенность качества и рыночный механизм».

Стандартизация в цифровую эпоху

Назрела необходимость во внедрении единых базовых стандартов качества оказания ИТ-услуг для трансграничного цифрового пространства. Для этого нужно создать саморегулируемую организацию в сфере ИТ, которая обеспечит разработку и развитие механизмов для создания, продвижения и совершенствования базовых стандартов и правил, регулирующих различные виды деятельности своих членов. Главными целями саморегулируемой организации в сфере ИТ должны стать: разработка согласованного определения, классификация и параметризация выделенного набора цифровых услуг в нормативно-правовом поле РФ. А в дальнейшем обеспечение постоянного контроля их качества цифровых услуг¹.

Сегодня очевиден факт, о котором много лет говорят специалисты ассоциации itsSMF России: доля ИТ в каждой индустрии возрастает, и происходит это все быстрее. Практически каждый значимый вид деятельности становится ИТ-бизнесом: процессы, цепочки и сети поставок, модели взаимодействия поставщиков и потребителей продуктов и услуг организуются по-новому, происходит радикальная трансформация целых отраслей. Это, в свою очередь, влияет и на образовательную деятельность: финансовое образование уже практически невозможно без знаний финтеха, совсем скоро и юридическое включит в себя обязательные знания «легалтеха» (LegalTech).

¹Впервые опубликовано на портале CIO.ru. Печатается с изменениями.

Мы видим также еще одну важнейшую управленческую тенденцию: бизнес-руководители начали создавать ИТ-системы без привлечения или с минимальным привлечением специалистов по информационным технологиям, используя готовые сетевые, как правило облачные, наработки. Такие же изменения видны и в среде пользователей: любой из нас, задействуя информационные сервисы (тоже, главным образом, облачные), может получить желаемые результаты.

Другими словами, от «пирамид» управленческой иерархии, регулирующей самую разную внутреннюю деятельность, связанную с автоматизацией, бизнес переходит к управляемому использованию различных внешних специализированных платформ. Это порождает совершенно новые риски, подчас неожиданные или недооцененные. Компании-платформы захватывают значительную долю рынка и видоизменяют конкуренцию. Возникает новая сложная экосистема, где взаимодействие и сотрудничество государства и бизнеса, в том числе международного, расширяются и усложняются. Эту экосистему сейчас принято называть «цифровой экономикой».

Яркий пример такого сотрудничества: с 2007 года по середину 2017-го агентство Innovate UK выделило на внедрение различных инноваций свыше 1,8 млрд фунтов, а партнеры и компании, разрабатывающие и внедряющие их, предоставили дополнительное финансирование на такую же сумму. Сотрудничество бизнеса с государством охватило более чем 7,6 тыс. организаций, а соответствующие проекты, по некоторым оценкам, добавили более 11,5 млрд фунтов к ВВП Великобритании и создали 55 тыс. новых рабочих мест.

Российские реалии

В самом начале 2018 года Правительство Российской Федерации выделило на деятельность, связанную с цифровой экономикой, свыше 0,5 трлн рублей, большая часть из которых будет потрачена на следующие направления: «Нормативное регулирование», «Формирование исследовательских компетенций и технологических заделов», «Информационная инфраструктура», «Информационная безопасность» и «Кадры и образование».

Следует отметить, что и частный бизнес, и государство продолжают вкладывать в ИТ колоссальные средства, нацеленные на другие программы и проекты, затрагивающие информатизацию или автоматизацию той или иной деятельности. Это означает, что сейчас как никогда важно уделять внимание координации всего комплекса деятельности по информатизации, вне зависимости от источника требований и целевой среды.

Особенно важно при решении этой задачи принять во внимание современный сервисный подход, то есть создание сервис-ориентированной экосистемы с обязательными элементами управления качеством ИТ-сервисов. Согласно принципам ITSM, управление этой экосистемой должно учитывать интересы и взаимодействие всех заинтересованных сторон. Ведь комплексные ИТ-услуги обеспечивают потребителям цифровизацию конечного результата, а не продуктов, входящих в состав этих сервисов. Именно сервисы являются основой инновационной экономики и одним из лавных факторов успешного международного экономического, инновационного, технологического и гуманитарного сотрудничества. Более того, в рамках данной экосистемы будут приняты меры по обеспечению цифровой безопасности и созданию технических заделов, обозначенных в Программе создания цифровой экономики.

От «пирамид» управленческой иерархии, регулирующей самую разную внутреннюю деятельность, связанную с автоматизацией, бизнес переходит к управляемому использованию различных внешних специализированных платформ. Это порождает совершенно новые риски.

Такой подход позволит управлять наиболее значимыми для цифровой экономики рисками и преимуществами, связанными с тем, что разработка и применение ИТ трансграничны по своей сути. Ведь главный фактор влияния на мировых цифровых рынках сегодня – мультинациональные игроки. Это означает, что, например, деятельность по созданию компонентов или целых сервисов может финансироваться Европейским союзом или инвесторами из США, само проектирование и разработку сервисов де-факто могут выполнить команды из России, Израиля, Сингапура, производство компонентов – запущено в Китае, а поддержка и ИТ-инфраструктура предоставлены Индией, Чехией или Сербией. Добавим сюда различия в налоговых требованиях, культуре ведения бизнеса, в часовых поясах и проч. Полученная картина хорошо отражает колоссальные трудности владельцев и руководителей в организации единого управления. Однако она дает лишь частичное представление о современных сложностях при реализации планов построения цифровой экономики.

Поэтому именно использование принципов построения современной сервис-ориентированной экосистемы позволит через фильтры контроля единых критериев качества сервисов и их компонентов, применяемых от момента планирования и проектирования до процесса их использования и совершенствования, обеспечить единое управление с учетом различных и не всегда прозрачных факторов влияния. Соответственно, параметры качества цифровых сервисов, получаемые при помощи грамотно спроектированной системы измерений и метрик, должны быть едины для всех участников сети создания сервисов, а также прозрачны и понятны для всех потребителей этих сервисов.

Параметры качества цифровых сервисов, получаемые при помощи грамотно спроектированной системы измерений и метрик, должны быть едины для всех участников сети создания сервисов, а также прозрачны и понятны для всех потребителей этих сервисов.

Цена и качество

Многолетний опыт работы ассоциации itSMF показывает, что оценить качество работы цифровых ИТ-услуг задача нетривиальная, не говоря уже о грамотном управлении ими. Внешняя спецификация сервиса, доступная заказчику, не дает уверенности в надлежащем качестве работы и состава его компонентов. Оптимально ли функционирует система управления, достаточна ли квалификация персонала, имеется ли нужное оборудование, обслуживается ли оно эффективно и в полном ли объеме, обеспечен ли требуемый уровень информационной безопасности? При выборе потенциального поставщика цифровой услуги необходимо также учитывать параметры внутренних спецификаций. Не у всех компаний – потребителей цифровых услуг есть в штате персонал, обладающий соответствующей квалификацией и имеющий запас времени для такой последовательной и скрупулезной работы. Да и вряд ли одной организации есть смысл иметь достаточную квалификацию по всем предлагаемым на рынке ИТ-услугам, тем более инновационным. Это экономически неэффективно. У части поставщиков попросту отсутствует мотивация для поддержания должного уровня качества, поскольку услуги и так потребляются, даже если страдает заказчик.

Бывает и по-другому: заказчик выдвигает поставщику ничем не обоснованные требования или претензии, и тот несет убытки, ведь клиент всегда прав... У потребителя свой подход: многие основным критерием качества услуг считают цену. Однако этот критерий, если он рассматривается отдельно от прочих, может совсем не отражать реальное качество услуги. Вспомним, сколько поводов для дискуссий дает 44-ФЗ. В условиях интенсивного инновационного развития ценовой критерий совсем не работает, поскольку глубокая повсеместная модернизация стоит огромных денег и требует грамотного управления для минимизации рисков и исключения ненужных затрат. Тут вряд ли уместно считать предлагаемый дешевый сервис качественным. Поэтому у большинства потребителей нет желания применять инновации первыми, и они ждут, когда кто-то другой их проверит. А ждать нельзя: в динамичной среде время важно во всем. Только очень оперативное принятие решений и мгновенное выполнение правильных действий приведут к прорывам, промедление же дает конкурентам возможность быстро добиться успеха и разделить нишу с чемпионами.

Свобода или запреты?

Во многих отраслях российской экономики до 80% рынка находится в руках профильных министерств и госкорпораций. Это – одна из причин того, что наша экономика пока еще неконкурентна, поскольку динамика развития рынка напрямую зависит от количества могущественных игроков и функционеров, за счет взаимодействия и конкуренции которых и обеспечивается движение вперед. Регулирование рынка тоже затруднительно, так как эксперты не имеют в своем распоряжении достаточных ресурсов, чтобы учесть в многочисленных актах, положениях, правилах и фильтрах все факторы влияния, а госаппарат весьма инертен в согласованиях и принятии окончательных решений.

С другой стороны, инновации не спрашивают разрешения, они открыты для применения и очень востребованы. Что лучше: идти по пути запретов или дать потребителям полную свободу действий? Истина, как всегда, где-то посередине. Нам нужен прорыв, но не нужна вседозволенность. Мы совсем не рады приезду некомпетентного водителя такси, вызванного через приложение, и уж тем более не рады опыту пользователей почившего сервиса Chatroulette, сталкивавшихся в самом прямом смысле с голыми маньяками.

Целостной картины будущего нет ни у кого, а значит, результат предстоящих изменений не предопределен. Поэтому к созданию эффективной кибер-

экономики нужно привлекать все заинтересованные стороны: требуется синергия государственной власти, частного бизнеса, науки и экспертных сообществ. В предельной формулировке закона необходимого разнообразия Уильяма Эшби оптимальное управление достигается при условии соответствия разнообразия управляющего воздействия разнообразию управляемого объекта и строгой однозначности управляющего воздействия. Множество мнений и подходов экспертов из всех областей позволяет выработать однозначные способы и подходы к управлению различными сервисами.

Саморегулируемая экосистема

У сервисной экосистемы, как и у любой другой, должна быть основа, платформа. Такой платформой могут стать сообщество и ресурсы его членов в форме отраслевой саморегулируемой организации. Для участников процесса построения киберэкономики очень важно уметь грамотно распоряжаться собственными ресурсами и оптимизировать внутренние процессы. Это позволит опередить конкурентов. Но не менее ценно и умение организовать взаимодействие с внешними сторонами. И важно не только повышение потребительской ценности экосистемы для отдельных игроков рынка, но и максимизация ее в целом.

Основой построения и функционирования предлагаемой сервис-ориентированной экосистемы может стать вновь созданная саморегулируемая организация в сфере ИТ (СРО ИТ), которая обеспечит разработку и развитие механизмов для создания, продвижения и совершенствования базовых стандартов и правил, регулирующих различные виды деятельности своих членов, включая контроль качества потребляемых и оказываемых ИТ-услуг. Эти правила, согласованные как внутри страны, так и на международном уровне, позволят осуществлять периодический контроль деятельности своих членов, в том числе в соответствии с правилами деловой этики. Работа такого объединения даст возможность на ранней стадии выявлять некомпетентность контрагентов и разрабатывать меры по противодействию злему умыслу недобросовестных партнеров.

Главными целями саморегулируемой организации в сфере ИТ должны стать разработка согласованного определения, классификация и параметризация выделенного набора цифровых услуг в нормативно-правовом поле РФ. При этом не следует ограничиваться традиционными ИТ-услугами, такими как SaaS, PaaS, IaaS, а стоит включить также информационные бизнес-услуги, полностью построенные на базе ИТ и оказываемые непосредственно потребителям не по гибридной схеме, а исключительно в рамках

Всемирной сети. Таким образом, качество формализованной цифровой услуги напрямую увязывается с бизнес-результатом, достижению которого должна способствовать эта услуга. Стороны, договаривающи-

Оценить качество работы цифровых ИТ-услуг задача нетривиальная, не говоря уже о грамотном управлении ими. В условиях интенсивного инновационного развития ценовой критерий совсем не работает, поскольку глубокая модернизация стоит огромных денег.

еся об уровне ИТ-обслуживания, однозначно понимают границы оказания услуг, однозначно определяют и интерпретируют понятие качества оказываемых отдельных услуг или их набора, а также способы измерения показателей качества, которые подтверждены сертификатом саморегулируемой организации в сфере ИТ. Этот сертификат – основа защиты прав потребителей в цифровой реальности, инструмент для решения спорных вопросов Роспотребнадзором, Ростехнадзором или Роскомнадзором.

Целостной картины будущего нет ни у кого, а значит, результат предстоящих изменений не предопределен. Поэтому к созданию эффективной киберэкономики нужно привлекать все заинтересованные стороны: требуется синергия государственной власти, частного бизнеса, науки и экспертных сообществ.

Саморегулируемую организацию в сфере ИТ можно рассматривать как точку сосредоточения и анализа экспертных управленческих знаний, создания и пропаганды свода лучших российских управленческих практик и уникальных наработок. Она может оказать серьезную помощь в гармонизации законов и стандартов в области оказания ИТ-услуг и обеспечении контроля их качества, повысить уро-

вень грамотности населения и специалистов в области управления киберэкономикой. Кроме того, обязательным компонентом саморегулируемой организации в сфере ИТ является финансовый фонд, призванный компенсировать потери потребителям ИТ-услуг; появляется возможность организовать систему добровольной сертификации качества оказываемых цифровых услуг среди ее членов. На базе такой системы можно проводить добровольную сертификацию образовательных программ и компетенций специалистов, работающих в области цифровой экономики. Активное применение здесь также найдут смарт-контакты и распределенный реестр, управляемый доверенным органом.

Поможет ITSM

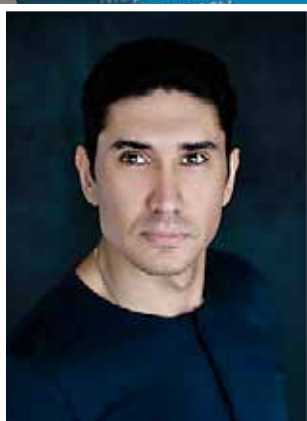
Главными целями саморегулируемой организации в сфере ИТ должны стать разработка согласованного определения, классификация и параметризация выделенного набора цифровых услуг в нормативно-правовом поле РФ.

Саморегулируемая организация в сфере ИТ не должна строиться с нуля – на российском рынке существует экспертиза в области управления качеством ИТ-услуг. Ассоциация организаций и специалистов в сфере управления информационными технологиями «ИТ сервис-менеджмент форум» (itSMF России) успешно ведет свою деятельность, в том числе на международной арене – уже почти двенадцать лет, имеет тесные связи с аналогичными отделениями по всему миру. Цель и стратегические ориентиры ассоциации – оптимизация управления деятельностью в области ИТ. Это предполагает повышение экономической эффективности как самих ИТ-услуг, так и работы внутренних и внешних ИТ-подразделений, обеспечение эластичности ИТ-сервисов даже при угрозах со стороны прорывных технологий, разрушающих традиционные подходы к управлению бизнесом. Опыт экспертов форума может быть напрямую применен при реализации ставших уже международными четырех принципов Индустрии 4.0.

Сервисный подход к управлению ИТ (ITSM) дает ответы на множество вопросов, связанных с организацией трансформации и управления релизами, с договорной деятельностью на протяжении всего срока действия контракта жизненного цикла ИТ-услуг и их среды, что требует от поставщиков услуг тщательного учета всех видов взаимодействий, включая децентрализацию принятия управленческих решений. Сервисный подход позволяет четко разграничить ответственность между функциональными единицами, организовать совместную работу инженеров и разработчиков в едином цифровом пространстве с возможностью делегирования некоторых функций киберфизическим системам. Он помогает выстроить учет цифровых моделей объектов эксплуатации, добиться прозрачности взаимосвязей всех компонентов, обеспечить их устойчивое функционирование, своевременное техническое обслуживание, внесение изменений в настройки систем и их частичную или полную замену, оперативную поддержку клиентов, управлять запасом расходных материалов, анализировать и своевременно получать точную информацию для принятия верного управленческого решения и т. п.

Что касается законодательной защиты, то участники киберэкономки получают реальную поддержку от Инновационного центра «Сколково». Эксперты itSMF России и ИТ-кластер «Сколково» уже сфокусировали свое внимание на аспектах LegalTech. Резиденты «Сколково» знают, как заключать смарт-контракты и разрешать споры контрагентов при помощи искусственного интеллекта, оставаясь в пределах экосистемы цифровых услуг.

Таким образом, саморегулируемая организация в сфере ИТ, созданная на базе Ассоциации организаций и специалистов в сфере управления информационными технологиями «ИТ сервис-менеджмент форум», обеспечит сбалансированную сервисную экосистему, которая окажет реальную помощь в создании цифровой экономики в России. При этом государственные структуры, российское бизнес-общество и ИТ-профессионалы получают единую точку доступа к российской, да и к любой другой согласованной и востребованной управленческой экспертизе и базе управленческих наработок. Они смогут участвовать в этой деятельности, что даст возможность России впервые минимизировать зависимость от западных управленческих «лучших практик» и выведет на первое место те достижения наших или работающих на наших рынках компаний, которые позволили им добиться лучших результатов. ♦



Руслан Смаков

Директор по развитию направления «Акелон Услуги» в компании «Акелон», член Ассоциации сертифицированных консультантов по управлению «Национальный Институт сертифицированных консультантов по управлению» и Ассоциации профессионалов управления бизнес-процессами (ABPMP Russian Chapter).

Деятельность любой организации подвержена влиянию различных факторов, что может привести к резкому падению прибыльности одних направлений бизнеса и резкому подъему других. Изменение бизнес-окружения может быстро поставить под сомнение актуальность и реалистичность целей организации, а неготовность к таким изменениям грозит не только потерей денег, репутации или бизнеса, но может стать и причиной упущенных возможностей. Гибкость организации, ее маневренность, мгновенная коррекция планов и целей требуют целостной системы управления. Статья основана на опыте реализации проекта построения сервисной модели управления в транснациональной компании.

Нарушая границы, или Как сервисная модель управления может повысить эффективность бизнеса в целом

Набили шишки, собрали грабли

ИТ-отдел уже умеет использовать процессы ITSM для оптимизации своей работы. Что-то работает, что-то приходится адаптировать, но в итоге все приносит свои плоды. В сухом остатке руководство ИТ-отдела получает возможность контролировать, замерять, анализировать и улучшать свою деятельность. А что дальше? Дальше осознание того факта, что **мы накопили достаточно опыта применения информационных технологий и знаний, чтобы расширить**

влияние ИТ в организации в целом. Что мешает нам использовать знакомые методы, подходы и накопленный опыт для повышения эффективности и производительности на всех уровнях компании, за рамками ИТ?

Зависимость бизнеса от информационных технологий с каждым годом увеличивается. Многофункциональные компании уже требуют от ИТ постоянной поддержки своих операций. Цифровые

каналы настолько пронизывают бизнес, что ИТ-отдел стал полноправным участником бизнеса.

Связанные одной целью

Серьезными показателями эффективной деятельности являются снижение количества ошибок, высокая скорость обмена информацией и стабильность работы бизнес-функции, процесса или подразделения. Корень проблем чаще всего не в ошибках конкретного человека, а в организации работы в целом. Когда процессы пересекают границы структурных подразделений, создаются стыки (дыры, слепые зоны, ничья территория) – скрытые и опасные зоны риска. Переходы от одного процесса к другому, из зоны ответственности одного подразделения под ответственность другого требуют наличия моста – качественных каналов передачи результатов. Этим мостом может стать система класса Enterprise Service Management, объединяющая в себе классические подходы, практики и системы управления ИТ-услугами и системы управления процессами BPM. Согласно определению Википедии:

Enterprise Service Management (ESM) – это вид программного обеспечения для управления бизнесом, как правило, набор интегрированных приложений, которые сервисная организация использует для сбора, управления, сохранения и анализа данных, имеющих для нее критически важное значение.

Система класса Enterprise Service Management обладает большим потенциалом для автоматизации процессов, которые объединяют бизнес-функции. ESM позволит создать систему мониторинга и контроля бизнес-процессов всей организации, обеспечить централизацию управления и трансформировать их в сервисную модель управления.

Формирование новых подходов к управлению означает взять на себя ответственность за организационно-культурные изменения в компании.

Сервисная модель управления

Начну с цитаты из статьи Каймара Кару, руководителя ITSM-направления компании AXELOS с 2014 по 2017 год, «Зачем сервис-менеджмент (в ИТ)»:

Управление услугами, или сервис-менеджмент, – это способ начать и поддерживать разговор с заказчиками, понять, в чем состоит ожидае-

мая заказчиками ценность (и постоянно обновлять это понимание), и постоянно повышать эту ценность, совершенствуя оказываемые услуги. Сервис-менеджмент подразумевает предоставление услуг, а не просто технологических решений или определенной функциональности. Сервисная модель предполагает целостный взгляд на формирование ценности, охватывающий все задействованные ресурсы и способности, и помогает избежать локальной оптимизации.

Формирование новых подходов к управлению означает взять на себя ответственность за организационно-культурные изменения в компании. Сервисная модель управления использует принципы сервисного подхода при организации повседневной работы сотрудников. Сервисная модель управления – это способ организации деятельности компании, при котором сотрудникам предоставляется единый инструмент для запроса необходимых в каждодневной работе услуг и сервисов, оказываемых внутренними подразделениями и/или внешними подрядчиками, где распределение инициированных сотрудниками запросов по ответственным исполнителям осуществляет информационная система.

Главная задача сервисной модели управления – обеспечить содействие основной бизнес-модели в получении конечных результатов. Руководству такой подход дает возможность «на ходу», без отрыва от производства менять структуру компании под изменяющиеся условия внутренней и внешней среды, делая бизнес более гибким и адаптивным. При такой модели «внутренняя кухня» бизнес-процессов для сотрудника/пользователя не видна и представляет собой каталог услуг и сервисов. Сотрудник даже не заметит, если ранее выполнявший функции внутренний отдел организации будет заменен на аутсорсинговую компанию.

Система класса Enterprise Service Management – это возможность управлять процессами организации через единый инструмент, обеспечивая качественную координацию и взаимосвязь, ведущую к общим целям компании. Централизация управления процессами позволит отслеживать прогресс выполнения работ, выявлять сбои и своевременно на них реагировать, расширять сотрудничество подразделений и контролировать качество передаваемых результатов, что повлияет на эффективность работы организации в целом.

Концепция Enterprise Service Management не нова. Традиционно использование ITSM ограничивалось стенами ИТ-подразделений, делая ИТ-операции более продуктивными, обеспечивая лучшую под-

держку ИТ-потребностей бизнеса. Опыт автоматизации рабочих процессов, которые помогают ИТ-отделу сэкономить время и снизить количество ошибок, может быть применим в других отделах и достичь тех же результатов. Концепция Enterprise Service Management обычно реализуется на базе программной платформы для управления корпоративными сервисами, которая позволяет отделам и подразделениям предоставлять услуги/сервисы внутренним пользователям более оптимизированным и экономичным способом.

Каталог услуг

Важнейший элемент сервисной модели управления – каталог услуг. Каталог услуг – это организованная цифровая презентация услуг, предоставляемых внутренними подразделениями (владельцами сервисов) клиентам (внутренним и внешним потребителям). На практике бывают случаи наличия в каталоге услуг/сервисов, оказываемых внешними подрядными организациями (см. рисунок). Цель каталога – создать единую витрину услуг, которые потребляются внутренними клиентами в повседневной работе.

Каталог услуг дает возможность пользователям получать доступ к услугам и сервисам, которые необходимы им в каждодневной работе. Важно, чтобы предоставляемая информация была в знакомых для потребителя терминах, максимально простой и без технических нюансов.

Переходы от одного процесса к другому, из зоны ответственности одного подразделения под ответственность другого требуют наличия моста. Этим мостом может стать система класса Enterprise Service Management.

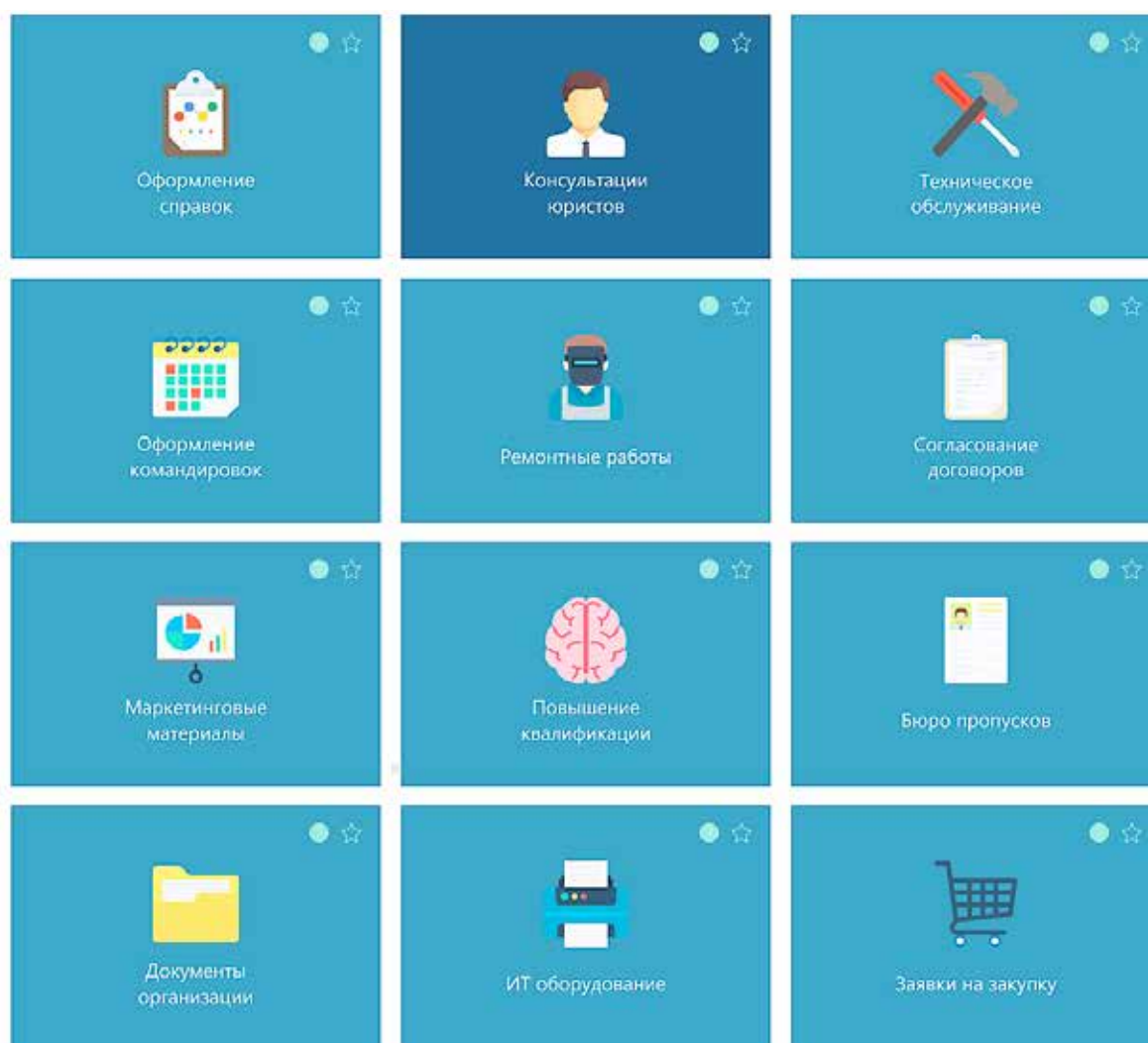


Рис.
Пример каталога услуг.

Каталог услуг состоит из внешней и внутренней частей. Внешняя – это портал, на котором расположены оказываемые подразделениями услуги и сервисы в понятных клиенту терминах, дающий возможность заказать необходимые в повседневной работе услуги. Услуги могут быть как банальными, так и уникальными:

- заказать справку 2НДФЛ;
- заказать канцелярские товары;
- провести техническое обслуживание оборудования;
- обеспечить утилизацию отходов;
- провести обучение персонала;
- разработать маркетинговые материалы и т.д.

К внутренней части каталога услуг можно отнести:

- а) информацию, четкие и понятные инструкции по предоставлению услуг:
 - название услуги;
 - соглашение об уровне сервиса (SLA) – договор между поставщиком услуги/сервиса и конечным потребителем, определяющий ожидаемый уровень обслуживания;
 - информацию о том, кто может запросить услугу (права доступа к услуге);
 - информацию о том, кто является владельцем сервиса;
 - график предоставления сервиса, условия и приоритеты.
- б) технические средства для предоставления услуг:
 - управление и наполнение каталога услуг;
 - автоматизация доставки запросов на предоставление услуг владельцам сервиса;
 - исполнение входящих запросов на предоставление услуг и предоставление результатов;
 - контроль, эскалация и уведомления о событиях.

Единый взгляд на существующие процессы организации и предоставляемые сервисы обеспечивает новый уровень прозрачности для непрерывной оптимизации.

Помимо предоставления наглядных и необходимых в работе бизнес-сервисов каталог услуг обеспечивает сервисной организации дополнительные преимущества:

- повышение удовлетворенности пользователей;
- повышение эффективности распределения ресурсов;

- сокращение сроков доставки результатов;
- упрощение процедуры предоставления услуг;
- снижение общих затрат на обслуживание;
- повышение коммуникаций и улучшение совместной работы.

Пять преимуществ сервисной модели управления и организации каталога услуг

1. Централизация. Организация подвергается влиянию внешней среды, а также осознанно или неосознанно сама оказывает влияние на нее. Единый взгляд на существующие процессы организации и предоставляемые сервисы обеспечивает новый уровень прозрачности для непрерывной оптимизации. Мы можем видеть, какие процессы стоит модернизировать, упростить, как и за счет чего снизить их стоимость и сроки исполнения, повышая при этом качество результатов. Прозрачность дает нам маневренность, гибкость, способность адаптироваться и адекватно реагировать на изменения окружающей среды.

2. Автоматизация. Минимизация человеческого участия увеличивает скорость процесса. Автоматизация позволяет объединить людей, процессы и технологии для более эффективной деятельности. Иницилируемые пользователем запросы на необходимую услугу могут передаваться сразу нескольким командам одновременно, что обеспечивает совместную работу для достижения общих целей.

3. Эффективность. Запросы на оказание услуг/сервиса, инициированные через портал, быстро отправляются на обработку соответствующей команде и/или исполнителю. Если возникает риск превышения сроков реагирования/ответа/исполнения, происходит эскалация запроса на вышестоящее руководство. Различные уведомления пользователя портала (инициатора запроса) о статусе и сроках исполнения запрошенных им услуг и сервисов позволяют ему вернуться к основным обязанностям и быть уверенным, что его запрос не останется без внимания. Предоставляя сотрудникам единую платформу обслуживания, вы даете им один источник решения проблем и упрощаете им жизнь, делая их работу продуктивнее, а самих сотрудников – счастливее.

4. Ответность. Мы не можем исправить проблемы, не зная о них. Систематическое отслеживание операций в реальном времени позволяет видеть текущий объем, статусы операций (регистрация, в работе, просрочено, отложено и пр.), что дает возможность улучшить предоставляемый сервис. Когда

мы видим места, замедляющие работу, мы можем принять соответствующие меры. Обратная связь от потребителей, в виде оценки удовлетворенности предоставленными результатами, позволит повысить качество предоставляемых услуг.

5. База знаний. Анализ поступающих запросов и структурирование ответов на них в статьях базы знаний позволят закрывать однотипные и повторяющиеся обращения. Самообслуживание на портале поможет сотрудникам самостоятельно находить ответы на вопросы, что должно снизить количество контактов с исполнителями и обеспечить снижение нагрузки на человеческие ресурсы. База знаний может включать в себя различный контент – статьи, видеоролики, пошаговые инструкции, ссылки на внешние интернет-ресурсы. Предоставляя пользователям доступ к базе знаний, вы даете им шанс найти необходимую информацию и быстро вернуться к работе. Описанный в базе знаний опыт позволит пользователям получить доступ к полезной и актуальной информации в режиме 24/7.

8 принципов организации каталога услуг

USER FIRST

Получаемый пользователем результат является промежуточным с точки зрения организации в целом. Постановка сервисной модели управления и организация каталога услуг начинаются с идентификации конечных пользователей, которые будут выступать потребителями услуг и сервисов. Многие ITSM-инициативы приносят пользу ИТ-подразделениям, но не всегда дают полную отдачу бизнесу. Это происходит тогда, когда управление услугами организуется с фокусом на технологиях, а не на потребителе и потребностях бизнеса. Портал с каталогом услуг – это упражнение в эффективных коммуникациях. Пользователям необходимо предоставить понятный и простой инструмент запроса требуемых услуг, иначе они продолжают использовать привычные неэффективные каналы.

Базовые требования к portalу каталога услуг:

- легкий и удобный интерфейс;
- понятно описанный и спроектированный каталог услуг;
- функции поиска услуг;
- информация о состоянии запроса (сроки исполнения, статус и пр.);
- доступ к распространенным вопросам и ответам;
- доступ к portalу с компьютера и мобильного устройства;
- адаптивный дизайн для мобильных устройств.

Система одного окна

Формирование запросов на оказание услуг с помощью единого корпоративного портала услуг позволит избежать потерь запросов от потребителей.

***Пример.** В одной крупной транснациональной компании причиной внедрения сервисной модели управления стала потеря заявок на техническую эксплуатацию оборудования, поступающих от удаленных представительств в центральный офис. Представительства жаловались на то, что центральный офис их не слышит. Информирование центрального офиса о проблемах с оборудованием осуществлялось на еженедельных совещаниях. Проблемы фиксировались, но терялись и оставались без должного внимания. Это приводило к тому, что оборудование без своевременного технического обслуживания переставало функционировать, и возникали критические ситуации.*

Используйте подход маленьких побед. Пилотный проект на одном подразделении позволит обкатать внедрение сервисного подхода за рамками ИТ и убедиться, что система эффективна.

Понятный каталог услуг может быть первой инстанцией, куда отправится сотрудник, если ему понадобится услуга/сервис или у него появился вопрос. Предоставление единого инструмента для запроса любой услуги/сервиса позволит не упустить и своевременно отреагировать на критически важные моменты, которые могут повлиять на работу всей компании.

Команда проекта

Выходя за рамки ИТ-отдела, мы оказываемся в зоне ответственности другого подразделения, и очень важно заручиться поддержкой руководителей подразделений. Предложив им войти в состав команды проекта, вы не только получите разрешение работать на их территории, но и совместно с ними спроектируете релевантный каталог услуг под предоставляемые подразделением услуги и сервисы. Такое сотрудничество позволит определить важные для конечного пользователя ожидаемые результаты. Командная работа даст возможность оптимально разработать маршруты распределения заявок на запрашиваемые услуги и сервисы по ответственным лицам и обеспе-

чит принцип WIN-WIN. Предоставление руководителям инструмента контроля за работой их подразделений позволит сразу же перевести операционную деятельность на сервисную модель управления, особенно если этим руководителям помимо своих сотрудников необходимо контролировать работу аутсорсинговых компаний.

Начните с малого

Используйте подход маленьких побед. Пилотный проект на одном подразделении позволит обкатать внедрение сервисного подхода за рамками ИТ и убедиться, что система эффективна. Начните с простых услуг – заказ справок, бронирование переговорной комнаты, замена лампочек и т.д. Устраните недочеты и после того, как вы увидите, что начало положено и услугами пользуются сотрудники компании – можно усложнять и обеспечивать

лей и жизнеспособность проекта. Этого ни в коем случае допускать нельзя. Сегодня рынок предлагает специализированные решения с высокой функциональностью построения бизнес-процессов, высокой адаптацией под специфику бизнеса и сильными возможностями по интеграции с уже существующими системами.

Привлечение экспертов

Фраза «Если хочешь что-то сделать хорошо, сделай это сам» может сыграть злую шутку, когда помимо основной деятельности ИТ-менеджерам приходится заниматься трансформацией управления компанией. Мы можем потратить драгоценное время, упустить что-то важное из виду из-за замысленности глаза, закопаться в деталях. Если все-таки у нас хватило упорства пройти путь до конца, есть риск, причем с высокой вероятностью, получить не то, что мы планировали изначально.

На рынке существуют эксперты, основная деятельность которых – постановка сервисной модели управления. Они имеют пул реализованных проектов, есть опыт, а у некоторых и собственные готовые разработанные ИТ-решения с необходимым набором функциональных возможностей и адаптации под вашу среду (ведь в каждой среде реализация будет уникальна). Со стороны экспертов – опыт и знания, с вашей стороны – требования, экспертиза по устройству вашего бизнеса, контроль за работами и прием результатов. Это правильная формула, часто приводящая к успеху. Совместная тщательная подготовка проекта позволит избежать автоматизированного хаоса.

Заинтересованные лица и защита проекта

Важно понимать, что постановка сервисной модели управления – это не ИТ-проект, а согласованное изменение подходов к управлению организацией с поддержкой информационных технологий. Сколько бы мы ни играли в своей ИТ-песочнице, чтобы воспользоваться преимуществами сервисного подхода на уровне всей компании, нужно выходить на уровень бизнеса. Любой проект требует своего заказчика. Даже если мы понимаем и видим выгоды от предлагаемых изменений, нам все равно нужно будет делать запрос на требуемые под масштабный проект ресурсы. Единственный выход из ситуации в условиях отсутствия ресурсов – это защита проекта держателем бюджета. После того, как мы заручились поддержкой руководителей других подразделений, запустили пилотный проект с небольшим количеством услуг в каталоге, пришло время подготовки к подведению результатов пилота. Это сбор данных и аналитика: сколько времени уходило на операции

Непрерывное совершенствование – это не просто измерение показателей, это понимание, почему нужно измерять показатели, это, в первую очередь, стратегия.

параллельное выполнение работ по одной услуге/сервису. Например, прием нового сотрудника требует организации рабочего места (компьютер, стол, стул, пропуск, визитки), выдачи необходимых реквизитов для пользования внутренними информационными системами, инициации обучения сотрудника и ввода в должность и т.д.

Платформа

Автоматизированная система позволяет упростить контроль, сэкономить время, но ни в коем случае не снимает ответственность за выполнение функций. Инструмент может помочь, а может и навредить. Часто стандартные системы Service Desk, которые мы привыкли видеть на рынке, для работы с обращениями пользователей по ИТ-заявкам не в силах удовлетворить всех предъявляемых функциональных требований со стороны бизнес-подразделений. В одних случаях это слишком сложный для рядового пользователя технический вид решения, в других функциональные ограничения и трудоемкость реализации сложных комплексных процессов.

Когда стоит выбор, чем же все-таки жертвовать, выбор часто падает на пользователя. Начинается борьба и принуждение работать так, как пользователю неудобно, падает продуктивность пользовате-

раньше и какие показатели сейчас, сколько ошибок было до и как изменилась ситуация на сегодняшний день. Покажите полученные данные держателю бюджета в понятном виде, покажите, чего вы достигли, что можно улучшить и что не работает вовсе. Если на этом этапе уже работаете с экспертами, то они помогут в подготовке необходимых, критически важных показателей для защиты проекта перед держателем проекта.

Непрерывное совершенствование

Для того чтобы работать на перспективу и создавать основу для непрерывного улучшения, следует начать с вопросов: каких бизнес-результатов мы хотим достичь? Какие на сегодняшний день бизнес-требования и какие будут в будущем? Непрерывное совершенствование – это не просто измерение показателей, это еще и понимание, почему нужно измерять показатели, это, в первую очередь, стратегия.

Шаги на пути к совершенствованию:

- определите и соберите данные, необходимые для постоянного улучшения;
- проанализируйте как негативные, так и позитивные тенденции;
- выявите возможности для улучшения;
- поделитесь информацией с командой;
- иницируйте процесс улучшений по циклу Деминга (PDCA);

- вернитесь к началу и повторите, поскольку это непрерывное совершенствование.

Заключение

Бизнес-процессы определяют то, как организация функционирует в целом, а значит эффективная организация потоков работ ведет к более высокой производительности и в конечном итоге к более высокой

Постановка сервисной модели управления – это не ИТ-проект, а согласованное изменение подходов к управлению организацией с поддержкой информационных технологий.

отдаче от инвестиций. Все проблемы и упущения непродуманной работы в конечном счете ложатся на клиентов организации. Внедрив сервисную модель управления компанией и каталог услуг, вы сможете четче видеть, как обстоят дела в вашей компании, что является причинами тех или иных событий, кто отвечает за каждый этап работы, и получите возможность влиять на результаты работ. Определите для себя цели, нарисуйте схему, разработайте логику, согласуйте их с командой и заказчиком, действуйте. ♦



Всероссийский конкурс

Подробнее на сайте
www.itsmforum.ru



Александр Шпер

Ведущий эксперт, тренер компании IT Expert, к. э. н., ITIL Expert. Обладает 18-летним опытом работы в области информационных технологий. В IT Expert с 2007 года занимается разработкой процессов управления ИТ и реорганизацией системы ИТ-управления в рамках консалтинговых проектов, проводит аудиты систем управления ИТ-услугами, разрабатывает, совершенствует и проводит симуляционные тренинги, авторские и сертификационные курсы. Участвовал в переводе глоссария ITIL и международного стандарта ISO 20000.

Два года назад в Альманахе itSMF России мы писали про контрольные карты Шухарта – прекрасный инструмент статистического управления процессами, который призван помочь нам принимать взвешенные и обоснованные решения на основании проведенных измерений. Именно благодаря этому инструменту мы можем переводить данные в информацию и знания, постепенно приобретая мудрость. Но в то же время контрольные карты Шухарта являются отличным примером визуализации информации – и именно о визуализации мы поговорим в этой статье.

Лучше один раз увидеть

Визуализация данных и информации – это представление в графическом виде какого-либо текста или чисел. Но нас, в первую очередь, будут интересовать результаты различных измерений, которые обычно хранятся в виде таблиц или, возможно, даже простого текста. Наверняка многим из вас приходилось строить различные диаграммы, рисовать графики и т. д., и уж совсем точно каждый из вас так или иначе сталкивался с таким представлением информации на работе или в обычной жизни. Давайте же разберемся, зачем мы это делаем, какие интересные

инструменты для этого существуют и какие подводные камни нас могут ожидать на пути визуализации.

ЗАЧЕМ вообще все это нужно

Несмотря на то, что мало кто задается вопросом целесообразности графического представления информации, мы все-таки решили начать именно с этого – с ответа на вопрос «зачем?». Мы выделили три основные цели:

- заставить мозг работать;
- сжать большой объем данных;
- выявить то, что невозможно осознать без картинки.

Конечно, за всем этим по-прежнему стоит одна глобальная цель – принятие правильных, взвешенных, рациональных решений по итогам анализа данных. И как раз достижение трех указанных выше целей визуализации и позволит нам более эффективно проводить анализ данных.

1. Мозг

Мы привыкли думать, что у мозга есть правая и левая части, каждая из которых отвечает за какой-то свой тип мышления. Есть даже распространенные названия – логическое (аналитическое) и креативное (творческое) мышление, за каждое из которых якобы отвечает определенное полушарие. На самом деле, если верить Татьяне Черниговской, ведущему российскому ученому в области изучения мозга и сознания, «по последним исследованиям, разница между полушариями не настолько жесткая, как казалось раньше, мозг всегда все равно работает целиком, стенок, отделяющих полушария, внутри нет». Да, в мозгу есть функциональные блоки и какая-то локализация функций. Но практически в любой осмысленной деятельности принимают участие все участки мозга. Как говорит Татьяна Черниговская: «Одним словом, во время выполнения какой-либо задачи в мозге не активируется какой-то отдельный участок — мозг всегда работает весь. То есть участки, которые за что-то отвечают, вроде бы есть и в то же время их как бы нет».

Тогда почему же так важно визуализировать данные и информацию? Вот лишь несколько причин.

Эффект превосходства картинки. Зрение главенствует над всеми нашими чувствами восприятия окружающего мира. Мы прекрасно помним образы и картинки. Различные исследования показывают разные данные, но вывод всегда один – картинка действу-

ет в разы сильнее, чем текст. Почему? Потому что всю свою «человеческую» жизнь наш биологический вид добывал себе еду, искал партнера для продолжения рода и обнаруживал угрозы (чтобы спастись от них) именно при помощи зрения.

По словам доктора Джона Медины, «различные исследования показывают, что распознавание и понимание информации удваиваются, если мы используем картинки, а не текст».

Кроме того, мы должны учитывать, что мозгу требуется дополнительная работа (а значит, и потребляемая энергия), чтобы сложить буквы в слова, слова – в фразы, фразы – в текст, а потом извлечь оттуда образ и смысл. Картинка сразу дает образ. Текст просто неэффективен для нашего мозга. Поэтому в большинстве случаев, независимо от обстоятельств, люди гораздо лучше усваивают и запоминают картинки, нежели слова.

Еще одно интересное исследование, подтверждающее, что наша способность обрабатывать конкретные образы гораздо выше, нежели перерабатывать абстрактные знания, описано в книге датского научного публициста Тора Норретрандερса «Иллюзия пользователя. Преуменьшение важности сознания». В ней приводится цитата из нейрофизиологического исследования, во время которого замерялась пропускная способность различных органов чувств. Обобщенные результаты показаны на рис. 1.

Обратите внимание: данные на второй диаграмме представлены в кбит/с, что в 1024 раза быстрее, чем при измерении в бит/с, как на первой диаграмме. Получается, мы преимущественно бессознательно обрабатываем поступающие данные: бессознательная пропускная способность зрения в 256 000 раз

«Рецепт мозга выглядит так: 78% воды, 15% жира, а остальное – белки, гидрат калия и соль. Нет ничего более сложного во Вселенной из того, что мы знаем и что сопоставимо с мозгом вообще».

Татьяна Черниговская, профессор СПбГУ, нейролингвист и экспериментальный психолог, доктор филологии и биологии, член-корреспондент Норвежской академии наук.

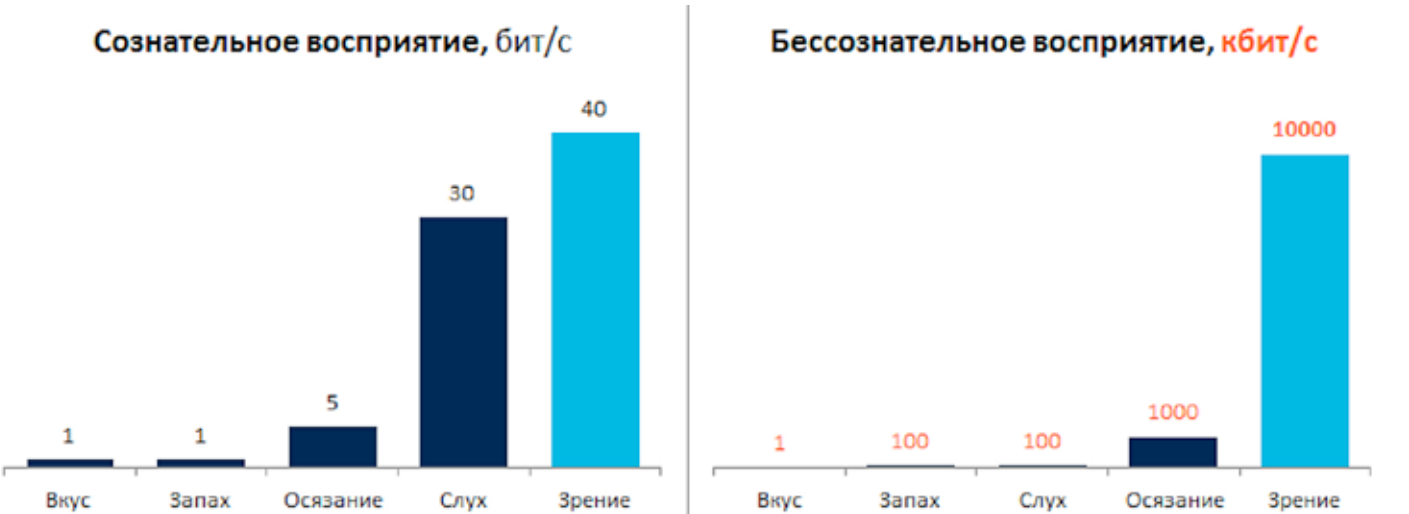


Рис. 1. Пропускная способность сознательного и бессознательного восприятия.



Рис. 2.

Динамика курса
доллара США
с 1 августа 2017 г.
по 28 августа
2018 г. (cbr.ru)

мощнее, чем сознательная составляющая этого же канала получения информации. И зрение всегда впереди остальных способов восприятия.

Вероятно, вы знаете старую английскую поговорку: «Одна картинка стоит тысячи слов». И уж точно вам знаком ее русский аналог: «Лучше один раз увидеть, чем сто раз услышать». Наглядные пособия в виде образов и картинок как раз и рассчитаны на зрительную пропускную способность, причем в большей степени на бессознательную. Текст же, который усваивается, главным образом, сознательно, мало того что перегружает мозг, так еще и поступает в него по гораздо более медленному (в 256 000 раз!) каналу.

Зрительная кора и отрезки прямых линий.

Вячеслав Дубынин, профессор МГУ, биолог, в своей лекции о нейроэстетике рассказал, что в соответствии с результатами научных исследований, проведенных еще в 1970 году¹, «первое, что узнает зрительная кора – это короткие отрезки прямых линий». То есть первое, на что реагирует наша зрительная система, – это короткие отрезки прямых под разными углами к горизонту. И что самое интересное, эта функция – не результат обучения, а врожденная особенность нейросети.

В лекции Дубынина по нейроэстетике дальше идет переход к кубизму, который может и не нравится

человеку, но все равно привлекает внимание, а для нас с вами важно, что наш мозг будет отлично реагировать и пытаться сразу обработать диаграммы (столбики, линии и т. д.). Потому что именно первичная зрительная кора скажет: «Ух ты! Прямые линии! Я хочу это рассматривать».

Эмоции. А вот это самая удивительная часть. Мы прекрасно понимаем, что картинки будят в нас эмоции, гораздо более яркие, чем текст или цифры. Эмоции могут быть сильными и не очень, плохими или хорошими, адекватными и не совсем, но главное в них то, что, оказывается, решения принимаются человеком только в процессе эмоционального переживания. То есть без эмоций наш мозг практически не способен принимать решения². «Чистый разум» Декарта не работает, увы!

Поэтому, если вы хотите, чтобы на основании ваших измерений и данных были приняты разумные, обоснованные решения – используйте хорошие и правильные картинки. Это поможет «включить» и заставить работать на полную катушку мозг тех, кто должен принимать эти решения.

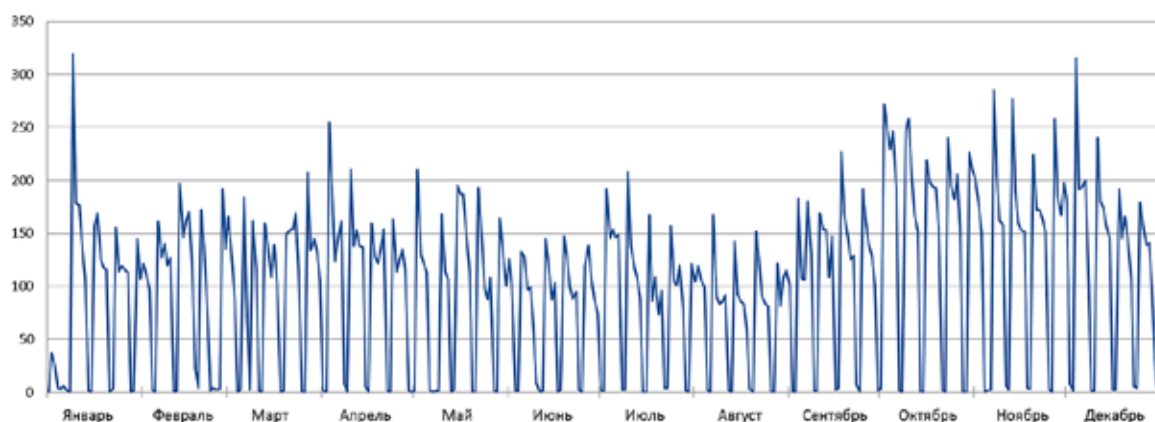
2. Большой объем данных

Нет, конечно же, речь сейчас не о big data. Давайте возьмем простой пример – курс доллара по отношению к рублю за последний год. Это более 200 точек. Если попытаться разместить все эти точки в текстовом или табличном виде, то помимо того, что этого не допустят редакторы, это все равно невозможно будет обработать/понять, просто посмотрев на эти числа – слишком много данных для нашего восприятия. А вот картинка – это наглядно и понятно (см. рис. 2).

А причем тут ITSM, процессы управления услугами и вообще ИТ? Хорошо, давайте возьмем пример из «нашей» жизни. На рисунке 3 показана ежедневная статистика по зарегистрированным инцидентам за 2017 год. Никаким другим способом, кроме как при помощи графика, охватить одним взглядом 365 точек просто не получится (см. рис. 3).

Рис. 3.

Статистика зарегистрированных инцидентов за 2017 год.



¹За это открытие авторы получили Нобелевскую премию.

²Мы намеренно не раскрываем дальше детали, так как на эту тему можно написать отдельную статью. Кто хочет углубиться в эту область, обратитесь к исследованиям доктора Антонио Дамасио.

3. Выявить невидимое

Невидимое в виде текста, цифр и таблиц и в то же время очевидное в виде графиков. Простой пример. Давайте специально рассмотрим те же данные, что были использованы в прошлой статье – это статистика вовремя решенных инцидентов, собранная за 20 недель.

Вот так она выглядит в текстовом виде:

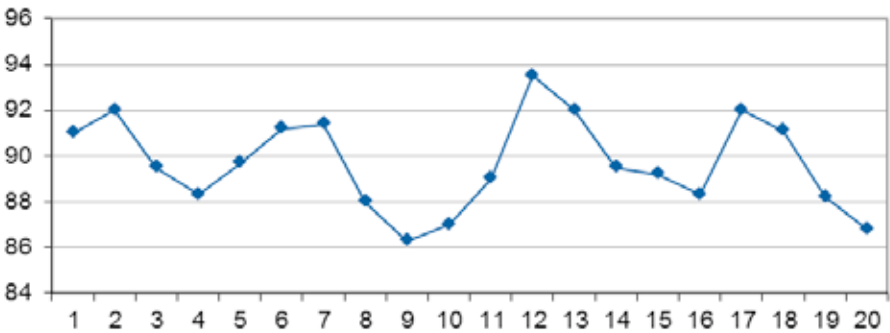
Неделя 1 – 91%; неделя 2 – 92%;
неделя 3 – 89,5%; неделя 4 – 88,3%; неделя 5 – 89,7%;
неделя 6 – 91,2%; неделя 7 – 91,4%; неделя 8 – 88%;
неделя 9 – 86,3%; неделя 10 – 87%; неделя 11 – 89%;
неделя 12 – 93,5%; неделя 13 – 92%; неделя 14 – 89,5%;
неделя 15 – 89,2%; неделя 16 – 88,3%; неделя 17 – 92%;
неделя 18 – 91,1%; неделя 19 – 88,2%; неделя 20 – 86,8%.

А то, как эта же статистика представляется в табличном виде, показано в таблице 1. Скажите, удастся понять какие-то тренды, выявить максимум/минимум? Думаю, что нет.

Таблица 1. Процент вовремя решенных инцидентов, статистика за 20 недель.

Неделя 1	Неделя 2	Неделя 3	Неделя 4	Неделя 5
91	92	89,5	88,3	89,7
Неделя 6	Неделя 7	Неделя 8	Неделя 9	Неделя 10
91,2	91,4	88	86,3	87
Неделя 11	Неделя 12	Неделя 13	Неделя 14	Неделя 15
89	93,5	92	91	89,5
Неделя 16	Неделя 17	Неделя 18	Неделя 19	Неделя 20
88,3	92	91,1	88,2	86,8

А теперь обратимся к графику (рис. 4) – сразу становится понятно, как работает этот процесс. По крайней мере, нам так кажется. На самом деле, чтобы понять, что происходит и хорошо это или плохо, надо построить контрольную карту Шухарта, но об этом мы уже рассказывали в деталях в ста-



тье «Когда пора сбивать высокую температуру? Умеем ли мы принимать правильные решения» в Альманахе 2016 года.

Кстати, этот график называется **«график хода процесса»** – пожалуй, самый простой и самый недооцененный инструмент визуализации. Но при этом один из самых важных. Не знаете, какую картинку построить – начните с графика хода процесса. И если вы обратите внимание на график на рис. 3, то сразу увидите, что данные на нем неоднородные (постоянные провалы в ноль – это выходные и праздники). При дальнейшем анализе данных целесообразно сначала разделить данные (провести стратификацию) на будние дни и выходные, а уже затем работать по отдельности с каждым массивом.

А вот еще интересный пример по тем же данным, что использованы в рис. 3: на одном графике отображено количество инцидентов в неделю (синяя линия) и среднее время решения одного инцидента за эту же неделю (оранжевая линия) – см. рис. 5.

Вот и думайте теперь, почему при снижении количества инцидентов во вторую неделю июня среднее время решения выросло в 2-3 раза. С другой стороны, за исключением майских праздников, летних отпусков (июнь-июль) и новогодних каникул, когда достаточно большое количество персонала уходит в отпуск, на протяжении остального года картина выглядит достаточно ровной и стабильной.

Рис. 4. Процент вовремя решенных инцидентов, статистика за 20 недель. График хода процесса.

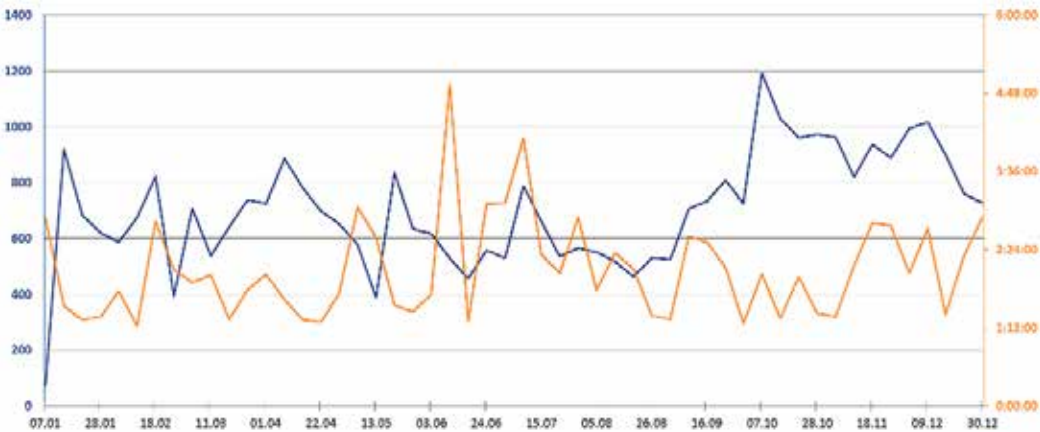


Рис. 5. Количество зарегистрированных за неделю инцидентов и среднее время решения инцидента.

ЧТО СТОИТ ИСПОЛЬЗОВАТЬ

Все мы знаем про линейные графики, столбиковые и круговые диаграммы и многие другие простейшие инструменты визуализации. Но есть гораздо менее распространенные и при этом очень полезные, эффективные и совсем несложные инструменты, которые позволяют легче анализировать данные в определенных условиях.

Рассмотрим три из них:

- диаграмма Парето;
- гистограмма;
- диаграмма «ящики с усами».

1. Диаграмма Парето

Знаменитое правило 80/20 у всех на слуху, многие отлично его понимают и применяют на практике. Но, как показывает опыт, мало кто умеет строить диаграмму Парето, основывающуюся на этом правиле.

Итак, диаграмма Парето – это визуальный метод ранжирования приоритетов, основанный на правиле 80/20: только 20% возможных причин приводят к 80% конечных результатов, поэтому именно

этим 20% и стоит заниматься в первую очередь. Остальные нужно отложить.

Диаграмму Парето можно строить по любым данным, если есть какие-либо категории группировки данных. Чтобы построить диаграмму Парето, категории нужно упорядочить по убыванию числа объектов в каждой категории. Далее по этим убывающим числам строится простая столбиковая диаграмма. Затем необходимо посчитать, какой процент от полной суммы всех объектов составляет каждый столбик, и по этим процентам построить кривую накопленного процента: то есть процент, который составляет первая категория, потом процент, который составляют первая и вторая категории вместе, потом первая, вторая и третья категории вместе и т. д. Эта кривая обязательно должна закончиться в точке 100%. Для ломаной накопленного процента используется независимая шкала (ось графика): слева обычно располагается шкала, по которой построены столбики с абсолютными значениями, а справа – по которой строятся точки накопленных процентов.

Рассмотрим два примера диаграммы Парето (рис. 6 и 7). В диаграмме Парето для расходов на авто-

Рис. 6.
Диаграмма Парето
для расходов
на автомобиль
по типу расходов.

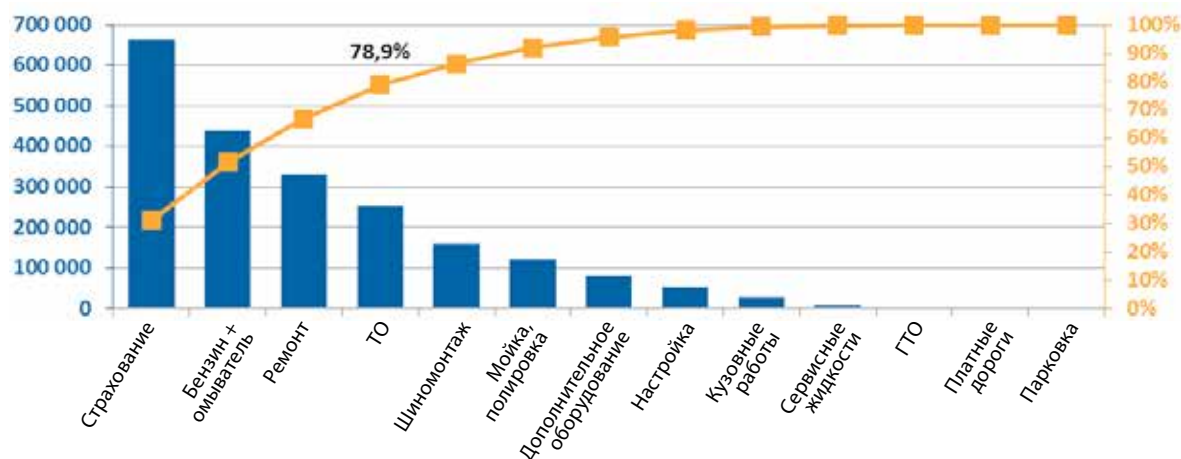
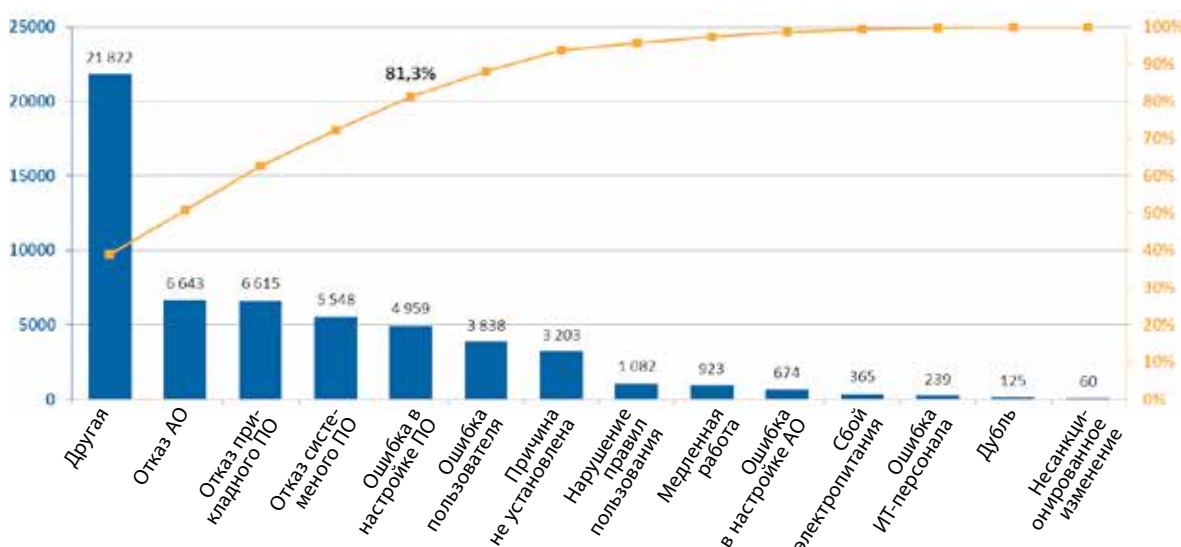


Рис. 7.
Диаграмма Парето
для причин
возникновения
инцидентов.



мобиль по типу расходов, наверное, нет ничего удивительного – 80% расходов составляют страховка, топливо, ремонт и ТО (рис. 6).

Теперь посмотрим на пример из области ITSM – процесс управления инцидентами, показатель «причина возникновения инцидента» (рис. 7, он построен по тем же данным, что были использованы для рис. 3 и 5). По этому графику мы уже прекрасно понимаем, что, скорее всего, наши ИТ-сотрудники просто не очень хотят разбираться в причине и заполнять это обязательное поле корректно. Потому что причина «Другая» в разы превосходит остальные. И это при том, что есть вариант «Причина не установлена». Поэтому наша первоочередная задача – добиться корректного заполнения этого поля, а уже затем можно смело браться за нашу инфраструктуру и прикладное ПО (если, конечно, расклад не поменяется кардинально после распределения «Других» по остальным категориям) – так мы избавимся от наибольшей части инцидентов. После чего уже можно будет переходить к работе с пользователями. Хотя если объединить категории «Ошибка пользователя» и «Нарушение правил пользования», то новая категория почти догонит категорию «Ошибка в настройке ПО». Но на общий расклад это никак не повлияет.

2. Гистограмма

Это столбиковая диаграмма, но не совсем обычная. К сожалению, часто под видом гистограмм нам показывают просто столбиковые диаграммы, что приводит к искажению информации. Но об этом мы поговорим в конце статьи. А пока вернемся к настоящим гистограммам.

В гистограмме высота столбиков равна числу объектов, попадающих внутрь границ данного столбика – тут все как обычно. А вот дальше начинаются особенности. Во-первых, гистограмму можно строить, если у вас есть более 100 точек. Во-вторых, чтобы построить гистограмму, нужно сначала выбрать, сколько столбиков в ней будет. Рекомендуется строить гистограммы с числом столбиков больше 5 и меньше 20. Внутри этого диапазона нужно выбирать так, чтобы границы интервалов были удобными. И главное – гистограмма строится с **равными значениями ширины** столбиков³. Если это правило не соблюдать, картинка будет обманывать.

Для примера мы снова вернемся к затратам на автомобиль. На рисунке 8 показана гистограмма ежемесячных расходов (построенная по данным из рис. 6).

По оси абсцисс указаны равные интервалы в тысячах рублей. Получается, что 60 месяцев из всей

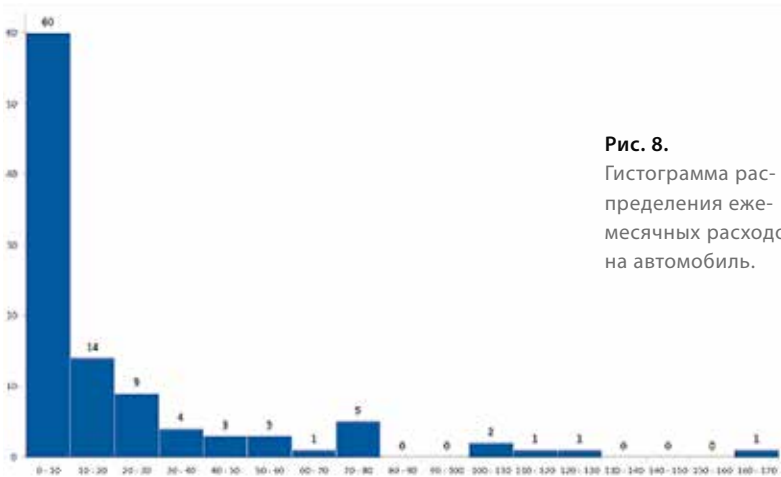


Рис. 8.
Гистограмма распределения ежемесячных расходов на автомобиль.

выборки на машину тратили от 0 до 10 тыс. руб., 14 месяцев – от 10 до 20 тыс. руб. и т. д. Что видно из такого графика? Во-первых, распределение вообще несимметрично – ежемесячно на автомобиль расходуются небольшие суммы, но очень редко бывают месяцы со значительными затратами. И вот этот так называемый хвост дает достаточно большое среднее значение – 20 408 руб., что попадает в третью колонку на нашем графике. Во-вторых, тут и в помине нет никакого нормального распределения, поэтому многие аналитические модели, базирующиеся на нем, просто не будут работать в этом случае.

Давайте снова вернемся к управлению инцидентами и построим гистограмму по статистике времени решения инцидентов за год (рис. 9). Это все те же самые данные, что были использованы для рис. 3, 5 и 7. Для наглядности были взяты данные только для инцидентов, решенных менее чем за двое суток.

График показывает не только то, что большинство инцидентов устраняются в первые два часа, но и интересное волнообразное распределение с суточным интервалом. В данной организации это связано с тем, что ИТ-сотрудники работают только в офисное время, поэтому те инциденты, которые не были устранены в первый же рабочий день, откла-

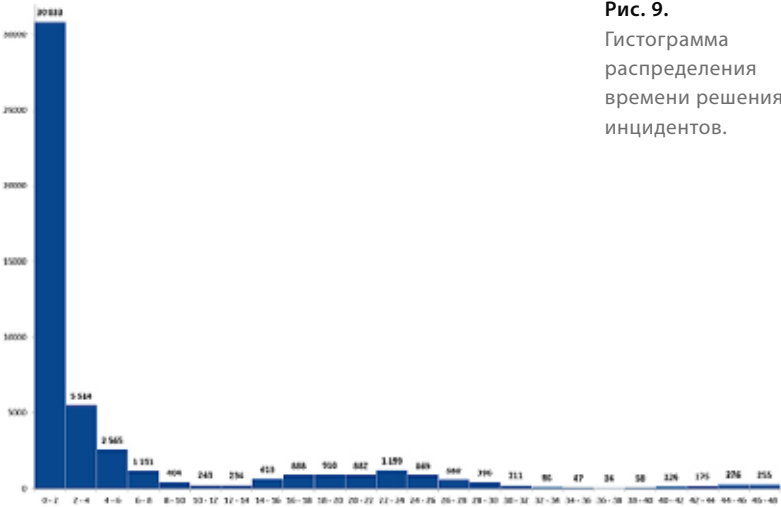


Рис. 9.
Гистограмма распределения времени решения инцидентов.

дываются на следующий, затем еще на один – и т. д., пока не будут решены. Для этих данных отдельный интерес может представлять гистограмма, построенная по первым двум и даже только по первому часу решения инцидентов. Но рассмотреть все возможные варианты в рамках данной статьи просто не представляется возможным.

3. «Ящики с усами»

Это, пожалуй, самый сложный инструмент из рассматриваемых в статье. Но так как последние версии MS Excel, а уж тем более все статистические пакеты ПО, справляются с построением этой диаграммы автоматически, то сложность данного инструмента ни в коей мере не должна вас пугать.

«Ящик с усами» (ЯСУ) – это диаграмма, показывающая вариабельность (разброс) данных в грубом приближении с помощью пяти реперных точек. Чтобы

построить диаграмму ЯСУ, сначала нужно упорядочить данные. Далее в уже упорядоченных данных надо найти медиану – число, стоящее посередине упорядоченного ряда. Если число данных четное, то медиана равна полусумме двух чисел, стоящих посередине.

Рассмотрим пример: пусть у нас есть следующий набор чисел {3, 11, 2, 7, 15}. Чтобы найти медиану, упорядочим их: 2, 3, 7, 11, 15. Медиана равна 7.

Если бы в исходном наборе не было числа 15, то упорядоченный ряд имел бы вид: 2, 3, 7, 11. Тогда медиана была бы равна $(3+7)/2 = 5$.

Далее необходимо найти медианы каждой половины данных – это будут первая и третья квартили, т. е. одна четвертая часть данных и три четверти данных. Их принято обозначать Q1 и Q3. Квартили дают нам значения границ ящика, медиана лежит где-то внутри ящика, а усы при самом простом способе построения соответствуют минимальному и максимальному значениям наших данных.

Диаграмма «ящик с усами» обладает рядом таких качеств, которые делают ее незаменимым инструментом анализа любых данных. Основные достоинства этой диаграммы:

1. Этот инструмент незаменим для сравнения нескольких выборок. Например, мы хотим сравнить вариабельность за год с помощью помесечных данных. Сравнение 12 гистограмм даст только один результат: головную боль и отчаяние из-за безнадёжности задачи. Или другой вариант – хотим сравнить нагрузку на первую линию с точки зрения количества обращений, приходящих каждый день. Только за одну рабочую неделю это будет пять графиков, а за месяц – уже больше 20. И без диаграммы ЯСУ тут никак не справиться.
2. Диаграмму «ящик с усами» можно построить любому количеству данных больше трех. Гистограмму не имеет смысла строить при числе данных меньше 50, но лучше иметь не менее 100 точек.

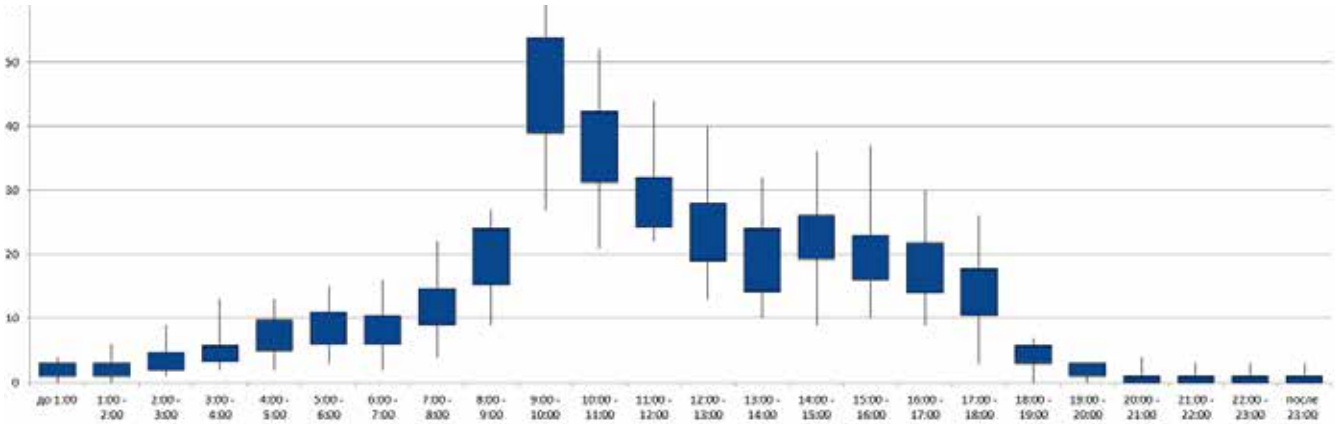
Медиана и среднее

Обратите внимание, что медиана – это срединное, а не среднее значение. То есть это такое число в выборке, что ровно половина из элементов выборки больше него, а другая половина – меньше. Медиана и среднее – это разные величины, имеющие разное смысловое наполнение. Например, для гистограммы на рис. 8 среднее равно 20 409 руб., а медиана – 6 991 руб. Для гистограммы на рис. 9 среднее равно 5 ч 14 мин, в то время как медиана – 1 ч 5 мин.

Каждое из этих чисел рассказывает что-то про наши данные, и дальше весь вопрос в том, что мы хотим узнать и для чего. Например, для данных по времени решения инцидентов медиана со значением 1 ч 5 мин будет говорить о том, что половину инцидентов за год (из выборки инцидентов, решенных в срок менее 2 суток) ИТ-сотрудники решили не дольше чем за 1 ч 5 мин, а еще половину – не меньше чем за 1 ч 5 мин. А половина в данном случае – это 24 274 инцидента. Согласитесь, это дает чуть лучшее представление о времени решения инцидентов, нежели среднее значение по тем же данным.

Рис. 10.
Количество зарегистрированных инцидентов с разбивкой по часам за одну рабочую неделю.





3. Диаграмма ЯСУ – это единственный инженерный способ анализа данных на наличие выбросов, не требующий никаких специальных критериев и статистических таблиц.

Давайте рассмотрим пример диаграммы «ящик с усами» как раз для определения нагрузки на первую линию поддержки в части инцидентов в зависимости от времени суток. Для этого возьмем данные за октябрь 2017 г. (это те же данные, что ранее использовались для рис. 3, 5, 7 и 9) и посмотрим, какое количество инцидентов регистрируется каждый час в течение дня – и так для всей рабочей недели (рис. 10).

На этой диаграмме, с одной стороны, достаточно хорошо видны общие тренды (пиковая нагрузка, спад обращений в обеденное время и т. д.), с другой стороны, даже пять кривых на одном графике – это явный перебор. Да и контрастные цвета уже почти закончились. А теперь представьте, что мы хотим проанализировать картину за весь октябрь, а это 22 кривых.

Но мы вполне можем это сделать при помощи диаграммы «ящик с усами» (рис. 11). Область внутри ящиков показывает нам половину всех точек для каждого часа из 22 дней. Остальные 50% – это усы снизу и сверху от ящика (по 25% точек соответственно). И такую картинку наш мозг вполне способен воспринять. Для того чтобы, например, в дальнейшем принимать решение, какое количество сотрудников в какое время должно быть в нашей техподдержке для максимально быстрого устранения инцидентов.

«Ложь, наглая ложь и статистика»

Картинки – это очень сильный инструмент воздействия. И если правильные картинки помогают нам принимать правильные решения, то неправильные картинки могут как сбить с толку и запутать, так и сильно навредить и привести к принятию неправильных решений.

Вообще, если вы хотите глубже изучить эту тему, то самым правильным источником информации, я даже не побоюсь этого слова – первоисточником, будет книга Даррелла Хаффа «Как лгать при помощи статистики». Мы же с вами рассмотрим всего несколько примеров неправильных картинок.

1. Вредные советы

Начнем с самых простых инструментов визуализации. Хотите сделать так, чтобы никто не понял, что изображено на картинке, где какие данные и как они соотносятся друг с другом? Используйте трехмерные графики (рис. 12). Добавьте прозрачности. И побольше, побольше разных массивов данных!

Это, кстати, те же данные, что были использованы для рис. 10 и 11 (выборка за две недели). Почему-то многие думают, что любовь к трехмерным играм и трехмерному кино обязательно должна экстраполироваться на диаграммы в отчетах. В лучшем случае это может вызвать снисходительную улыбку, в худшем – недоумение и раздражение руководства.

Или рассмотрим столбиковые диаграммы. Казалось бы, уж их-то вряд ли можно чем-то испортить, если не использовать 3D. Но нет ничего невозможного.

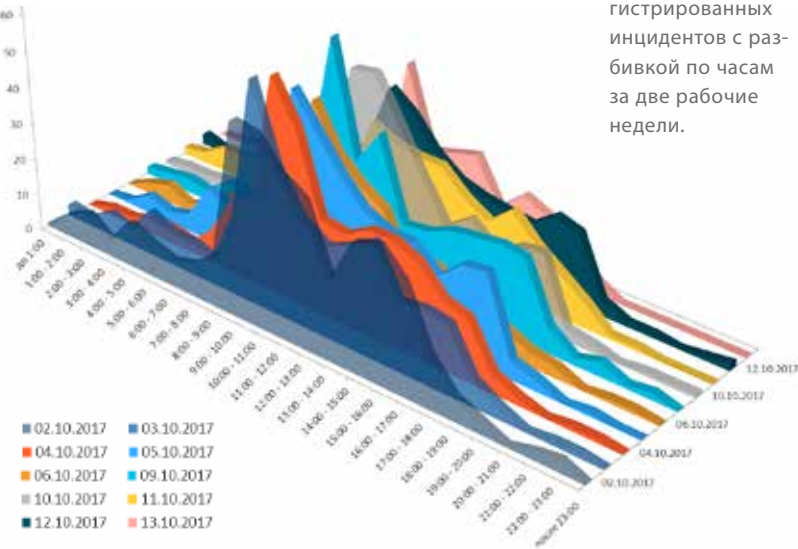


Рис. 11.
Количество зарегистрированных инцидентов с разбивкой по часам за октябрь 2017 г. в рабочие дни.

Рис. 12.
Количество зарегистрированных инцидентов с разбивкой по часам за две рабочие недели.

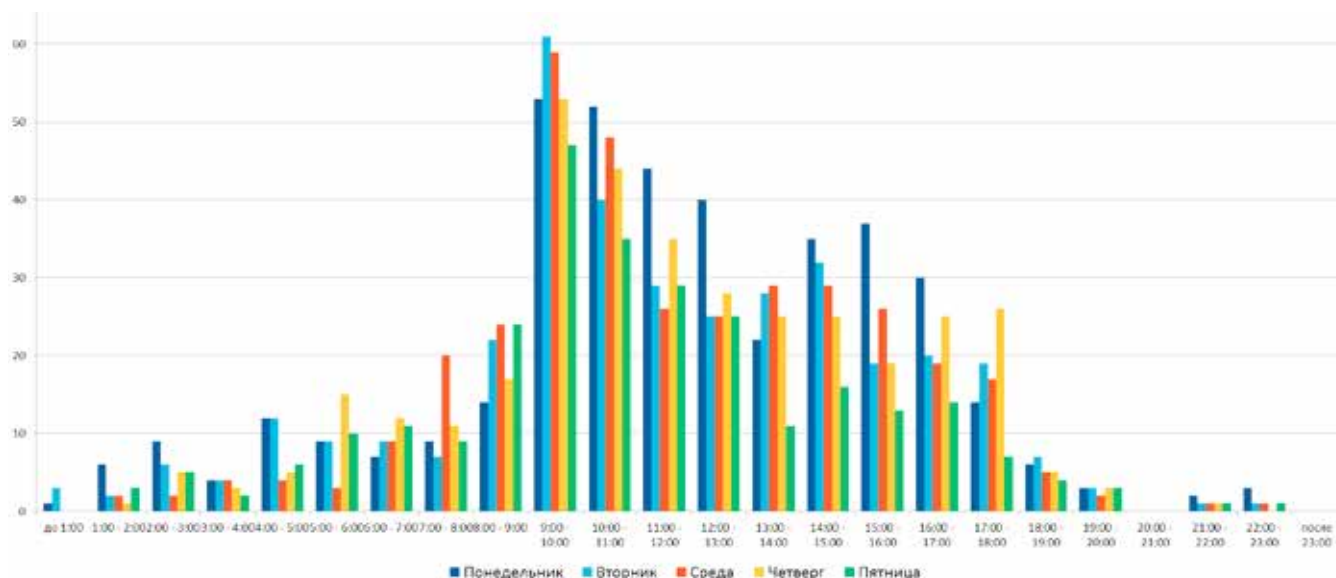


Рис. 13. Количество зарегистрированных инцидентов с разбивкой по часам за одну рабочую неделю.

можно, благо нам с радостью подсовывают вот такие шаблоны (рис. 13). А это точно те же данные, что и на рис. 10.

Скажите спасибо, что мы не стали использовать объемные конусы или пирамиды с вертикальным распределением всех массивов данных в одном объекте. Помните принцип «**Keep it simple**»? Так вот, он очень важен во всем – в том числе и в вопросах визуализации.

2. Обман без обмана

Мы рассмотрели примеры диаграмм, которые вообще не стоит использовать (к ним также относятся круговые диаграммы и в некоторых случаях пузырьковые⁴). Но даже простым линейным графиком из двух точек и одной кривой можно изрядно озадачить читателя. Или попросту обмануть. Давайте попробуем: на рисунке 14 показан график снижения среднего времени решения инцидентов.

Казалось бы, отличный прогресс! Но наблюдательный читатель сразу обратит внимание на ось ординат. И с легкостью построит правильный график по тем же данным (рис. 15). И сразу становится ясно, что картина

не такая уж и радужная, а наши успехи не столь значительны, как могли показаться на рис. 14.

Справедливости ради стоит заметить, что на первом графике (рис. 14) нас вроде бы даже и не обманывают – мы же видим шкалу и значения по оси ординат. Только вот большинство, увидев значительный спад на самой картинке, порадуется ему и вряд ли будет внимательно изучать диаграмму и искать точку отсчета (ноль) на этом графике. Потому что картинка – это сильный инструмент воздействия. И не все пользуются им правильно и честно – кто-то намеренно, а кто-то просто по незнанию или неопытности.

3. Большой «обман»

Ну и, наконец, обещанный в середине статьи пример гистограммы, которая вовсе не гистограмма (рис. 16). Что самое обидное – это официальная статистика⁵, которая преподносится нам, чтобы нас как-то подбодрить и поддержать. Давайте построим этот же график, убрав все лишнее (рис. 17).

А что будет, если эти данные превратить в настоящую гистограмму (с допущением, что внутри ука-

Рис. 14. График снижения среднего времени решения инцидентов. Сравнение 2016 и 2017 гг.

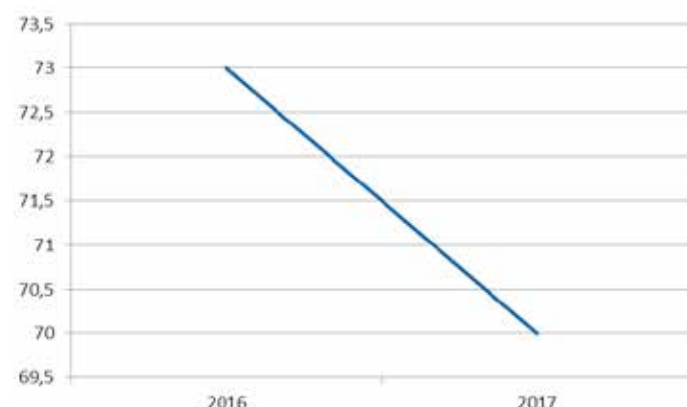
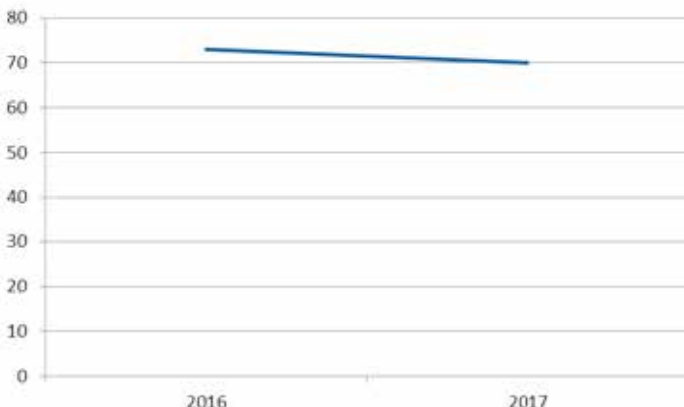


Рис. 15. График снижения среднего времени решения инцидентов. Сравнение 2016 и 2017 гг. Правильный график.



⁵<https://www.vedomosti.ru/economics/articles/2017/04/04/684008-rabotniku-ne-hvataet-na-edu>

занных выше «двойных» и пограничного интервалов распределение процентов равномерное)? Результат показан на рисунке 18. Выводы по этим графикам делайте сами. Главное – используйте для этого гистограмму, а не то, что приводится первоисточником в качестве нее.

КАК использовать картинки во благо, при этом не наступать на грабли и не стать жертвой обмана

Действительно, картинки имеют великую силу влияния на людей. И мы должны уметь ее использовать – но использовать правильно и честно. Поэтому:

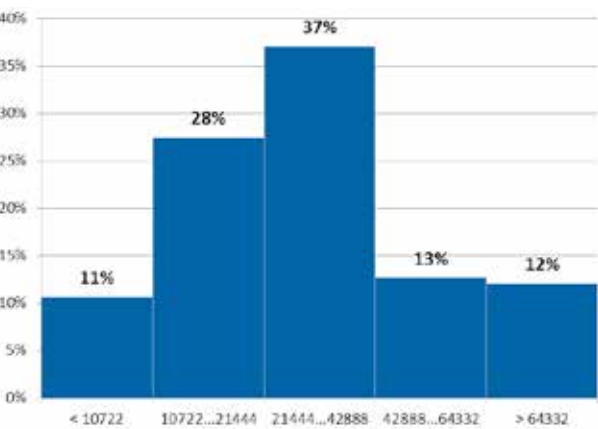
- 1. Тратьте время на анализ данных и их визуализацию! **Картинки всегда сильнее текста и таблиц.**
- 2. Используйте простые и эффективные инструменты. Не забывайте про простейший – **график хода процесса!**
- 3. Работайте аккуратно – **не обманывайте** себя и других. И не давайте обманывать себя!

Как не дать себя обмануть? Наверное, так же, как и в обычной жизни: используйте здравый смысл и критическое мышление. Плюс тренируйтесь самостоятельно анализировать данные и строить правильные картинки. Если вы будете хорошо понимать, как это работает изнутри, никто не сможет ввести вас в заблуждение.

Начинать анализ любой картинки, которую вам предлагают, всегда можно с самых простых вещей. Приведем лишь небольшой список того, что стоит держать в голове:

- Корреляция не означает причинно-следственную связь.
- Причина может казаться следствием, и наоборот – используйте системный подход для анализа.
- Соединение несовместимых понятий, объектов, сущностей всегда приводит к неверным выводам.
- Сложение процентов, процентных пунктов, манипуляции с индексами – повод задуматься.

Рис. 17. Распределение наёмных работников по зарплате в 2016 г., %. Просто столбиковая диаграмма.



- Использование понятия «среднее» при неравномерном распределении (а если вам неизвестно распределение, то вы не имеете право считать его равномерным) – повод углубиться в первичные данные.
- Отсчет осей не от нуля без соответствующего пояснения – надо быть начеку.
- Излишняя точность (за пределами сугубо научной литературы): 2–3 знака после запятой – удобная маскировка обмана.

На самом деле, чтобы не стать жертвой статистической (статической манипуляции), вам нужно всего лишь быть внимательным и уметь задавать (себе или выступающему) правильные вопросы. Не забывайте об этом!

И помните, **картинки не дают нам ответов** на вопросы, что происходит в процессе. Они всего лишь позволяют глубже его понять и лучше проанализировать. И именно для этого они и нужны. ♦

Рис. 18. Распределение наемных работников по зарплате в 2016 г., %. Гистограмма.

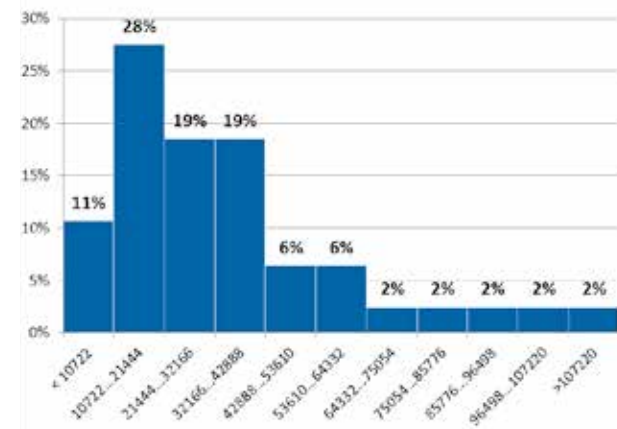


Рис. 16. Распределение наемных работников по зарплате в 2016 г., %. Исходная картинка.



Алексей Поперлюков

Генеральный директор «ИТСК». С 2009 года работает в структурах «Газпром нефти». Прошел путь от специалиста технической поддержки до директора по сервисам «ИТСК». С 2014 по 2017 годы работал в должности начальника управления общекорпоративных сервисов департамента информационных технологий, автоматизации и телекоммуникаций «Газпром нефти». С декабря 2017 года генеральный директор «ИТСК».

По мере роста наукоемкости технологий нефтедобычи все большую роль в повышении эффективности бизнеса играет ИТ-функция. С каждым годом увеличивается количество информационных систем, ИТ-услуг, растут требования к эффективности специалистов и срокам решения задач. Во многих компаниях сегодня как никогда остро стоит вопрос обеспечения высокого уровня предоставления ИТ-услуг. Вместе с тем специалисты по ИТ-сервису, достигнув определенного уровня качества в своей работе, в какой-то момент обнаруживают, что предел возможностей исчерпан. Что делать в этой ситуации и какие организационно-технические меры можно предпринять для перехода на следующий этап развития? Расскажем об опыте компании «ИТСК», дочернего ИТ-подразделения «Газпром нефти».

Новые подходы к развитию ИТ-сервисов в «Газпром нефти»

Программа «7+1»

В последние годы качество оказания ИТ-услуг специалистами «ИТСК» пользователям «Газпром нефти» достигло такого состояния, когда согласованные ранее показатели SLA практически утратили свою информативность – их значения приблизились к максимальным, а изменения от месяца к месяцу оказались столь незначительными, что оценка по ним перестала давать ИТ-специалистам информацию к размышлению и действиям.

Для дальнейшего развития ИТ-сервиса требовались новые подходы. Так в нашей компании родилась идея создания стримов, которые смогли бы обозначить новые направления движения. Таких стримов наша экспертная команда нашла восемь: семь регулярно измеряемых в натуральных показателях и один в финансовых:

- введение интегрального показателя качества сервиса;
- повышение удовлетворенности пользователей;

- снижение числа обращений пользователей;
- развитие сервисов самообслуживания;
- снижение среднего времени решения обращений пользователей;
- снижение количества инцидентов в бизнес-услугах;
- повышение производительности труда специалистов компании;
- сокращение удельных затрат на оказание услуг.

Таким образом, была сформирована программа, которую мы назвали «7+1». По каждой из обозначенных целей были созданы паспорта расчета показателей, разработана и введена в работу система аналитической отчетности «Айсберг», сформирована команда, в которую вошли лидеры по направлениям, энтузиасты и «волонтеры», заинтересованные в реализации намеченных инициатив. Всего в рамках программы «7+1» было выполнено порядка ста организационно-технических мероприятий.

Важным элементом стало создание управляющего комитета программы «7+1», состоящего из руководителя, ответственных по целям, приглашенных экспертов. Регулярные встречи всех участников реализации программы, открытые обсуждения и обмен мнениями, координация общих усилий позволили нам в короткие сроки существенно продвинуться в достижении наших целей.

Ниже мы подробнее остановимся на этих направлениях и мерах, которые предпринимались для достижения целевых показателей в каждом из них.

Введение интегрального показателя качества сервиса

Прежде всего, был введен интегральный показатель качества оказания ИТ-услуг. Он включает в себя количественный показатель претензий по качеству оказываемого сервиса, SLA по инцидентам, запросов на обслуживание и стандартным изменениям. Формула расчета интегрального показателя:

$$M_{\text{инт.цел}} = \sqrt[N]{M_{1\text{цель}} * M_{2\text{цель}} * M_{3\text{цель}} * M_{4\text{цель}}}$$

Где:

M1 – метрика «количество претензий» (пример приведен в таблице 1);

M2 – метрика «своевременность решения инцидентов»;

M3 – метрика «своевременность выполнения запросов на обслуживание» (request to fulfill);

M4 – метрика «своевременность решения стандартных (normal) изменений».

Для достижения целевого значения этого показателя принято решение о создании в компании единого

Количество обоснованных претензий за отчетный период	% выполнения обязательств по метрике за отчетный период
0–1	100
2–4	70
5–7	40
8–10	10
Более 10	0

Таблица 1.
Метрика «количество претензий».

органа управления сервисом – «Сервисного офиса». В корпоративной системе управления ИТ-сервисами «Газпром нефти» проведена доработка для обеспечения контроля уровня услуг – создан модуль «Мероприятия по улучшению уровня услуг», что позволило значительно сократить разрыв между ожидаемым

Перед «ИТСК» стоит задача унификации подходов к управлению ИТ-услугами с учетом географической распределенности подразделений и ресурсов компании, специфики ключевых направлений деятельности, особенностей формирования ИТ-бюджетов.

и фактическим качеством обслуживания и сфокусироваться на таких вопросах, как обучение пользователей, системное тестирование, обеспечение документационной поддержки. Инициировать создание подобных мероприятий сможет любой сотрудник подразделения, осуществляющего поддержку, а также любой представитель заказчика. Саму функцию контроля за своевременным исполнением мероприятий возложили на ответственного сервис-менеджера за услугу.

Повышение удовлетворенности пользователей

Для повышения эффективности работы в данном направлении был изменен подход к расчету показателя удовлетворенности пользователей (customer satisfaction index, CSI). Формальный опрос представителей заказчиков о качестве предоставляемых ИТ-услуг был заменен на показатель, который состоит из доли обращений, которые оценены непосредственными потребителями услуг, и среднеарифме-

тической оценки, ими выполненной по обращению работы. Целевое значение показателя удовлетворенности пользователей – не менее 4,8.

К концу 2017 года каждое седьмое обращение автоматизировано, каждый четвертый запрос на обслуживание шаблонизирован, а удельная стоимость затрат на предоставление ИТ-услуг вместо запланированных 2% снижена на целых 7%.

Затем были созданы условия для развития компетенций в Единой корпоративной диспетчерской службе, а также выделено специальное подразделение для проведения независимого аудита качества предоставления сервисов и регулярной отчетности службам заказчиков и сервисному офису. Кроме того, для достижения данной цели была доработана корпоративная система управления ИТ-сервисами «Газпром нефти» и проведена автоматизация работы с негативными оценками пользователей.

Снижение числа обращений пользователей

Важное направление движения вперед – постепенное уменьшение числа обращений пользователей (до 5% в год) за счет автоматизации предоставления сервисов, согласований, шаблонизации запро-

сов пользователей. Для этого был проведен аудит пула обращений по 20 услугам, в которых отмечено максимальное количество обращений по темам SAP, Web-системы и СЭД.

Приняты следующие меры:

- согласование и предоставление доступа к бизнес-приложениям переведены в систему IDM;
- разработан и внедрен процесс «Управление знаниями», на портале самообслуживания создан соответствующий модуль, а также открыт доступ к документации по услугам (с разделением уровней доступа от пользователей до компетентных специалистов).

Развитие сервисов самообслуживания

В этом направлении мы стремились к тому, чтобы более 70% бизнес-услуг были обеспечены сервисами самообслуживания на этапе проектирования. Для этого в 2017 году был создан портал самообслуживания пользователей ИТ-услуг в «Газпром нефти», который позволяет консолидировать все ИТ-сервисы на одной площадке с личным кабинетом. В личном кабинете пользователя «Газпром нефти» можно получить необходимую информацию, начиная с показателей SLA, количества обращений и заканчивая количеством отпечатанных копий на принтерах. Кроме того:

- сервисы структурированы на управляемые и неуправляемые;
- функция согласования и предоставления доступа к ресурсам, системам была передана в IDM.

Снижение среднего времени решения обращений пользователей

По нашему мнению, один из важных показателей для повышения эффективности ИТ-сервисов – доля

Портал самообслуживания пользователей ИТ-услуг «Газпром нефти»

Портал самообслуживания пользователей «Газпром нефти» формирует единое информационное пространство для взаимодействия пользователей ИТ-услуг со службой технической поддержки, а также интеграции информации из различных ИТ-систем, получения сведений по ИТ-активам, нормативно-методической документации, отчетности, данных из базы знаний и других.

Портал рассчитан на широкую аудиторию – от специалистов, пользователей корпоративных услуг до руководителей высшего звена. Он оснащен удобным интерфейсом и широким функционалом. Для каждого пользователя создан личный кабинет, из которого он может управлять своими ИТ-услугами, используя типовые шаблоны заявок: заказывать ИТ-оборудование и ПО, корпоративную связь, запрашивать доступы, проверять наличие числящегося за ним аппаратного и программного обеспечения и контролировать ход оказания услуг. Также на портале размещены выпуски издания IT-News, из которого пользователи могут узнать новости о развитии информационных технологий в «Газпром нефти».

Что можно делать с помощью портала самообслуживания пользователей?

- отправлять запросы в поддержку и обращения различного типа (инцидент, запрос на обслуживание, запрос на изменение);
- получать информацию о ходе обработки своих заявок;
- участвовать в согласовании обращений;
- оценивать работу службы поддержки и качество услуг;
- проводить аудит активов различного типа (программное и аппаратное обеспечение), находящихся во владении;
- получать информацию о проводимых плановых работах.

В перспективе планируется объединить портал самообслуживания с базой знаний, а также реализовать такие функции, как самообслуживание специалиста и расширение отчетности по сервисам.

Центр управления ИТ-операциями «УПитер»

Цель создания Центра управления ИТ-операциями – объединение на одной площадке всех инструментов мониторинга и управления ИТ-услугами для повышения качества ИТ-сервисов и эффективности использования ИТ-инфраструктуры компании. Это позволило повысить доступность и надежность ИТ-услуг, автоматизировать операции, повысить производительность систем и персонала, а также уменьшить операционные расходы.

В центре организованы рабочие места для оперативного дежурного, лидеров по направлениям и бизнес-приложениям. «УПитер» имеет прямую видеосвязь с омским подразделением Центра управления инфраструктурой и сервисами «Газпром нефти», что позволяет мгновенно реагировать на события и принимать оперативные управленческие решения.

В результате создания центра «УПитер»:

- время обнаружения критичных инцидентов сократилось на 33%;
- время восстановления сервиса сократилось на 17%;
- время перевода критичных инцидентов в статус «устранен» сократилось на 68%;
- время прерывания сервиса сократилось на 32%.

В настоящее время на базе «УПитера» ведется разработка чат-бота для оперативной поддержки ключевых ИТ-пользователей по базовым запросам. Еще одна из перспективных инициатив – создание виртуальных комнат для консультации пользователей по бизнес-приложениям.

обращений, которые были приняты в течение первых 30 минут. Для его снижения мы организовали ранжирование обращений по сложности и обеспечили автоматическую маршрутизацию обращений на специалистов, минуя маршрутизацию, осуществляемую функцией service desk. Вместе с этим ведется постоянная работа по постепенному увеличению доли обращений, решаемых на первой линии техподдержки.

Снижение количества инцидентов в бизнес-услугах

Цель по данному направлению – доля инцидентов и средневзвешенный показатель количества инцидентов и систем в общем количестве оказываемых услуг не должны превышать 5%. Для выполнения этой задачи в конце 2017 года был создан Центр управления ИТ-операциями «УПитер». В этом центре на одной площадке объединены функции управления ИТ-процессами, мониторинга ИТ-услуг и управления ИТ-инфраструктурой. Под одной крышей собраны лидирующие эксперты по бизнес-приложениям и направлениям работы в области ИТ.

Были перезапущены процессы управления критичными инцидентами и проблемами. Вместо привычных эшелонированных команд поддержки в центре «УПитер» использована новая модель формирования команд, получившая название «рой» (Swarming). За период менее года эта модель формирования команд показала отличные результаты, снизив показатель среднего времени восстановления услуг после сбоя.

Центр «УПитер» призван не только управлять текущей ситуацией с предоставлением ИТ-сервисов, но и заранее планировать работу и развитие всех направлений ИТ. Такой подход позволил повысить качество ИТ-сервисов и эффективность использования ИТ-инфраструктуры компании.

Повышение производительности труда специалистов компании

Показатель производительности труда специалистов компании является одним из ключевых параметров, отражающих эффективность деятельности производственных подразделений. Кроме того, показатель отражает уровень зрелости управления производственными процессами. Целевой показатель – повышение производительности труда специалистов компании не менее чем на 5% в год.

В результате создания центра «УПитер» время обнаружения критичных инцидентов сократилось на 33%, время восстановления сервиса сократилось на 17%, время перевода критичных инцидентов в статус «устранен» сократилось на 68% и время прерывания сервиса на 32%.

Основой для оценки уровня производительности труда являются данные о количестве объектов обслуживания, приходящихся на одного специалиста производственного подразделения, обеспечивающего предоставление услуг конечному потребителю. Для проведения анализа производительности труда специалистов была создана аналитическая система план-фактного анализа трудозатрат «СПФА» на базе BI-инструментов компании Qlik.

Сокращение удельных затрат на оказание ИТ-услуг

Отдельным стримом мы выделили финансовую составляющую предоставления ИТ-услуг, в частности сокращение удельных затрат на их оказание. Цель по этому направлению – сокращение удельных затрат на оказание ИТ-услуг не менее чем на 2% в год. Для этого мы, в первую очередь, разделили затраты по сервисам, а также провели аудит потребностей по поддержке приложений вендорами.

Итоги программы «7+1» и ее продолжение

К концу 2017 года были достигнуты следующие результаты (рис. 1):

- каждое седьмое обращение автоматизировано;
- каждый четвертый запрос на обслуживание шаблонизирован;
- созданы портал самообслуживания ИТ-пользователей «Газпром нефти» и Центр управления ИТ-операциями «УПитер»;
- удельная стоимость затрат на предоставление ИТ-услуг вместо запланированных 2% снижена на целых 7%!

В 2018 году программа «7+1» получила свое логическое продолжение (см. таблицу 2). Ряд целей

Таблица 2. Стримы (направления движения) показателей на 2018 год.

№	Цели на 2018 год
1	Управление знаниями
2	Уровень удовлетворенности пользователей по услугам/дочерние организации
3	Роботизация и искусственный интеллект
4	Удельное снижение количества запросов на обслуживание (цель – на 5%)
5	Реализация гибких подходов к организации рабочего процесса
6	Повышение эффективности
7	Среднее время решения обращений пользователей (цель – меньше 30 минут)
8	Сокращение удельных затрат на предоставление ИТ-услуг

Рис. 1. Результаты программы «7+1».



и показателей претерпели изменения, но общий вектор на повышение эффективности ИТ-сервиса сохранился.

Роботизация и искусственный интеллект

Одно из важных направлений работы в 2018 году – роботизация и внедрение искусственного интеллекта в предоставление ИТ-услуг. За пять месяцев текущего года в Контактном центре более 100 тысяч обращений было обработано в автоматическом режиме – без участия ИТ-специалистов. Это стало возможным благодаря внедрению специального робота технической поддержки.

Робот для автообработки обращений, поступающих от ИТ-пользователей «Газпром нефти» по e-mail и через портал самообслуживания, был внедрен в конце 2017 года – сначала в режиме опытной эксплуатации, а затем в промышленной.

На данный момент этот робот может выполнять такие операции, как: корректировка описания и классификация обращений по ключевым словам, маршрутизация обращений на рабочие группы специалистов линий техподдержки. С января по май 2018 года робот помог с каждым седьмым обращением, поступившим в Контактный центр. Кроме того, около 50 тысяч обращений было автоматически маршрутизировано на рабочие группы специалистов линий поддержки.

Очевидным преимуществом наличия робота является уменьшение времени, за которое обращения пользователей попадают к специалистам: интеллектуальная система за считанные секунды классифицирует и маршрутизирует поступающие к ней сообщения. Кроме того, сокращаются трудозатраты сотрудников Контактного центра, и теперь они могут сосредоточить свое внимание на более сложных задачах.

Разработка робота выполнена специалистами компании «ИТСК» на платформе корпоративной системы управления ИТ-сервисами с использованием языков JavaScript, RAD и технологии Simple Search с применением Regular Expressions.

Интерактивная система «Айсберг»

Нужно сказать еще об одном важнейшем элементе развития ИТ-сервисов, предоставляемых «ИТСК». Для эффективного управления предоставлением ИТ-услуг необходимы аналитические инструменты, которые могут показать «узкие места» и найти точки приложения усилий для повышения эффективности работы. При этом необходимо учитывать неоднород-

ность и распределенность ИТ-ландшафта в группе компаний «Газпром нефть». Многие подразделения территориально разнесены, имеют собственные ИТ-бюджеты и управляются разными командами. Это, с одной стороны, накладывает некоторые ограничения на стандартизацию подходов, но, с другой стороны, необходима их же унификация и обеспечение прозрачности.

Кроме того, для обеспечения продуктивного взаимодействия с бизнесом работа по увеличению его вовлеченности как в сами информационные технологии и их возможности, так и в процесс предоставления ИТ-услуг должна быть непрерывной. Только такая взаимная интеграция функций обеспечивает конструктивное развитие в условиях цифровой трансформации бизнеса.

За пять месяцев текущего года в Контактном центре более 100 тысяч обращений было обработано в автоматическом режиме – без участия ИТ-специалистов.

Таким образом, перед «ИТСК» стоят следующие задачи:

- унификация подходов к управлению ИТ-услугами с учетом географической распределенности подразделений и ресурсов компании, специфики ключевых направлений деятельности, особенностей формирования ИТ-бюджетов;
- вовлечение в работу ИТ-функции представителей бизнеса;
- повышение прозрачности работы ИТ-функции для бизнеса;
- мониторинг эффективности и оптимизация ИТ-услуг и сервисов.

Для их решения в блоке логистики, переработки и сбыта «Газпром нефти» был разработан ряд BI-инструментов, предназначенных как для обеспечения сервисной функции ИТ, так и для предоставления агрегированной информации бизнес-пользователям по количественным и качественным показателям данных внутри систем. В процессе развития BI-инструменты были объединены в единую иерархическую систему, получившую название «Айсберг». Иерархическая структура системы построена

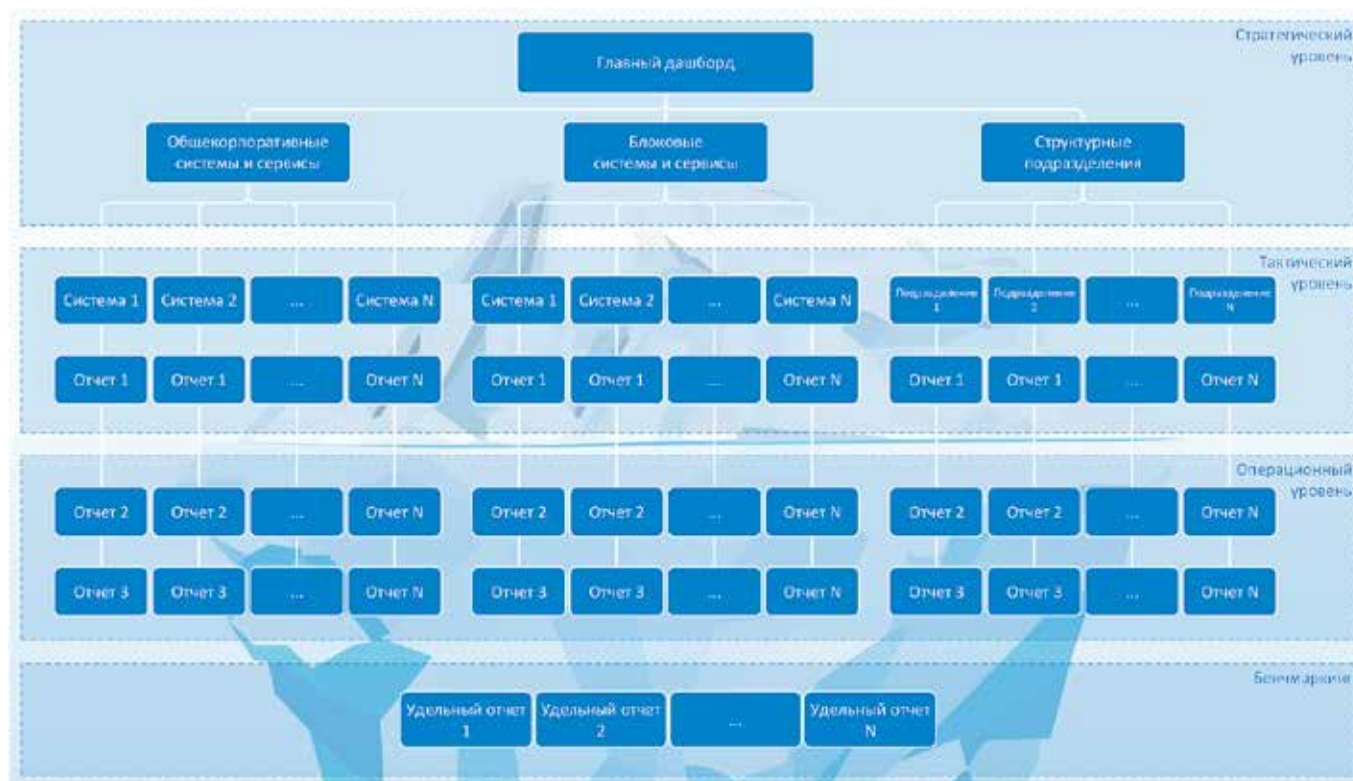


Рис. 2.
Иерархическая
структура
системы
«Айсберг».

на основе продуктового подхода и в зависимости от принадлежности к подразделениям (рис. 2).

«Айсберг» представляет собой четырехуровневый интерактивный комплекс аналитических инструментов. Каждый из уровней предоставляет информацию определенного класса, адресно предназначенную потребителям различного уровня ответственности.

Первый (стратегический) уровень аналитических инструментов предоставляет высшему менеджменту компании информацию об информационных системах, структурных подразделениях, а также общекорпоративных сервисах, потреби-

Второй (тактический) уровень информирует как руководство, так и представителей бизнеса, ответственных за ключевые бизнес-процессы, об изменениях ключевых показателей в информационных системах и автоматизируемых ими процессах.

Третий (операционный) уровень содержит исчерпывающую специализированную и наиболее детальную информацию о состояниях, статусах, основных показателях, их динамике. Этот уровень предназначен для глубокого анализа ситуации менеджерами продуктов и информационных систем, а также сотрудниками ИТ в подразделениях компании.

Инструменты, представленные на четвертом уровне (бенчмаркинг), позволяют производить сопоставительный анализ сервисных показателей структурных подразделений, отслеживать динамику изменения удельных показателей, выявлять тенденции эффективного функционирования, а также использовать их для принятия решений и тиражирования лучших практик с целью повышения качества сервиса.

Такой комплексный подход к организации аналитических инструментов дает возможность предоставлять актуальную, исчерпывающую информацию бизнесу по ключевым направлениям, позволяет объединять подходы всей сервисной команды, а также обеспечивает единое видение целей и единую оценку происходящих событий. «Айсберг» оптимизирует про-

Система «Айсберг» оптимизирует процессы подготовки и публикации отчетности, гарантирует оперативность предоставления информации.

телями которых являются все сотрудники подразделений. Информация представлена только в виде лаконичных, агрегированных показателей «на одном экране» (рис. 3.)



Рис. 3. Представление первого уровня аналитических инструментов.

цессы подготовки и публикации отчетности, гарантирует оперативность предоставления информации. Способность в реальном времени производить анализ поступающей информации и оперативно выявлять отклонения от референтных значений, позволяет не только снижать влияние негативных факторов на многие бизнес-процессы, но и повышает качество управленческих решений со стороны бизнеса.

Помимо всего вышеперечисленного в системе предусмотрены элементы предиктивной аналитики. Дальнейшее же их развитие поможет обеспечить одно из самых главных требований к сервису – сокращение затрат и оптимизацию ресурсов за счет прогнозирования изменения картины по ИТ-сервисам, что имеет огромное значение в развитии направления систем поддержки принятия решений.

Выполнение всех перечисленных инициатив, безусловно, повышает эффективность управления ИТ-услугами. Однако данный перечень не окончательный. Мы будем стремиться к организации ИТ-услуг на уровне лучших мировых практик. Более того, я уверен, что в некоторых аспектах мы сможем стать первопроходцами. Наша главная задача – высокое качество оказания услуг и доверие к компании со стороны потребителей. А значит, развитие ИТ-сервиса в нашей компании продолжится. ♦

**Работа над следующим
выпуском
Альманаха itSMF
уже началась**

Уважаемые авторы!
Ждем ваши статьи и заявки
на участие во всероссийском
конкурсе статей
**«ITSM в России: практические
ценности»**

СТАТЕЙ КОНКУРС
itSMF
Russia

Информация на сайте www.itsmforum.ru



Владимир Аношин

Старший преподаватель, ведущий тренер-консультант. Обладает 30-летним опытом работы в области информационных технологий, имеет опыт разработки и организации поддержки, в том числе 10 лет – на руководящих позициях. В IT Expert более 10 лет – с 2006 г. Аккредитован в качестве тренера по большинству курсов по ITIL. Соавтор многих проводимых IT Expert курсов по ITIL, COBIT и управлению IT-персоналом.

Однажды Роза Бертэн, личная портниха французской королевы Марии-Антуанетты, слегка подновив старое платье королевы, предложила его королеве, и та с удовольствием его приняла. «Новое – это хорошо забытое старое», – прокомментировала этот случай портниха.

Из мемуаров (1824) Розы Бертэн.

ITIL и риски быстрого изменения приложений. Как организовать поддержку в новых условиях?

ITIL и новые вызовы времени

В последнее время часто говорят о необходимости быстрой разработки и внедрения изменений в приложениях и о кризисе традиционной ИТ-поддержки, которая описана в лучших практиках, в том числе в библиотеке ITIL.

Так что же, подходы ITIL поколеблены и он безнадежно устарел или не все так плохо? Попробуем в этом разобраться и понять, что в ITIL останется

незыблемым, а что необходимо совершенствовать согласно современным вызовам. В своих рассуждениях я буду использовать идеи, которые уже были высказаны до меня авторитетами в области разработки, только мы рассмотрим их через призму задач поддержки.

В менеджменте, как, впрочем, и в остальных областях человеческой деятельности, люди часто пытаются отказаться от уже существующего и построить

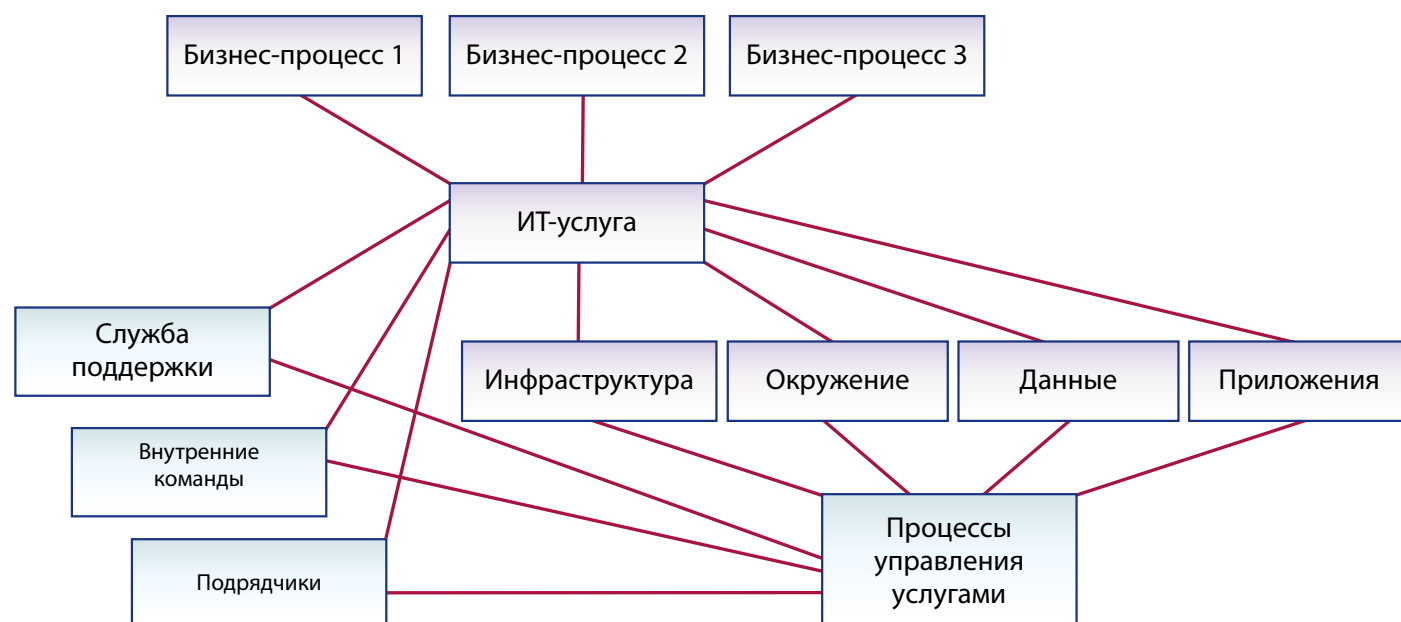


Рис. 1.
Общая схема
услуги.

все с чистого листа. Но, как мне кажется, многое из того, что требуется для быстрых изменений, можно взять из уже существующих практик или они будут являться их следствием с необходимыми корректировками в соответствии с текущим моментом (и это согласуется с принципами, изложенными в книге «ITIL Practitioner»). Конечно, в текущей версии ITIL вопросы разработки освещены, скажем так, весьма скудно. Будем надеяться, что этот недостаток исправят в следующей версии ITIL, а пока попробуем показать, какие изменения могут претерпеть лучшие практики в данной области.

ITIL – новые нюансы ценности ИТ-услуги для заказчика

Как говорит ITIL, услуга – это способ предоставления ценности заказчикам через содействие им в получении конечных результатов, которых заказчики хотят достичь без владения специфическими затратами и рисками. ИТ-услуга включает в себя информационные технологии, процессы и людей. Бизнес-приложение является частью услуги. Несомненно, важной, но только частью.

На общей схеме услуги мы видим, что ИТ-услуга поддерживает бизнес-процессы и может включать ИТ-инфраструктуру, приложения, данные, физическое окружение, команды поддержки и подрядчиков (рис. 1).

Одним из основополагающих принципов ITIL является понятие ценности предоставляемой услуги. А ценность, в свою очередь, складывается из полезности и гарантии.

Полезность часто выглядит как новый функционал, необходимый заказчику. В современных условиях ценностью для заказчика также является возможность быстрого предоставления и использования нового функционала, в некоторых случаях на конкретный период.

Но ценность ИТ-услуг без гарантии – ничто: нужно гарантировать, что параметры ИТ-услуги всегда будут оставаться в согласованных пределах, то есть риски, возникающие в ходе предоставления такой новой возможности, будут выявлены и управляемы. К тому же на организацию поддержки влияют факторы, являющиеся следствием организации разработки приложений. Исходя из этого, попробуем рассмотреть, какие наиболее значимые дополнительные риски для гарантии предоставления и поддержки ИТ-услуг возникают и какие есть мероприятия и механизмы, которые помогут с ними справиться.

Новые факторы организации разработки приложений, которые могут повлиять на организацию поддержки

Потребность в быстрой разработке приложений вызвала к жизни такие новаторские подходы организации разработки, как *scrum* или микросервисы. Кратко опишем их, выделив факторы, влияющие на организацию поддержки:

- приложение разрабатывается путем нескольких итераций, каждая из которых имеет законченный бизнес-функционал и отражает текущие требования заказчика (вспоминаются шесть шагов постоянного совершенствования услуги, описываемого ITIL);

Поступательность в единстве с преемственностью составляет суть диалектического развития.

Конспект лекций по философии

- разбиение приложения на законченные компоненты (или микросервисы), отражающие конкретные требования бизнеса и способные самостоятельно предоставлять ценность для бизнеса;
- законченные компоненты приложений будут разрабатываться кросс-функциональными командами, имеющими полный набор необходимых навыков в построении пользовательских интерфейсов, архитектуры приложений и управления проектами;
- в ходе замены одной версии компонента на другую возможен так называемый «канареечный тест», в ходе которого какое-то время обе версии компонента параллельно работают в среде эксплуатации, чтобы можно было подстраховаться (образно говоря, «два бронепоезда», идущих параллельными курсами, пока команда по частям пересаживается из одного в другой);
- представитель команды разработчиков может владеть продуктом на протяжении всего срока

В ITIL говорится о несоответствии скорости реакции ИТ на новые требования бизнеса. Но ситуация еще более обостряется. Теперь ряд услуг будут изменяться очень часто, возможно, практически непрерывно.

его жизни (это хорошо коррелирует с жизненным циклом услуги и ролью владельца услуги в ITIL); она не распускается по окончании проекта, а сопровождает его дальше по жизни; совсем крайний пример реализации – команды отвечают за все аспекты ПО, которое они разрабатывают, включая поддержку его в режиме 24/7 (то есть предлагается разработчиков переселить на вторую линию поддержки, если пользоваться терминологией ITIL);

- максимальная автоматизация всех процессов разработки, включая тестирование и развертывание в среде эксплуатации, а также разработка и использование более продвинутого инструмента, общего для разработчиков и системных администраторов;
- переход к модели «инфраструктура как код (IaC)», по которой процесс настройки инфраструктуры аналогичен процессу программирования приложений и ведет к стиранию границ между написанием приложений и созданием сред для этих приложений;
- приложения должны быть спроектированы таким образом, чтобы они могли работать при отказе отдельных компонентов (частичное функционирование) и имели возможность быстрого отката как

всего приложения, так и отдельного компонента в случае неудачного изменения;

- большой акцент на мониторинге приложения в режиме реального времени, как на техническом, так и на бизнес-уровнях, для выявления возможных отклонений как можно раньше.

Какие новые риски возникают в области поддержки?

Новые подходы и практики разработки приложений, несомненно, оказывают влияние на поддержку и вызывают новые риски. Рассмотрим эти риски, используя приведенный в ITIL подход жизненного цикла услуги. В небольшой статье трудно учесть все риски, но тем не менее попробуем разобрать некоторые из них:

- неготовность ИТ следовать требованиям бизнеса к частому, практически непрерывному изменению функционала уже используемых им бизнес-приложений;
- разобщенность миров разработчиков и специалистов поддержки;
- хаотичная разработка приложений, ведущая к критичным сбоям в ИТ-инфраструктуре;
- увеличение вероятности ошибок в разрабатываемом приложении;
- остальная ИТ-инфраструктура не такая гибкая, как приложения;
- высокая вероятность неудач при внедрении большого количества маленьких, но очень частых изменений;
- недостаточный уровень контроля за изменяемыми компонентами;
- повышение вероятности проблем, вызванных частыми изменениями;
- недостаток знаний линий поддержки о внесенных изменениях;
- недостаток знаний пользователей о внесенных изменениях;
- необходимость поддержки нескольких параллельных решений.

Далее мы подробно разберем все эти риски и опишем, какие меры уменьшения их воздействия можно взять из существующих практик (например, ITIL) и как их скорректировать в соответствии с новыми реалиями.

Риск: неготовность ИТ следовать требованиям бизнеса к частому, практически непрерывному изменению функционала бизнес-приложений

Данный риск и раньше рассматривался в ITIL. Говорилось о несоответствии скорости реакции ИТ на новые требования бизнеса. Ситуация еще более обостряется. Ряд услуг теперь будут изменяться

очень часто, возможно, практически непрерывно. Какие же меры по уменьшению этого риска можно предпринять?

Меры. Согласно рекомендациям лучших практик, для начала необходимо определить заказчиков, у которых будут такие требования, и услуги, которые будут затрагиваться этими требованиями. Заметим также, что, согласно подходам по совершенствованию ITIL, такая деятельность должна выполняться постоянно для мониторинга изменений как по заказчикам, так и по услугам.

Если рассматривать существующую модель ITIL, то это может выглядеть как часть дополнительной деятельности по анализу влияния на бизнес (business impact analysis). После определения таких услуг необходимо перестроить в них работу – создать новые модели (pattern) управления для других этапов жизненного цикла услуги, в особенности это касается этапа преобразования услуги.

В качестве примера создания одной из таких новых моделей управления попробуем определить новый тип изменения, который отвечал бы требованиям к частому изменению приложений, а значит и ИТ-услуг, если рассматривать это в рамках терминологии ITIL. Для простоты назовем данный новый тип изменения «быстрым» (quick). В существующей библиотеке ITIL наиболее близким к нему является стандартный тип изменения, но не только. Например, «быстрое изменение»¹ также несет черты экстренного изменения ITIL (инициация в любой момент, отложенное документирование), а кроме того, включает в себя элементы пилотного внедрения. Опираясь на ранее разобранные новые практики разработки приложений, можно говорить о том, что быстрое изменение:

- не должно выходить за рамки одного компонента;
- его инициация, согласование и утверждение должны проходить с минимумом формальностей;
- тестирование, внедрение и откат должны быть максимально автоматизированы;
- внедрение изменения должно происходить максимально быстро и прямо на работающем приложении или с минимальной остановкой того компонента, который изменяется;
- тестирование, которое невозможно провести на стадии разработки компонента, может выполняться на начальной стадии эксплуатации, возможно, на ограниченной целевой группе пользователей;
- сразу после внедрения должен осуществляться целевой мониторинг работы измененного компонента, как на уровне ИТ-инфраструктуры, так и на уровне бизнес-логики;

- если в ходе эксплуатации уровень отклонений (ошибки или отрицательные отзывы) превышает некий определенный уровень, то должен происходить откат в первоначальное состояние, возможно, автоматический, или должны инициироваться быстрые корректирующие меры и проводиться следующая итерация изменения;
- документирование данных в ходе изменения должно быть минимальным, полное документирование всех данных должно происходить после проведенного изменения.

Разработка приложений, ведущаяся хаотично, без какого-либо плана, рано или поздно приведет к критичным сбоям. Нужен механизм, ограничивающий хаос

Риск: хаотичная разработка приложений, ведущая к критичным сбоям в ИТ-инфраструктуре

Разработка приложений, ведущаяся хаотично, без какого-либо плана, рано или поздно приведет к критичным сбоям в работе ИТ-инфраструктуры и прерыванию ИТ-услуг, критичных для бизнеса. Нужен механизм, ограничивающий хаос, нужен «управляемый хаос».

Меры. Если распространить идеи быстрой разработки приложений на практику поддержки, то можно предложить путь, дающий возможность защитить гарантию услуги от данного риска – разделить все активы ИТ-услуг на два типа:

- «базовые компоненты» – будут являться ключевыми для всей ИТ-инфраструктуры, изменяться нечасто и только после тщательной оценки всех рисков, которые несут в себе эти изменения для бизнеса;
- «динамические компоненты» – будут изменяться часто, практически постоянно, и должно быть сделано все, чтобы их изменения не принесли больших рисков для бизнеса.

Соответственно, изменение «базовых компонентов» происходит как проекты или как нормальные (normal) изменения ITIL, изменение «динамических компонентов» потребует новых подходов. «Динамические компоненты» можно рассматривать как расширение понятия «компонент приложения», но «динамический компонент» должен рассматриваться в рамках всей услуги, а не только одного приложения. В общем случае в «динамический компо-

¹Далее в этой статье я буду применять термин «быстрое изменение» без кавычек.

нент» услуги могут входить компоненты всех элементов услуги, связанные с компонентом приложения². В чем-то динамический компонент можно рассматривать как реинкарнацию понятия единицы релиза, о котором говорится в ITIL. Данное разделение может быть определено через политики жизненного цикла услуг и введено в классификацию активов услуг.

В ITIL уже есть владелец услуги, отвечающий за услугу на протяжении ее жизненного цикла. Ему явно не хватает команды по услуге, в которую органично впишутся разработчики.

Риск: разобщенность миров разработчиков и специалистов поддержки

Традиционно лучшие практики разработки приложений и их поддержки были разделены. Специализация привела к тому, что миры разработки и операций разделились, это привело к дополнительному риску. Однако ужесточение требований к скорости реализации требований бизнеса с сохранением гарантий качества предоставляемых услуг требует от разработчиков и специалистов поддержки соединить свои усилия и наконец-то понять, что их миры должны как минимум сближаться, а как максимум – слиться в некий симбиоз.

Меры. Одна из новых практик разработки приложений – формирование команды по продукту. С точки зрения лучших практик поддержки это аналогично команде по услуге. В ITIL уже есть владелец услуги, отвечающий за услугу на протяжении ее жизненного цикла. Ему явно не хватает команды по услуге, в которую органично впишутся разработчики.

В состав команды по услуге могут входить:

- аналитики, способные транслировать требования бизнеса (бизнес-процессов) в требования к приложению и к услуге в целом;
- команда разработчиков, владеющая знаниями по всем аспектам приложения;
- специалисты по ИТ-инфраструктуре;
- специалисты, владеющие знаниями по поддержке и пользователям;
- представители подрядчиков.

Отмечу, что данное изменение не противоречит подходам ITIL, а только дополняет их. Поддержке

придется жить в беспокойном мире максимальной полезности, в котором частые изменения будут рутинными событиями, а разработчикам – привыкать заботиться о гарантиях предоставляемых услуг.

Риск: увеличение вероятности ошибок в разрабатываемом приложении

Давление заказчиков по сокращению времени реакции на их новые требования приводит к уменьшению срока разработки приложений, что неминуемо повышает риски того, что в жизненную среду попадут приложения с серьезными ошибками.

Меры. Конечно, этот риск в первую очередь будет минимизироваться самими разработчиками через применение более совершенных систем разработки и тестирования, использование проверенных наборов стандартных библиотек, максимальную близость сред разработки, тестирования и эксплуатации, максимальную автоматизацию постоянно совершенствующейся системы тестирования, но тем не менее такой риск не может быть полностью компенсирован только командами разработки.

Здесь я хотел бы высказать мысль, которая может вызвать возражения – в ряде случаев ошибки в динамических компонентах будут проходить в среду эксплуатации, и нам придется отлавливать их уже там. Тогда меры уменьшения этого риска должны помочь поддержке выявить ошибку быстро и незаметно, чтобы ее появление не создавало паники в рядах поддержки и решалось бы как незначительный рядовой инцидент, почти не влияющий на бизнес.

И здесь основная часть работы должна быть сделана на этапе разработки. Одним из принципов новых подходов в разработке является разработка приложения с возможностью продолжать работу при частичном сбое и (или) возможность быстрого отката при полном отказе. В ходе разработки можно заложить в приложение элементы встроенного мониторинга, которые могут стать частью общей системы мониторинга поддержки как на уровне ИТ-инфраструктуры, так и на бизнес-уровне.

При возникновении данных событий:

- пользователи могли бы получать формируемые самим приложением рекомендации по особенностям продолжения работы с этим приложением с учетом уровня их компетенции (user experience);
- первая и вторая линии поддержки могли бы получать адаптированную под них информацию как от самого приложения, так и от обслуживающей это приложение ИТ-инфраструктуры.

²Далее в этой статье под термином «компонент» для краткости я буду иметь в виду термин «динамический компонент» услуги (далее буду использовать его без кавычек) и рассматривать только этот тип компонентов.

Риск: высокая вероятность неудач при внедрении большого количества маленьких, но очень частых изменений

Данный риск возникает в самом интересном месте жизненного цикла – когда разработанный динамический компонент внедряется в среду эксплуатации. Этот риск понижается принципами независимости и малоразмерности внедряемого объекта. Но сильно повышается от того, что это будет делаться часто, местами непрерывно.

Меры. Для начала попробуем более детально разобрать, к каким проблемам это может привести, как может выглядеть реализация такого быстрого изменения и какие механизмы могут быть задействованы. Неудачи быстрых изменений с точки зрения поддержки будут следующие:

- ИТ не успеет внедрить быстрое изменение, нужное бизнесу, в срок, необходимый бизнесу;
- после внедрения, если бизнес примет решение об отказе от данного изменения, ИТ не сможет сделать это в срок, требуемый бизнесом;
- работа ИТ-услуги будет нарушена целиком или частично вследствие ошибок, внесенных быстрыми изменениями в ИТ-инфраструктуру, и мы не сможем заметить это вовремя и принять меры к исправлению ситуации.

Исходя из подходов быстрой разработки приложений и положений ITIL в области преобразования ИТ-услуг, быстрое внедрение динамического компонента в среду эксплуатации может включать в себя следующие шаги и механизмы:

1. Появляется заявка бизнеса на изменение, возможно, неформальная. И если эта заявка относится к динамическому компоненту, то она принимается, кратко фиксируется в базе быстрых изменений, и по ней начинается работа.
2. Как в ходе разработки, так и на выходе перед внедрением разрабатываемый компонент подвергается автоматизированному тестированию, позволяющему выявить ошибки.
3. Как только компонент разработан и прошел тестирование, устанавливается время запуска релиза и о нем оповещаются все заинтересованные стороны.
4. Необходимые сведения о быстром изменении заносятся в базу знаний по быстрым изменениям, где должно быть зафиксировано место проводимого изменения (изменяемый компонент), интервал «опасного» времени, в течение которого изменение не будет считаться успешно внедренным, характер изменения – постоянный или временный с обязательным откатом, целевая группа, для которой будет доступно данное изменение, что изменилось (изменения интерфейса, логики работы и т.д.), понятным для поддержки и пользователей языком.

5. Производится автоматический релиз компонента и, при необходимости, переключение целевой группы пользователей на внедренный компонент.
6. Параллельно поддержка оповещается о проведенном изменении, ей предоставляются описание проведенного изменения и другие необходимые сведения из базы знаний для работы с пользователями, а также проводятся необходимые изменения в базе знаний для пользователей.
7. Сразу после релиза начинается мониторинг измененного компонента, включающий в себя фиксацию отклонений в работе как самого компонента, так и отклик на его работу со стороны пользователей.
8. В случае обнаружения отклонений в объемах, превышающих допустимый, дается команда, возможно, автоматическая, на откат изменения и (или) запуск компенсирующих мер, например быстрое исправление ошибок и запуск следующей итерации релиза.
9. Заявка на откат изменения может также поступить от бизнеса и не являться следствием отклонений в работе измененного компонента, а будет результатом решения самого бизнеса на основе своих бизнес-критериев.

Ошибки в динамических компонентах будут проходить в среду эксплуатации, и нам придется отлавливать их уже там. И необходимо выявить ошибку быстро и незаметно, чтобы ее появление не создавало паники в рядах поддержки.

10. По окончании интервала «опасного» времени изменение считается успешным, что должно быть подтверждено бизнесом и зафиксировано.
11. Если на изменяемый компонент переводилась только некая целевая группа пользователей («канареечный тест»), то делается перевод всех пользователей на измененный компонент, и ненужный более компонент удаляется.
12. В заключение документируются все необходимые данные по проведенному изменению.

Риск: недостаточный уровень контроля за изменяемыми компонентами

В поддержке нам будет необходимо контролировать не одно, а много быстрых изменений и работать с откликами на них в виде инцидентов и обращений пользователей. Количество инцидентов и событий



Рис. 2.
Статусы
динамического
компонента.

в поддержке будет расти, сложность их классификации тоже, и мы рискуем потерять контроль над динамическими компонентами. Мы не поймем, какие из компонентов являются динамическими и на какой стадии изменения находится тот или иной динамический компонент, если он изменяется.

Меры. Здесь нам поможет система мониторинга, связанная с базой конфигураций, в которой должны будут фиксироваться все динамические компоненты

Количество инцидентов и событий в поддержке будет расти, сложность их классификации тоже, и мы рискуем потерять контроль над динамическими компонентами.

и их связи с поступающими после быстрого изменения обращениями и инцидентами. Это будет развитием идей ITIL о постоянном контроле за работой ИТ-инфраструктуры. Возможными статусами динамических компонентов могут быть следующие (рис. 2):

- компонент стабилен;
- компонент готовится к изменению и меняется;
- компонент изменен, начинается опасный период эксплуатации, возможны ошибки;
- опасный период закончился, изменение признано успешным и стабильным;
- отклонения больше допустимого или изменение признано бизнесом неуспешным;
- откат к предыдущему стабильному состоянию или коррекция с повторным изменением.

Таким образом, может быть сформирована карта мониторинга, обеспечена возможность сообщать об изменениях всем заинтересованным сторонам и наблюдать за ростом отклонений, а также возможность своевременной реакции на это, в том числе автоматической.

Риск: неготовность к решению проблем, вызванных быстрыми изменениями

Внедрение в практику быстрых изменений с большой степенью вероятности потребует увеличения скорости решения проблем и смещения деятельности по их решению в проактивную область. Как можно уменьшить этот риск, опираясь на существующие лучшие практики?

Меры. Реактивными мерами решения проблем, вызванных изменениями динамических компонентов, вероятно, будут либо откат к первоначальному состоянию, либо (если это допускает ситуация) быстрая коррекция и повторное внедрение. Необходимо заранее разработать типовые обходные решения проблем, ибо времени на нахождение их решения будет мало.

Проблемы также могут заключаться в следующих областях:

- в неправильном определении границ динамических компонентов – изменения динамических компонентов ведут к нарушению работы базовых компонентов;
- потеря стабильности работы изменяемых приложений – применяемая методика разработки и внедрения приводит к накоплению ошибок, и далее справиться с ними с учетом потока новых изменений становится все тяжелее;
- узкие места, создающие задержки в ходе проведения изменений – необходима постоянная работа по выявлению и ликвидации таких мест;
- различия между тестовой средой и средой реальной эксплуатации, которые ведут к слишком позднему выявлению ошибок, связанных с проводимыми изменениями – тестовая среда и режимы испытаний должны быть максимально приближены к реальности, возможно, путем эмуляции;
- недостатки коммуникаций между всеми заинтересованными лицами, которые будут неизбежно увеличиваться в случае частых изменений.

Риск: недостаток знаний линий поддержки о внесенных изменениях

При росте количества инцидентов и событий в рамках поддержки увеличивается нагрузка на первую линию поддержки. Первая линия поддержки должна быть в курсе статусов изменений всех динамических

компонентов в режиме online, чтобы оказывать поддержку пользователей и формировать статистику по успешности проводимых быстрых изменений.

Меры. Нужно понимать, кто на первой и на второй линиях поддержки и в какой форме должен получать постоянно актуальную информацию об изменениях всех динамических компонентов. Необходимо организовать обмен знаниями, которые отвечали бы следующим требованиям:

- новые знания должны быть быстро оформлены (параллельно с разработкой);
- новые знания должны быть обновлены и доведены до адресатов в нужное время;
- их форма должна быть удобной для быстрого понимания и использования;
- распространение знаний может быть многошаговым, если внедрение будет также в несколько шагов;
- новые знания должны быть быстро отозваны и заменены на исходные как в местах хранения, так и в головах адресатов, если это будет необходимо.

Риск: недостаток знаний пользователей о внесенных изменениях

Если проводимые быстрые изменения будут менять интерфейс или логику работы услуги, то они могут вызвать поток обращений пользователей, вызванных неосведомленностью о таких изменениях или нехваткой знаний или умений, которые будут следствием таких изменений.

Меры. Необходимо организовать консультации пользователей по проводимым изменениям (чем на практике часто пренебрегают) и своевременное обновление баз знаний для пользователей. Один из вариантов – организовать точечную блиц-поддержку, встроив в изменяемое приложение или страницу сайта интерфейс взаимодействия с поддержкой в виде чата или звуковой связи. Также необходимо продумать всплывающие подсказки и ссылки на базу знаний по наиболее часто возникающим у пользователей вопросам. Одна из трудностей здесь – определение, что необходимо менять в базе знаний в ходе быстрого изменения и реализация быстрого внедрения таких обновлений.

Риск: необходимость поддержки нескольких параллельных решений

В ходе изменения динамического компонента возможна схема, при которой какое-то время обе версии компонента параллельно работают в среде эксплуатации. Тогда будет необходима поддержка двух групп пользователей: пилотной, у которой новая версия, и остальных – со старой версией. Такая ситуация создает дополнительный риск.

Меры. Поддержка должна быть в курсе, какая версия компонента установлена у обращающегося в поддержку пользователя, следовательно, нужен механизм контроля всех пользователей пилотной группы и времени перевода остальной части пользователей на новую версию.

Первая линия поддержки должна быть в курсе статусов изменений всех динамических компонентов в режиме online.

Риск: остальная ИТ-инфраструктура не такая гибкая, как быстро изменяемые приложения

Несмотря на то обстоятельство, что «железо» является одним из самых тяжело изменяемых элементов ИТ-услуги, придется меняться и ему.

Меры. Автор не является глубоким экспертом в области методов построения гибкой ИТ-инфраструктуры, но полагает, что успехи, демонстрируемые в последнее время в области виртуализации ИТ-инфраструктуры, помогут делу и позволят снизить данный риск. Здесь уместно вспомнить такие подходы, как инфраструктура как услуга (Infrastructure-as-a-Service, IaaS) и архитектура Serverless, расширяющиеся границы гибкости ИТ-инфраструктуры. Поскольку динамические компоненты услуги могут включать в себя и «железо», то часть требований бизнеса может быть осуществлена за счет гибкой перенастройки части ИТ-инфраструктуры в ходе быстрых изменений, и это должно быть отражено в лучших практиках.

ITIL – что же дальше?

Несомненно, новые подходы и практики разработки приложений оказывают влияние на поддержку и вызывают новые риски. Лучшие практики потому и должны быть лучшими, что они обязаны отвечать на новые вызовы времени. Как мы видим, в ITIL есть основа, которую можно и нужно развивать.

Существующие описания организации быстрой разработки больше фокусируются на организации разработки приложений, а существующую версию ITIL часто и, наверно, справедливо критикуют за высокоуровневость и чрезмерную научность. Но многие положения и идеи ITIL, уже имеющиеся там, будучи конкретизированы и соединены с новыми подходами разработки, дадут необходимые ответы на вопрос: «Как нам организовать поддержку в этих новых условиях?». Будем же надеяться, что новая версия ITIL расскажет нам об этом, и пожелаем ITIL удачи в этом деле. ♦

Если вы решили первым стать в рядах своих сограждан – никогда не догоняйте устремившихся вперед. Через пять минут, ругаясь, побегут они обратно, и тогда, толпу возглавив, вы помчитесь впереди!

*Стихотворение
Григория Остера*

DevOps можно рассматривать как набор практик и инструментов, направленных на повышение скорости и гибкости, эдакий «мешок» с полезными организационными и техническими идеями, из которого можно вынимать идеи и практики и применять их там, где это кажется эффективным. И в таком понимании есть очень большая доля истины, о чем мы и поговорим далее.

Однако авторы данной статьи – сторонники системного мышления. Мы считаем, что это помогает бороться со сложностью рассматриваемых явлений. С одной стороны, такой подход позволяет осуществлять анализ, дает возможность думать по очереди обо всех важных элементах, с другой стороны, делать это не упуская из виду необходимость конечного синтеза, не теряя взаимовлияний и взаимосвязей отдельных элементов и постоянно думать о целом как о системе.

Не являясь экспертами в описываемой области, мы решили взглянуть на DevOps как на систему и задались вопросом, можно ли выделить в DevOps ключевые принципы – те идеи и положения, что остаются неизменными для различных идей и практик, «появляющихся из мешка», а также для различных ситуаций, задач и компаний, использующих DevOps. Те идеи и положения, без принятия которых и следования которым эту идею, практику и деятельность вряд ли можно отнести к области DevOps. Иными словами, то, что можно назвать ядром и базовым фундаментом подхода DevOps.

«Слон» DevOps

Первый вопрос – а имеют ли смысл наши поиски ключевых принципов и положений DevOps? Разве DevOps не может быть объединением в чем-то сходных, но все-таки разных идей и направлений? Не системой, не единым подходом, а просто объединением различных идей, направленных к одной цели, не красиво и гармонично составленным букетом, а пучком собранных в разных местах и в разное время полевых цветов. Что ж, в таком букете может быть своя прелесть...

Интересно, что подобное уже встречалось в истории с ИТ-практиками и совсем недавно и получило название Agile-«слон» (рис. 1). Приведем цитату из книги «Постигая Agile. Ценности, принципы и методологии»¹:

...В этой ситуации очень подходит история «Слепые и слон». В раздробленном agile-видении человек использует только те методы, которые влияют на его работу (так же, как каждый из слепых ощущает только одну часть слона). Например, разработчики концентрируются на разработке, тестировании, рефакторинге и автоматической



сборке. Руководители проекта – на доске задач, скорости реализации проектов и соблюдении графика. ...Руководители используют ежедневные встречи и ретроспективы для управления и улучшения команды. ...Внедряя все эти методы по отдельности, вы, безусловно, улучшите положение вещей, потому что agile-практики действительно очень полезны. Проблема в том, что все – разработчики, менеджеры проектов, бизнес-пользователи и лидеры команды – видят проект с разных точек зрения и концентрируются только на тех методах, с которыми непосредственно связаны...

Рис. 1.

Agile-«слон», который больше, чем сумма различных практик.

Источник:
«Постигая Agile. Ценности, принципы и методологии». Эндрю Стеллман и Дженнифер Грин.

Интересно, что целый ряд экспертов придерживаются аналогичного взгляда на подходы и практики Agile, используемые в DevOps. По их мнению, не нужно объединять разнородные практики в единую систему, нужно оставить как есть. Ведь каждый раз практики и инструменты DevOps используются в различных компаниях и ситуациях, для выпуска разнообразных продуктов, и далеко не всегда можно найти принципы организации деятельности, адекватные для многих ситуаций. Так мы приходим к метафоре «слона DevOps» – родного брата agile-«слона» (так сказать, «молочного брата», только несколько выросшего и возмужавшего).

Такую позицию можно понять, и с определенной точки зрения она оправдана. Во-первых, вряд ли стоит преждевременно систематизировать то, что еще не готово к систематизации, ту область знания, которая пока еще активно развивается и похожа на бурлящий котел. Другими словами, сложность и разнообразие ситуаций с развитием ИТ так велики, что пока о систематизации говорить рано. Во-вторых, именно разнообразие подходов и инструментов можно рассматривать как одну из ценностей, формирующих культуру DevOps. Эта культура создавалась в условиях небольших команд единомышлен-

¹Эндрю Стеллман и Дженнифер Грин. «Постигая Agile. Ценности, принципы и методологии». Манн Иванов и Фербер, Москва, 2018.

ников, которые нацелены на постоянное развитие в изменяющейся среде. Такие команды уважают, прежде всего, индивидуальность каждого ее участника и уникальность команды.

Разнообразие подходов и инструментов можно рассматривать как одну из ценностей, формирующих культуру DevOps. Эта культура создавалась в условиях небольших команд единомышленников, которые нацелены на постоянное развитие в изменяющейся среде.

Для таких команд неприемлемы строгие методологии, создающиеся на основе четких и обязательных последовательностей процессов и процедур, и превращаются в исчерпывающие и формальные предписания о том, как необходимо действовать. В результате они, как правило, становятся жесткими, фактически убивают креативность и инициативу, заменяя их процессами, практиками, инструментами и обязательными методиками.

Вместо строгой методологии – эвристика

Главное, на наш взгляд, что жесткие методологии, которые создаются на основе четких и обязательных последовательностей процессов и процедур, хорошо зарекомендовали себя в условиях относительной стабильности и невысокой скорости изменений. Процессы и практики – это надежно с точки зрения управления и привлекательно для менеджера. Даже если и нет результатов работы, все делается правильно, все «прикрыты». Большинство практик DevOps, напротив, сложились в достаточно экстремальных ситуациях, когда требования к скорости проведения изменений в ИТ-системах и выпуска продуктов были очень малыми и при этом требования к качеству и устойчивости работы систем – достаточно высокими.

С этой точки зрения можно понять заявления ряда экспертов, что DevOps – это не методология в классическом ее понимании, а ее противоположность, она не должна быть комплексом тесно и логично связанных обязательных компонентов. Логически выверенный системный подход, отталкивающийся от некоторого непротиворечивого набора принципов или модели, заменяется эвристикой – приемами и подходами, опирающимися на дух исследования и творчества. В эври-

стическом подходе основной упор делается на создании условий для исследований и креативности, на доверии, уважении к людям и способности самоорганизации. Это подход мягкого управления ограничениями и условиями работы, он ничего не должен навязывать и мало к чему принуждать.

Еще раз отметим, что такой способ построения новой области знаний понятен и оправдан, однако в нем есть и отрицательные моменты. Если мы видим DevOps (и Agile) только как «мешок» интересных идей и практик, то, увы, рискуем упустить синергетический эффект от применения различных практик, в ходе взаимодействия людей с разными подходами, культурой и идеями. Причем, как правило, именно этот синергетический эффект и создает новые возможности и результаты.

Заметим, что многие адепты Agile критикуют его понимание только как набора различных практик и пишут о достижении некоего синергетического видения своего «слона». Приведем еще одну цитату из книги Эндрю Стеллмана и Дженнифер Грин:

Таким образом, хотя этот «слон» и состоит из множества применяемых практик, все же он существенно больше, чем просто сумма всех частей.

Итак, видя в DevOps только набор различных практик и инструментов, мы рискуем потерять эффекты от их применения как единого комплекса. Если, как в том анекдоте про ощупывающих слона слепцов, мудрый король нам объяснит, что все правы и наш «слон DevOps» обладает всеми этими качествами, однако он больше, чем сумма этих качеств. Объяснит, что «слон DevOps» – это... Увы, дальше продолжить пока не получается, мудрый король пока не находится и «слон DevOps» – это пока только «слон DevOps».

Поэтому вернемся к нашей идее и постараемся выделить в DevOps ключевые принципы – те идеи и положения, которые остаются неизменными для различных идей и практик, ситуаций, задач и компаний. Мы выделили три наиболее интересных набора принципов DevOps.

Шесть принципов DevOps с точки зрения DASA

Эксперты ассоциации DASA (DevOps Agile Skills Association) составили [список принципов DevOps](#). Авторы так определяют свое видение:

Существует множество определений DevOps. И многие из них адекватно объясняют один или более аспектов, важных для предоставления ИТ-услуг. Однако вместо попыток сформулировать наше собственное исчерпывающее опре-

деление мы считаем правильным подчеркнуть шесть принципов DevOps, которые важны для тех, кто применяет DevOps к организации работы или переходит на него.

Принципы, которые выделили авторы, показаны на рисунке 2. Это интересный набор принципов, и на первый взгляд он выглядит весьма привлекательно, но в то же время ясно демонстрирует то отсутствие системности, которое сейчас наблюдается в DevOps. По нашему мнению, недостатки этого свода принципов следующие:

- 1. Критерии выделения принципов непонятны. В список включены принципы, работающие с совершенно разнородными элементами системы управления – общая философия деятельности (деятельность, ориентированная на заказчика и постоянное совершенствование), организация деятельности (ориентация на конечный результат и ответственность от начала до конца), культура мышления сотрудников (ориентация на конечный результат), команда (кросс-функциональные автономные команды) и ИТ-инструменты (автоматизируется все, что можно). Вряд ли столь пестрый набор можно объединить какими-то более-менее четкими основаниями.
- 2. Принципы сильно неоднородны по уровню понятий, которыми они оперируют. Например, инстру-

менты автоматизации поставлены в один ряд с общей философией деятельности и культурой мышления. Безусловно, все это важнейшие активы ИТ-организации, однако вряд ли будет правильным смешивать в одном наборе принципов систему управления жизненным циклом ПО и культуру мышления.

- 3. В одном принципе объединено несколько разнородных идей – в принцип 1 включены деятельность, ориентированная на заказчика, и постоянное внедрение инноваций. Это разные идеи, хотя между ними есть определенная связь. Но почему инновации должны рассматриваться только в ключе ориентации на заказчика, другие инновации разве не инновации?

Итак, этот свод из шести принципов, неоднородный, неоднозначный, и его сложно считать фундаментом подхода DevOps.

Пять принципов DevOps

Весьма интересный набор принципов приведен в книге Олега Скрынника «DevOps для ИТ-менеджеров». Автор проделал огромную работу, изучив множество книг по DevOps, и выделил пять ключевых принципов, «без принятия и применения которых от DevOps остается совсем мало смысла» (рис. 3). Отметим, что эти пять принципов макси-

Рис. 2. Шесть принципов DevOps с точки зрения DASA.

Принцип 1. Деятельность, ориентированная на заказчика (customer-centric action)

Короткие циклы обратной связи с реальными клиентами и конечными пользователями сегодня крайне важны. И всю деятельность по созданию ИТ-продуктов и услуг надо сфокусировать на клиентах. DevOps-организации должны работать как lean-стартапы, которые постоянно внедряют инновации, максимизируя удовлетворенность заказчика.

Принцип 3. Ответственность от начала до конца (end-to-end responsibility)

DevOps-команды должны отвечать за полный жизненный цикл продукта, от концепции до вывода из эксплуатации. Эти стабильные команды несут ответственность за созданные и поставляемые ими ИТ-продукты или услуги.

Принцип 5. Постоянное совершенствование (continuous improvement)

Постоянная адаптация к изменяющимся обстоятельствам (потребности заказчиков, требования законодательства, новые технологии). Важнейшая часть культуры DevOps – непрерывное сокращение потерь, оптимизация скорости, затрат, упрощение поставки и совершенствование продуктов и услуг. Неотъемлемая часть постоянного совершенствования – это экспериментирование и обучение на ошибках. Хорошее жизненное правило – делать чаще то, что получается плохо.

Принцип 2. Ориентация на конечный результат (create with the end in mind)

Необходимо отказаться от каскадной модели разработки программного обеспечения («водопада») и процессно-ориентированных моделей, в которых каждое подразделение и сотрудник выполняют только конкретную роль/функцию, не видя общей картины. Компании должны сфокусироваться на создании продуктов, которые будут поставляться реальным заказчикам, и у всех сотрудников должно быть инженерное мышление, которое требуется для представления этих продуктов.

Принцип 4. Кросс-функциональные автономные команды (cross-functional autonomous teams)

Команды должны быть полностью независимыми на протяжении всего жизненного цикла продукта/услуги, иметь сбалансированный набор компетенций и состоять из специалистов более широких профилей.

Принцип 6. Автоматизируется все, что можно (automate everything you can)

Думайте об автоматизации не только процессов разработки программного обеспечения (непрерывной поставки, включающей непрерывную интеграцию и непрерывное развертывание), но также всего инфраструктурного ландшафта (через управление версиями инфраструктуры и работе с ней как с кодом).

мально практичны, с минимумом красивых лозунгов. Наверно, поэтому они несильно пересекаются с шестью принципами, выделенными экспертами DASA.

Это весьма полезный набор принципов, однако его тоже нельзя считать идеальным. Можно отметить следующие особенности данного свода принципов:

1. Критерии выделения принципов неоднозначны, включены принципы, работающие с разнородными элементами системы управления – организация деятельности (поток создания ценности), культура мышления сотрудников (определение завершения и все должно храниться в системе контроля версий) и ИТ-инструменты (конвейер развертывания и автоматизированное управление конфигурациями). Хотя разнородность здесь уже меньше, чем в первом списке.
2. Инструменты автоматизации поставлены в один ряд с культурой мышления. Более того, из пяти принципов два связаны с обеспечением ИТ-деятельности. Безусловно, роль информационных технологий в DevOps очень высока. ИТ – это все-таки лишь инструмент достижения целей, а принципы, на наш взгляд, должны оперировать более высокоуровневыми понятиями.
3. Принципы пересекаются друг с другом. Хотя, возможно, именно в этом одна из особенностей данного свода принципов. Например, поток создания ценности (принцип 1) обеспечивает, в числе прочего, концентрацию сотрудников на предоставлении ценности для конечного заказчика,

но о том же прямо говорит и определение завершения (принцип 5).

4. Наконец, самое важное – список неполон, ряд важных идей остались за кадром. Например, из какого принципа следует практика постоянной обратной связи с конечным заказчиком, которая, очевидно, играет в DevOps важнейшую роль? На наш взгляд, наличие постоянной петли обратной связи, а если быть более точным, не одной, а нескольких петель – это важнейший элемент, необходимый для постоянного совершенствования. И из принципа определения завершения наличие петли обратной связи непосредственно не следует. Или на каком принципе основана практика создания кросс-функциональных команд? Конечно, можно сказать, что это следует из принципа фокусировки на всем потоке создания ценности, однако подобные логические связи слишком косвенны и представляются надуманными.

Поэтому, мы считаем, эти пять принципов нельзя назвать ключевыми для DevOps, хотя они, безусловно, являются важными. Отметим еще, что первые четыре принципа ближе к практикам и инструментам, а не к общим положениям, что и определяет особенности этого набора принципов – он выглядит максимально практичным.

Три принципа DevOps

Наконец, приведем еще один список принципов. Его автор – Джин Ким, один из авторов известного в мире DevOps-романа «Проект «Феникс»». В своей статье «Три пути: принципы, лежащие в основе DevOps» [\(Gene Kim](#)

Рис. 3.

Пять принципов DevOps, выделенных Олегом Скрынником.

Принцип 1. Поток создания ценности

Поток создания ценности – это последовательность действий, которые ИТ-организация осуществляет для удовлетворения запроса пользователя. Моделирование этого потока (картирование) и определение реальных метрик выполнения каждого шага этого потока дают наглядное представление деятельности и общий для всей группы контекст работы, позволяя концентрироваться на ценности для конечного заказчика. А также помогают выявлять и устранять узкие места.

Принцип 2. Конвейер развертывания

Под конвейером развертывания понимается комплекс инструментальных систем, максимально автоматизирующих проведение изменений в какой-либо прикладной системе, от разработки до запуска в продуктивной среде. Конвейер развертывания позволяет сократить время на проведение изменений, повысить качество и контроль хода изменений.

Принцип 3. Все должно храниться в системе контроля версий

Системы контроля версий применяются разработчиками довольно давно, однако в DevOps области использования таких систем расширяются на все, что связано с ИТ-системой. Это дает очень высокий уровень контроля за всем, что имеет отношение к ИТ-систем, причем с единой точки зрения.

Принцип 4. Автоматизированное управление конфигурациями

Этот принцип говорит о том, что любое изменение операционной, тестовой и разработческой среды выполняется только скриптами, хранящимися в системе контроля версий. Это ограничивает права администраторов по изменению параметров среды, прежде всего операционной.

Принцип 5. Определение завершения

Этот принцип говорит, что работа завершена не тогда, когда какой-то из сотрудников в цепочке создания ценности выполнил свою часть работы, а когда отработала вся цепочка и заказчик начал получать ожидаемую ценность.

Принцип 1. Системное мышление Фокусировка на производительности всей системы, в отличие от выполнения конкретной работы отдела или отдельного сотрудника. Основное внимание надо уделять потоку создания ценности для бизнеса, которое обеспечивает ИТ.	Принцип 2. Усиление петли обратной связи Создание быстрой петли обратной связи необходимо для непрерывного совершенствования процесса или системы.	Принцип 3. Создание культуры постоянного экспериментирования и обучения Это создание культуры, которая способствует постоянному экспериментированию, учету рисков и обучению на ошибках. Повторение и постоянная практика являются основой мастерства.
--	---	---

The Three Ways: The Principles Underpinning DevOps)

он выдвинул свои принципы (рис. 4).

Надо отметить, что эти три принципа (или три пути) DevOps лишены многих недостатков предыдущих списков. Они основаны на единой логике развития связей, которая хорошо показана на рисунках из оригинальной статьи Кима (рис. 5). Все они относятся к общей философии деятельности и носят культурный, а не инструментальный характер. Наконец, они не дублируют друг друга. Тем не менее и у такого свода принципов есть пара недостатков.

- 1. Недостаточная полнота. По крайней мере, ни из одного принципа не следует практика организации кросс-функциональных команд с широкими компетенциями или фокусировка на устойчивости и самовосстановлении системы, идет в противовес традиционной фокусировке на надежности.
- 2. Ни слова про принципы применения информационных технологий в DevOps. С одной стороны, это хорошо, поскольку нам представляется, что предыдущие перечни принципов уделяли этому моменту чрезмерное внимание. С другой стороны, может создаться впечатление, что все это реально реализовать без конвейеров развертывания и подобных технологий. Однако практика показывает, что нет, и это, по всей видимости, должно быть отражено в принципах.

Демократия DevOps

Существуют и другие своды ключевых принципов DevOps, но тех, что мы привели, достаточно, чтобы сделать несколько выводов. К сожалению, побывав на «острове» DevOps, мы не нашли ответа на, с нашей точки зрения, важнейший вопрос – можно ли выделить ключевые принципы, «ядро» этого подхода?

Единой точки зрения на ключевые принципы, или «ядро», DevOps нет. Разные эксперты выделяют различные наборы принципов DevOps, сделанные разными группами авторов, однако ни один из них не является общепринятым не только де-юре, но и де-факто.

- 1. DevOps нельзя считать сформировавшимся системным подходом или системой взглядов. В том, что представляет собой DevOps, не видно логически стройной системы, свойственной популярным «лучшим» практикам и стандартам.
- 2. В этом смысле на DevOps можно смотреть как на демократию, которая опирается на множество мнений. Возможно, именно это и есть один из принципов этого подхода, одна из его ценностей.

Шесть лучей DevOps

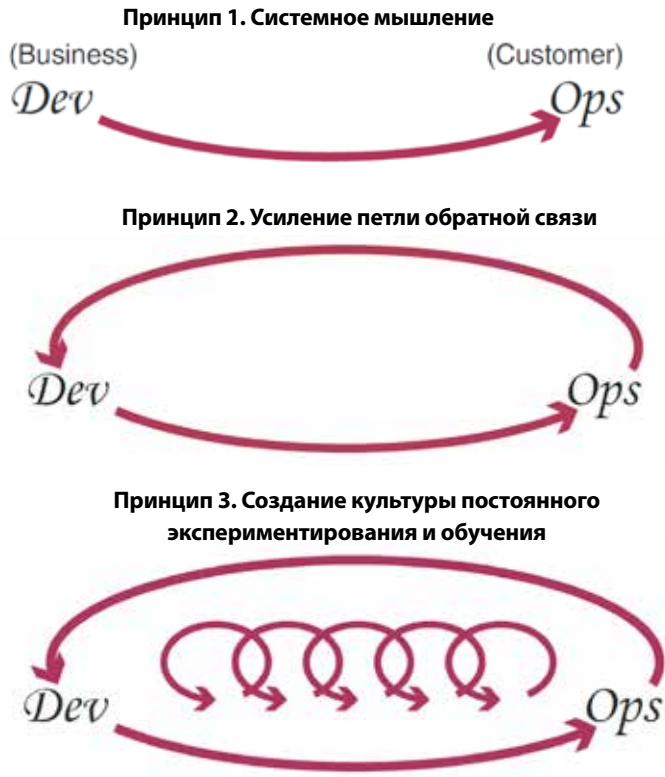
Только очень ленивые «дилетанты» не хотят изложить свое видение нового для них явления, описав это в путевых заметках о прекрасном путешествии. Наш вариант видения сегодняшнего «острова» DevOps не исключение, мы назвали его «Шесть лучей DevOps».

Он базируется на понимании, что подходы и практики, объединенные названием DevOps, родились в нестандартных, близких к экстремальным условиям, когда традиционные практики управления развитием и эксплуатацией не смогли обеспечить решение задач. В результате в этих условиях не получается опереться на методологически выверенный логичный подход, отталкивающийся от некоторого логически строгого набора принципов или некоторой модели.

Жесткость подобных моделей противоречит всему, что ценно в DevOps: высокой гибкости и экстремальной скорости изменений, бурному развитию технологий, культуре, ориентированной на неформальное взаимодействие и минимум ограничений. Жизнь показала, что применение привычной, «жесткой»

Рис. 4. Три принципа DevOps Джина Кима.

Рис. 5. Иллюстрация трех принципов DevOps Джина Кима.



методологии в условиях высокой неопределенности не привет к желаемому результату и не обеспечит развитие бизнеса и технологий.

структуры и методы управления, начнут тормозить дальнейшее движение по пути DevOps.

Согласованность практик DevOps

Если посмотреть на DevOps как на шесть направлений, то можно увидеть, что изменения и подходы, предлагаемые в каждом из направлений, внутренне по большей части согласованы и непротиворечивы. Однако все они должны развиваться сбалансированно, при этом согласованность и взаимодополняемость (или комплементарность²⁾) практик, использующихся в различных направлениях – это большой вопрос, который должен осознаться участниками сообщества. Сама по себе взаимодополняемость и непротиворечивость практик и подходов не возникнет.

Например, максимально широкое использование ИТ и обеспечение единой согласованной информации потребует определенного уровня формализации взаимоотношений и процедур. И непонятно, каким образом ИТ-инструменты должны сочетаться с самоорганизацией команд, ведь некоторые ограничивающие рамки в виде стандартов все равно будут необходимы. Как совместить свободу самоорганизации и формализацию взаимоотношений хотя бы в области взаимно необходимой информации?

Другой пример – подходы бережливости и постепенного развития отчасти противоречат креативности и смелости экспериментирования. Как они должны совмещаться и в каком соотношении находиться в «бульоне» DevOps? Каким образом обеспечить согласованность и непротиворечивость различных практик? Этот вопрос сегодня остается открытым. И, на наш взгляд, это основной вызов, на который должно ответить дальнейшее развитие DevOps.

В результате DevOps не может быть построен как комплекс тесно и логично связанных элементов. Вместо этого основной упор делается на создании условий для исследований и креативности, на доверии, уважении к людям и способности самоорганизации.

При этом мы уверены, что даже в самом творческом подходе должны проявиться ростки будущей системности и внутренней логики. Наиболее значимым источником информации стало для нас общение с коллегами на конференции DevOpsDays Moscow 2018, успешно занимающимися разработкой, внедрением и эксплуатацией ПО в России. Беседы с ними в значительной степени определили следующее видение сегодняшнего DevOps (по крайней мере, в России).

DevOps – это активно развивающийся набор новых подходов и практик, идущих одновременно по шести ключевым направлениям, перспективам или, если угодно, «лучам»: культуре, сотрудниках, структуре (организационной, ответственности и т.д.), методах управления, ИТ-инструментах и информации, а также взаимодействию с внешним окружением (рис. 6).

Однако развитие по этим шести направлениям происходит не независимо. Важно, что развитие DevOps не может быть успешным без одновременного и более-менее сбалансированного развития во всех шести направлениях. Если какое-то из направлений, например люди, будет развито сильнее, то другие –

Рис. 6.

Шесть «лучей» DevOps.



Кроме того, как это часто бывает, идеалы DevOps могут расходиться с реальностью. Методы управления, тяготеющие к известной «бирюзовой» организации, предполагают идеальные условия, которые пока труднодостижимы в большинстве компаний. Осознавая необходимость изменений и привлекательность культуры открытости и доверия, на практике мы осознаем тот факт, что применение этих принципов требует значительных ресурсов и времени для достижения прекрасных идеалов. Далеко не многие готовы пригласить в свою компанию идеального «слона» DevOps. DevOps, как философия и культура, неприменима в структурах, для которых важны традиционные подходы и ценности.

ITSM и/или DevOps: аргументы здравого смысла

Выбор подхода и принципов управления – типичная задача для ИТ-менеджера. Например, как будем управлять изменениями, инициируем проект или выполним эти изменения в рамках сопровождения? Во многих организациях этот выбор определяется формальными процедурами или решением руководителя. Однако это не спасает от проблем, которые могут возникнуть в связи с неудачным выбором. Подход DevOps привлекателен тем, что как бы исключает проблему выбора, позволяя ИТ органично изменяться вместе с бизнесом. Но при этом создает и ряд новых проблем...

Как отражение новых тенденций и «свежий» подход, DevOps привлекает обаянием комплексности и новизны, которое поддерживается маркетинговыми усилиями профессиональных сообществ и производителей ПО. Причем разнообразие уже зарекомендовавших себя подходов и практик дает ощущение легкости применения в конкретной организации: «Возьмем из лучшего опыта то, что нам нужно, и все получится! В DevOps наверняка найдется то, что нам подойдет и сейчас необходимо, получим результат и дальше посмотрим что-то еще!».

Мы не согласны с таким подходом, поскольку он несет значительные риски. Конечно, в «слоне» DevOps есть много интересных идей и практик, но вот подойдут ли он конкретной компании – это еще вопрос. Выше мы уже писали о сложности согласованности и непротиворечивости различных практик, и особенно сложно совмещать практики, возникшие в относительно стабильной среде (например, ITSM) и экстремальных условиях.

Более того, даже есть уверенность, что новые практики могут подойти и прижиться в организации, проведение таких изменений – отдельная и непростая задача. Изменение подхода к управлению в целом или даже изменение процессов управления

ИТ неизбежно приводит к дополнительным затратам и снижению эффективности в период проведения организационных преобразований. Не стоит забывать, что любое организационное преобразование требует от бизнеса инвестиций и встречает сопротивление, как со стороны ИТ-специалистов, так и других заинтересованных сторон.

Все направления DevOps должны развиваться более-менее сбалансированно, при этом согласованность и взаимодополняемость используемых практик – это большой вопрос.

Принимая решение о выборе того или иного подхода, которое неизбежно приведет к изменениям, нужна предварительная оценка необходимости и возможности предполагаемых изменений. Для этого, по нашему мнению, целесообразно задать себе несколько вопросов и поискать ответы на них в рамках своей организации:

1. Насколько нам нужна скорость в ущерб эффективности и стабильности? Чем мы готовы пожертвовать ради высокой скорости?
 2. Чем мы можем пожертвовать, совмещая DevOps с традиционным подходом ITSM, уже используемым в организации?
 3. Могут ли практики DevOps эффективно работать без обязательного изменения архитектуры ИС?
 4. DevOps требует изменения культуры, но готовы ли вы и ваши коллеги изменяться?
 5. Дадут ли мне денег на эти изменения?
- Ниже мы последовательно рассмотрим эти вопросы.

Насколько нам нужна скорость в ущерб эффективности и стабильности³? Чем мы готовы пожертвовать ради высокой скорости?

Скорость проведения изменений, в соответствии с изменяющимися требованиями бизнеса, является наиболее значимой ценностью от использования подходов и практик DevOps. Для достижения этого необходимо сделать значительные инвестиции в автоматизацию процессов тестирования и развертывания, а также тратить значительные ресурсы на устранение хрупкости инфраструктуры и управление техническим долгом. Но, прежде чем сделать все это, целесообразно провести детальный анализ направлений развития бизнеса, требований заказчиков и пользователей к ИТ. Цель анализа – понять, насколько важна скорость изменений и является ли она значимым требованием для развития бизнеса. Приведем два примера.

³Качество результата деятельности здесь мы предполагаем как минимум неизменным. Иначе зачем проводить такие изменения?

Пример 1. Стабильность важнее гибкости.

Основные бизнес-процессы организации автоматизированы с помощью достаточно стабильной сложной ИС, разработанной на основе платформенного ПО. В ней работает большинство пользователей. Эта система обеспечивает накопление и хранение значительных объемов информации. Хранение исторической информации обеспечивает формирование прогнозов, на осно-

Целесообразно провести детальный анализ и понять, насколько важна скорость изменений и является ли она значимым требованием для развития бизнеса.

ве которых принимаются важные управленческие решения и формируется корпоративная отчетность. Потребность в скорости изменений невысока, а потеря информации от неудачно выполненных изменений и снижение производительности критичны для основной массы пользователей. Понятно, что в такой ситуации стабильность ИС является главным бизнес-требованием, а необходимые в этом случае значительные инвестиции в конвейер развертывания не повлияют на развитие бизнеса за счет использования ИТ.

Пример 2. Гибкость важнее всего. Другой наиболее характерный пример – компания приняла решение о развитии направления интернет-торговли и выводе на рынок новой линейки продуктов и услуг. Основой ИТ для этого направления бизнеса является новый сайт компании. Состав продуктовой линейки не определен и постоянно меняется в соответствии со спросом на товары. Услуги «привязаны» к продуктам и выводятся на рынок вместе с товарами. Соответственно, постоянно изменяются требования к ИТ, связанные с продажами через новый сайт.

Для бизнеса крайне важно как можно скорее вывести на рынок новые товары и связанные с ними услуги. Заказчик готов инвестировать, понимая, что бизнес изменяется, а ИТ – основной инструмент достижения поставленных целей. Требования к новому сайту также постоянно меняются, но существуют в рамках ограничений, которые определены уже автоматизированными бизнес-процессами. Потери от срочных изменений соразмерны с потерями, связанными с неверными бизнес-решениями. Поэтому так важно быстрое исправление ошибок. В этой ситуации можно предположить, что преимущества от вывода на рынок новых продуктов и услуг покрывают затраты на DevOps-команду и обеспечивают доходы от нового направления бизнеса с требуемой эффективностью и допустимым уровнем рисков, связанных с изменениями.

Итак, на наш взгляд, скорость проведения изменений, безусловно, важный фактор, но ситуации, с которыми сталкиваются ИТ-менеджеры, достаточно разнообразны, и совсем не везде требуется такая скорость.

Чем мы можем пожертвовать, совмещая DevOps с традиционным подходом ITSM, уже используемым в организации?

В отечественной практике управления ИТ широко применяются подходы и практики ITSM. Во многих крупных и средних компаниях внедрены и используются: каталог услуг, процессы в соответствии с рекомендациями ITIL и требованиями международных стандартов по управлению ИТ-услугами. Эти организации сделали значительные инвестиции для перехода на сервисный подход в управлении ИТ, организацию процессов и их автоматизацию. Все это успешно работает, обеспечивая согласованный уровень предоставления ИТ-услуг. При таком подходе существует четкое разделение ответственности между бизнесом и ИТ-подразделением за проводимые изменения.

DevOps, напротив, стремится всячески объединить ИТ и бизнес, включить бизнес-сотрудников в команды. Но разделение на бизнес и ИТ – это привычно, это то, с чем мы живем. И далеко не всегда функциональные заказчики и потребители услуг готовы разделить с ИТ-специалистами ответственность за проводимые изменения и связанные с ними затраты, а также возможные потери. Специалисты функциональных подразделений не готовы участвовать в команде, даже если существует объективная потребность в большом количестве срочных изменений и применение гибких методов разработки может быть единственным решением этих проблем. Прежде всего, это связано с тем, что такой подход требует другого уровня корпоративной культуры, изменения привычных взглядов. Гораздо охотнее руководители функциональных подразделений включают в свои подразделения «собственного» разработчика, который будет «мгновенно» реагировать на возникающие запросы.

Отвечая «взаимностью» на такие желания, многие ИТ-руководители совершенно несклонны трансформировать поставленные и автоматизированные процессы управления изменениями и релизами. Однако организация DevOps-команд требует перемен в уже сложившихся взаимоотношениях с бизнесом, изменения каталога услуг и процесса управления уровнем услуг. Одной из существенных проблем является использование DevOps-командами других подходов к планированию работ, нормированию и учету рабочего времени. Все это усложнит управление ИТ-подразделением в целом. Особенно при организации проектов и выполнении комплекса работ специалистами нескольких подразделений. Например,

не все изменения конфигурации рабочих мест пользователей, которые необходимы для развертывания ПО и дополнительного оборудования, могут быть выполнены в автоматическом режиме.

На наш взгляд, чтобы избежать такого усложнения, придется чем-то жертвовать, отказываясь от каких-то инструментов, частей и элементов традиционных практик, а также от уже используемых средств автоматизации процессов. Более того, у нас есть опасения, что различные по своей культуре подходы – ITSM и DevOps – не смогут существовать вместе (см. «луч» культура на рис. 6). Хотя многие эксперты, как в ITSM, так и в DevOps, считают, что построенные по ITSM-практикам процессы можно модифицировать и совместить с практиками DevOps. Однако мы предполагаем, что «механически» совместить столь различные культуры и взгляды на ведение деятельности не получится. Культурные различия столь значительны, что, вероятнее всего, при совмещении подходы, а вслед за ними и практики, как ITSM, так и DevOps, придется существенно менять, приспособивая их друг к другу. Здесь мы, как и многие коллеги, возлагаем надежды на выход новой версии ITIL, которая даст ответ на эти и многие другие вопросы.

Могут ли практики DevOps эффективно работать без обязательного изменения архитектуры ИС?

Наиболее значимое техническое достижение DevOps – это конвейер развертывания, и ряд экспертов считают, что оно является одной из основ DevOps. Однако сложность его создания и эффективность использования в значительной степени зависят от архитектуры ИС. Развертывание изменений в монолитном классическом приложении требует значительно большего количества ресурсов и времени, чем изменений в приложении, которое имеет, например, микросервисную архитектуру. Кроме того, необходимо автоматизировать развертывание ряда тестовых сред, которые нужны как в ходе выполнения разработки, так и для проведения тестирования перед развертыванием в продуктивной среде.

Сложность автоматизации этих процессов в случае монолитного классического приложения может быть высока. Чтобы получить преимущества от создания конвейера развертывания, возможно, потребуется изменение архитектуры ИС, например переход на микросервисную архитектуру. Однако для больших корпоративных ИС это порой совершенно невозможно.

Именно поэтому анализ существующей архитектуры ИС, включая все компоненты, на предмет возможности реализации эффективного конвейера развертывания позволит принять технически обоснованное решение о применимости нового подхода.

DevOps требует изменения культуры, но готовы ли вы и ваши коллеги изменяться?

Применение новых подходов к управлению – это всегда организационное изменение, сопряженное с рисками противодействия, непринятия коллегами и пользователями, отсутствия поддержки. Практики DevOps основаны на определенной культуре работы в команде. Причем в эту команду входят не только разработчики и менеджеры, но и представители заказчика, специалисты, осуществляющие эксплуатацию ИС. Часто это очень разные люди, разговаривающие на разных языках, имеющие различные зоны ответственности, разную мотивацию.

Есть опасения, что различные по своей культуре подходы – ITSM и DevOps – не смогут существовать вместе. По крайней мере, «механически» совместить столь различные культуры и взгляды на ведение деятельности не получится.

Учитывая эти факторы, ИТ-менеджер должен спросить себя: «Готов ли я лично к таким серьезным изменениям? Готовы ли мои коллеги так меняться? Готов ли я требовать изменений от них?». Конечно, много говорится о самоорганизующихся командах, но никогда не стоит забывать, что организационные изменения не бывают успешными без поддержки со стороны руководства и бизнеса.

Дадут ли мне денег на эти изменения?

Наконец, последний вопрос – о деньгах и вероятности получения инвестиций на эти изменения. Даже если заказчик осознает бизнес-необходимость внедрения нового подхода, который требует не только организационных изменений и перемен корпоративной культуры, а значительных инвестиций в технологии, и последующего привлечения значительных ресурсов. Для достижения высокой скорости всегда требуется большая мощность! Оцените, насколько бизнесу это будет «по карману».

Конечно, как «дилетанты» мы не могли в этой статье дать исчерпывающие ответы на вопросы: есть ли ключевые принципы DevOps, насколько этот подход может стать системной методологией, «быть или не быть», то есть использовать или не использовать DevOps в условиях конкретной организации? Формулируя вопросы, мы только обозначаем области, в которых, по нашему мнению, можно поискать ответы на эти вопросы. ♦



**Синтия Харви
(Cynthia Harvey)**

Независимый писатель и редактор,
уже более пятнадцати лет пишущий
про информационные технологии
и отрасль ИТ.

Когда ИТ-подразделения переходят на подход DevOps, они часто сталкиваются с непростыми организационными и культурными проблемами. Любой заслуживающий внимания эксперт в области DevOps скажет вам, что простое использование инструментов DevOps не превратит вашу ИТ-группу в DevOps-команду. Вашей организации необходимы культурные изменения. И, увы, менять людей намного сложнее, чем технологии.¹

Девять человеческих проблем, отравляющих DevOps

При первой реализации подходов и инструментов DevOps большинство ИТ-групп сталкиваются с проблемами, связанными с культурой и/или структурой их организации. Эти проблемы могут препятствовать гибкости, затруднить взаимодействие между разработчиками и операционным отделом, что является отличительной чертой подхода DevOps.

К счастью, препятствия можно преодолеть, и организации, успешно решающие эти проблемы, получают значительные преимущества. В отчете [Interop ITX 2017 State of DevOps](#)² компании, перешедшие на DevOps, отметили, что они улучшили совместную работу (46%), ускорили развертывание программного обеспечения (39%), сократили время на обслуживание приложений (39%), улучшили каче-

¹Cynthia Harvey. [9 'People Problems' Plaguing DevOps](#). Статья предоставлена порталом ReallTSM и компанией Cleverics. Публикуется с некоторыми изменениями.

²Interop – международная ежегодная выставка-конгресс, посвященная информационным технологиям, проводимая в нескольких странах.

ство и производительность приложений (38%), снизили затраты (25%) и увеличили доходы (20%).

Такие исследования и другие опросы отражают некоторые из наиболее распространенных «человеческих проблем», негативно влияющих на работу организаций, переходящих на DevOps. Ниже мы рассмотрим эти проблемы и некоторые общие пути их решения.

Отсутствие экспертизы

В опросе Interop организациям, не планирующим использовать методологии или инструменты DevOps, предложили назвать основные причины, которые привели к такому решению. Почти треть (31%) респондентов заявили, что использованию DevOps препятствует нехватка опыта. Аналогично, когда ИТ-директоров попросили назвать основные проблемы реализации стратегии DevOps в своих организациях, 23% сказали, что им не хватает правильных навыков в областях развития и операций. Для организаций, которые оказались в этой ситуации, помочь ликвидировать недостаток квалификации своих команд могут программы обучения DevOps, или же они могут нанять новых сотрудников, уже знакомых с DevOps.

Организационная сложность

Еще одна проблема DevOps, которую отмечают 22% респондентов опроса Interop, – это организационная сложность. Большинство предприятий, особенно тех, что обладают крупными ИТ-командами, имеют структуру, разработанную для традиционных водопадных методов разработки. Часто даже названия должностей и должностные инструкции ИТ-отдела способствуют созданию барьеров, разделяющих разработку и операционный департамент.

Поскольку изменить уже существующую организационную структуру сложно, многие эксперты и DevOps-консультанты рекомендуют экспериментировать, при этом начинать лучше с небольшого участка, охватывающего маленькую часть ИТ-персонала. Подбираются инструменты и процессы DevOps, которые подходят для организации, и тогда можно постепенно масштабировать DevOps-инициативы, вовлекая в них все больше членов команды.

Плохое лидерство

Отчет 2017 State of DevOps, который опубликовали DevOps-вендоры Puppet и DORA, показал, что DevOps-команды наиболее успешны, когда у них есть «трансформирующие лидеры», приверженные подходу DevOps. Отчет определяет трансформирующих лидеров как тех, кто имеет видение, вдохновляет,

стимулирует, поддерживает, а также кого признают в качестве лидера. Отчет обнаружил, что у высокопроизводительных команд с большой долей вероятности есть трансформационные лидеры, а у малоэффективных команд – нет. Резюме: «Достичь успеха гораздо легче, если у вас есть поддержка эффективных лидеров».

Но в отчете также есть и предостережение: «Мы обнаружили доказательства того, что наличие лидеров с трансформационными характеристиками недостаточно для достижения высоких результатов DevOps ... Лидеры не могут самостоятельно достичь результатов. Успех DevOps также зависит от подходящей архитектуры, хорошей технической практики, использования принципов бережливого управления и всех других факторов, которые мы изучали на протяжении многих лет».

Большинство предприятий имеют структуру, разработанную для традиционных водопадных методов разработки. Часто даже названия должностей и должностные инструкции ИТ-отдела способствуют созданию барьеров, разделяющих разработку и операционный департамент.

Отсутствие независимости

Еще один вывод из отчета Puppet и DORA – DevOps-команды наиболее успешны, когда они обладают высокой степенью автономии. Команды, скорее всего, достигнут цели в области непрерывной поставки приложений, когда у них будет возможность проводить масштабные изменения в системах, завершать их работу и разворачивать свои приложения, не получая разрешения от кого-то вне команды. С другой стороны, организации, которые должны следовать требованиям, накладываемым на них руководителями вне команды, часто создавали «продукты, которые на самом деле не радуют и не привлекают клиентов, и не дают ожидаемых результатов в бизнесе».

Авторы отчета добавляют: «Наш анализ показал, что способность команды тестировать новые идеи, создавать и обновлять спецификации в процессе разработки (без необходимости одобрения за пре-

делами команды) является важным фактором для прогноза эффективности организации, измеряемой с точки зрения прибыльности, производительности и доли рынка».

Негибкие процессы

В опросе Coleman Parkes Research 29% лидеров бизнеса указали на «слишком жесткие процессы» как ключевое препятствие для реализации DevOps. Любая DevOps-инициатива требует от команд работать по-новому и предполагает, что у них есть возможность для устранения или изменения процессов, не отвечающих целям DevOps-инициативы.

Вероятно, именно поэтому 39% респондентов в исследовании Interop заявили, что планируют перепроектировать процессы, чтобы продвинуть свои DevOps-практики.

DevOps-команды наиболее успешны, когда они обладают высокой степенью автономии. Команды, скорее всего, достигнут цели, когда у них будет возможность проводить масштабные изменения в системах и разворачивать свои приложения, не получая разрешения от кого-то.

Организационная несогласованность между разработкой и операциями

Еще одним серьезным препятствием для DevOps является отсутствие организационной согласованности между группами разработчиков и операционным отделом. В опросе Interop почти четверть (23%) респондентов заявили, что большой проблемой для реализации DevOps в их организациях было то, что «роли и обязанности в рамках разработки и операций не согласованы».

Достижение успеха в использовании подходов и инструментов DevOps может потребовать сложных структурных изменений, таких как переопределение ролей и обязанностей или изменение организационной структуры. Однако единовременное проведение радикальных изменений не лучший подход для крупных команд. Начиная с небольшой команды и постепенно масштабируя практики, можно сделать переход на DevOps более мягким к членам команды и в конечном счете более успешным.

Соппротивление изменениям

В любой команде есть люди, которые охотно пробуют что-то новое. Это может быть просто чертой характера или обусловлено желанием сохранить статус-кво. Опрос, проведенный Coleman Parkes Research, показал, что 34% ИТ-лидеров считают, что «культура и мышление/сопротивление изменениям» были основным препятствием для принятия DevOps.

Для этих организаций эксперты рекомендуют говорить о потенциальных преимуществах DevOps, а также использовать стратегии управления изменениями для достижения взаимопонимания до и во время перехода к методам и практикам DevOps.

Отсутствие общих инструментов и обмена знаниями

Одним из отличительных признаков DevOps является то, что разработчики и операционный отдел тесно взаимодействуют и сотрудничают. Однако достичь этих тесных рабочих отношений может быть затруднительно, если организации не имеют инструментов и процессов, способствующих сотрудничеству. **Опрос, проведенный компанией Atlassian**, показал, что 80% организаций используют общие инструменты для команд разработчиков и операционных отделов. Это хороший первый шаг к сотрудничеству.

Однако у большинства еще нет инструментов, которые бы облегчали обмен знаниями между этими двумя группами. В отчете говорится, что только 17% компаний динамически предоставляют открытую информацию смежным подразделениям, например на сайте или на форуме. А в подавляющем большинстве компаний делятся информацией только по запросу и только в статичном формате.

Новые барьеры

Философия DevOps направлена на устранение существующих барьеров в ИТ-подразделениях. Однако иногда компании, переходящие на DevOps, обнаруживают, что они создают новые барьеры. Эти организации создали небольшие команды для экспериментов с практиками и инструментами DevOps, но потом не смогли их масштабировать. В результате преимущества DevOps были небольшими и изолированными в пределах этой маленькой группы.

В отчете Coleman Parkes Research отмечается: «Хотя применение DevOps продолжает расти, его обычно применяют на небольших участках. Это приводит к небольшому обмену знаниями и лучшими практиками. Чтобы действительно быть эффективным, «DevOps-мышление» должно быть встроено в культуру ваших ИТ». ♦



**Майк Даффи
(Mike Duffy)**

Консультант DevOps имеет опыт работы с разными компаниями, от небольших до самых крупных. Сотрудничал с Honeycomb TV, Capito Systems, Universal Music Group, Clifford Thames, BSkyB. Автор книг DevOps Automation Cookbook и Learning DevOps: Continuously Deliver Better Software.

Подход DevOps позволил командам разрабатывать и обновлять программное обеспечение, делать это быстро, легко и без традиционных разногласий (или стычек) между «стражами райских врат» – операционной командой и наступающими ордами разработчиков. Находясь на гребне волны инноваций в автоматизации, мониторинге и создании ПО, DevOps способствовал практически повсеместному использованию виртуализации и контейнеризации. В широком смысле DevOps стал глотком свежего воздуха в отрасли, последним крупным изменением подходов в которой была методология Agile. Я долгое время усердно внушал людям веру в его чудеса. Но теперь для DevOps пришло время умереть.¹

Мы должны убить DevOps

Прежде, чем вы решите, что я сошел с ума, послушайте меня. DevOps, как методология, имеет крайне важное значение для поставки современных технологий. Мы не должны возвращаться в «обнесенные стенами сады прошлого» – туда, где функционально разделенные технологические департаменты смотрят друг на друга сквозь туман недоверия и взаимных сомнений в квалификации и иногда подбрасывают друг другу релизы. Методология DevOps

превосходна и в настоящий момент созрела для расширения с помощью других методологий, например таких, как MetOps.

Однако использование аббревиатуры DevOps в названии должности или в качестве дополнения к ней нужно прекратить. Помимо нелепости самой идеи назвать должность в честь методологии (вы же, например, не называете сотрудников «гибкими») это

¹Mike Duffy. [We need to kill DevOps](#). Статья предоставлена порталом RealITSM и компанией Cleverics. Публикуется с некоторыми изменениями.

противоречит ключевому принципу DevOps: объединение разработчиков и специалистов по эксплуатации. Называя кого-то «DevOps» или «DevOps Engineer», вы по-прежнему делаете их «другими», хотя они сидят там же, где и разработчики.

В результате для большинства разработчиков крупных предприятий ничего не изменилось. У них, как и раньше, есть «страж райских врат», просто теперь он ближе к разработчикам. Во многих организациях DevOps-инженеры преобразовались в отделы мини-операций. Они все так же «проталкивают» релизы, просто теперь на более короткой дистанции.

DevOps-инженеры являются универсалами. Избавляясь от операционных подразделений, вы теряете глубокие и сконцентрированные на инфраструктуре навыки.

Это, может, и создаст разногласия внутри команды, но традиционное противоборство разработки и эксплуатации теперь принимает вид столкновений внутри маленьких команд, а не глобальных «войн». Все это вызывает неприятный побочный эффект в виде появления у технического директора уверенности в том, что отдел эксплуатации не нужен. Не правда ли, если у нас столько людей с приставкой «Ops» в должности, то у нас нет необходимости в дополнительных ops-департаментах?

Но проблема в том, что DevOps-инженеры являются универсалами. Избавляясь от операционных подразделений, вы теряете глубокие и сконцентрированные на инфраструктуре навыки, а также дробите функции, которые должны быть централизованными. Каждая команда, с которой я работал, прошла по этому пути и в результате столкнулась с серьезными проблемами, вызванными тем, что в узкой области работали специалисты-универсалы.

Как же мы оказались в такой ситуации, если все пели дифирамбы DevOps и автоматизации? Как правило, это связано с корпоративной культурой и инерцией. В какой-то момент разработчики заклеили как слишком небезопасных, чтобы быть допущенными к инфраструктуре беспрепятственно и без надзора. Аргумент, во многом аналогичный тому, что никто не должен иметь доступа к созданию кода приложения и его выпуску в продуктивную среду без должного контроля. Это давно было обеспечено с помощью

современного SDLC (жизненного цикла разработки программного обеспечения). Но разве автоматизация не обещала полное устранение этой проблемы и наступление эры «свободы разработчика»? Не в этом ли весь смысл методологии DevOps?

Именно так, автоматизированная инфраструктура и выпуск ПО могут управляться так же, как программный код. Код инфраструктуры специфицируется бизнес-аналитиком, проверяется архитектором, реализуется разработчиком вместе с модульными тестами и механизмами мониторинга, контролируется службой качества и, наконец, пройдя через различные тестовые среды, выпускается. Когда необходимо добавить новую функцию в инфраструктуру, формируется отдельная ветвь разработки, функция разрабатывается, затем делается запрос на развертывание, который проверяется и принимается. Нет никакой причины для того, чтобы ограничивать разработчика приложения в возможностях по созданию отдельной ветви разработки кода инфраструктуры, внесению улучшений и регистрации запроса на развертывание. Этот запрос затем может быть рассмотрен разработчиками инфраструктуры. Если это «хорошее» изменение, оно принимается и развертывается. Если его необходимо доработать, разработчики инфраструктуры дадут обратную связь, и разработчик приложения сможет внести корректировки и повторно отправить запрос на развертывание. Если запрос отклонен, разработчики инфраструктуры должны открыто объяснить причины.

Итак, если все это сейчас возможно, почему у многих разработчиков по-прежнему есть ощущение, что они ждут чего-то от «DevOps-команды»? На мой взгляд, причин тут две.

В первую очередь, многие компании, утверждающие, что они следуют методологии DevOps и развивают автоматизацию, по факту этого не делают. Довольно часто они лишь прикрываются DevOps-инженером в команде (или, что еще хуже, самой DevOps-командой, как фиговым листом, чтобы показать, что они молодцы и идут в ногу со временем). Копните глубже, и вы увидите все те же старые процессы. В результате вместо операционного отдела у таких компаний масса бегущих операционных мини-групп, которые контролируют отдельные компоненты и которым предоставлен эксклюзивный доступ к инфраструктуре, причем для создания компонентов вручную. Я все еще жду встречи с компанией, которая может объективно утверждать, что полностью все автоматизировала. В реальности на определенном уровне, будь то обновление записей DNS, управление сетью или что-то еще, автоматизации нет и сотрудникам нужно преодолевать бюрократические преграды, чтобы вносить изменения в эту часть инфраструктуры. Надо заметить,

что виноваты в этом не DevOps-инженеры компаний. Однако такие бюрократические преграды воспринимаются как операционный «тормоз».

Вторая причина – это нетерпеливые разработчики. Все хорошие разработчики хотят поставить пользователям код, который, конечно, является функциональным и безопасным, а кроме того, классным и «крутым». Их невероятно разочаровывает, если они не могут делать этого из-за ожидания новых вычислительных мощностей, средств хранения данных, создания сетей и т.д. Однако DevOps-инженеры часто защищают этих разработчиков от большей части корпоративной бессмыслицы. В результате разработчики не осознают, что неотъемлемые сложности бухгалтерского учета, обеспечения безопасности или соблюдения политик делают трудным лишь путь к той точке, в которой вы сможете «развернуть несколько серверов» в консоли AWS.

Другая проблема заключается в том, что разработчики приложений могут «зависать» в ожидании реализации разработчиками инфраструктуры новых возможностей, необходимых для дальнейшего движения. Иногда их реализация может быть сложнее, чем создание новой функции приложения, может включать уточнение требований и упрощение существующего бизнес-процесса, прежде чем его можно будет автоматизировать. Когда автоматизация становится возможной, необходимый код, как правило, оказывается достаточно тривиален, но при этом заставляет пересмотреть сформировавшиеся за десятилетия подходы к выделению бюджетов на создание новых мощностей или к управлению DNS-записями.

Решение этой проблемы заключается в привлечении разработчиков приложений. Если разработчик приложения вынужден приостановить свою работу в ожидании новой функции инфраструктуры, появление которой задерживается вследствие затянутого процесса, то позовите разработчика приложения на встречи, где обсуждается данный процесс. Это поможет ему осознать существующие нетехнические барьеры и лучше понять процесс.

Итак, как нам сотрудничать эффективнее? Я бы посоветовал начать с распространения «сверху-вниз» следующих правил:

1. Есть только разработчики. Нет DevOps, QC и прочих. Существуют разработчики инфраструктуры, разработчики приложений, разработчики тестов и так далее.
2. Все изменения инфраструктуры осуществляются с помощью кода, и весь код инфраструктуры доступен для разработки.

Первое правило избавляет нас от синдрома противопоставления «мы-они». Нет различных ролей, есть

разработчики с различной специализацией. Они управляются одинаково, следуют тем же процессам, что и другие разработчики, и просто являются членами команды разработчиков. Если разработчик приложения не может выполнить свою работу из-за отсутствия инфраструктуры, это просто случай плохого управления спринтом; необходимая функциональность инфраструктуры должна быть идентифицирована, оценена и реализована до того, как ее отсутствие станет барьером. То же относится к любой функции приложения. Такое изменение процессов не будет слишком затруднительным.

Второе правило реализовать несколько сложнее. Когда мы говорим, что единственный способ изменения инфраструктуры – с помощью кода, это правило должно действовать в рамках всего предприятия. Хотите новую DNS-запись верхнего уровня? Это изменение кода. Хотите выделить новое устройство хранения? Код. Хотите завести нового пользователя? Тоже код. И так далее.

Нет причин, по которым вы не можете разрешить любому разработчику в компании изменять код и выдавать запрос на его развертывание, если эти элементы автоматизированы. Почему бы не позволить добавлять нового пользователя в рамках проектов разработки? Ведь проекты регистрируются, включают в себя автоматизированный аудит, могут быть скорректированы и отклонены. Почему бы не позволить добавлять ресурсы верхнего уровня через код? Нет никаких непреодолимых или просто веских причин, почему этого нельзя делать. Предлагаемый подход избавит вас от недостатка автоматизации внутри компании, а также будет

**Есть только разработчики.
Существуют разработчики
инфраструктуры, разработчики
приложений, разработчики тестов
и так далее.**

стимулировать сотрудничество с операционными командами, которые следят за инфраструктурой. Если код является «lingua franca»² при осуществлении изменений, происходящее и его причины прозрачны и понятны для всех заинтересованных лиц.

Итак, примите DevOps-методологию и искорените DevOps-должности. Теперь мы все разработчики, просто отличаемся специализацией. Мы все хотим одного и того же – возможности сосредоточиться на написании классного программного обеспечения, которое делает удивительные вещи. ♦



Всеволод Шадрин

Руководитель проектов и менеджер продукта Naumen ITAM. Опыт работы в сфере информационных технологий более 9 лет. Имеет сертификаты ITIL OSA, RCV и IAITAM CHAMP. С 2012 года руководит проектами по разработке и автоматизации управления ИТ в компании NAUMEN. Выполнял проекты для компаний СО ЕЭС, Интер РАО, Мособлгаз, МОЭСК, Мосэнергосбыт.

Не раз в проектах приходилось сталкиваться с идеалистическим подходом, когда стремятся работать по принципу «внедряем всё и сразу». Практика показывает, что с многофункциональными ИТ-системами это сложно выполнимая задача. Всегда предлагаю есть такого «слона» по частям, поэтапно. Универсальной последовательности этапов внедрения не существует, но выделяется костяк, который подходит большинству компаний. Не вошедшие в скелет проекта процессы будут индивидуальными особенностями реализации. Давайте вместе разберемся в следующих вопросах: с чего начинается внедрение процессов управления ИТ-активами, из каких шагов оно состоит и каковы точки роста.

Есть С.Л.О.Н.а по частям. Внедрить ITAM и не подавиться

Что говорит статистика

Интерес к IT Asset Management (ITAM) стабильно растет. Это подтверждает мировая статистика. Например, в результатах исследования «Управление ИТ-услугами следующего поколения» от аналити-

ческой компании Enterprise Management Associates отмечен стабильный рост внимания к вопросам автоматизации процессов управления ИТ-активами со стороны бизнеса (рис. 1). Посмотрим, как большой «слон» ITAM чувствует себя в российской практике.



Какие компоненты входят в ИТАМ-практику

Сначала разберу, на что похож «слон» – так условно называю совокупность компонентов, из которых состоит ИТАМ-практика. Вспоминается древняя индийская притча о слоне и шести слепцах.

Первый слепой протянул руку и коснулся бока слона: «Какой гладкий! Слон похож на стену». Второй слепой протянул руку и коснулся хобота слона: «Какой круглый! Слон похож на змею». Третий слепой протянул руку и коснулся бивня слона: «Какой острый! Слон похож на копье». Четвертый слепой протянул руку и коснулся ноги слона: «Какой высокий! Слон похож на дерево». Пятый слепой протянул руку и коснулся уха слона: «Какой широкий! Слон похож на опахало». Шестой слепой протянул руку и коснулся хвоста слона: «Какой тонкий! Слон похож на веревку».

Чтобы не быть похожими на слепцов, посмотрим общим взглядом на ИТАМ-«слона» и разложим его на ключевые компоненты (см. таблицу). Их можно назвать С.Л.О.Н. ИТАМ.

От чего зависит выбор модели управления ИТ-активами

Практика управления ИТ-активами имеет множество аспектов и не каждой компании требуется внедрять абсолютно все подходы и области управления. Важнее определить, для чего в компании реализуется управление ИТ-активами, какие цели ставит перед собой конкретная компания. Это напрямую влияет на «внешний вид» С.Л.О.Н.а, а значит и на выбор последовательности шагов и итоговую картину модели управления ИТ-активами.

Опираясь на свою практику внедрения ИТАМ-решений, предложу следующий набор целей, которые можно задать на старте:

Рис. 1. В каких ситуациях наиболее важно использование CMDB/CMS?

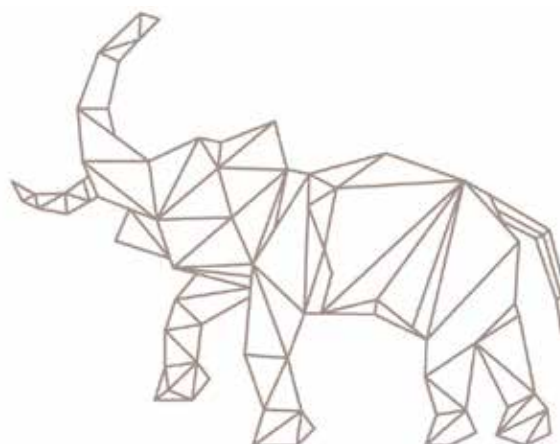
С.	Соглашения , которые определяют: • процессы • внутренние и внешние договорные отношения • фиксирующие документы (например, факты затрат на ИТ-активы) • политики для установки правил учета ИТ-активов или отдельных операций по ним • бюджеты на ИТ-активы.
Л.	Лицензии . Учет и управление лицензиями.
О.	Оборудование . Учет рабочих мест, периферии, серверов.
Н.	Нематериальные активы . Учет виртуальной инфраструктуры, софт, знания.

Таблица. Ключевые компоненты ИТАМ – С.Л.О.Н.



1. Минимизировать внеплановые закупки ИТ-активов при оптимальном размере складского запаса.
2. Снизить процент секвенции бюджета при ежегодном планировании ИТ-бюджета, как следствие – улучшить обоснование затрат на ИТ.
3. Сократить время на получение актуальной информации по ИТ-активам.
4. Оптимизировать затраты на лицензирование.

Достижению этих целей поможет автоматизация функциональных частей ИТАМ. Их специфику рассмотрим далее.



С.Л.О.Н.а можно «съесть», но не целиком и не за один подход.

Из каких частей состоит управление ИТ-активами

Поделю С.Л.О.Н.а ИТАМ на крупные части:

- идентификация и отслеживание изменений по ИТ-активам;
- планирование закупок и списания ИТ-активов;
- построение ресурсно-сервисной и ресурсно-финансовой моделей учета ИТ-активов;
- управление лицензиями и поставщиками.

Разберем эти части в формате «цели → шаги → результаты».

Идентификация и отслеживание изменений по ИТ-активам

Первым базовым этапом при внедрении ИТАМ должна стать агрегация информации об ИТ-активах компании. Ставим цели:

- сформировать централизованный источник актуальной информации об оборудовании, ПО и т.д.;
- получить актуальную информацию об ИТ-активах, включая: количество, виды, учетные параметры и местоположение;
- установить связь ИТ-активов с основными средствами как мостик для дальнейшего развития.

Для реализации целей двигаемся по шагам с учетом ITSM-процессов (рис. 2.):

1. Выбираем средство автоматизации.
2. Разрабатываем классификатор ИТ-активов.
3. Составляем сценарии обновления базы ИТ-активов.
4. Проводим инвентаризацию и наполняем базу ИТ-активов.
5. Устанавливаем связь активов и основных средств (нематериальные активы, малоценка).

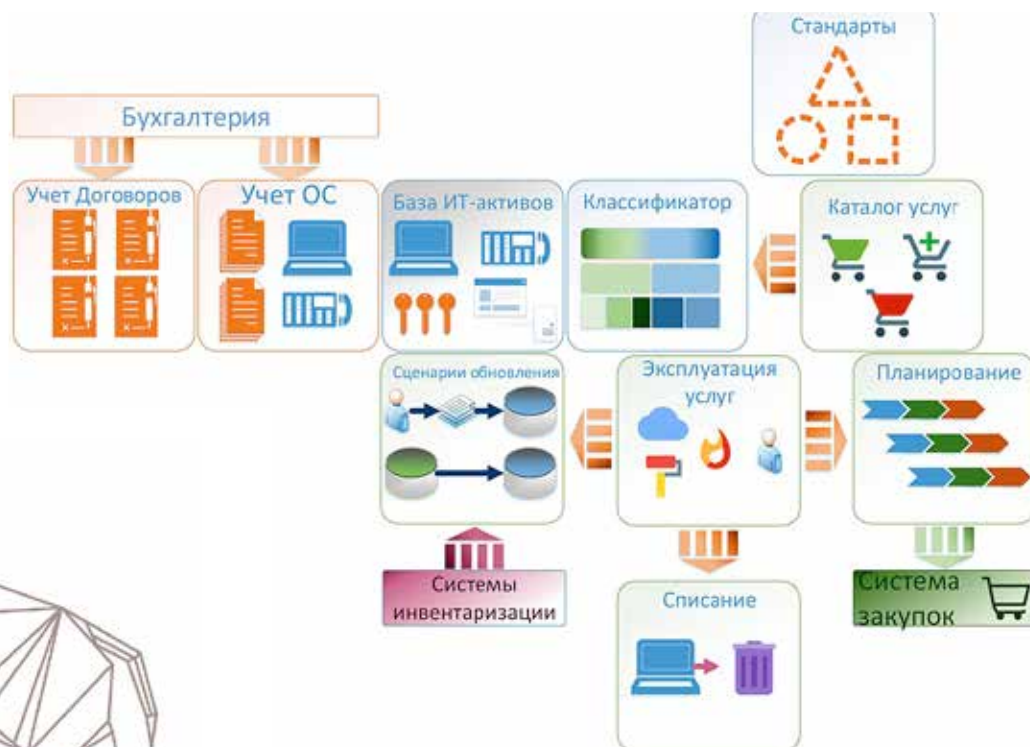
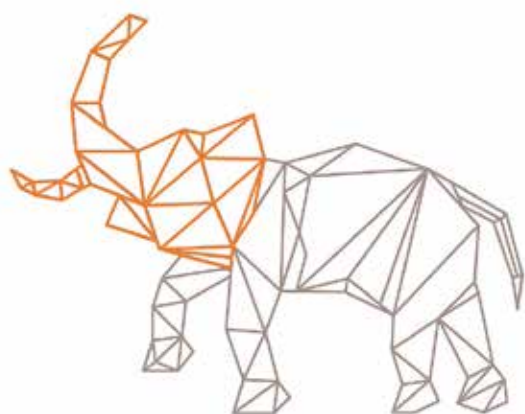
Важно отметить, что управление ИТ-активами рассматривается мной в постоянной связке с каталогом услуг (в рамках которых осуществляется поддержка ИТ-активов внутри ИТ) и процессами эксплуатации услуг (управление инцидентами, запросами на обслуживание и изменениями).

Рис. 2.
Идентификация
и отслеживание
изменений
по ИТ-активам.



В результате работа со сформированным единым источником актуальной информации об ИТ-активах позволит:

- повысить утилизацию оборудования;
- уменьшить количество закупаемого оборудования;
- упростить процесс инвентаризации;
- сократить время на поиск нужной информации.



Построение ресурсно-сервисной и ресурсно-финансовой моделей учета ИТ-активов

Далее расширяем вектор использования накопленной информации об ИТ-активах: учитываем все договоры с подрядчиками, в рамках которых несем затраты на ИТ-активы, связываем ИТ-активы с Основным средством. Чего хотим достичь:

- получить информацию о целях использования ИТ-активов (работу какой услуги они поддерживают);
- собрать информацию о стоимости владения ИТ-активом и себестоимости предоставления услуг;
- разработать механизмы планирования затрат в рамках бюджетных статей.

Планирование закупок и списания ИТ-активов

Допустим, мы накопили и сформировали базу ИТ-активов, наладили процесс актуализации информации в ней. Далее наступает этап управления жизненным циклом ИТ-активов и контроля всех процессов, связанных с их планированием, поступлением и выбытием.

Ставим цели:

- оптимизировать процесс закупки ИТ-активов;
- повысить контролируемость процесса списания ИТ-активов.

И идем по шагам (рис. 3.):

1. Проводим стандартизацию ИТ-активов.
2. Разрабатываем сценарии планирования закупок.
3. Интегрируемся с системами закупок.
4. Прописываем сценарии списания ИТ-активов.

Эффектами выполнения указанных шагов будут:

- консолидация закупок и получение выгодных цен со стороны подрядчиков;
- оптимизация складского запаса;
- консолидация по договорам поддержки;
- упрощение механизмов контроля за многообразием видов техники;
- упрощение операций с ИТ-активами.

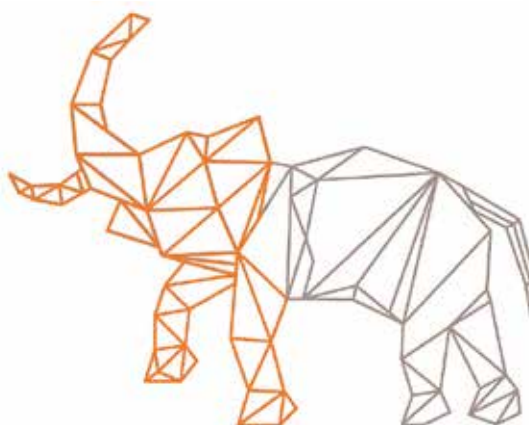


Рис. 3.
Планирование закупок и списания ИТ-активов.

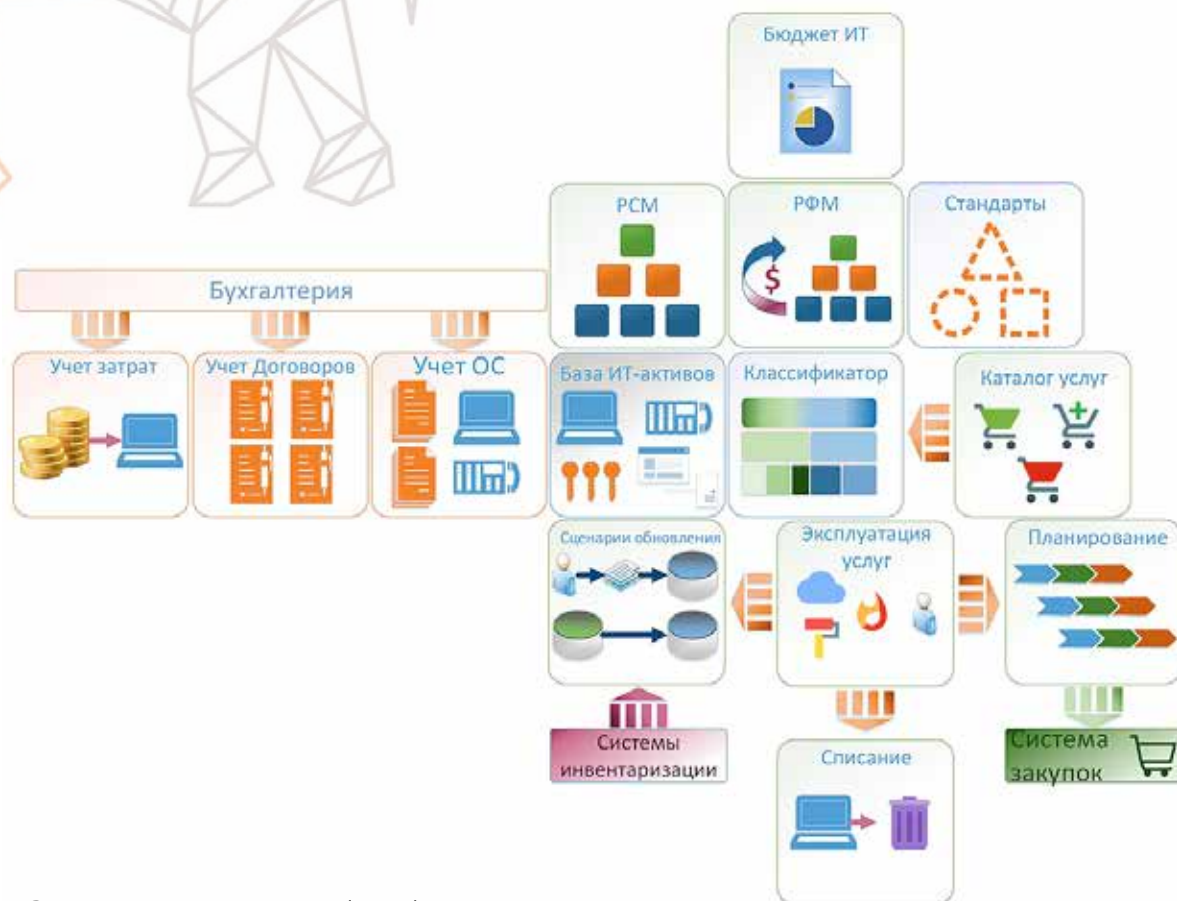


Рис. 4.
Построение
ресурсно-сервисной
и ресурсно-
финансовой моделей
учета ИТ-активов.

Выполняем следующие шаги (рис. 4):

1. Загружаем информацию по договорам, затратам, амортизации и остаточной стоимости ИТ-активов из бухгалтерских систем (связываем через Основное средство).
2. Вносим информацию о связях между ИТ-активами, тем самым формируем ресурсно-сервисную модель.
3. Используем связи ресурсно-сервисной модели и информацию из бухгалтерских систем, чтобы сформировать финансовую модель распределения затрат по ИТ-активам.
4. Планируем бюджет на ИТ.

Итоги контроля целей использования ИТ-активов:

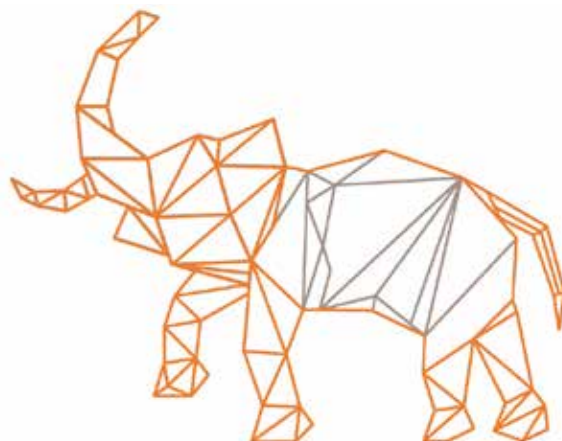
- снижение холостой работы ИТ-активов;
- контроль критичных ИТ-активов;
- улучшение утилизации активов.

Следствие оптимизации расходов на ИТ:

- выделение дорогих услуг, направлений деятельности и операций;
- принятие решения об аутсорсинге;
- улучшение планирования бюджета;
- контроль исполнения бюджета.

Управление лицензиями и поставщиками ИТ-активов

Важным этапом является работа с поставщиками ИТ-активов. Это отдельная часть ИТАМ-«слона», где учитываются все обязательства поставщиков, дого-



воры с ними, включая условия поддержки. Вновь определяем цели:

- оптимизировать затраты на лицензии;
- организовать процесс контроля за поставщиками.

Что делаем (рис. 5):

1. Собираем данные по лицензиям.
2. Формируем лицензионную модель.
3. Унифицируем правила взаимодействия с поставщиками.
4. Проводим категоризацию и измерение качества сотрудничества с поставщиками.

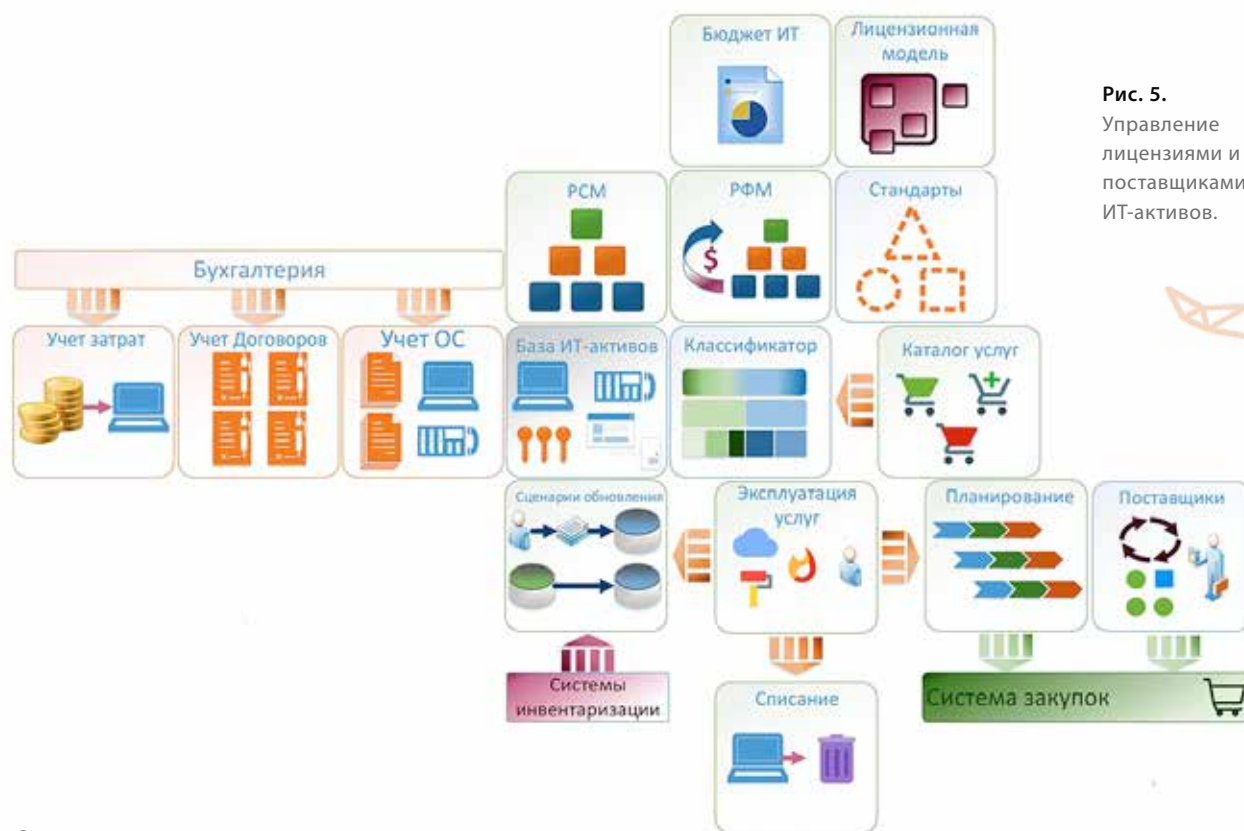


Рис. 5.
Управление
лицензиями и
поставщиками
ИТ-активов.

Следствием оптимизации затрат на лицензии будут:

- консолидация лицензий, поиск более выгодных схем лицензирования;
- сокращение количества лицензируемых продуктов, их унификация;
- улучшение условий лицензирования за счет консолидации информации;
- повышение утилизации лицензий.

При организации контроля за поставщиками получаем:

- измеримость и прозрачность процессов работы с поставщиками;
- экономию затрат за счет консолидации и стандартизации;
- упрощение операций по взаимодействию с поставщиками;
- оперирование договором как самостоятельным активом.

Развитие системы управления ИТ-активами

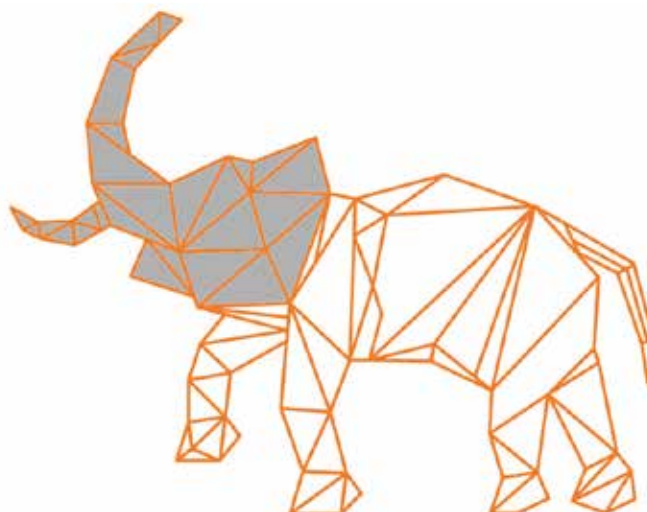
Когда крупные части С.Л.О.Н.а внедрены и автоматизированы, можно задуматься над развитием ИТ-системы. К точкам роста я отношу:

- 1. Разработку политик** – от требований к поставщикам к внутренним требованиям. Поддержку и развитие корпоративных стандартов по ИТ-активам.
- 2. Использование ИТ-активов в проектной деятельности** – услуги, инсталляции, лицензии.

3. Управление знаниями в ИТ-активах – база статей с типовыми решениями инцидентов и запросов на обслуживание, документация, инструкции, F.A.Q.

В итоге получаем многокомпонентную картину собранных воедино и взаимосвязанных частей ИТАМ-«слона».

Многокомпонентный характер процесса управления



Лаборатория ITSM-решений



itSMF России приглашает профессионалов ITSM принять участие в воркшопе «Лаборатория ITSM-решений».

Формат

Анализ существующих форматов коммуникаций внутри сообщества ITSM показал преобладание формы «спикер – аудитория слушателей». В выступлениях часто рассказывается о реальных кейсах, однако этого не достаточно для развития навыков специалистов в области управления ИТ. Новый формат коммуникаций – воркшоп «Лаборатория ITSM-решений» – заполняет этот пробел. Основа воркшопа – формат общения, используемый в стартап-командах, состоящих из групп единомышленников и предполагающий активность всех членов команды.

Кейс и его владелец

В каждом воркшопе решается один или несколько кейсов. Одним из ключевых факторов успеха воркшопа является участие носителя проблемы (кейса). Представитель любой компании может предложить на обсуждение реальную проблему, задать цель и определить условия (исходные предпосылки, ограничения, допущения) выработки решений. Он же оценивает качество предложенного решения, его работоспособность и полезность.

Подать заявку на участие и узнать о результатах предыдущих игр можно на сайте www.itsmforum.ru.

Преимущества

- Для «владельца кейса» участие в воркшопе – это уникальная возможность организовать на площадке itSMF мозговой штурм по поиску вариантов решений одной из своих проблем, шанс найти талантливых молодых специалистов в свою компанию.
- Для компании, предоставившей свой кейс – это возможность поиска ИТ-специалистов, а также помощь в развитии компетенций молодых специалистов в области ITSM.
- Для участников – это уникальная возможность попробовать применить свои знания и навыки для решения реальных проблем, а также заявить о себе.

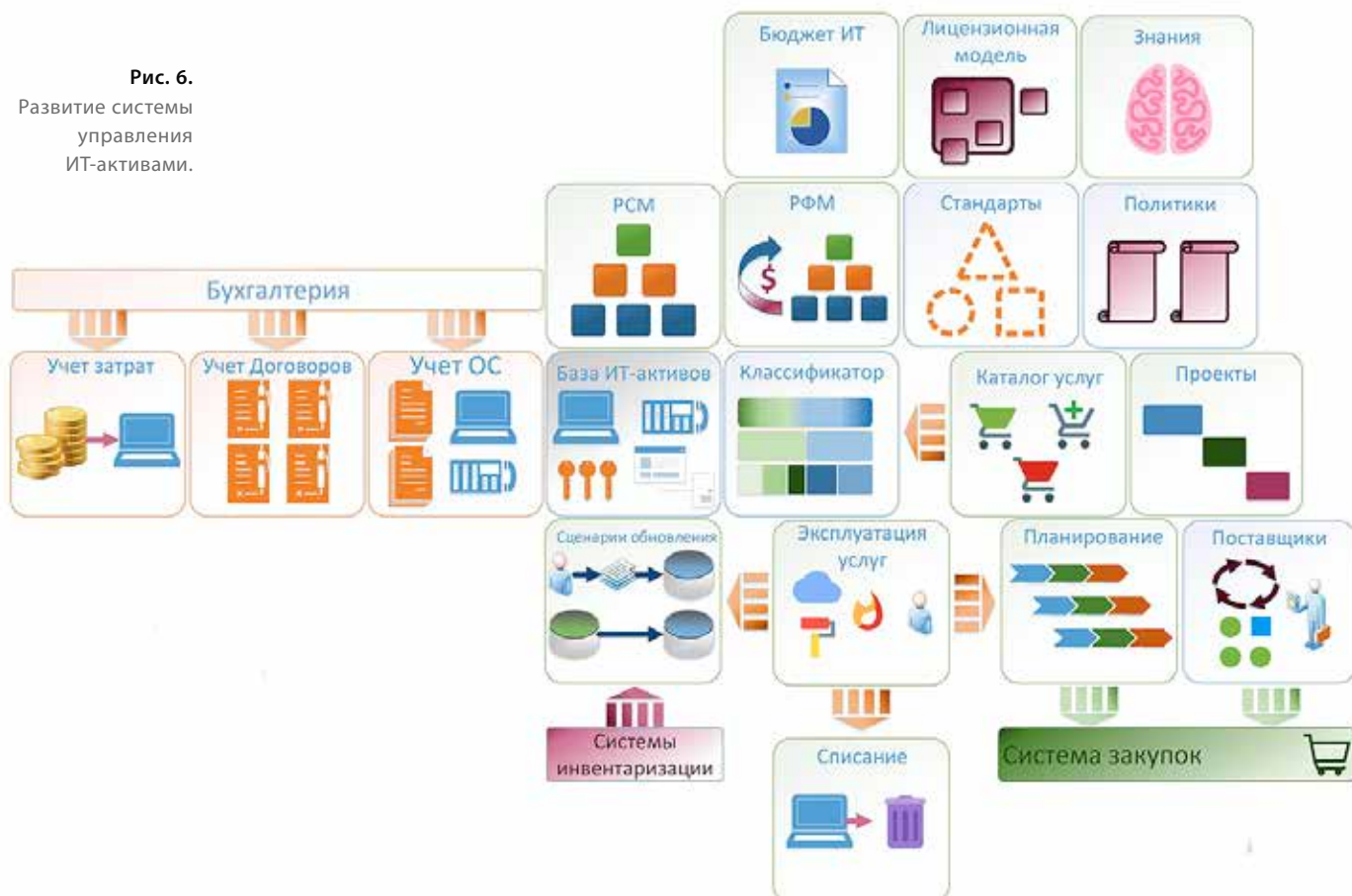
Что в итоге

ИТ-активами и практика внедрения ITAM-решений подводят к следующим выводам.

1. Крайне важно уже в начале пути понимать, каких целей хочешь добиться и на каком этапе. Без этого придется постоянно решать текущие задачи, не понимая дальнейшего глобального развития процесса.
2. Ваши цели и их критичность должны четко ложиться на последовательность того, что вы делаете.
3. Связка с процессами эксплуатации крайне важна и для процессов управления ИТ-услугами, и для процессов управления ИТ-активами.
4. С.Л.О.Н.а можно «съесть», но не целиком и не за один подход. ♦

Рис. 6.

Развитие системы управления ИТ-активами.





Рик Джослин (Rick Joslin)

Признанный эксперт в области Knowledge Centered Support и консультант по управлению услугами. С 2006 по 2016 годы был Executive Director of Certification & Training в HDI. Сертифицированный инструктор по Knowledge Centered Support, руководит проектами внедрения этой методологии.

Вы стремитесь совершенствовать свое управление знаниями. Для этого необходимо, прежде всего, сделать первые и главные для любой инициативы шаги: ответить на вопрос, поддержит ли ваша инициатива достижение бизнес-стратегии, определить заинтересованные стороны, оценить, что уже есть и работает хорошо, а что нужно улучшить, сделать с нуля и даже от чего можно отказаться. Эти шаги помогут понять, каким будет результат вашей инициативы и ее успешность. Нет единой метрики, отражающей работу процесса управления знаниями. Вам необходим набор метрик, которые позволят отслеживать отклонения в процессе, прогнозировать будущие результаты и своевременно вносить коррективы¹.

Как правильно выбрать метрики управления знаниями

Влияние управления знаниями на бизнес

Причина, по которой встал вопрос о совершенствовании управления знаниями, – это совершенствование бизнеса. Ваша ключевая цель – не повышение качества знаний. Прежде чем фокусироваться на улучшении процессов и практик управления знаниями, нужно понять, почему возникла в этом необходимость. Почему знания важны для бизнеса? Задавайтесь вопросом «почему?» снова и снова, до тех пор, пока ответ не будет связан с бизнес-целями.

Основная цель инициативы – совершенствование бизнеса. Чтобы дойти до ИТ-составляющей вашей инициативы, можно воспользоваться разными техниками. Например:

- Почему?
- Для повышения качества знаний.
- Почему?
- Для повышения доли инцидентов, разрешенных при первом обращении клиента.
- Почему?
- Для сокращения среднего времени решения инцидента.

¹Rick Joslin. [How to Select the Right Knowledge Management Metrics](#). Статья предоставлена порталом RealITSM и компанией Cleverics. Публикуется с некоторыми изменениями.

- Почему?
- Для повышения степени удовлетворенности клиентов и их производительности.
- Почему?
- Для повышения лояльности клиентов и снижения стоимости разрешения инцидента.
- Почему?
- Для увеличения продаж и повышения прибыли.

Этот метод известен под названием «Пять почему». Однако «почему?» необязательно должно быть пять.

Другие варианты ответа на вопрос «почему?» могут быть следующими:

1. Для улучшения производительности труда и повышения удовлетворенности сотрудников с целью снижения текучки кадров, чтобы сократить затраты на поиск новых и повысить прибыль.
2. Для снижения среднего времени обработки обращения, с целью повышения производительности труда аналитиков, снижения стоимости инцидента и получения большей выгоды.
3. Для повышения уровня самообслуживания, снижения объема звонков, уменьшения показателя отказов, повышения степени удовлетворенности клиентов.

Как только вы изучите ответы, сможете определить несколько важных для вас показателей, которые нужно измерять.

Метрики для оказываемых услуг

Знания собираются, совершенствуются и используются снова в процессах управления услугами, таких как управление инцидентами. Например, специалисты службы поддержки используют имеющиеся знания для решения известных инцидентов, если в организации уже возникала такая проблема, ее зафиксировали и решение нашли. Это становится элементом актива знаний. Если инцидент неизвестен, специалисты службы поддержки должны разработать решение, – добавить новые знания в базы знаний для использования в перспективе.

Наиболее распространенные метрики организации мониторинга знаний:

- **доля обращений к базе знаний**, процент обращений, закрытых с использованием статей из базы знаний;
- **индекс создания знаний**, процент закрытых обращений, для которых были созданы статьи в базе знаний;
- **доля повторного использования знаний**, процент обращений, закрытых с использованием существующих статей базы знаний.
- **индекс изменений знаний**, процент обращений, закрытых с использованием существующих знаний,

при которых специалист службы поддержки обновил статью или добавил комментарии с целью ее улучшения.

Эти метрики измеряют действия, связанные с управлением знаниями. Это начало, но его недостаточно. Фокусировка на количестве статей базы знаний может привести к засоренности базы, а также к обращению ради самого обращения, что приведет к неверным данным. Поэтому показатели действий, связанных с управлением знаниями, должны дополняться показателями качества, полученными в процессе контроля качества.

1. Качество знаний (или индекс качества статей) – результат процесса мониторинга знаний, в ходе которого специально обученные люди рассматривают случайные выборки статей знаний исходя из набора критериев качества знаний. Оценка рассчитывается для каждого автора статей базы знаний за определенный период, и специалисты предоставляют им обратную связь для повышения их навыков.

2. Качество ссылок (или точность ссылок) – процент обращений, в которых была использована соответствующая статья базы знаний. Эта метрика является результатом процесса мониторинга обращений, во время которого специалисты просматривают случайную выборку обращений. При прохождении статьи базы знаний через контроль качества критерии связи с обращениями обновляются.

Если вы используете методологию поддержки, основанной на знаниях (Knowledge Centered Service, KCS), вы можете оценить уровень компетенции специалистов. Мониторинг и выявление взаимосвязей с бизнес-метриками позволяют оценить степень проникновения культуры обмена знаниями.

Самообслуживание клиентов

Когда ваша организация решит предлагать клиентам прямой доступ к знаниям, станут необходимы новые показатели. Для мониторинга работы с базой самообслуживания важны следующие показатели.

1. Время до публикации. Среднее время с момента создания статьи базы знаний до ее доступности через сервис самообслуживания. Цель – сократить это время и предоставить клиентам доступ к новым знаниям как можно быстрее.

2. Процент опубликованных статей. Процент статей базы знаний, доступных через сервис самообслуживания. [Руководство Knowledge Centered Service v6 Practices Guide](#) ставит цель – незамедлительную публикацию 90% новых знаний в сервисах самообслуживания.

3. Процент известных знаний. Процент обращений, решенных с помощью существующих знаний. Это теоретический максимальный процент от всего объема заявок, который потенциально может быть разрешен с помощью сервиса самообслуживания. Стратегия «сдвига влево» предполагает, что все большая доля статей базы знаний доступна для самообслуживания.

4. Доля решений на нулевом уровне поддержки. Процент заявок, решенных при обращении в поддержку с помощью статей базы знаний, уже опубликованных через сервис самообслуживания. Это процент от объема обращений, которые можно передать на самостоятельное решение пользователей при условии повышения лояльности клиентов.

Измерение успешности сервисов самообслуживания достаточно сложная задача. Весьма вероятно, что вход будет осуществляться через браузер и закрытие его страницы нельзя считать успешным или неудачным использованием базы знаний. Также исследования показали, что лишь небольшой процент клиентов участвует в опросе после прочтения статьи базы знаний. С учетом такой ситуации какие показатели могут помочь оценить успешность самообслуживания клиентов?

1. Уникальные сессии, количество посещений базы знаний. Отклонения этого показателя могут указывать на изменение уровня использования портала самообслуживания.

2. Веб-обращения (из сервисов самообслуживания). Число обращений, созданных клиентами с помощью портала самообслуживания. Это может означать, что клиенты не могут найти полезные статьи базы знаний. Процент веб-обращений по сравнению с уникальными сеансами показывает минимальный процент неудач в самообслуживании. Если 40% сеансов приводят к веб-обращению, то по крайней мере 40% попыток самообслуживания не дают клиенту найти полезные знания. Результаты 60% неизвестны. Вы не знаете, нашел ли клиент то, что искал, или просто перестал искать.

3. Показатель успеха самообслуживания. Поскольку напрямую определить это невозможно, некоторые организации проводят периодические опросы среди клиентов. Например: «Если вы использовали базу знаний в сервисах самообслуживания, какой процент посещений привел к тому, что вы нашли полезные знания?»

4. Несостоявшиеся обращения (или обращения, решенные самостоятельно). Число обращений, не созданных благодаря успешному самообслуживанию. Общее число несостоявшихся обращений

равно количеству уникальных сеансов успешного самостоятельного решения.

Еще одним индикатором успешности сервиса самообслуживания является влияние на оказываемые услуги. Это влияние можно выявить, если изменения в потреблении услуг окажутся связанными с доступностью и продвижением сервисов самообслуживания. Например:

- **объем обращений** должен уменьшаться по мере роста уникальных сеансов самообслуживания;
- **доля инцидентов, разрешенных при первом обращении клиента,** должна уменьшиться, поскольку простые в решении обращения, использующие знания, передаются в сервисы самообслуживания;

Создание базы знаний самообслуживания может иметь огромную ценность для клиентов и организации. К сожалению, нет показателей, которые непосредственно и точно измеряют эту ценность.

- **среднее время обработки обращения** должно расти по мере того, как простые и быстрые в разрешении обращения закрываются с использованием сервисов самообслуживания, высокий процент сложных обращений остается для вспомогательного обслуживания;
- **процент известных знаний** уменьшается по мере перехода известных проблем к самообслуживанию;
- **доля решений на нулевом уровне поддержки** снижается по мере перехода известных задач в сервисы самообслуживания;
- **стоимость обращения** будет расти по мере уменьшения объема обращений и увеличения их сложности.

Создание базы знаний самообслуживания может иметь огромную ценность для клиентов и организации. К сожалению, нет показателей, которые непосредственно и точно измеряют эту ценность.

Улучшение продуктов и услуг компании

Ценность знаний не ограничивается использованием знаний для оказания услуг и самообслуживания. Понимание ценности статей базы знаний увеличивается по мере того, как культура обмена знаниями проникает в компанию. Оценивая повторное использование статей базы знаний, проще обнаружить проблемы, что ведет к улучшению продуктов и услуг. ♦

От популяризации ITSM



Многолетнее и успешное сотрудничество itSMF России с образовательными учреждениями — это стратегическое партнерство, основанное на взаимных интересах и общих ценностях. Постоянно расширяется круг ВУЗов, которые стали полноправными членами форума. В этих ВУЗах ITSM входит в программы подготовки бакалавров и магистров по целому ряду образовательных направлений как полноправная дисциплина. Образовательная деятельность в них строится на основе современных информационных технологий и мощной ИТ-инфраструктуры.

Форум также открыт и для сотрудничества с ВУЗами, пока не являющимися постоянными членами itSMF России. Мы заинтересованы во взаимодействии со всеми образовательными учреждениями, которые осуществляют подготовку специалистов в области информационных и коммуникационных технологий, и рассматриваем это сотрудничество как одно из важнейших направлений нашей деятельности.

С целью популяризации ITSM среди студентов мы проводим открытые лекции и тематические семинары, участвуем в конференциях, проводимых ВУЗами. Все это стало постоянной практикой нашей совместной деятельности. В планах — организация интернет-трансляции открытых лекций и проведение вебинаров по тематике ITSM. Это позволит студентам и преподавателям региональных высших учебных заведений принять участие в этих мероприятиях.

Мы видим новые и важные тенденции в образовании и поддерживаем развитие практико-ориентированного обучения через регулярные мастер-классы и мероприятия в рамках новой программы «Лаборатория ITSM-решений» (Workshop ITSM Labs). Обобщение положительного опыта проведения подобных мероприятий подтверждает перспективность развития такого направления сотрудничества с ВУЗами.

Постоянным проектом форума, направленным на популяризацию и применение сервисного подхода в управлении ИТ, является проведение ежегодного конкурса выпускных квалификационных работ студентов по тематике ITSM «Лига юниоров ITSM». Он рассчитан на демонстрацию знаний и навыков выпускников в соответствии с государственным образовательным стандартом по профильным специальностям, связанным с ITSM. В жюри конкурса входят представители ВУЗов и эксперты itSMF России. Торжественное вручение гра-



— к стратегическому партнерству

мот лауреатам происходит на Всероссийской конференции — главном профессиональном событии года.

Центральным ежегодным мероприятием в рамках сотрудничества с образовательными учреждениями является проведение круглого стола itSMF России с ВУЗами. В его программу мы традиционно включаем обсуждение наиболее важных и актуальных проблем развития профессионального ИТ-образования в нашей стране:

- текущее состояние и перспективы развития профессиональных и образовательных стандартов;
- опыт, проблемы и направления разработки образовательных программ по специальностям и дисциплинам, связанным с ITSM;
- обобщение опыта и перспективы развития совместных мероприятий и форм сотрудничества itSMF России с ВУЗами;
- практические аспекты взаимодействия ВУЗов и работодателей.

itSMF России, как сообщество профессионалов, постоянно отслеживает новые тенденции, законодательные инициативы и подходы к организации образовательной деятельности. Мы заинтересованы в сотрудничестве и налаживаем взаимодействие с другими профессиональными сообществами с целью участия в профессионально-общественной аккредитации образовательных программ и разработке профессиональных и образовательных стандартов.

Взаимовыгодное информационное партнерство — это постоянная практика успешного сотрудничества. Форум готов оказать информационную поддержку в продвижении проводимых мероприятий и образовательных услуг.

Мы постоянно ищем новые формы взаимодействия с ВУЗами и возможности реализации совместных проектов, расширяя область применения сервисного подхода.

Присоединитесь к сообществу профессионалов.

Владимир Павлов

Заместитель председателя форума,
Руководитель комитета по работе с ВУЗами





Джон Холл (Jon Hall)

Главный менеджер продуктов BMC Remedy. Его интересы сфокусированы на проникновении DevOps и программно-управляемой инфраструктуры на крупные предприятия, а также на воздействии этих изменений на каналы технической поддержки.

Фреймворк Cynefin привлекает значительное внимание DevOps-сообщества, а также вызывает интерес у специалистов в области ITSM, особенно с учетом грядущей эволюции (или замены) привычных сводов знаний, таких как ITIL. Этот всеобщий интерес к Cynefin напоминает аналогичный интерес к подходу Swarming, который отвергает привычные разобщенные или иерархические структуры команд в пользу гибких и адаптивных рабочих групп. Сходство Swarming с ключевыми элементами концепции DevOps неоспоримо, и первый идеально подходит в качестве методологии организации технической поддержки в эпоху DevOps. В статье показано, как техническая поддержка может быть организована на основе фреймворка Cynefin и подхода Swarming¹.

Совместное использование фреймворка Cynefin и подхода Swarming для лучшего управления инцидентами

Cynefin: быстрый обзор

Cynefin – интересный фреймворк, базирующийся на теории сложности и принципе, согласно которому разные ситуации требуют существенно разных подходов. Его создатель, Дейв Сноуден (Dave Snowden), кратко описывает цель использования фреймворка

Cynefin так: «осознать сложность, чтобы действовать».

Сноуден создал свой фреймворк в 1999 году, когда работал в IBM, а в 2003-м опубликовал широко известную статью [Complex Acts of Knowing \(«Сложные акты познания»\)](#).

¹Jon Hall. [Combining Cynefin with Swarming for better Incident Management](#). В материале также использована еще одна статья Джона Холла [ITSM, DevOps, and why three-tier support should be replaced with Swarming](#). Статьи предоставлены порталом RealITSM и компанией Cleverics. Публикуются с некоторыми изменениями и сокращениями.

Так что же такое Synefin? Во-первых, давайте поговорим о произношении! Synefin – это валлийское слово, не имеющее точного перевода на английский язык (его ближайший эквивалент часто называют «средой обитания», хотя Сноуден утверждает, что это «множество факторов в нашей среде и наш опыт, которые влияют на нас способами, которые мы никогда не сможем понять»).

Synefin позволяет классифицировать любую проблему, ситуацию и связанный с ней сценарий принятия решения по четырем областям или доменам (рис. 1)². Опишу кратко области Synefin³.

Очевидная и усложненная – это области порядка, в которых имеются повторяющиеся взаимосвязи между причиной и следствием. В каждой из областей порядка, как говорит сам Сноуден:

Одинаковые симптомы всегда имеют одинаковые причины их возникновения, раз за разом. Разница в том, что в очевидной области каждый может видеть, что это [за причины], в то время как в усложненной области вам требуется провести анализ, чтобы найти их. В усложненной области вы можете использовать лишь хорошую практику, тогда как в очевидной области у вас есть лучшая практика, и это действительно важное различие.

Сложная и хаотичная – это области беспорядка. В **сложной области** понимание проблемы или ситуации требует экспериментов и исследований. Работа сложных систем нарушается по сложным, непонятным нам причинам. Утверждать что-то, касающееся ситуации или проблемы, можно только в отношении настоящего, а не будущего. Верное решение в этой ситуации сразу не видно и становится очевидным только после ее исследования. Возможно, со временем сформируется достаточный объем знаний о ситуации и ограничениях, чтобы можно было классифицировать ситуацию уже не как сложную, а как усложненную.

Хаотичная область – это область, где эксперименты и исследования не помогут, сценарии развития таких ситуаций драматичны и практически ничем не ограничены. Приоритетом в такой ситуации является ограничение распространения хаоса и сокращение ущерба. Именно на достижении этих целей сосредоточено любое первоначальное решение. Цель решений и действий – «обуздание» ситуации и перемещение проблемы в любую из других областей фреймворка.

СЛОЖНАЯ

*исследовать,
наблюдать,
ответить*

**Ограничения,
определяющие
возможности**

**Возникающие
практики**

ХАОТИЧНАЯ

*действовать,
наблюдать (ощущать),
ответить*

**Нет эффективных
ограничений**

**Неизвестанные
практики**

УСЛОЖНЕННАЯ

*распознать,
проанализировать
и ответить*

**Управляемые
ограничения**

**Хорошие
практики**

ОЧЕВИДНАЯ

*увидеть,
классифицировать
и ответить*

**Фиксированные
ограничения**

**Лучшие
практики**

Отметим, что есть еще и пятая область – посередине четырех доменов. Она отражает состояние непонимания того, в какой области находится текущая ситуация. Приоритетом в этом случае является поиск фактов, необходимых для перемещения в другую область.

Краткий обзор идей Swarming

Объединенные сообщества могут преодолевать дисциплинарные и организационные барьеры, которые препятствуют сотрудничеству, обучению и прогрессу.

*Дон Танскотт
и Энтони Д. Уильямс, Wikinomics.*

В настоящее время развивается подход под названием Swarming⁴. Он появился в конце последнего десятилетия и позиционировался как новый подход к организации технической поддержки. Первым адептом подхода Swarming стала компания Cisco, которая анонсировала свою новую «Модель для распределенного сотрудничества и принятия решений» в документе **Digital Swarming** в 2008 году. Модель была впоследствии принята Консорциумом по инновациям в сфере услуг (Consortium for Service Innovation) и получила название **Intelligent Swarming**.

Рис. 1.
Фреймворк
Synefin.

² Вообще говоря, Сноуден создал свой фреймворк как инструмент для осмысления ситуации или проблемы, для отражения ее ключевых элементов и аспектов, которые влияют на нее. Фреймворк Synefin можно использовать для классификации управленческих ситуаций по четырем доменам, однако Сноуден настаивал, что основной результат – это проявление доменов из самой природы элементов управленческой ситуации. То есть домены – это результаты анализа ситуации и проблемы, а не предопределенная заранее структура. Прим. ред.

³ Отличный обзор этих областей от самого Сноудена можно найти в первых десяти минутах [этого подкаста](#) 2015 года.

⁴ Англ. роение, от swarm (рой, в т.ч. пчелиный).

Проблемы трехуровневой организации поддержки в свете DevOps

Классической организационной структурой управления является трехуровневая иерархия поддержки⁵.

1. **Уровень 1** – первая линия службы поддержки, служба Service Desk, которая непосредственно принимает входящие сообщения заказчиков (как правило, отвечая на телефонные звонки). Большинство служб поддержки созданы для обеспечения среднего уровня технической поддержки, в рамках которой обрабатываются типовые обращения. Цель – обеспечение постоянной поддержки пользователей на определенном уровне. Значительное число входящих вопросов при этом решаются на первой линии.
2. **Уровень 2** – второй уровень поддержки, тесно связанный с Service Desk, но обладающий более продвинутыми навыками общего или специализированного назначения. Специалисты второго уровня могут, например, обладать знаниями по поддержке распространенных операционных систем или аппаратных средств и, следовательно, иметь возможность решать более сложные задачи, затрагивающие распространенные технологические решения.
3. **Уровень 3** – специализированные группы поддержки, сосредоточенные на конкретных технологиях и приложениях. В компаниях, которые разрабатывают и используют прикладное программное обеспечение, широко распространена практика выделения групп поддержки уровня 3, отвечающих за конкретные приложения или службы.

Такая структура широко распространена, что обусловлено наличием ряда преимуществ:

- заказчикам предоставлено «единое окно» для взаимодействия с ИТ-поддержкой, независимо от характера обращения;
- специалисты с общими техническими навыками, необходимыми для работы в поддержке уровней 1 и 2, доступны на рынке труда, кроме того это упрощает вывод на аутсорсинг одного или обоих этих уровней, что также часто встречается;
- специализированные технические ресурсы могут быть изолированы от прямого контакта с пользователями, заявки к ним попадают только после корректного анализа.

Путь обращения клиента в такую структуру поддержки начинается и заканчивается на первой линии. Однако неизбежно поступление обращений, которые не могут быть разрешены поддержкой уровня 1. Такие обращения передаются на уровни 2 и 3 через процесс эскалации.

Специалисты второго уровня, как правило, обрабатывают меньше обращений, чем их коллеги первого

уровня, но эти задачи, как правило, сложнее и требуют больше времени на решение. Обращения, которые доходят до третьего уровня (либо путем вторичной эскалации с уровня 2, либо непосредственно с уровня 1), как правило, составляют небольшую долю в общем объеме входящих обращений, но вместе с тем являются самыми сложными, требующими наличия продвинутых навыков специалистов и еще больше времени для решения.

DevOps быстро развивается как фундаментальный вызов устоявшимся «ортодоксальным» подходам, объединяя ранее разделенные роли и борясь с неэффективностью и излишней бюрократией.

Подвергать сомнению настолько распространенную структуру непросто. Однако в свете принципов DevOps и подхода Swarming видны некоторые существенные недостатки такой многоуровневой структуры службы поддержки.

1. **Многоуровневая поддержка создает очереди задач.** Хотя поддержка первого уровня стремится быть реактивной и в реальном времени, любая заявка, которая не может быть решена на этом уровне, сразу же попадает в очередь. Ее сущность меняется, превращаясь из текущей задачи в запись в бэклоге задач. Таким образом, уровни 2 и 3 являются накопителями незавершенных задач. Однако, согласно философии бережливого производства (Lean), лежащей в основе DevOps, это большая проблема. Практики Lean требуют решительных шагов по снижению систематического объема незавершенных задач. Сама по себе проблема формирования пула незавершенных задач является серьезным препятствием для проникновения DevOps-практик.
2. **Разделенная на уровни поддержка мешает корректной маршрутизации.** DevOps ориентирован на повышение ответственности и автономности. Разработчикам предлагается взять на себя ответственность за поддержку своего собственного кода. Наиболее эффективные DevOps-организации благодаря этому добиваются значительно более высокой скорости обработки обращений (например, согласно отчету [2016 State of DevOps Report](#) скорость обработки обращений повышается в 24 раза, а количество ошибок при проведении изменений ПО сокращается в 3 раза).

⁵ Заметим, что во многих организациях заказчики могут решить свою проблему с помощью систем автоматизированного самообслуживания, которые часто называют «нулевым уровнем» поддержки.

Однако DevOps не даст ожидаемого эффекта, если по пути к нужному эксперту обращение по-прежнему проходит несколько уровней поддержки, на каждом из которых выполняются попытки его решения (рис. 2). Как сказал один менеджер по поддержке BMC, когда мы обсуждали реализацию подходов Swarming в их службе поддержки: «Почему мы поставили наших лучших людей в конце процесса?».

3. **Многоуровневая поддержка приводит к «футболу».** В контексте многоуровневой поддержки передача обращения между уровнями выполняется посредством изменения команды, которая назначена ответственной за ее решение. Обращение переназначается напрямую от одной команды другой кем-то из участников первой. Участники вновь назначенной команды впервые видят обращение, когда оно попадает в их очередь. К сожалению, часто бывает, что обращение возвращается назад – либо для предоставления уточнений, либо потому, что маршрутизация была выполнена не по назначению.

DevOps основывается на сотрудничестве между специалистами, участвующими в операционной деятельности, и их коллегами, решающими задачи развития. Вертикальные и горизонтальные барьеры, присущие многоуровневой поддержке, и пассивная передача заявок между линиями – это полная противоположность кросс-функциональному сотрудничеству, являющемуся основой DevOps.

4. **Многоуровневая поддержка не решает проблему перегрузки профильных экспертов.** В то время как одним из положительных качеств многоуровневой поддержки является предотвращение маршрутизации легко решаемых заявок специалистам более высокой квалификации, она не защищает ключевых специалистов от большого объема сложных задач. Управление ИТ-услугами страдает от таких «героев». Это, как правило, весьма умные люди, которые,

на первый взгляд, вносят огромный вклад в успех организации, неоднократно являя чудо решения архисложных проблем. На самом же деле такой «герой» – это перегруженная, подверженная сбоям точка отказа, преднамеренный или непреднамеренный хранитель эксклюзивных знаний, которые следовало бы как можно шире распространить в организации. Многоуровневая поддержка, являющаяся линейной и разделенной структурой, не делает ничего, чтобы предотвратить культ таких «героев», скорее даже усиливает его.

По мере того, как предприятия начинают использовать практики DevOps, мы вновь и вновь наблюдаем сценарий, при котором ключевые специалисты (и члены DevOps-команд) находятся в конце цепочки эскалации наиболее сложных заявок. С точки зрения подхода DevOps ключевые разработчики при этом не участвуют в развитии, а занимаются «тушением пожаров», эскалированных им с других линий поддержки (и уже отложенных на некоторое время).

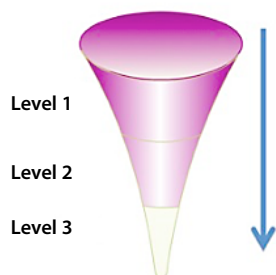
Недостатки организационной структуры затрудняют или делают невозможной успешную интеграцию новых практик DevOps с уже существующими структурами технической поддержки. Традиционная структура активно наполняет очередь незавершенными работами, ограничивает автономию и гибкость. Способствует разобщенности, так как формирование жесткой вертикальной и горизонтальной структур приводит к созданию барьеров для кросс-функционального сотрудничества, являющегося залогом практик DevOps. Эти подходы несовместимы с философией DevOps, что становится ключевой проблемой по мере того, как все больше крупных компаний с традиционной организационной структурой стремятся использовать возможности DevOps.

Рис. 2.
Ключевые специалисты находятся в конце цепочки эскалации наиболее сложных заявок.

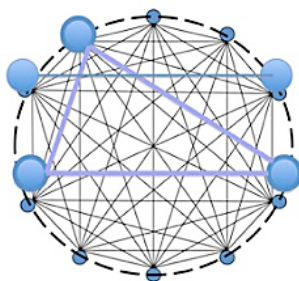


Старая модель: Streaming

Новая модель: Swarming



Процесс, основанный на **эксплуатации**



Процесс, основанный на **сотрудничестве**

Рис. 3. Традиционная модель трех уровней поддержки и модель сетевого сотрудничества Swarming.

Некоторые из его основных принципов прямо противоположны «ортодоксальным»:

- не должно быть многоуровневых групп поддержки;
- не должно быть эскалаций от одной группы к другой;
- задача должна перейти сразу к человеку, который, скорее всего, сможет ее решить;
- человек, который рассматривает заявку, является тем, кто ведет ее до разрешения.

Подход Swarming идеально подходит в качестве методологии организации технической поддержки в эпоху DevOps. Он недвусмысленно отвергает трехуровневую «ортодоксальную» модель, предлагая взамен модель сетевого сотрудничества (рис. 3). Более

подробно о проблемах трехуровневой организации поддержки в свете DevOps и противоречиях этих подходов рассказано во врезке. Модель сетевого сотрудничества Swarming реализуется через структуру, получившую название «рой» (swarm). Каждый рой представляет собой небольшую команду, сосредоточенную практически в реальном времени на входящем потоке различных заявок от заказчиков.

Пример реализации подхода Swarming на практике

На данный момент не существует единого подхода к формированию Swarming-структуры, поскольку это относительно новая и пока мало используемая концепция. Однако ниже я приведу пример, основанный на реализации подхода Swarming в технической поддержке компании BMC. Он является типовым и его появление привело к значительному улучшению ситуации с реализацией идей Swarming.

Как только какая-то проблема не решается сразу в точке контакта с клиентом, далее можно использовать подход Swarming. В BMC за быстрым первичным анализом следует распределение заявок в один из двух роев: рой «Критичность 1» и диспетчерский рой (рис. 4).

Рой «Критичность 1»

Основной фокус: обеспечить немедленный ответ и решить задачу или проблему как можно скорее.

Рой «Критичность 1» (Severity 1 Swarm) сосредоточен на наиболее критичных обращениях и используется для небольшого процента проблем, которые являются достаточно значимыми, чтобы оказать влияние на бизнес. Рой координирует решение критичных задач, привлекая «правильных» людей, чтобы обеспечить решение настолько быстро, насколько возможно (рис. 5). Лидер роя добавляет необходимых людей в рабочую группу, и они вместе работают над задачей, пока она не будет решена. Этот рой – аналог практики «war room» в традиционных структурах поддержки. В целом эта практика не отличается от процесса решения крупных инцидентов, типичного для традиционной многоуровневой поддержки.

Рис. 4. Первичный анализ и пример маршрутизации в Swarming.



Рис. 5. Состав роя «Критичность 1» (Severity 1 Swarm).



Лидер роя
Коммуникации



Другие участники
Исследования, тесты, координация



**Опытный эксперт
и аналитик**



**Менее опытные эксперты
и аналитики**

Рис. 6.
Состав
диспетчерского
роя (Dispatch
Swarm).

Диспетчерский рой

Основной фокус: какие из новых обращений можно решить немедленно, прямо сейчас.

Вторичный фокус: проверка обращений перед их назначением группам поддержки, организованным по продуктовым линейкам.

Диспетчерский рой (Dispatch Swarm)⁶ играет две роли:

- он отбирает задачи, которые могут быть решены действительно быстро, в первую очередь занимается обращениями, которые можно решить немедленно, игнорируя все, что невозможно решить очень быстро;
- его эксперты выступают валидаторами качества данных для проблем и задач, которые будут переданы дальше в группы поддержки, организованные по продуктовому принципу.

В диспетчерский рой маршрутизируется гораздо больший объем заявок, чем в рой «Критичность 1». Этот рой образует небольшая группа сотрудников поддержки, как правило, с различным уровнем опыта, которая решает проблемы по мере их поступления. Эта группа встречается часто, по несколько раз на протяжении дня (как вариант, каждые 60–90 минут), и мониторит входящий поток обращений.

Благодаря отбору задач, которые могут быть решены быстро, а также быстрой маршрутизации, можно значительно сократить время, затрачиваемое на решение значительного количества попавших в этот рой задач. Диспетчерский рой устраняет ключевой недостаток многоуровневой поддержки: большинство задач может быть решено очень быстро, если они своевременно попадут к «правильным» специалистам, однако они стоят в очереди незавершенных задач. В результате чего решение «пятиминутной» задачи может занять несколько дней.

У диспетчерского роя также есть целый ряд дополнительных преимуществ. Включение в него персонала начального уровня дает им доступ к знаниям, которые в обычной ситуации становятся доступными только после перевода в одну из специализированных команд (рис. 6). Вместе с тем специалисты третьего уровня поддержки становятся ближе к клиентам.

Диспетчерский рой обеспечивает быстрое разрешение значительного числа задач (в ВМС, как правило, это порядка 30%), но оставшиеся задачи поступают в очереди традиционных групп поддержки, организованных по продуктовым линейкам. Многие из этих обращений окажутся достаточно простыми для постоянных членов групп и будут решены в их рамках. Однако другие обращения (возможно, снова около 30%) окажутся достаточно сложными, и для их решения потребуются внимание со стороны лучших специалистов поддержки, независимо от структурирования групп поддержки. Для таких задач удобно использовать рой третьего типа: бэклог-рой.

Бэклог-рой

Основной фокус: рассмотрение сложных обращений, переданных из групп поддержки по продуктовым линейкам.

Вторичный фокус: замещение отдельных квалифицированных экспертов в определенных областях.

Бэклог-рой (Backlog Swarm) объединяет группы квалифицированных и опытных кросс-функциональных технических специалистов (рис. 7), которые обычно являются членами продуктовых групп поддержки. Рой снимает такие ограничения традиционных структур, как географическое расположение и принадлежность к тому или иному департаменту. Таким образом, ключевые специалисты могут сфо-

Рис. 7.
Состав бэклог-роя
(Backlog Swarm).



**Опытный эксперт
и аналитик**



**R&D-инженеры
с высокой
квалификацией**

⁶ Иногда этот рой называют Triage Swarm.

кусироваться на наиболее сложных случаях (рис. 8). Они встречаются ежедневно, специально для решения сложных вопросов, которые в противном случае рискуют застрять в очередях или перенаправляться от одной продуктовой группы к другой в поисках решения. Задачи передаются в этот рой

Подход Swarming идеально подходит в качестве методологии организации технической поддержки в эпоху DevOps. Он недвусмысленно отвергает трехуровневую «ортодоксальную» модель, предлагая взамен модель сетевого сотрудничества.

инженерами и командами групп поддержки по продуктам, которым не разрешается напрямую привлекать квалифицированных экспертов по отдельным вопросам. Вместо этого они должны всегда передавать сложные задачи в бэклог-рой.

Применение подхода Swarming совместно с фреймворком Synefin

Не существует единой, окончательной структуры Swarming, этот подход по определению, динамичен. Предприятия, которые используют подход Swarming для организации технической поддержки, имеют несколько типов роев, чтобы применять различные сценарии разрешения инцидентов. Одним из примеров является описанная выше структура трех роев. Тем не менее даже эти структуры могут быть непостоянными. Как и в DevOps, в подходе Swarming команды ориентированы на достижение результатов ценой большей гибкости, а не приверженности строгим правилам.

Можем ли мы взять классификацию на области фреймвока Synefin и организовать рои в соответствии с их логикой? Давайте попробуем.

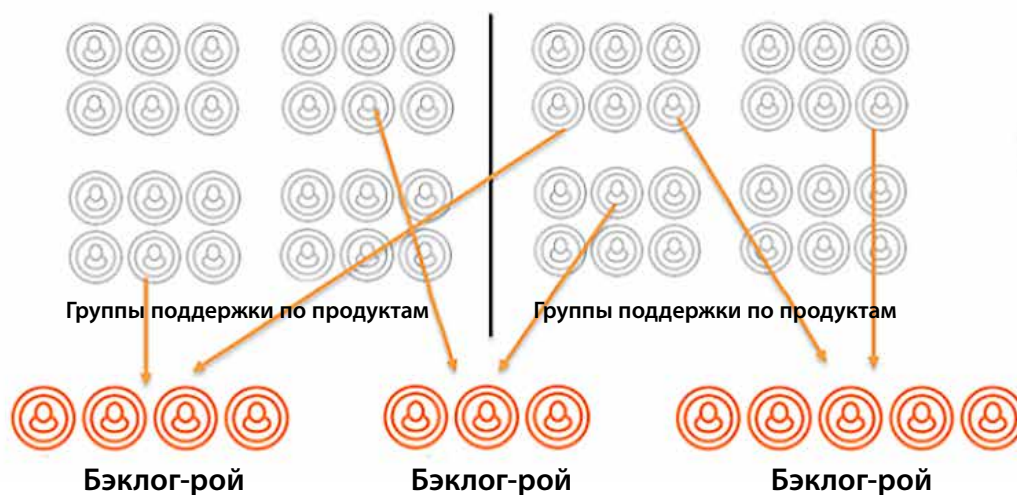
Задачи из очевидной области. Для простых задач из очевидной области Synefin рекомендует следующее: *увидеть, классифицировать и ответить*. Решение таких задач, как правило, уже известно и понятно, и после осмысления ситуации его легко можно идентифицировать. Идентификация решения задачи дает возможность самостоятельного ее решения пользователем или это сделает сотрудник поддержки на основе простого шаблона или документа из базы знаний. Здесь не требуется применение подходов Swarming. В идеале, пути решения очевидных задач, которые возникли впервые, должны быть занесены в базы знаний.

Задачи из усложненной области. В усложненной области фреймворка Synefin проблема уже сложнее, она не идентифицируется и не разрешается однозначно, однако сохраняется четкая взаимосвязь между причиной и следствием. Следовательно, решение все равно можно найти, записать и повторить при необходимости. Однако необходимо вместо простой категоризации требуется анализ. Synefin рекомендует следующее: *распознать, проанализировать и ответить*.

Если предположить, что способность индивидуума выполнять этот анализ возрастает благодаря знаниям и опыту, то хороший пример организации такого анализа – диспетчерский рой, в котором опытные сотрудники работают с менее опытными, обеспечивая тем самым распространение навыков. Получение детальных знаний о возникающих проблемах в первый раз обеспечивает дальнейшее организационное обучение и более быстрый ответ в следующий раз, когда случится аналогичная комбинация причин и следствий.

Рис. 8.

Объединение лучших специалистов поддержки в бэклог-рой независимо от структуры групп поддержки.



Задачи из сложной области.
Для задач, лежащих в сложной области, решения не предопределены, четкая взаимосвязь между причиной и следствием уже не сохраняется, все представляется запутанным. Поиск решений таких задач требует экспериментов и итераций. Synefin рекомендует следующее: *исследовать, наблюдать, ответить*. Возможно, потребуются собрать данные из нескольких источников, часть из которых впоследствии может быть отброшена.

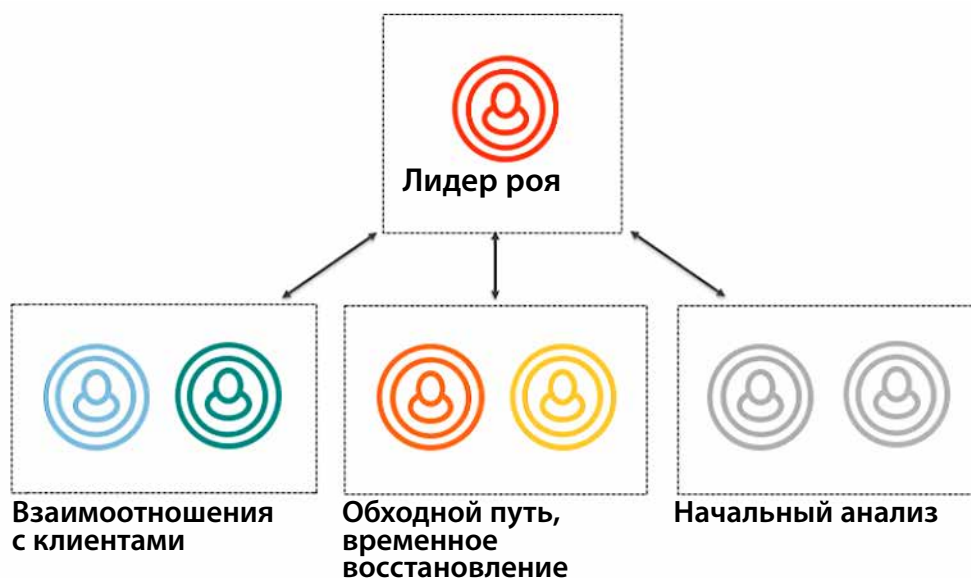
Для решения сложных задач идеи *Swarming* подходят гораздо лучше, нежели традиционная многоуровневая модель поддержки. В то время как последняя больше ориентирована на переназначение задач между фиксированными группами сотрудников с отдельными ролями, *Swarming* делает команду, занимающуюся решением, динамичной, позволяя ей определять текущую стадию исследований и понимания проблемы, управлять своим временем и иметь возможность вести исследования и эксперименты.

На рисунке 9 показан возможный пример работы роя под руководством лидера роя и при помощи ассистента. После первоначального анализа фактов рой немного расширяется. Он привлекает специалистов, способных строить гипотезы о причинах, вызывающих проблему. Затем проводятся исследования параллельно по двум направлениям, при этом рой делится на две подгруппы (наличие помощника позволяет руководителю поддерживать диалог с другой подгруппой). После отказа от непродуктивных гипотез и исключения членов роя, в которых больше нет необходимости, финальный состав роя определяет действия,



Рис. 9.
Пример использования
роя для исследования и
решения сложной задачи.

Рис. 10.
Организация
нескольких
различных суб-роев,
ориентированных
на различные
задачи.



которые необходимо сделать для решения сложной задачи. После этого возможны дальнейшие итерации.

Задача из хаотичной области. Поддержка решения задач из хаотичной области должна стремиться перевести ситуацию из хаотической области в сложную (или усложненную). Однако ключевой приоритет – остановить «потерю крови», которая неизбежна в хаотической ситуации. Одной из важнейших задач лидера в этой ситуации является установление порядка. Хаотичная ситуация может потребовать от лидера роя организации нескольких различных суб-роев, ориентированных как на борьбу с критической ситуацией, так и на поиск информации, достаточной, чтобы переместить проблему в сложную область (рис. 10).

объединяя ранее разделенные роли – операционные и решающие задачи развития, – и борясь с неэффективностью и излишней бюрократией.

Следует как можно скорее изменить многоуровневую модель поддержки, используя подход, который помогает воспользоваться преимуществами концепции DevOps для ИТ-поддержки в масштабах предприятия. Swarming может быть таким подходом.

Второй важный момент – Swarming отлично сочетается с принципами Cynefin. В частности, он предлагает подход, который намного лучше работает с беспорядочными областями Cynefin: сложной и хаотической. Эти области описывают меняющиеся ситуации, требующие адаптивных и итеративных подходов, к которым жесткую, разделенную на уровни модель организации поддержки чрезвычайно сложно адаптировать.

Кроме того, фреймворк Cynefin открывает перспективы непрерывного совершенствования организации посредством совершенствования методов Swarming. Наибольших успехов сможет добиться организация, ориентированная на улучшение и нацеленная на:

- упрощение ответов на очевидные задачи с помощью базы знаний и самообслуживания;
- активное перемешивание команды для решения усложненных задач, обеспечения лучшего обмена экспертными знаниями и формирование знаний по конкретным направлениям;
- наблюдения за результатами исследований различных суб-роев для сложных проблем, чтобы постоянно совершенствовать методы исследований и поиска неизвестных ранее решений. ♦

Swarming отлично сочетается с принципами Cynefin. Он предлагает подход, который намного лучше работает с беспорядочными областями Cynefin: сложной и хаотической.

Резюме

Сходство Swarming с ключевыми элементами концепции DevOps неоспоримо, и проблемы, которые этот подход призван решать, неразрывно связаны с проблемами, решаемыми с помощью DevOps. DevOps быстро развивается как фундаментальный вызов устоявшимся «ортодоксальным» подходам,

ФОРУМ – ЭТО МЫ! ПРИСОЕДИНЯЙТЕСЬ!

В сложившейся экономической ситуации особенно ценным является умение принимать решения, обеспечивающие надежное и эффективное использование ресурсов. Поэтому участие в работе itSMF России именно сегодня – верное решение. Членство в форуме – почетный и значимый статус, дающий широкие преимущества.

Для компаний, уже внедряющих ITSM или только разрабатывающих собственную ITSM-стратегию, форум – это:

- ♦ участие специалистов в конференциях, мастер-классах и круглых столах;
- ♦ обсуждение наболевших тем в рамках мероприятий форума;
- ♦ общение с ведущими экспертами;
- ♦ доступ к лучшим международным практикам ITSM;
- ♦ возможность повысить мотивацию сотрудников.

Ведущие российские системные интеграторы и IT-консультанты с нами, потому что:

- ♦ мы – не конкурентная среда, а сообщество успешных профессионалов;
- ♦ активность специалистов в форуме в качестве экспертов – показатель высокого уровня подготовки кадров направления ITSM в компании;
- ♦ мы предлагаем уникальную возможность целевого позиционирования услуг.

Для производителя оборудования и ПО членство в форуме – это:

- ♦ возможность проведения вендорских тематических мероприятий на площадках форума с гарантированно целевой аудиторией;
- ♦ уникальная возможность целевого позиционирования продуктов и услуг;
- ♦ участие в подготовке и издании русскоязычных материалов по тематике ITSM.

Специалистам в области ITSM членство в форуме открывает возможности:

- ♦ участия в мероприятиях сообщества профессионалов ITSM;
- ♦ неформального общения с ведущими экспертами в области ITSM;
- ♦ повышения квалификации;
- ♦ публикаций в периодических и онлайн-изданиях;
- ♦ повышения личного престижа в отрасли через участие в проектах форума в качестве эксперта.

Высшие учебные заведения и студенты, вступающие в форум, пользуются особыми преимуществами, среди которых:

- ♦ эксклюзивные лекции и семинары для студентов последних курсов профильных факультетов;
- ♦ совместные мероприятия по популяризации практик ITSM и поднятию престижа образования и деятельности в этой области.

Приглашая вас вступить в itSMF России сегодня и встать в один ряд со специалистами ведущих российских и международных компаний-заказчиков, вендоров, консультантов и интеграторов, мы предлагаем уникальную услугу. Быть частью независимого, не ориентированного только на коммерческие интересы сообщества, в конечном итоге, означает значительную выгоду для каждого нашего участника, для всех нас и для развития ITSM в России.

