

2. Rudenko Yu. N. Nadezhnost' sistem energetiki i ih oborudovaniya. Nadezhnost' system snabzheniya [Reliability of energy systems and their equipment. Reliability of heating systems]. Mos-cow, Nauka Publ., 2000. 351 p.
3. Sennova Ye.V., Kirbhin S.N., Shymanskaya A.O. Metodologiya i algoritm rascheta pokazateley nadezhnosti teplosnabzheniya potrebiteley i rezervirovaniya teplovyh setey pri razrabotke shem teplosnabzheniya [Methodology and algorithm for calculating indicators of reliability of heat supply to consumers and reservation of heat networks in the development of heat supply schemes]. *Novosti teplosnabzheniya*. Available at: <http://www.nts.ru/> (accessed: 16.10.2018).
4. Skvortsov M.S. Metodika optimizatsii sistem s setevoi strukturoy. [Methods of optimization of systems with network structure]. *Trudy SPIIRAN*, 2011, vol. 1, no. 16, pp. 231-242.
5. Cherkesov G.N. Otsenka nadezhnosti system s uchetom ZIP [Evaluation of systems' reliability with regard to spare parts and accessories]. St. Petersburg, BHV-Peterburg Publ., 2012. 480 p.
6. Cherkesov G.N. *Nadezhnost' apparatno-programmykh kompleksov*. [Reliability of hardware and software systems. Tutorial]. St. Petersburg, Piter Publ., 2005, 479 p.
7. Ushakov I.A. *Probabilistic Reliability Models*. Wiley, 2012. 248 p.
8. Polovko A.M., Gurov S.V. *Osnovy teorii nadezhnosti* [Fundamentals of the reliability theory]. St. Petersburg, BHV-Peterburg Publ., 2006. 704 p.
9. Baikhel't F., Franken P. *Nadezhnost' i tehnikeskoye obsluzhivaniye. Matematicheskiy podhod* [Reliability and maintenance. Mathematical approach]. Moscow, Radio i sviaz Publ., 1988. 392 p.
10. Barlow P., Hunter L. Optimal'nyi poryadok provedeniya profilakticheskikh rabot [Optimal order of preventive maintenance]. Optimal'nyye zadachi nadezhnosti. Moscow, Standarty Publ., 1968, pp. 244-255. (In Russian).
11. Peschansky A.I. Polumarkovskaya model' tehnikeskogo obsluzhivaniya monotonnoy sistemy s uchetom vozrasta i otklyucheniye elementov [Semi-Markov model of maintenance of a monotonous system with regard to age and its elements' deactivation]. *Sistemnyye tehnologii*, 2009, vol. 61, no. 2, pp. 29-41.
12. Obzherin Y.E., Peschansky A.I. Semi-Markovian Model of Monotonous System Maintenance with Regard to its Element Deactivation and Age. *Applied Mathematics*, 2010, vol. 1, no. 3, pp. 234-243. DOI: 10.4236/am.2010.13029.
13. Obzherin Y.E., Peschansky A.I. Kalendarnoye tehnikeskoye obsluzhivaniye sistem s proizvol'noy strukturoy [Calendar maintenance of systems with an arbitrary structure]. *Kibernetika i sistemnyy analiz*, 2006, no. 2, pp. 69-86.
14. Korlat A.N., Kuznetsov B.N., Novikov M.I., Turbin A.F. *Polumarkovskiy modeli vosstanavlivayemykh sistem i system massovogo obsluzhivaniya* [Semi-Markov models of restorable and queuing systems]. Kishinev, Shtiintsa Publ., 1991. 209 p. (In Russian).
15. Korolyuk V.S., Turbin A.F. *Protsessy markovskogo vosstanovleniya v zadachah nadezhnosti sistem* [Markovian restoration processes in the system reliability problems]. Kiev, Nauk. dumka Publ., 1982. 236 p. (In Russian).

Received 28.11.2018

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 681.518: 339.13

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ БИЗНЕС-ПРОЦЕССА РАЗРАБОТКИ И КАСТОМИЗАЦИИ ИНТЕГРАЦИОННЫХ РЕШЕНИЙ В ИНТЕРЕСАХ УПРАВЛЕНИЯ ИТ-КОМПАНИЕЙ

Димов Э.М., Маслов О.Н., Хаджиева С.В.

Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ
E-mail: ssv2507@mail.com

В современных условиях ИТ-компании системные интеграторы вынуждены бороться не только за возможность быть первыми на рынке с новым разработанным на инновационных технологиях продукте или сервисе, но и за такой важный ресурс, как персонал. Предметом исследования является процесс формирования команды, реализующей разработку и кастомизацию интеграционных решений, который протекает в сложной нерефлекторной системе социально-экономического типа. Для исследования влияния распределения исполнителей по командам на результат функционирования бизнес-процесса предложено использовать метод статистического имитационного моделирования по версии Димова-Маслова. Авторами изучена специфика деятельности ИТ-компаний при реализации современных интеграционных решений, описана модель функционирования исследуемого бизнес-процесса, описана постановка задач моделирования в интересах управления, которые можно решить с помощью

метода статистического имитационного моделирования. Использование рекомендаций по совершенствованию способов формирования команд, разрабатывающих интеграционные решения, на основании результатов имитационного моделирования позволит выявить узкие места, учесть влияние случайных факторов на бизнес-процесс разработки и кастомизации интеграционных решений, отследить подробную статистику и поведение системы во времени и проиграть сценарии при различных вариантах укомплектования команд.

Ключевые слова: статистическое имитационное моделирование, метод Димова-Маслова, нерелефторная система, формирование команды, ИТ-компания, системный интегратор, сервис-ориентированная архитектура

Введение

Для современного бизнеса умение сотрудников работать в команде приобретает очень большое значение. Команда, работающая слаженно, продвигает компанию вперед, в то время как плохо функционирующая команда препятствует развитию компании. По сравнению с административным управлением командный метод работы является более эффективным. Однако переход на него – процесс нелегкий, он требует перестройки мышления, поведения и способов принятия решений. Особенно остро вопросы командообразования и командоуправления стоят в сфере информационных технологий (ИТ), где имеет место высокая степень неопределенности в постановках задач от бизнес-заказчиков – лиц, принимающих решения (ЛПР) [1].

Целью исследования является совершенствование деятельности ИТ-компания путем повышения качества принимаемых управленческих решений при формировании команд, разрабатывающих интеграционные решения, на основе результатов статистического имитационного моделирования (СИМ) по методу Димова-Маслова (МДМ). В качестве объекта исследования рассматривается ИТ-компания системный интегратор. Предмет исследования – процесс формирования команды, реализующей разработку и кастомизацию интеграционных решений.

Кастомизация (customization) – это адаптация массового продукта под запросы конкретного бизнес-заказчика путем частичного изменения продукции под конкретный запрос. Термин происходит от английского customer – клиент, потребитель.

Статья посвящена важнейшему этапу СИМ по МДМ – постановке задачи моделирования исследуемого бизнес-процесса в интересах управления им. Данный этап содержит вербальное описание процесса работы системы управления с включением СИМ-модели в контур управления вместе с ЛПР-оператором для оперативного поиска и реализации воздействий, управляющих социально-экономической системой с целью обеспечения наивысшей эффективности ее функционирования в каждый текущий период времени.

Описание объекта исследования

Основные бизнес-процессы ИТ-компания представлены на рисунке 1. Такие компании разрабатывают и реализовывают собственные консалтинговые решения с одновременной интеграцией электронных приложений в рабочие процессы. Этим они отличаются от компаний, представляющих и продвигающих на ИТ рынке лишь какой-то определенный комплекс программного обеспечения, целенаправленно поставляя под него оборудование или оказывая услуги по его установке и наладке.

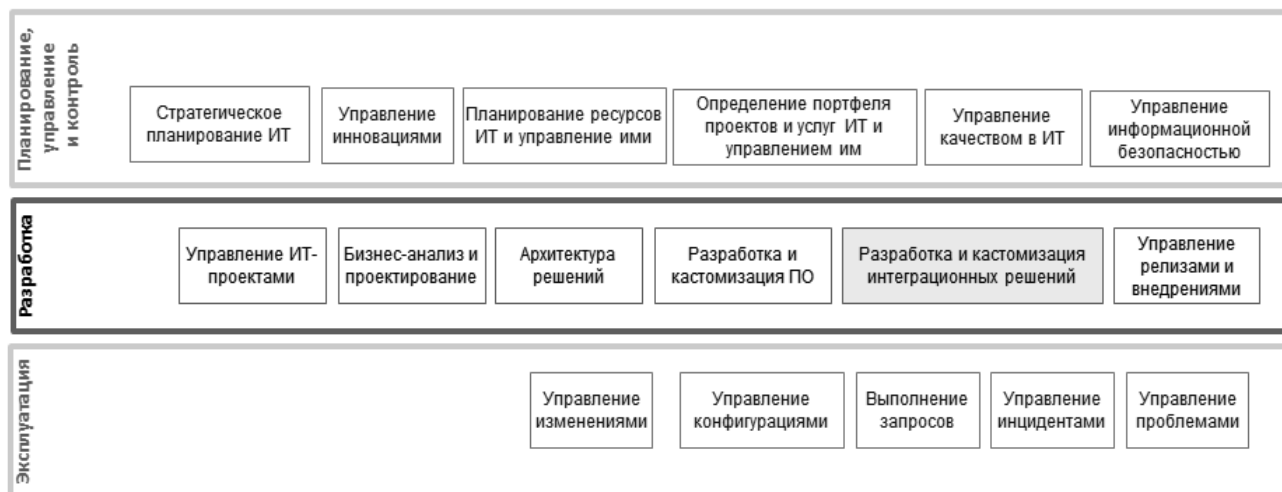


Рисунок 1. Карта основных бизнес-процессов ИТ-компания

Системный интегратор предлагает комплексное решение под конкретного заказчика, которое начинается с разработки ИТ-стратегии, построения инфраструктуры, внедрения прикладных компьютерных систем и приложений, а заканчивается автоматизацией и обслуживанием рабочего процесса. Такая компания специалистов объединяет работу программистов и оптимизаторов, решает уникальные инженерные и маркетинговые задачи [2].

Специфика предметной области

Воплощение в жизнь принимаемых управленческих решений зачастую требует оперативного изменения бизнес-процессов, для чего необходимо резко повысить степень гибкости и адаптивности имеющихся корпоративных информационных систем, поддерживающих бизнес-процессы. Без внедрения современных аналитических и интеграционных решений это невозможно. Интеграционные решения являются технологическим фундаментом для цифровой трансформации.

Традиционные архитектурные подходы и стили решений на базе замкнутых коммерческих или заказных систем поддержки отдельных направ-

лений деятельности не были ориентированы на эффективную реализацию сквозных интегрированных бизнес-процессов, которые должны опираться на функциональные сервисы множества корпоративных информационных систем компании, ее подразделений, филиалов и внешних контрагентов. Многие компании видят выход в том, чтобы переориентировать деятельность и ее информационно-технологическую поддержку на управляемую бизнес-требованиями сервис-ориентированную архитектуру (Service-Oriented Architecture, далее SOA), предложенную компанией IBM [3].

Согласно [4], SOA – это «архитектурный стиль для создания ИТ-архитектуры предприятия, использующий принципы ориентации на сервисы для достижения тесной связи между бизнесом и поддерживающими его информационными системами». Архитектурный подход SOA поддерживают практически все мировые лидеры ИТ-индустрии. Интерес к использованию SOA во многом определяется наличием в компаниях мощной системы средств автоматизации, оперативную модернизацию которой проводить становится все сложнее и сложнее.

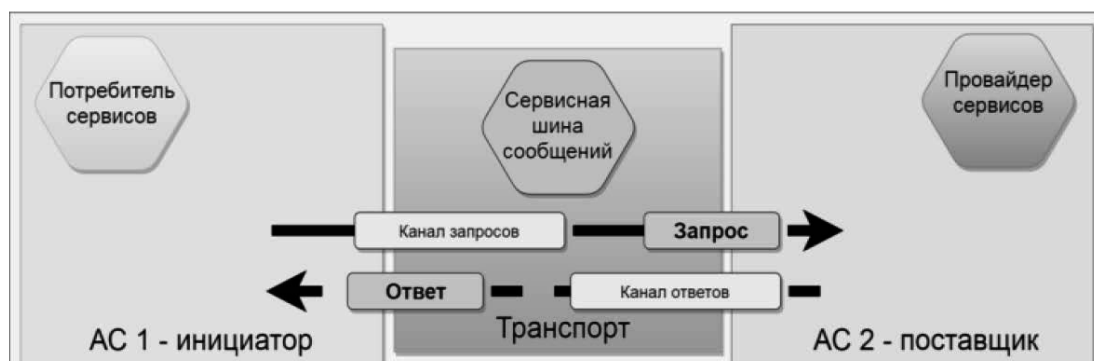


Рисунок 2. Интеграция через ESB

Сервисы – это базовые компоненты архитектуры, которые представляют видимый ресурс, выполняющий повторяющуюся задачу и описанный внешней инструкцией. Сервисы позволяют вызывать функциональность из других приложений и подсистем.

Интеграционная сервисная шина предприятия, ESB (Enterprise Service Bus), играет роль транспорта для сервисов и служит для формирования единой среды взаимодействия различных приложений предприятия. АС-инициатор (сервис или модульное приложение) отправляет запрос

на сервисную шину, которая преобразует сообщение в формат, поддерживаемый в точке назначения, и перенаправляет запрос АС-получателю. АС-получатель обрабатывает полученный запрос и возвращает ответ в сервисную шину, которая преобразует сообщение в формат отправителя (см. рисунок 2).

Все взаимодействие идет через сервисную шину, так что если она падает, то с ней падают и все остальные системы. Другими словами, ESB является ключевым посредником и достаточно сложным компонентом системы [5].

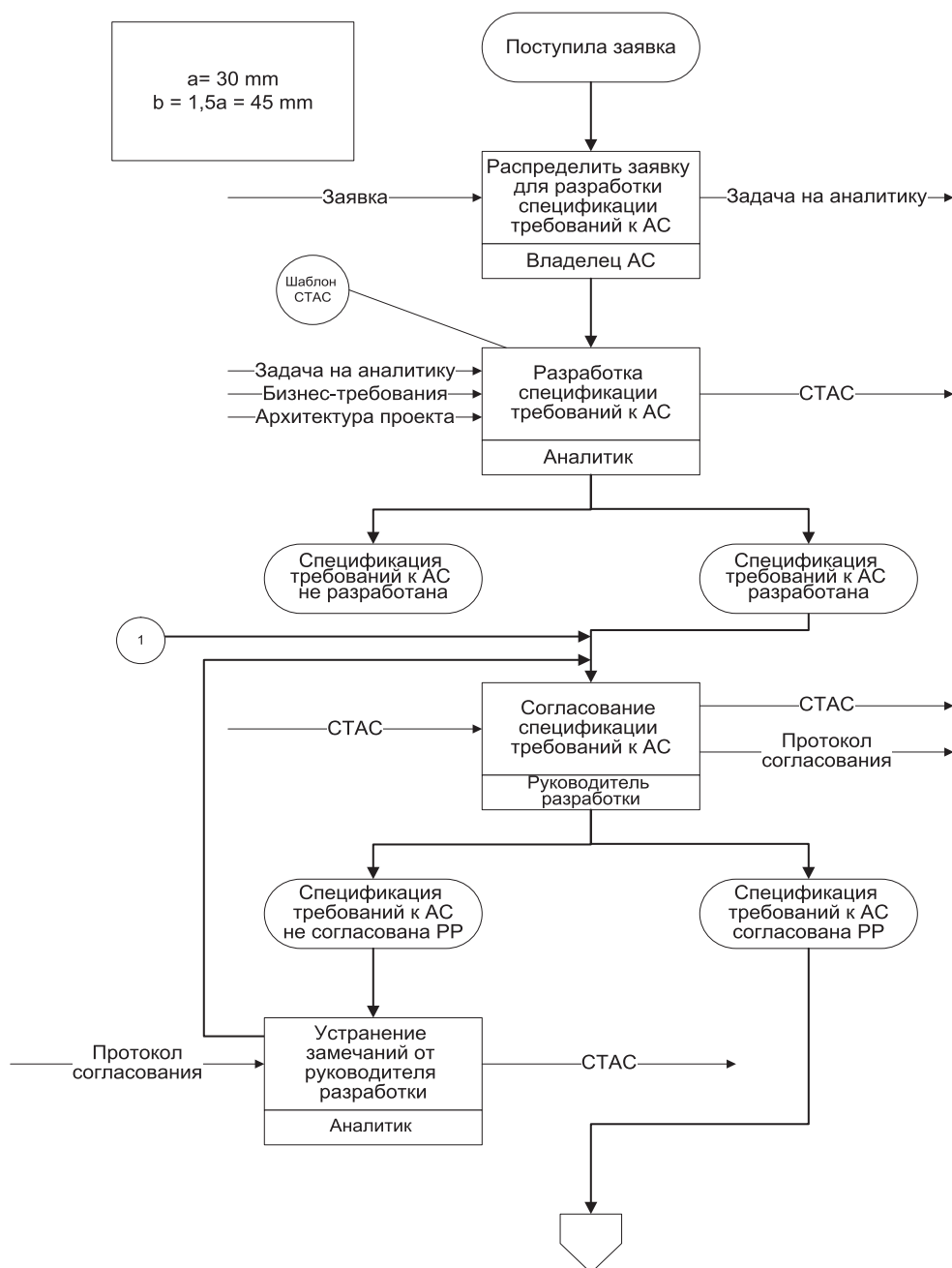


Рисунок 3. Схема бизнес-процесса разработки и кастомизации интеграционного решения

Анализ бизнес-процесса разработки и кастомизации интеграционных решений в интересах управления

Формализованная модель бизнес-процесса разработки и кастомизации интеграционных решений (см. рисунок 1) в нотации блок-схема представлена на рисунке 3. Бизнес-процесс начинается с поступления в отдел интеграции АС заявки на разработку интеграционного решения. Получив заявку, владелец АС распределяет задачу на проведение аналитики. Аналитик разрабатывает документ «Спецификация требований к АС». В документе фиксируется перечень необ-

ходимых к выполнению работ: разработка, доработка или использование сервиса (сервисов).

В процессе подготовки спецификации требований к АС аналитик опирается на информацию, описанную в бизнес-требованиях и концептуальной архитектуре. В случае отсутствия этих документов или неполноты информации в них – аналитик консультируется непосредственно с ИТ-аналитиком, бизнес-аналитиком, ИТ-архитектором, руководителем проекта.

Также аналитик сервиса выстраивает коммуникации с аналитиками смежных АС. Смежные АС являются поставщиками и потребителями

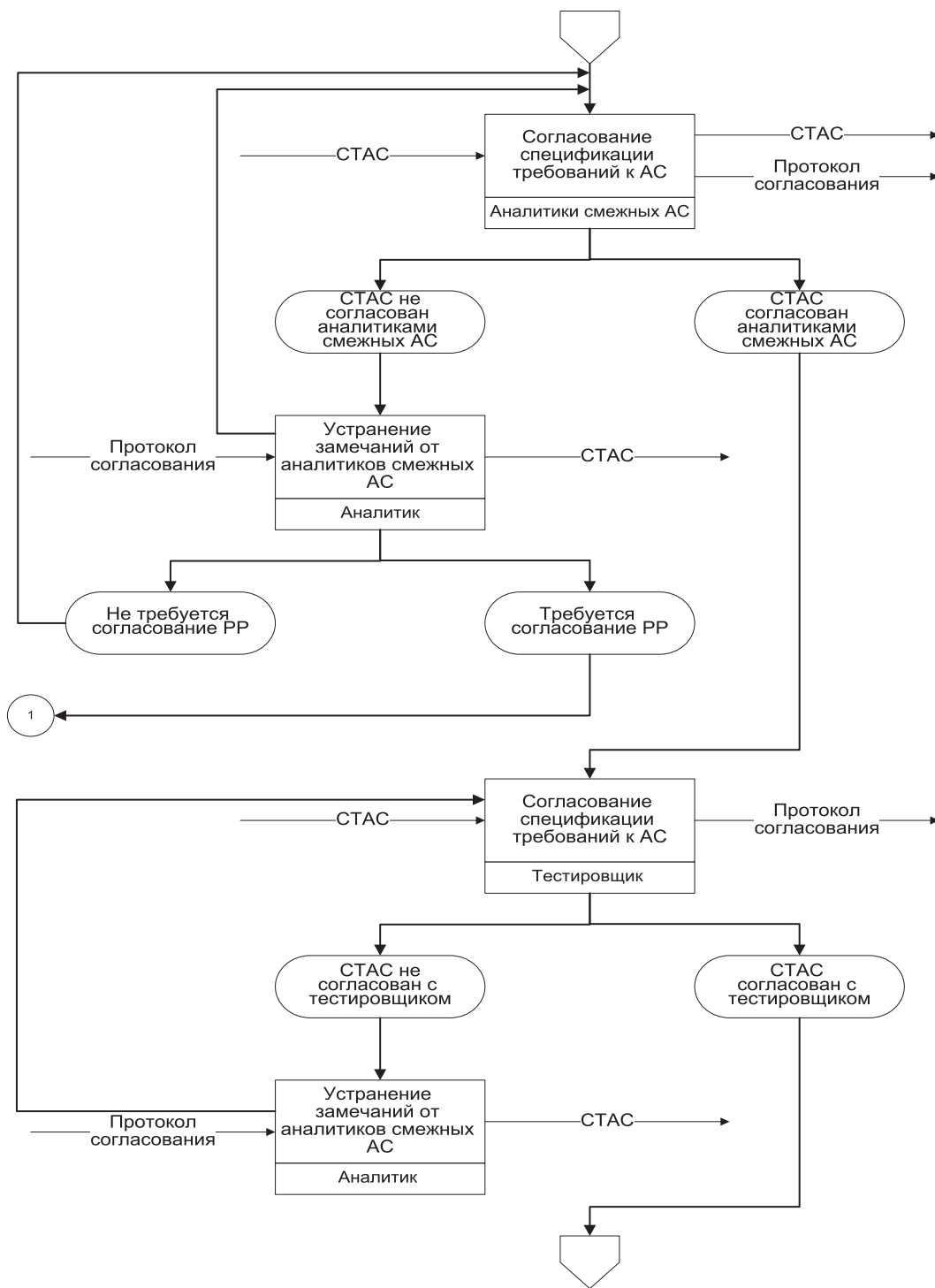


Рисунок 3 (продолжение)

разрабатываемого сервиса. В случае если информации недостаточно, то аналитик останавливает задачу до момента предоставления информации и берет в работу новую задачу.

Если информации достаточно для функционального проектирования сервиса, аналитик фиксирует информацию о назначении сервиса в спецификации требований к АС. Кроме документа аналитик готовит схему данных сервиса, с помощью которой АС-поставщик сможет

корректно сформировать сообщения, которые затем будут приняты и прочитаны сервисом для последующего выполнения инструкций, заложенных в сообщении, как правило, это передача данных АС-потребителю. После подготовки спецификации требований к АС аналитик сервиса последовательно согласовывает документ со всеми заинтересованными лицами: руководителем разработки, смежными АС, отделом тестирования.

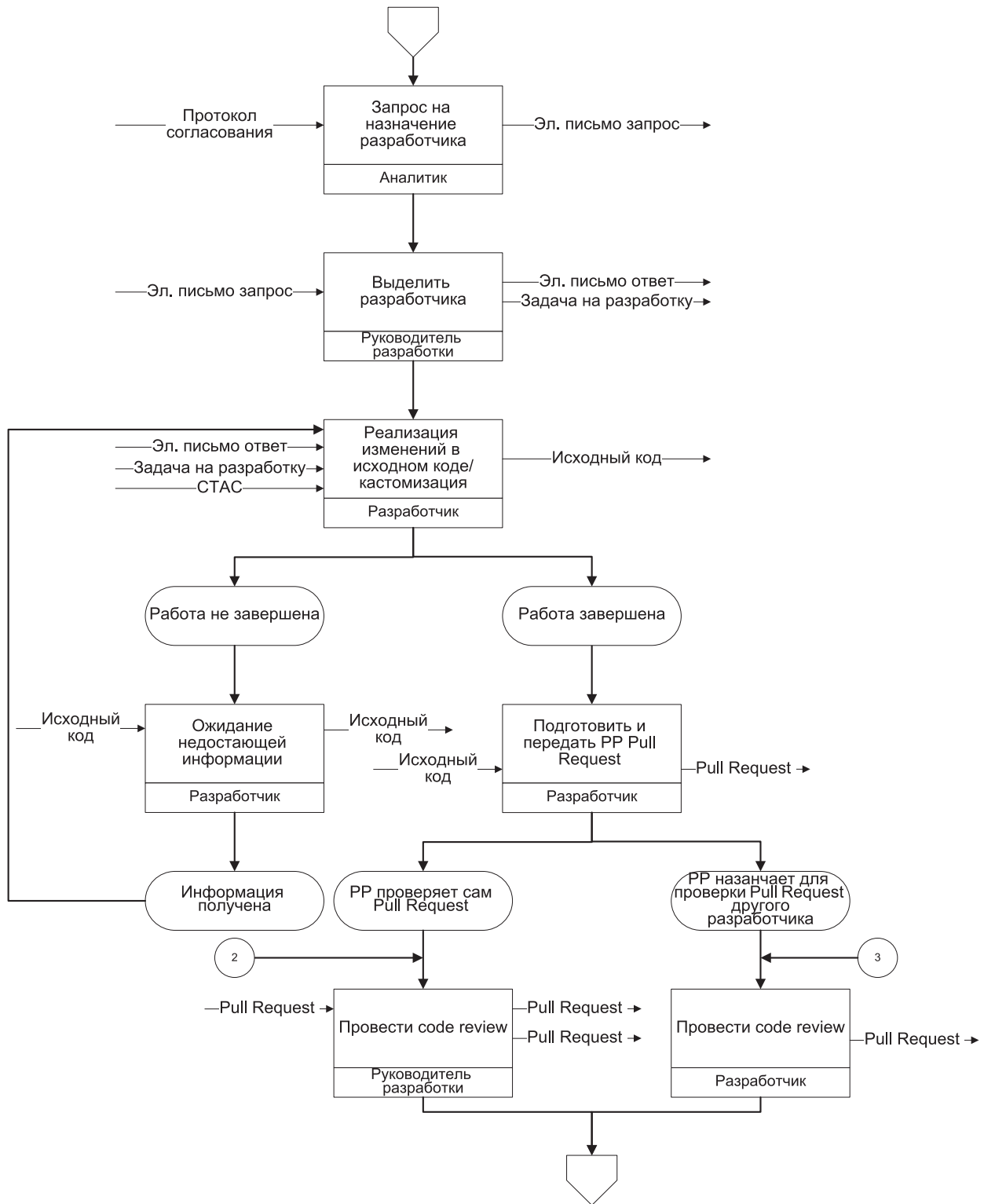


Рисунок 3. (продолжение)

В первую очередь требования должны быть согласованы со стороны руководителя разработки. Если у руководителя разработки имеются замечания, то документ возвращается аналитику сервиса. Аналитик сервиса фиксирует протокол замечаний и приступает к их устранению. Если подготовленное аналитиком реше-

ние возможно к реализации, то руководитель разработки согласовывает спецификацию требований. После успешного согласования с руководителем разработки спецификация требований должна быть согласована со смежными АС, которые будут использовать разрабатываемый сервис. Далее спецификация требований

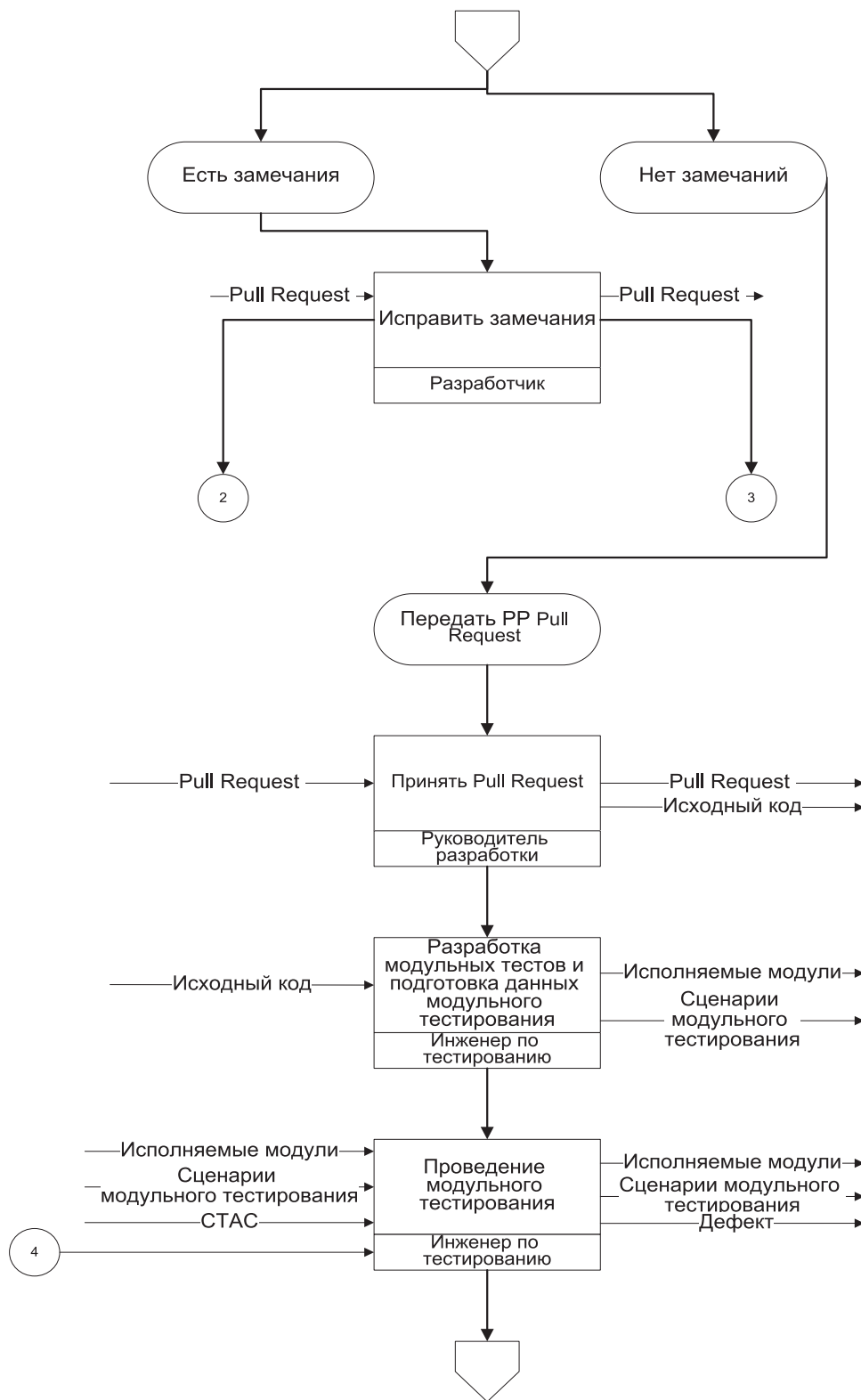


Рисунок 3 (продолжение)

согласовывается с отделом интеграционного тестирования.

В случае успешного согласования отдел интеграционного тестирования включает задачу в свой перечень работ и ожидает выпуск дистрибутива на стенд тестирования. После согласования

спецификации требований к АС со всеми заинтересованными лицами начинается этап программной разработки интеграционного решения. Для этого аналитик делает запрос руководителю разработки на назначение разработчика. Руководитель выбирает разработчика и назначает задачу.

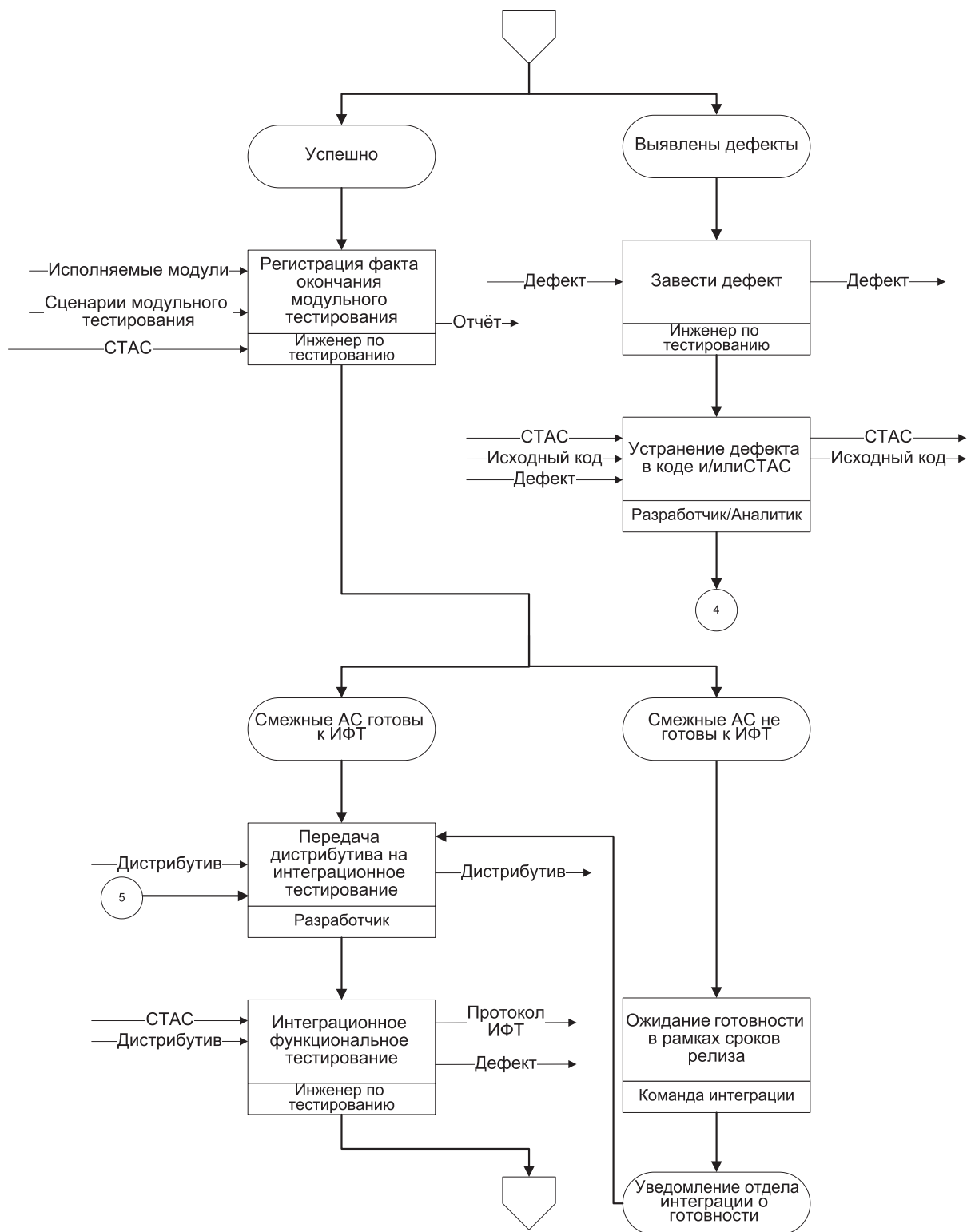


Рисунок 3 (продолжение)

Разработчик получает задачу, проверяет наличие входных артефактов, как правило, это спецификация требований к АС, но в отдельных случаях может понадобиться информация технического характера от разработчиков смежных АС.

Разработчик вносит необходимые изменения в исходный код. Результатом выполнения задачи яв-

ляется Pull Request – запрос к руководителю разработки, управляющему репозиторием, на выполнение изменений из репозитория разработчика. Далее Pull Request подвергается проверке исходного кода программы с целью обнаружения и исправления ошибок, которые остались незамеченными в начальной фазе разработки. Целью просмотра явля-

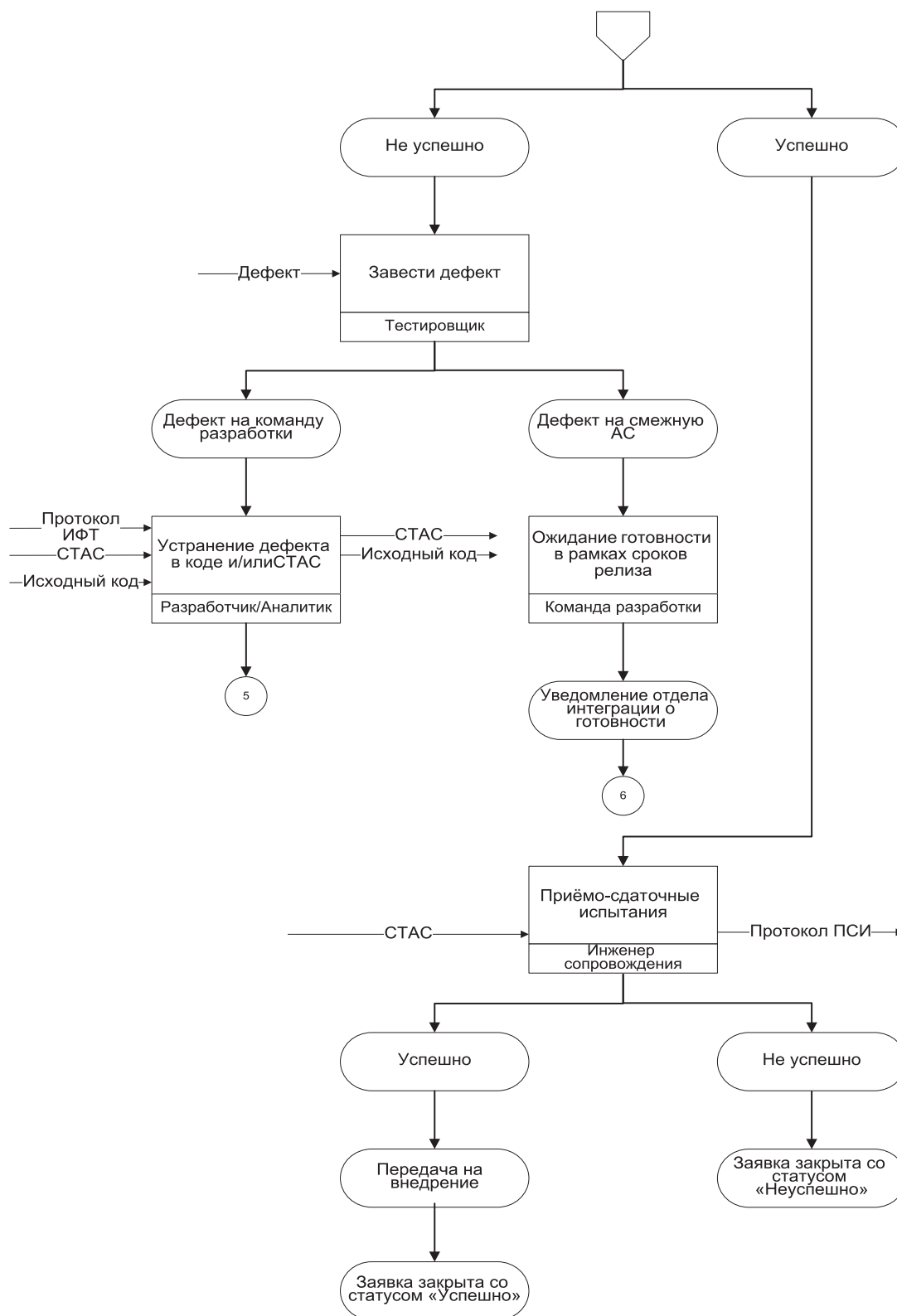


Рисунок 3 (окончание)

ется улучшение качества программного продукта и совершенствование навыков разработчика. Данная процедура проверки кода называется Code Review.

Проверку исходного кода может проводить сам руководитель разработки или он может делегировать проверку другому разработчику.

После устранения всех замечаний (при их наличии) разработчик передает Pull Request руководителю разработки для включения его в основной код.

Следующий этап предусматривает проведение тестирования. Первый этап тестирования называется

модульным и проводится каждой АС (АС 1, АС 2, АС интегратором) независимо друг от друга. Успешное прохождение модульного тестирования всеми участниками интеграционного решения является стартом для подготовки работ к интеграционному функциональному тестированию (ИФТ), при котором отдельные программные модули объединяются и тестируются в группе. В качестве входных данных используются модули, над которыми было проведено модульное тестирование. Если в ходе тестирования были выявлены ошибки, инженер по тестированию заводит дефект на АС, которая является причиной дефекта.

Результатом успешного завершения ИФТ является работоспособное интеграционное решение, что подтверждает протокол ИФТ, подготовленный инженером по тестированию. Протокол ИФТ также является документом, обеспечивающим начало приемосдаточных испытаний (ПСИ). Если в ходе ПСИ у комиссии возникают замечания функционального или программного характера относительно разработанного решения, то все указанные замечания фиксируются в протоколе ПСИ и прикладываются к заявке. Далее инициатор заявки, как правило, руководитель проекта закрывает заявку окончательно или с возможностью доработки. В случае успешного прохождения ПСИ в протоколе фиксируются рекомендации к внедрению решения в промышленную эксплуатацию. Этапы ПСИ и внедрения находятся за рамками исследуемого бизнес-процесса и рассмотрены только в качестве логического понимания завершения работ над заявкой.

Постановка задачи СИМ

В конкурентной борьбе ИТ-компании соревнуются не только за долю на рынке, но и за инвестиции и персонал. Для получения конкурентных преимуществ ИТ-компании вынуждены трансформировать бизнес. Трансформация подразумевает переход компании с одного типа управления ИТ-проектами на другой. Для цифровой трансформации характерен переход на гибкие методологии в управлении ИТ-проектами, ориентированные на организацию работ небольшими командами.

Процесс формирования команды – задание, требующее высокой управленческой компетенции. При его осуществлении требуется не только наличие правильно подобранных, высококвалифицированных ИТ-специалистов, но и людей, желающих работать вместе, сообщая, как команда [1]. Определений «команды» существует несколько десятков, например, М. Армстронг приводит следующее определение: «Команда – это небольшая группа людей, взаимодополняющих и взаимозаменяющих друг друга, которые собраны для совместного решения задач производительности и в соответствии с подходами,

посредством которых они поддерживают взаимную ответственность» [6]. Применительно к сфере ИТ под командой понимают кроссфункциональную группу в 5-10 чел., которая наделена полномочиями и способна определить, разработать и протестировать инкремент ценности решения за обозначенный временной интервал. Включает следующие роли: команда разработки (аналитики, разработчики, тестировщики), Scrum-мастер и владелец продукта.

В своем развитии команды проходят несколько этапов, которые сменяют друг друга (см. таблицу 1). На этапе формирования команда не определяет свой состав сама, она формируется сверху, от руководства. Фактическую производительность команды можно оценить только на этапе функционирования команды. Получить прогнозные значения производительности команды возможно, если на этапе формирования команды ЛППР будет иметь данные о результатах того или иного распределения исполнителей по командам.

У каждого участника команды есть две роли: функциональная и командная. Функциональные роли относятся к должностным обязанностям и охватывают навыки, умения, знания и опыт. Командные роли отражают личностные качества, которые являются устойчивой характеристикой сотрудника и остаются неизменными в разных командах.

Таблица 1. Этапы формирования команд

Этап	Описание этапа
Формирование	На этапе формирования происходит создание команды и постановка целей, распределение и закрепление функциональных ролей.
Бурление	На этапе «бурления» участники осознают цели. Здесь возможны конфликты и противостояние между членами команды.
Нормализация	На этапе нормализации члены команды «притираются» друг к другу и начинают двигаться сонаправлено.
Функционирование	На этапе функционирования команда становится самоуправляемой и способной оптимизировать свою производительность.
Расформирование	Этап расформирования наступает, когда цели, поставленные перед командой, достигнуты.

Рекомендации по распределению командных ролей описывают качества, которыми должна обладать сформированная команда, однако вопросам распределения сотрудников на команды по функциональным ролям уделено недостаточно внимания [6-8].

Оценку производительности команды, риск срыва сроков разработки и внедрения невозможно рассчитать по аналитическим моделям, поскольку на эти показатели влияют несколько факторов, среди которых есть стохастические. Например, количество поступивших заявок на релиз, длительность подготовки требований, длительность согласования требований, длительность разработки, вероятность обнаружения дефекта при тестировании, длительность исправления дефекта в документации, длительность исправления дефекта в коде и др.

В гибких командах работа ведется параллельно: контроль и «власть» распределены между командами. Решения, из которых складывается конечный результат, принимаются каждой командой отдельно, разделяя, но не размывая ответственность между всеми командами. Распараллеливание процедур сбора и обработки информации приводит к децентрализации процедур принятия решений, то есть к созданию самостоятельно функционирующих подсистем, что означает появление в системе иерархической структуры. Для управления социально-экономической системой, обладающей иерархической структурой, необходимо рассмотреть исследуемый бизнес-процесс разработки и кастомизации интеграционных решений как систему нерелекторного типа.

Нерелекторная система – это система, которая может неоднозначно, многовариантно реагировать на одно и то же внешнее воздействие. Управление в таких системах связано с наличием человеческого фактора, что значительно усложняет процесс управления. Рассмотрим пример нерелекторной системы [9-10].

Имеется объединение N промышленных предприятий (концерн или трест), выпускающих однотипную продукцию. Назовем такое объединение центром. Обозначим через P_i продукцию, выпускаемую i -ым предприятием из общего их числа n , то есть $i \in [1; n]$. Результат функционирования центра определяется результатами функционирования отдельных производителей. Оценки этого результата могут быть различными. Центр также производит продукцию, и его целевая функция однозначно определяется продукцией производителей

$$J = J(P_1; P_2; P_3 \dots P_n) \quad (1)$$

Центр не имеет права назначать объемы производства P_i , но может влиять на эти объемы. Величина продукта, произведенного i -ым производителем, определяется объемом его фондов x_i и количеством рабочей силы L_i :

$$P_i = f_i(x_i; L_i). \quad (2)$$

Функция $f_i(x_i; L_i)$ в теории управления именуется производственной функцией. Существуют различные способы аппроксимации производственной функции. Доход i -го производителя J_i равен стоимости произведенной продукции за вычетом накладных расходов. Для простоты будем считать, что накладные расходы состоят только в оплате рабочей силы. Если обозначить через ω_i фиксированную ставку заработной платы, то каждая величина J_i будет равна

$$J_i = c P_i - \omega_i L_i \quad (3)$$

Если величина фондов x_i также фиксирована, то объем продукции определяется количеством рабочей силы L_i . Таким образом, L_i является управляющим параметром, который полностью находится в распоряжении производителя. Для того чтобы управлять действиями производителей, центр должен располагать способами эффективного воздействия на них.

Рассмотрим простейший способ: распределение экзогенного ресурса U , который полностью находится в распоряжении центра и может расходоваться, например, на инвестиции, на создание основных фондов производителя и т.п. Тогда задача центра, которую можно назвать задачей планирования, состоит в таком распределении ресурса U , которое приводит к максимуму функцию (1):

$$U = \sum_{i=1}^N u_i. \quad (4)$$

При этом эффективность распределения U будет зависеть не только от действий центра, но и от значений L_i , которые выбираются производителями. Таким образом, складывается игровая ситуация, в которой, по определению Ю.Б. Гермейера, имеется $N + 1$ игроков: N производителей и центр [11].

Для того чтобы эффективно реализовать процедуру управления, игрокам необходимо условиться о «порядке ходов», «гипотезе информированности» и «гипотезе поведения». В данной игровой ситуации право первого хода принадлежит центру – который делает этот ход, передавая i -му произво-

дителю ресурс u_i и учитывает, что производитель с этого момента знает величину u_i .

Основным теперь становится вопрос о гипотезе поведения игроков: предположим, что центр знает (или считает, что знает) интересы производителей, полагая, что они описываются целевой функцией (1). Тогда гипотеза центра о поведении производителя состоит в том, что он должен так выбрать свое управление L_i , чтобы максимизировать доход (1). Применительно к сфере исследуемого бизнес-процесса центром является объединение N -команд, ведущих разработку для одного или нескольких ИТ-проектов. Выпускаемой продукцией являются готовые интеграционные решения, которые разрабатываются каждой командой в отдельности. Целевая функция будет зависеть от производительности всех команд.

Исследование данного процесса привело к выводу о том, что наиболее перспективным методом его моделирования в интересах совершенствования управления является СИМ по МДМ [12].

Применение СИМ по МДМ предполагает использование наиболее развитых (по сравнению с другими подходами) способов целенаправленного повышения эффективности управления сложными системами на основе моделирования динамики их функционирования с учетом случайных факторов, влияющих на принятие управленческих решений.

Прикладная ценность СИМ по МДМ заключается в том, что он позволяет на основе хорошо апробированных и относительно простых графических средств, без потери необходимой точности и достоверности, формализовать исследуемые сложные системы на выбранном уровне их абстрагирования, а затем перейти от содержательного описания сложной системы к непосредственной программной реализации СИМ-модели. Целесообразность разработки и практического применения данной модели обусловлено необходимостью использовать метод альтернативных вариантов (сценариев) при проектировании и анализе эффективности сложной системы, предусматривающий сравнение между собой разных форм организации сложной системы с последующим выбором и реализацией наилучшей из них. Рассматриваются следующие сценарии:

Сценарий 1. В стандартном сценарии заявка поступает в отдел интеграции, где назначается на любого аналитика, далее на разработчика. После заявка поступает в отдел тестирования и распределяется на любого тестировщика. Выбор исполнителя осуществляется руководителем разработки, исходя из имеющейся у него информации о

занятости исполнителей. Исполнитель каждой роли «аналитик», «разработчик», «тестировщик» выполняют строго обозначенный перечень работ, обязанностей, которые соответствуют компетенциям роли.

Сценарий 2. Здесь рассматривается вариант реорганизации структуры управления в компании в условиях цифровой трансформации бизнеса, которая предполагает объединение отделов разработки и тестирования с целью разделения исполнителей на команды. Вопрос укомплектования команд является актуальным и сложным, так как необходимо учесть влияние большего числа случайных факторов и условий. Предполагается разделить N исполнителей на k команд, при этом должны выполняться условия:

- команда обязательно должна включать представителя каждой роли «аналитик», «разработчик», «тестировщик»;

- команды должны быть примерно равные между собой по производительности (то есть нельзя, например, чтобы в одной команде были только опытные сотрудники, а в другой команде объединить «новичков»). При этом нужно учесть распределение исполнителей не только по командным ролям, но и по функциональным ролям;

- необходимо учесть, что исполнители N территориально находятся в разных городах G , поэтому важно исключить ситуацию, когда исполнитель оказался бы единственным участником команды в городе G_n ;

- необходимо оценить влияние на общую производительность принципа функционального укомплектования команд. Функциональное распределение предполагает, что каждая команда будет специализироваться на каком-то конкретном виде интеграции (функции). Здесь может возникнуть ситуация, когда команда будет наращивать опыт в интеграции с выделенными ей функциями, достигнет максимума, что позволит быстро, с минимальным количеством ошибок проектировать интеграционные решения, но при этом есть и отрицательная сторона – закрепление за командой конкретных видов интеграции не позволит наращивать компетенции и опыт по другим видам интеграций;

- необходимо оценить влияние на общую производительность принципа горизонтального развития компетенций, который предполагает возможность одной функциональной роли выполнять другие роли. То есть аналитик может при необходимости выполнить некоторые задачи разработчика, если, например, разработчик перегружен, или может протестировать самостоятельно

но часть кода, если тестировщик заболел, ушел в отпуск или просто перегружен работой и т.д.

Целью исследования сложной системы является предсказание ее поведения в будущем – на основе прошлых и настоящих фактических данных о ней, с учетом возможной динамики параметров внешней среды, влияния внешних и внутренних случайных факторов. В качестве входных данных ЛППР будет вводить данные о желаемом распределении исполнителей по командам, задавать условия, ограничения и требования, предъявляемые к командам. Далее будет эмулироваться обработка заявок за период моделирования (один спринт или один релиз) получившимися командами.

Моделирование позволяет выявить узкие места, учесть влияние случайных факторов на бизнес-процесс разработки и кастомизации интеграционных решений, отследить подробную статистику и поведение системы во времени и проиграть сценарии при различных вариантах укомплектования команд.

Практическая значимость работы заключается в возможности принятия ИТ-менеджерами обоснованных решений на стадии формирования команд разработки с целью повышения эффективности работы команд в условиях трансформации бизнеса и перехода на командное управление.

Заключение

В статье рассмотрена функциональная характеристика объекта исследования и специфика предметной области, разработана содержательная и формализованная модели бизнес-процесса разработки и кастомизации интеграционных решений, разработаны и описаны задачи управления сложной системой, которые можно решить с помощью СИМ по МДМ.

Литература

1. Новиков В.Е. Формирование эффективной команды ИТ-проекта // «Системы управления бизнес-процессами» // URL: <http://journal.itmane.ru/node/235> (д.о. 12.02.2019).

2. Системный интегратор // URL: <https://it-ping.ru/stati/sistemnyij-integrator> (д.о. 12.02.2019).
3. Smart SOA: разумный подход ИТ-лидера // Инновации в технологиях и бизнесе. – 2008. – №4. // URL: https://www.osp.ru/data/893/418/1237/ibm_4_10%204 (д.о. 22.11.2018).
4. Портье Б. Обзор терминологии SOA. Ч.1. Сервис, архитектура, управление и бизнес-термины // IBM developer Works URL: ibm.com/developerWorks/ru/ (д.о. 20.12.2018).
5. Интеграция приложений и сервисов. ВСС Group (ООО «Би.Си.Си.») // URL: <http://bcc.ru/> (д.о. 12.02.2019).
6. Вольфсон Б.И. Гибкие методологии разработки. – СПб: Питер, 2017. – 144 с.
7. Стелман Э., Грин Дж. Постигая Agile, Ценности, принципы, методологии. Пер. с англ. – М.: Манн, Иванов и Фербер, 2018. – 448 с.
8. Селюк А. В., Денисова С.С. Управление проектной командой. – Тюмень: Изд. ТГУ, 2013. – 216 с.
9. Димов Э.М., Маслов О.Н., Пчеляков С.Н., Скворцов А.Б. Новые информационные технологии: подготовка кадров и обучение персонала. Ч.2. Имитационное моделирование и управление бизнес-процессами в инфокоммуникациях, – Самара: Изд-во СНЦ РАН, 2008. – 350 с.
10. Ануфриев Д.П., Димов Э.М., Маслов О.Н., Трошин Ю.В. Статистическое имитационное моделирование и управление бизнес-процессами в социально-экономических системах. – Астрахань: Изд. АстИСИ, 2015. – 365 с.
11. Гермейер Ю.Б. Игры с непротивоположными интересами (теория принятия решений при неполном единстве). – М.: Изд. МГУ, 1972. – 256 с.
12. Ануфриев Д.П., Димов Э.М., Маслов О.Н., Халимов Р.Р. Сравнительная эффективность методов и средств информационной поддержки управленческих решений // Инфокоммуникационные технологии. – 2014. – Т.12. – №1. – С. 54-67.

Получено 15.11.2018

Димов Эдуард Михайлович, д.т.н., профессор Кафедры прикладной информатики (ПИ) Поволжского государственного университета телекоммуникаций и информатики (ПГУТИ). Тел. 8-906-340-37-41. E-mail: e.m.dimov@gmail.com

Маслов Олег Николаевич, д.т.н., профессор, заведующий Кафедрой ПИ ПГУТИ. Тел. 8-917-950-05-13. E-mail: maslov@psati.ru

Хаджиева Светлана Владимировна, аспирант Кафедры ПИ ПГУТИ. Тел. 8-927-731-47-26. E-mail: ssv2507@mail.ru

SIMULATION MODELING BUSINESS PROCESS OF DEVELOPMENT AND CUSTOMIZATION OF INTEGRATION DECISIONS IN THE INTERESTS OF IT-COMPANY MANAGEMENT

Dimov E.M., Maslov O.N., Khadzhieva S.V.

Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation

E-mail: ssv2507@mail.ru

IT system integrator companies are struggling not only for the opportunity to be the first in the information technology market, but also for employees. The subject of the research is the process of distributing employees to teams. To assess the effectiveness of the distribution of employees to teams, it was proposed to use the method of statistical simulation modeling according to Dimov-Maslov. The authors studied the specifics of the activities of IT companies, described a business process model and performed a formulation of modeling problems in the interests of management. Using the obtained results will allow identifying bottlenecks in the functioning of the business process, taking into account the influence of random factors, tracking the statistics and system behavior in time and losing scenarios for various manning options.

Keywords: *statistical simulation modeling, Dimov-Maslov method, non-reflector system, team building, IT system integrator company, service-oriented architecture*

DOI: 10.18469/ikt.2019.17.1.08

Dimov Eduard Mikhailovich, Povolzhskiy State University of Telecommunications and Informatics, 77, Moskovskoe shosse, Samara 443090, Russian Federation; Professor of the Department of Applied Informatics, Doctor of Technical Science, Professor. Tel. +79063403741. E-mail: e.m.dimov@gmail.com

Maslov Oleg Nikolayevich, Povolzhskiy State University of Telecommunications and Informatics, 77, Moskovskoe shosse, Samara, 443090, Russian Federation; the Head of Department of Applied Informatics, Doctor of Technical Science, Professor. Tel. +79023710624. E-mail: maslov@psati.ru

Khadzhieva Svetlana Vladimirovna, Ph. D Student of Applied Informatics Department, Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation. Tel. +79277314726. E-mail: ssv2507@mail.ru

References

1. Novikov V.E. Formirovanie effektivnoy komandy IT-proekta [Formation of an effective IT project team] *Sistemy upravleniya biznes-processami*. Available at: <http://journal.itmane.ru/node/235> (accessed 10.11.2018).
2. Sistemnyy integrator. Available at: <https://it-ping.ru/stati/sistemnyij-integrator> (accessed 10.11.2018).
3. Smart SOA: razumnyy podhod IT-lidera [Smart SOA: IT Leader Reasonable Approach]. *Innovacii v tekhnologiyah i biznese*, 2008, no. 4. Available at: https://www.osp.ru/data/893/418/1237/ibm_4_10_4 (accessed 10.11.2018).
4. Porte B. Obzor terminologii SOA. CH.1. Servis, arhitektura, upravlenie i biznes-terminy [SOA terminology. Part 1 Service, architecture, management, and business terms]. *IBM developer Works*. Available at: ibm.com/developerWorks/ru/ (accessed 10.11.2018).
5. Integraciya prilozheniy i servisov [Integration of applications and services. BCC Group]. Available at: <http://bcc.ru/> (accessed 10.11.2018).
6. Wolfson B.I. Gibkie metodologii razrabotki [Flexible development methodologies]. St. Petersburg, Piter Publ., 2017. 144 p.
7. Stellman A., Greene J. *Learning Agile: Understanding Scrum, XP, Lean, and Kanban*. O'Reilly Media, 2013. 420 p. (Russ ed.: Stelman A., Green J. Postigaya Agile, Cennosti, principy, metodologii. Moscow, Mann, Ivanov and Ferber Publ., 2018. 448 p.)
8. Selyuk A.V., Denisova S.S. *Upravlenie proektnoy komandnoy* [Management of project team]. Tyumen, TSU Publ., 2013. 216 p.
9. Dimov E.M., Maslov O.N., Pcheljakov S.N., Skvorcov A.B. *Novye informatsionnye tehnologii: podgotovka kadrov i obuchenie personala. Ch. 2. Imitatsionnoe modelirovanie i upravlenie biznes-processami v*

- infokommunikacijah* [New information technologies: personnel training. P.2. Simulation modelling and management of business processes in infocommunications]. Samara, SNC RAN Publ., 2008, 350 p.
10. Anufriev D.P., Dimov E.M., Maslov O.N., Troshin Y.V. Statisticheskoe imitacionnoe modelirovanie i upravlenie biznes-processami v social'no-ehkonomicheskikh sistemah [Statistical simulation modeling and management of business processes in socio-economic systems]. Astrahan, AstrISMI Publ. 2015. 365 p.
11. Germeier Yu.B. *Igry s neprotivopolozhnyimi interesami (teoriya prinyatiya resheniy pri nepolnom edinstve)* [Games with opposing interests (decision theory with incomplete unity)]. Moscow, Moscow State University Publ., 1972. 256 p.
12. Anufriev D.P., Dimov E.M., Maslov O.N., Halimov R.R. Sravnitel'naya effektivnost' metodov i sredstv informacionnoy podderzhki upravlencheskih resheniy [Comparative effectiveness of methods and means of information support for management decisions]. *Infokommunikacionnye tehnologii*, 2014, vol. 12, no. 1, pp. 54-67.

Received 15.11.2018

УДК 658.5

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ БИЗНЕС-ПРОЦЕССА «ПОИСК И ВЫБОР ПОСТАВЩИКА ПРОДУКЦИИ»

Богданова Е.А., Михаленко Ю.А.

*Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ
E-mail: helen.bogdan@mail.ru, mihalenko97@mail.ru*

В статье приведено обоснование применения статистического имитационного моделирования как одного из методов обработки больших объемов данных, что позволяет существенно повысить эффективность принимаемых решений в различных сферах деятельности. Рассматриваются вопросы, связанные с разработкой статистической имитационной модели бизнес-процесса поиска и выбора поставщика продукции на примере организации по производству молочной продукции. Приведена блок-схема бизнес-процесса, его детальное описание, а также проведен анализ рассматриваемого бизнес-процесса в интересах имитационного моделирования, в ходе которого выделены наиболее значимые для моделирования данного процесса случайные величины. Для создания модели были определены законы распределения всех выделенных случайных величин, разработан моделирующий алгоритм и план эксперимента с моделью. Приведены результаты эксперимента, проведенного на имитационной модели.

Ключевые слова: статистическое имитационное моделирование, цифровая экономика, бизнес-процесс, случайные величины, алгоритм моделирования

Введение

В современных реалиях мировая экономика, как в целом и все социально-экономические системы (СЭС), постоянно подвергаются изменениям и различного рода трансформациям под влиянием ряда факторов: от природных катаклизмов, появления фундаментальных разработок и технологий и вплоть до разного рода кризисов. В условиях непрерывного развития перед каждой организацией на первый план становится вопрос грамотного и эффективного управления предприятием, что представляет собой достаточно трудный и кропотливый процесс, требующий определенной компетентности в области рационального сочетания разнообразных методов управления.

Лицо, принимающее решение (ЛПР), должно руководствоваться достоверной и полной ин-

формацией для выбора оптимального пути развития организации. Для этого ЛПР необходимо проанализировать огромный объем данных. И в этой связи данные являются основой экономического анализа, исследующего закономерности функционирования современных СЭС. Как утверждает ряд экспертов, в настоящее время для экономического агента становится важным не сам факт обладания каким-либо ресурсом, а наличие данных об этом ресурсе и возможность их использовать с целью планирования своей деятельности [1].

Исходя из этого, под «цифровой экономикой» следует понимать современный тип хозяйствования, в котором ключевым фактором производства являются данные в цифровом виде и методов управления ими как определяющий ресурсы в сфере производства, распределения, обмена и потребления. Обработка больших объемов дан-