

3.5. Развертывание и внедрение

Фазы жизненного цикла информационных систем

Управление жизненным циклом информационных систем

3.5. Развертывание и внедрение

- ▶ Закупка и настройка требуемой ИТ-инфраструктуры
- ▶ Ввод начальных остатков
- ▶ Обучение пользователей
- ▶ Развертывание системы на рабочих местах
- ▶ Основные виды тестирования
- ▶ Опытно-промышленная эксплуатация
- ▶ Приемочные испытания и интеграционное тестирование

Этап развертывания системы является одним из наиболее важных с точки зрения распределения ответственности за работы между заказчиком и подрядчиком. Исполнитель выполняет основную задачу — установку модуля на рабочий сервер с перенесением на него итоговой конфигурации с тестового сервера. Однако при верном последовательном обучении и вовлечении специалистов заказчика в процесс создания и внедрения системы остальные операции могут быть осуществлены ими:

- предшествующее работам обеспечение охвата всех автоматизируемых рабочих мест пользователей функционирующей ЛВС;
- настройка рабочего сервера;
- определение перечня и количества рабочих мест, которые необходимо развернуть для ОПЭ;
- развертывание рабочих мест пользователей с определением прав доступа;
- подготовка наиболее актуальной информации по каждому модулю для ввода в систему (включая подготовку, дополнение и выверку справочников, которые могли потерять актуальность за время между тестовой эксплуатацией и развертыванием системы);
- организация поддержания данных системы в актуальном состоянии (в том числе при интеграции с другими системами).

Целью данного этапа является подготовка модуля системы к опытно-промышленной эксплуатации — подготовка конечных пользователей, ввод начальных данных, подготовка приказов по предприятию заказчика о передаче модуля системы в опытно-промышленную эксплуатацию.

11

На этом этапе нужно:

- обеспечить безусловное выполнение условий готовности модулей системы к сдаче в опытно-промышленную эксплуатацию (закрепленных в отдельном документе);
- ввести и выверить начальные данные (например, начальные остатки) по каждому модулю для ввода информации в систему;
- отработать на рабочих местах пользователей модуля системы практические действия пользователей в параллельном режиме с имеющимися приложениями;
- разработать, согласовать с подрядчиком и утвердить регламент взаимодействия подразделений заказчика, участвующих в опытно-промышленной эксплуатации модуля системы;
- осуществить интеграцию модуля с другими модулями или внешними системами, внедренными ранее;

- подготовить и издать приказ по предприятию заказчика о передаче модуля системы в опытно-промышленную эксплуатацию, который должен содержать:

- наименование модуля системы, проходящего опытно-промышленную эксплуатацию;
- наименование компании-исполнителя;
- сроки проведения опытно-промышленной эксплуатации;
- список должностных лиц со стороны заказчика и исполнителя, ответственных за проведение опытно-промышленной эксплуатации модуля;
- перечень подразделений предприятия заказчика, участвующих в проведении опытно-промышленной эксплуатации;
- порядок и сроки перевода персонала заказчика на работу в условиях функционирования модуля системы.

К моменту ОПЭ уже должны быть проведены следующие работы:

- модули системы успешно перенесены и функционируют на рабочем сервере и автоматизированных рабочих местах пользователей (с разделением прав доступа);
- подготовлены, дополнены и введены недостающие справочники по каждому модулю системы, проведена выверка введенных в систему недостающих справочников;
- заказчиком утверждены итоговые документы.

3.5.1. Закупка и настройка требуемой ИТ-инфраструктуры

Важно учитывать, что для всех предприятий чрезвычайно высока ценность хранимой и обрабатываемой информации, а также стоимость времени простоя бизнеса. Поэтому к техническому обеспечению ИТ-систем компании должны предъявляться жесткие требования:

- по производительности — обеспечение приемлемого времени реакции с точки зрения пользователя, использующего данную систему;
- надежности и доступности — пользователи должны продолжать работу при выходе из строя единичных устройств, система не должна простаивать и не должно быть потерь данных;
- катастрофоустойчивости — простой системы должен быть минимальным, а все данные должны быть сохранены; допускается некоторое ухудшение производительности и увеличение времени реакции системы;
- безопасности — система работает с данными, составляющими коммерческую тайну, и должна обеспечивать должный уровень защиты, вплоть до аттестации и сертификации решений и оборудования;

- масштабируемости — система должна иметь возможность приспособиваться к увеличивающейся нагрузке со сравнительно малыми архитектурными изменениями или совсем без них.

Все эти требования к характеристикам решения в процессе подготовки ИТ-инфраструктуры детализируются на гораздо более подробном уровне. Ниже мы рассмотрим пример описания конкретных действий по обеспечению масштабируемости решения.

Пример

Основной объем обработки проводится при помощи рабочих процессов сервера приложений (диалог, фон, обновление, очередь или подкачка данных). Эти процессы запускаются на инсталляциях сервера приложений (первичная и дополнительная инсталляции дополнительного сервера приложений, за исключением процессов постановки в очередь, которые запускаются на инсталляции центральных сервисов). Вся обработка, связанная с базой данных, контролируется инсталляцией базы данных. Это значит, что критически важно обеспечить высокую масштабируемость инсталляций сервера приложений и базы данных. Масштабируемость этих инсталляций обеспечивается следующим образом.

- Сервер приложений масштабируется по горизонтали путем добавления дополнительных серверных хост-узлов, на которых выполняются дополнительные инсталляции приложений. Отправка запросов на инсталляции сервера приложений выполняется при помощи сервиса отправки сообщений, работающего на инсталляции центральных сервисов.
- Инсталляция базы данных масштабируется по вертикали путем адаптации необходимых аппаратных ресурсов (увеличения мощности процессора, расширение памяти и т.д.).

К ИТ-инфраструктуре в таком случае можно отнести следующие компоненты.

Вычислительная инфраструктура:

- базовые инфраструктурные сервисы. Описание и процесс организации таких инфраструктурных сервисов, как IP-телефония, электронная почта, резервное копирование, антивирусная защита и антиспам, служба каталогов AD, служба удаленных рабочих столов, сетевая печать и пр. Данные приложения автоматизируют вспомогательные задачи конечных пользователей, поэтому к этому классу в первую очередь относятся приложения, связанные с пользовательскими коммуникациями;
- серверное и пользовательское оборудование, системы хранения данных (СХД). Определение перечня пользовательского и серверного оборудования и систем хранения данных;
- серверные площадки и центры обработки данных (ЦОД). Планирование типовых центров обработки данных и серверных комнат, вплоть до составления некоей усредненной модели серверной комнаты/ЦОД;
- системное ПО. К этой категории относятся описания операционных систем и программных платформ как для серверов, так и для оборудования, доступного конечному пользователю. Помимо этого, в данном разделе рассматриваются службы виртуализации.

Сетевая инфраструктура:

- инфраструктура локальных вычислительных сетей (ЛВС). Данный компонент должен представлять собой типовое решение по организации внутренней локальной вычислительной сети. Именно этот элемент является ключевым для обеспечения связи между оборудованием в офисах, филиалах, ЦОДах и других типах помещений. Важно, что в случае производственной или любой другой компании со значительной долей технологических систем проводится физическое и логическое разделение локальной сети на два сегмента, каждый из которых эксплуатирует только собственное сетевое оборудование. Таким образом, в корпоративной сети работают все бизнес-пользователи, а в технологической располагаются основные сервисы (оборудование, пользователи), поддерживающие технологическую составляющую и процессы компании;
- корпоративные сети передачи данных (КСПД);
- телефония, ВКС, подключение к сети Интернет.

Инженерная инфраструктура: устройства бесперебойного питания, электропитания, охлаждения, кабельная инфраструктура.

Можно упомянуть, что среди современных тенденций построения ИТ-инфраструктуры — использование нескольких уровней абстракции для вычислительных комплексов, сетей передачи данных с целью повышения гибкости и обеспечения регулирования потребления ресурсов (что снижает в конечном итоге суммарную стоимость оборудования).

Действия на этапе построения необходимой для системы ИТ-инфраструктуры складываются из следующего.

1. Закупка оборудования — данный шаг обычно значительно увеличивает время подготовки технической платформы, так как закупка может быть связана с проведением конкурсов, заключением договоров, ожиданием поставки оборудования. Все эти шаги могут приводить к потерям времени, от нескольких недель до нескольких месяцев.

2. Подготовка инфраструктуры — расчет возможности установки оборудования в серверное помещение, выделение и конфигурирование сетевых портов, планирование мощностей компонентов инженерной инфраструктуры серверных и пр.

3. Аппаратное конфигурирование — получение оборудования, его конфигурирование и установка (в том числе тестовый сервер, рабочие станции, вспомогательное оборудование).

4. Установка системного программного обеспечения — система управления базами данных, операционная система, генераторы отчетов (либо установка базового программного обеспечения, реализующего слой виртуализации).

5. Конфигурирование пользовательских рабочих мест — рекомендуется определить несколько стандартных конфигураций пользовательских рабочих мест, которые позволяли бы обеспечивать приемлемое качество работы пользователей. Это дает возможность не только упростить поддержку клиентских рабочих мест, но и, например, настроить систему мониторинга для проверок ключевых параметров работы приложений на всех конфигурациях пользовательских рабочих мест, используемых в компании.

В результате команда внедрения со стороны подрядчика получает развернутую функционирующую стендовую локальную вычислительную сеть, удовлетворяющую требованиям ТЗ, и программно-аппаратную платформу, уже готовую к последующей установке системы.

3.5.2. Ввод начальных остатков

Данный этап, актуальный для систем финансового управления, является логическим продолжением проведенной ранее миграции данных (со вводом справочников и первичной информации). Формирование начальных остатков проводится уже после настройки параметров приложения. На их основе составляется начальный баланс, с помощью которого проверяется, насколько корректно были введены данные. В зависимости от системы при вводе и выверке данных может проводиться либо редактирование сальдо счета (субсчета), либо формирование фиктивных проводок. Делаются записи по дополнительным регистрам. При этом проводится тщательный контроль реакции программы на все операции, проверки сбалансированности сальдо, проверки каждого счета (в том числе аналитических).

3.5.3. Обучение пользователей

Для минимизации сопротивления, которое может встречаться со стороны специалистов организации при внедрении «очередной» системы, обучение проводится в три основных этапа.

1. Выделение и обучение ключевых пользователей.
2. Обучение рядовых пользователей.
3. Повышение квалификации или переобучение.

Рассмотрим эти этапы подробнее. В первую очередь необходимо объяснить принципы работы уже настроенной системы перед первым тестированием с ключевыми пользователями.

Следующая веха, инсталляция на стендовый сервер, важна как промежуточный этап, на котором снова будут привлекаться ключевые пользователи, только уже для описания работы и интерфейсов в гораздо более детализированном виде, для идентификации и устранения всех остающихся критических замечаний перед ОПЭ. Разумеется, к этому моменту все ИТ-специалисты, которые будут заниматься поддержкой системы, уже должны в деталях знать внутреннюю архитектуру системы и быть способными проводить ее доработку (или формировать требования на доработку) в случае необходимости.

И наконец, наиболее масштабное обучение организуется уже для всех остальных сотрудников, которые будут работать с системой. В данном случае система представляется не с точки зрения внутренней архитектуры, а с точки зрения пользователя, т.е. описываются принципы ввода, обработки и получения данных.

Для организации процесса обучения необходимо выполнить следующие шаги.

Шаг 1. Выделение и обучение ключевых пользователей. В первую очередь выделяются ключевые пользователи — один-два специалиста от каж-

дого структурного подразделения (деление организации может выполняться не с организационной, а с логической точки зрения).

Важно, что в зависимости от масштаба и специфики системы эти специалисты могут работать не только с пользовательскими аспектами системы, но и самостоятельно разворачивать функциональные компоненты и модули, конфигурировать их по требованиям пользователей. Именно поэтому их обучение, аттестацию и вовлечение в работу крайне важно проводить гораздо раньше ОПЭ, ведь основное число сотрудников увидят систему и ознакомятся с принципами работы в ней лишь во время опытно-промышленной и промышленной эксплуатации.

Проведение учебных курсов по основам работы с системой для сформированной группы ключевых пользователей (по функциональным направлениям) организуется специалистами исполнителя (включая аттестацию). В качестве формы организации обучения может быть выбрана очная (преподавателем от команды исполнителя в офисе компании заказчика или подрядчика) либо дистанционная (когда личное присутствие преподавателя нецелесообразно или невозможно). Все популярнее становится проведение вебинаров и телеконференций, позволяющих экономить временные и денежные ресурсы.

Их поддержка будет крайне важна на следующем этапе при проведении тренингов для остальных пользователей: помимо предоставления комментариев по пользованию системой и о необходимости доработок, они могут помочь остальным пользователям разобраться с работой в системе и не обращаться каждый раз к команде сопровождения со стороны подрядчика.

Шаг 2. Обучение рядовых пользователей. К этому моменту необходимо, чтобы достаточное количество специалистов группы внедрения от заказчика было способно самостоятельно разворачивать функциональные компоненты системы или модуля на рабочих местах пользователей, проводить обучение конечных пользователей, оказывать консультации по стандартной конфигурации системы, а также сопровождать ее в оперативном режиме.

Принципы проведения тренингов могут значительно различаться в зависимости от масштабов компании, проекта/системы и пожеланий заказчика, однако на практике они чаще всего проводятся несколько раз, не более чем для 10–15 человек одновременно. При этом группы обычно формируются под каждую отдельную роль пользователя. Основные темы включают общий обзор задач системы, рассмотрение ее интерфейса, основных возможностей, затем специфической для конкретной роли пользователя функциональности и секции «вопрос – ответ». Иногда после этого проводится настройка необходимых параметров (например, интеграции с учетными записями Outlook, настройка синхронизации) уже на местах пользователей, однако необходимость этого зависит от конкретной системы.

Шаг 3. Повышение квалификации и переобучение. Следует учитывать также необходимость последующего периодического обучения с целью повышения квалификации сотрудников.

Условно можно выделить три основные категории такого обучения.

1. Плановое обучение после изменений в системе (при добавлении дополнительных возможностей, внедрении нового модуля и т.п.).
2. Плановое повышение квалификации: для совершенствования знаний работников, уже обладающих определенными профессиональными навыками работы с системой.
3. Обучение по запросу — относится к сфере поддержки пользователей на этапе эксплуатации системы и проводится в случае, если пользователь в своей работе столкнулся с конкретными проблемами или нуждается в проведении методического занятия.

Отметим, что для работы с некоторыми классами систем (как правило, если их деятельность связана с управлением объектами, представляющими в случае выхода из-под контроля опасность для жизни и здоровья людей) просто проведения обучения недостаточно. В таких случаях организуются дополнительные тестирования знаний, успешное прохождение которых дает допуск к пользованию системой при предоставлении персонального сертификата. Примером могут являться системы, автоматизирующие управление безопасностью железнодорожного движения и авиаперевозок, запуск космических объектов, химическое производство и подобные виды деятельности.

И разумеется, при планировании обучения следует учитывать такие аспекты, как опыт сотрудников, предполагаемые и видимые знания, а также навыки руководства и управления, организационные способности.

3.5.4. Развертывание системы на рабочих местах

Следующим шагом организуется развертывание модуля для дальнейшей его опытно-промышленной эксплуатации. Успех данного этапа целиком зависит от проведенной ранее работы. Так, если уже был реализован пилотный проект в другом подразделении, филиале или в рамках другой функциональной области, этот опыт поможет компании воспользоваться рекомендациями по развертыванию системы и выполнить оптимальные настройки. Кроме того, инсталляция проводится еще до тестирования, а значит, результаты используются исключительно в целях совершенствования системы в так называемой песочнице (*sandbox*).

«Песочница» (*sandbox*) — закрытое для доступа извне виртуальное пространство, в котором можно работать с ПО без риска изменения системных файлов, в связи с чем они используются для запуска кода, еще не прошедшего тестирование.

Настройки «песочницы» и ее элементы дублируют основные элементы и возможности реальной системы для первоначальной отладки и проверки кода в виртуальной среде перед его переносом в продукт. Это помогает избежать значительного числа рисков и сбоя настроек сложной конфигурации системы.

Набор ресурсов системы, выделяемый для «песочницы», жестко ограничен в части доступа к объему памяти, к сети, обмену информацией с основной системой. Это обусловлено в первую очередь нестабильностью работы тестируемого кода и значительной нагрузкой на систему. По этой причине «песочницы» часто используют исключительно для проверки отдельных критических частей кода.

Сделанные в «песочнице» настройки используются при полноценном развертывании модуля системы после тестирования перед ОПЭ. Модули системы устанавливаются на стендовый сервер с проведением основных необходимых настроек под подлежащие автоматизации бизнес-процессы предприятия заказчика. По результатам этих действий подрядчик подготавливает описание оптимальных настроек сервера БД и рекомендации по развертыванию рабочих мест, которое и проводится следующим шагом.

Перед развертыванием системы необходимо формирование командой внедрения следующей информации:

- выделение основных блоков функциональности, их запусков, взаимозависимостей;
- выделение основных блоков работ, требуемых для запуска функциональности системы, и их взаимозависимости;
- расстановка контрольных точек для верхнеуровневого мониторинга процесса;
- информация по исполнителям работ (зонам ответственности).

При переходе к стадии эксплуатации должны быть осуществлены следующие действия:

- проверка необходимых предпосылок для перехода в продуктивную систему;
- описание мероприятий по переходу в продуктивную систему;
- описание критериев, на основании которых будет приниматься решение о начале работы пользователей;
- описание порядка действий в случае возникновения ошибок, сбоев или нарушения работы системы.

При развертывании системы на рабочих местах выполняется следующее:

- установка и подключение технических средств;
- установка и конфигурация ПО;
- развертывание баз данных, служебных программ;
- доработка интерфейса и процессов, настройка профилей пользователей, настройка отчетов;
- установка прав доступа;
- пуско-наладочные работы и окончательная отладка конфигурации.

Потенциальные сложности могут возникнуть также, если несколько смежных подпроектов (например, в части разных модулей системы) реализуются несколькими подрядчиками. Тогда следует уделять пристальное внимание управлению интеграцией всех проектов на протяжении всего времени их реализации и при подготовке к ОПЭ – в особенности.

3.5.5. Основные виды тестирования

Прежде чем приступить к промышленной эксплуатации системы, важно проверить ее работоспособность во всех предусмотренных спецификацией режимах, а также в экстремальных условиях. Если планируется одновременный распределенный ввод данных сотней пользователей, необходимо, как минимум, проверить правильность обработки системой одновременной активности именно не менее сотни пользователей. Подобная идея лежит в основе нагрузочного тестирования.

Нагрузочное тестирование необходимо для предсказания поведения системы в реальных и экстремальных условиях, выявления ошибок, отслеживания производительности и доступности при изменении различных параметров работы с системой. Среди других его задач:

- оценка производительности и работоспособности приложения на этапах:
 - разработки,
 - опытно-промышленной эксплуатации,
 - сопровождения и эксплуатации (выпуск новых релизов, патчей);
- оптимизация приложений;
- подбор оптимальной программно-аппаратной платформы и конфигурации сервера.

Перечислим еще несколько ключевых категорий тестирования.

Тестирование производительности (*load testing*) – моделирование ожидаемой интенсивности использования путем распределенной работы большого числа пользователей с различными модулями системы.

Среди возможных целей этого типа тестирования:

- определение времени выполнения операций с заданной интенсивностью (на разных нагрузках);
- определение максимально достижимой производительности системы.

Стрессовое тестирование (*stress testing*) – определение стабильности системы при интенсивности работы, превышающей плановые или стандартные значения. Стрессовое тестирование по своей сути проверяет, возвращается ли (и насколько быстро) система после запредельной нагрузки к нормальному режиму работы, тестируется способность системы к регенерации.

Стрессовое тестирование особенно важно для компаний, в которых стоимость отказа системы в экстремальных условиях может быть очень велика, а стандартного тестирования разработчиками недостаточно для эмуляции тех условий, в которых произойдет отказ системы.

Среди возможных целей этого типа тестирования:

- оценка динамики производительности при нестандартных ситуациях: аварийном изменении серверной конфигурации либо резком повышении числа пользователей и выполняемых одновременно операций;
- определение условий и скорости возвращения системы к нормальному режиму работы после окончания стрессового тестирования.

Тестирование надежности (*reliability testing*) — определение длительности бесперебойной и безошибочной работы системы. Цели этого типа тестирования следующие:

- оценка стабильности системы при многочасовом тестировании со стандартным средним уровнем нагрузки для определения:
 - утечек памяти,
 - некорректных конфигурационных настроек,
 - случаев перезагрузки сервера и других требующих устранения проблем.

Конфигурационное тестирование (*configuration testing*) — оценка степени влияния на производительность изменений в конфигурации и различной балансировки нагрузок. Конфигурационное тестирование относится как к серверному уровню (совместимость с окружением), так и к клиентскому уровню (например, кросс-платформенное или кросс-браузерное тестирование). Цели этого типа тестирования:

- оценка обработки ошибок и исключительных ситуаций, а также «узких мест» отдельных модулей и компонентов системы при непропорциональных нагрузках;
- определение оптимальной конфигурации оборудования, которая бы обеспечивала требуемые характеристики производительности и отклика системы;
- проверка определенных компонентов системы на предмет совместимости с указанным в спецификации оборудованием, операционными системами и программными продуктами внешних поставщиков.

Последовательности действий для проведения данного вида тестирования такова.

1. Создается матрица покрытия с описанием всех возможных конфигураций системы.
2. Выполняется приоритезация конфигураций.
3. И наконец, в ходе тестирования в соответствии с имеющимися приоритетами организуется проверка всех основных конфигураций.

Объемное тестирование (*volume testing*) — тестирование программного обеспечения системы на предмет стабильности обработки определенного объема данных. Цели этого типа тестирования:

- измерение динамики выполнения операций при увеличении объемов данных в базе данных, постепенном росте числа запросов пользователей;
- определение максимального числа пользователей, которые могут работать с приложением без снижения производительности.

Отсутствие или ошибки проведения подобных видов тестирования значительно увеличивают риски и могут даже приводить к ситуациям, когда после внедрения срок эксплуатации систем не превышает даже года в связи с невозможностью конфигурации системы справляться с необходимыми задачами и нагрузками. Именно для нивелирования подобных рисков необходимо своевременно организовывать тестирование еще на этапе внедрения (хотя его проведение возможно (и часто проводится) и для уже эксплуатируемых систем).

3.5.6. Опытнo-промышленная эксплуатация

Опытнo-промышленная эксплуатация представляет собой тестирование в полной функциональности и полной нагрузке для определенного количества пользователей. Ее основной целью является апробирование работы пользователей в системе в реальных производственных условиях. Это означает, что если по спецификациям системы предполагается, что она будет

обрабатывать 500 000 записей в день, необходимо проверить корректность обработки именно этого количества записей. При этом важные задачи:

- тестирование модуля системы в условиях, максимально приближенных к реальным условиям промышленной эксплуатации (в том числе, при необходимости, в интеграции с другими модулями или внешними системами) — нагрузочное тестирование, рассмотренное ранее;
- проведение множественных расчетов по фактической производственной информации с применением соответствующих программно-аппаратных средств, предусмотренных техническим проектом (проектным решением);
- достижение наиболее полного охвата бизнес-процессов, автоматизируемых подразделений предприятия.

При условии соблюдения ранее определенных требований к предыдущим этапам и их успешном завершении в первую очередь необходимо провести:

- ввод наиболее актуальной информации по каждому модулю ввода в систему (подготовка которой проведена на предыдущем этапе);
- отработку действий пользователя в параллельном режиме с уже имеющимися приложениями (это проводится непосредственно на рабочих местах пользователей и включает дополнительную, но необходимую нагрузку для пользователей, которые в короткий период времени должны адаптироваться к работе в другой системе (а также помочь адаптировать ее для оптимальной и эффективной работы));
- интеграцию внедряемого модуля с уже существующими модулями и (или) внешними системами, внедренными ранее;
- разработку и согласование регламента взаимодействия внутренних подразделений предприятия-заказчика в рамках ОПЭ и дальнейшей промышленной эксплуатации (помимо схемы и порядка передачи функций с обязательным определением зон ответственности за актуальность и корректность данных).

Пример

Несмотря на то что официально «владельцем» системы будет являться одно подразделение (или даже одно ответственное лицо), ввод и редактирование информации будут осуществляться распределенно и часто одновременно различными категориями пользователей. Соответственно, существует практика определения зон ответственности за различными подразделениями, которые вне зависимости от источника ввода будут обладать исключительными правами на удаление и изменение информации. Этот механизм может быть организован многими способами, самым «простым» из которых является четкое определение алгоритма взаимодействия подразделений в рамках работы с системой (сотрудники будут сами знать, какую информацию в какой момент они будут вводить и обрабатывать). Во многом подобный механизм исходит из принципов «презумпции невиновности», предполагая что каждый специалист достаточно компетентен и отвечает за каждое совершенное им действие (как в целом в работе, так и в системе).

Другим, менее распространенным, способом управления взаимодействием подразделений (и главное — контроля данных) является распределение прав доступа таким образом, что у наиболее важной информации будет только один источник ввода/проверки. И хотя другие пользователи могут изменять поле (например, вно-

сить данные об инвестициях и договорах), в системе эта информация будет «висеть» в режиме ожидания до момента ее одобрения/авторизации действия ответственным за поле пользователем. Однако, несмотря на значительную гарантию достоверности данных, подобная схема крайне неэффективна с точки зрения использования ресурсов, особенно в условиях большого числа и масштаба операций.

Основная часть (как по объему, так и по трудозатратам) проводимых в ОПЭ мероприятий предусматривает сопровождение заказчиком системы на рабочих местах пользователей с формированием перечня замечаний ключевых пользователей (при их «переводе» из бизнес-терминологии в более техническую область) и устранением этих замечаний.

Пример

Замечания пользователей:

В список партнеров нельзя добавить университеты (нет пункта в списке).

Требование к реализации подрядчиком:

Создать в типе организаций «Прочие» наследующий его свойства подтип «Университет» с дополнительными полями «Ректор», «Факультет», «Кафедра». Настроить общедоступный отчет по всем университетам.

Устранением замечаний и внесением изменений в систему на этом этапе (длящемся, как и в случае тестовой эксплуатации, не менее одного месяца) занимается команда внедрения со стороны исполнителя, которая помимо простого соответствия системы всем требованиям и бизнес-процессам должна удостовериться в устойчивости работы системы при нагрузке, максимально приближенной к реальной промышленной нагрузке этапа эксплуатации (с исключением риска отказа системы в момент ее пика). Чтобы перейти к полноценной промышленной эксплуатации, система также должна безошибочно функционировать в параллельном режиме и при интеграции с внешними приложениями в течение всего срока ОПЭ. При наличии аналогичных *legacy*-приложений об успешном внедрении можно говорить, если результаты обработки данных в них и в новой системе совпадают и (или) различия объяснимы и некритичны.

Для окончания ОПЭ необходимо подтверждение того, что модуль системы функционирует в параллельном режиме с существующими внешними приложениями (при их наличии) в течение срока опытно-промышленной эксплуатации. Вторым условием является подтверждение, что любые отклонения (в случае их обнаружения) признаны некритичными и объяснимыми.

3.5.7. Приемо-сдаточные испытания и интеграционное тестирование

Именно передача системы из ОПЭ в промышленную эксплуатацию (в частности, путем проведения приемо-сдаточных испытаний) является целью комплексного проекта по автоматизации деятельности компании. Приемо-сдаточные испытания такого типа называют также интеграционным тестированием. Такое тестирование обеспечивает комплексную проверку реализации проектных решений системы.

Цели интеграционного тестирования:

- проверка корректности функционирования бизнес-процессов и функций;
- уточнение настроек бизнес-процессов в системе;
- уточнение и доработка настроек пользовательского интерфейса;
- проверка функций начальной загрузки в систему основных и переменных данных;
 - проверка полноты переносов настроек посредством транспортной системы;
 - проверка работоспособности интерфейсов с внешними системами;
 - проверка корректности проектной и эксплуатационной документации;
 - определение правильности функционирования системы на реальном объеме данных в реальном времени.

Тестирование проводится на продуктивном наборе данных для проверки работоспособности всей системы и правильности настройки интерфейсов взаимодействия с внешними системами.

Как уже было сказано, именно этот этап является переходным между этапом ОПЭ и реальной эксплуатацией. С организационной точки зрения перед началом промышленной эксплуатации необходимо удостовериться в наличии следующего набора документов:

- рабочий журнал опытно-промышленной эксплуатации с перечнем замечаний ключевых пользователей и результатами их устранения (заполняемый командой внедрения и согласовываемый затем с представителями заказчика);
- протокол об окончании ОПЭ и готовности к промышленной эксплуатации совместно с Актом приемки-передачи работ;
- приказ по предприятию заказчика о приеме системы в промышленную эксплуатацию (с составом функций, описанием программно-аппаратной платформы, списком ответственных за работу модуля и списком подразделений-пользователей, а также с порядком и сроками перехода персонала на работу в системе, включая аспект разработки и принятия новых форм документов в случае необходимости).

После этого использование системы уже не имеет признаков проектной деятельности и ограничений по срокам, содержанию или стоимости. Однако такие последующие этапы ЖЦИС, как сопровождение эксплуатации, модернизация и утилизация системы, будут по своей сути являться проектами и могут также выполняться не только за счет внутренних ресурсов компании, но и (что случается гораздо чаще) с привлечением внешних подрядчиков.