

КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

corpinfosys.ru
Научно-популярный сетевой журнал

Першин Д.С.
Стартапы

Прядильников Е.В.
Стратегия реализации в проектах имплементации корпоративных
информационных систем

Степанов Д.Ю.
Стратегия управления изменениями в проектах имплементации
ERP/ERP2-систем

Терентьев И.М.
Стратегия анализа в проектах внедрения ERP-систем

Петров С.В.
Использования SQL-запросов и команд языков программирования
для разработки ERP-систем

Выпуск №2
2018 год

Научно-популярный сетевой журнал
«Корпоративные информационные системы»
(Corporate Information Systems)

Рецензируемое научно-популярное сетевое издание corpinfosys.ru, публикует оригинальные авторские статьи по актуальным вопросам корпоративных информационных систем и технологий. Свидетельство регистрации СМИ ЭЛ № ФС 77-71053 от 13.09.2017. Периодичность издания - четыре выпуска в год. Издаётся с марта 2018 года.

Миссия журнала - объединение теории с практикой анализа, проектирования, разработки, тестирования и внедрения корпоративных информационных систем из-за образовавшегося разрыва между двумя важнейшими составляющими проекта. Тематика журнала не ограничена рассмотрением лишь одного класса систем, обсуждаются всевозможные информационные, экономические, биотехнические и кибернетические системы, действующие в масштабе корпорации.

Принимаются к публикации работы не только научных сотрудников, специализирующихся по тематике журнала, но и руководителей проектов, консультантов, разработчиков, ключевых и конечных пользователей, непосредственно участвующих в процессе реализации корпоративных информационных систем. Достоверность, актуальность и новизна являются обязательными атрибутами публикуемых статей.

Важнейшими принципами издания являются доступность и открытость. Регламентированная и быстрая процедура рецензирования, а также электронный формат печати статей обеспечивают реализацию принципа доступности. Использование механизмов потока работ и операций в процессе обработки заявок на публикацию и возможность отслеживания статуса публикации задают принцип открытости. Рубрики журнала:

- теоретические основы корпоративных информационных систем;
- функционал корпоративных информационных систем;
- обзор современных корпоративных информационных систем;
- особенности реализации корпоративных информационных систем;
- интеграция корпоративных информационных систем;
- управление проектом внедрения корпоративных информационных систем;
- опыт внедрения корпоративных информационных систем.

Здравствуйте, уважаемые читатели!

Если вы знакомитесь с этим материалом, значит тематика корпоративных информационных систем Вам так же интересна, как авторам статей и членам редакционной комиссии.

Второй выпуск журнала содержит подборку статей касающихся стартапов в IT-индустрии, стратегии реализации в проектах имплементации корпоративных информационных систем, стратегии управления изменениями в проектах имплементации ERP/ERP2-систем, стратегии анализа в проектах внедрения ERP-систем, использования SQL-запросов и команд языков программирования для разработки ERP-систем.

Целью нашего журнала являются актуальность, доступность и открытость информации по тематике корпоративных информационных систем. Выражаю благодарность всем тем, кто способствует и помогает в подготовке выхода очередного выпуска.

Желаю Вам успешных проектов внедрения корпоративных информационных систем и наличия времени для обмена опытом на страницах нашего издания.

С наилучшими пожеланиями,
главный редактор журнала,
к.т.н., доц. РТУ МИРЭА
Степанов Дмитрий Юрьевич



Теоретические основы корпоративных информационных систем

Першин Д.С.
Стартапы.....1

Прядильников Е.В.
Стратегия реализации в проектах имплементации корпоративных информационных систем.....10

Степанов Д.Ю.
Стратегия управления изменениями в проектах имплементации ERP/ERP2-систем.....17

Терентьев И.М.
Стратегия анализа в проектах внедрения ERP-систем.....23

Функционал корпоративных информационных систем

Петров С.В.
Использования SQL-запросов и команд языков программирования для разработки ERP-систем.....30

Стартапы

Першин Дмитрий Сергеевич

Аннотация: в статье в ознакомительном формате дана характеристика понятия стартап и его видов. Рассмотрены этапы развития стартапов и источники их финансирования. Вопрос «Стартапы в IT-индустрии» - это один из наиболее перспективных и доступных способов продвижения своих проектов и привлечения сторонних источников финансирования для молодых и амбициозных IT-компаний, представлен в статье в формате методики/рекомендаций по их становлению и развитию.

1. Стартап - понятие и виды

Стартап (от англ. startup startup, буквально - «стартующий») - компания с короткой историей операционной деятельности. Это коммерческий проект, основанный на какой-либо идее и требующий финансирования для развития. Основная цель любого стартапа - быстрое развитие и получение прибыли. В качестве основы деятельности используются новейшие технологии, новые бизнес-модели, которые ещё никем и нигде раньше не опробованы.

«Стартапами» сегодня обычно называют IT-компании, которые выходят на рынки с новыми технологиями или высокотехнологичными услугами/товарами, на которых пока нет установленных «правил игры» и «игроков», но есть ажиотажный спрос или интерес к оказываемой услуге или товару (так называемые «интернет-проекты»), или новое дело. Но это далеко не так.

Стартап - это не любой новый бизнес и не интернет-проект. Его суть в поиске воспроизводимой и обязательно масштабируемой бизнес-модели и источников прибыли, а также в развитии проекта с нуля. Обычные компании используют модели бизнеса, которые уже существуют, они просто приспособливают их под себя, заранее понимая, как будут зарабатывать деньги. А стартапы ищут новые модели [1].

Наиболее полное определение этого явления дал Стив Бланк, предприниматель и профессор Стэнфорда, Беркли и Имперского колледжа: «Временная организация, созданная для поиска бизнес-модели, которая является повторяемой и масштабируемой». Слово «start-up» фактически означает новый проект или новую компанию. Это первоначальная стадия развития любого технологического нововведения, которое существует только на уровне идеи. В дальнейшем эта идея

развивается, привлекаются инвесторы, и начинается стадия практического преобразования в реальный продукт.

Стартапы в настоящее время широко востребованы и используются в IT-технологиях (агрегаторы, приложения и порталы), довольно часто создаются в торговле (маркетплейсы, интернет-магазины и др.), финансовых и платежных технологиях, медицине, в образовании, сфере обслуживания, развлечений и досуга. В России стартапы наиболее распространены и используются в социальной сфере, сфере услуг, развлечений и досуга.

Основные характеристики стартапа:

- Вновь созданный коммерческий проект, связанный с продвижением новой высокотехнологичной идеи и новой бизнес-модели, направленный на получение высокой прибыли.
- Над созданием и развитием идеи, бизнес-модели работает небольшая группа молодых единомышленников-профессионалов (группа разработчиков-новаторов) всего в несколько человек (стартаперы).
- Основанная особенность проекта - его высокотехнологичность и новизна. Секреты и новый взгляд на уже известные технологии являются источником получения высокой прибыли от проекта.
- Задача стартапа - существенное упрощение и улучшение качества жизни человека за счет совершенствования конкретных его аспектов, чем и объясняется привлекательность проекта и повышенный спрос на результаты его внедрения.
- Возможность быстрого роста бизнес-модели проекта, ее масштабирования и повтора.
- Отсутствие финансовых ресурсов на начальном этапе для воплощения и развития коммерческого проекта.

Стартапы по технологичности и наукоемкости можно условно разделить на два направления:

- Startup на основе разработки и внедрения новых бизнес-моделей за счет создания новых технологий, технических инноваций, высокотехнологичных услуг/товаров. Это разработка и внедрение нового технологического решения, которое никогда и никем не применялось - инновационные стартапы. К ним можно отнести интернет-проекты, например, такие как социальная сеть «Facebook», онлайн-энциклопедия «Википедия» и др.

- Startup на базе разработки и внедрения новых бизнес-моделей путем внедрения инновационных идей и технических инноваций в отраслевые виды бизнеса (переработка идей): традиционные стартапы. Например, McDonald's стал типичным стартапом, создатели которого в свое время нашли инновационную идею - ресторанный франчайзинг, интернет-магазины Ozon, Amazon и др.

Стартапы по степени риска вложений и влияния на рынок товаров, работ и услуг можно разделить на три условных вида [2]:

- Startup качественные копии - наименее рискованная ниша использования стартап-проекта, где не нужно ничего придумывать, а просто делать полную копию и адаптировать под свой рынок. Все крупные соцсети рунета пришли к успеху через этот принцип;
- Startup агрессивные новички - модель, в которой на развитый, популярный рынок приходит молодой проект и начинает вытеснять старожил данного бизнеса. Зачастую это реализуется через демпинг - продажа услуг/товаров по заниженным ценам;
- Startup неожиданно сильные проекты - самая уникальная, но рисковая модификация, опережающая по доходности остальные виды стартапов. Инвесторы и разработчики этого проекта выигрывают позиции на рынке благодаря новаторским предложениям.

Главные составляющие успешного коммерческого стартап-проекта:

- высокая рентабельность идеи стартапа;
- уникальность бизнес-модели;
- актуальность и востребованность у потребителей;
- объективная оценка рисков;
- привлекательность для инвесторов.

2. Этапы развития стартапов и источники их финансирования

Каждый стартап в своем развитии от идеи до масштабирования и захвата значимой доли рынка проходит несколько этапов. К основным этапам, которые проходит стартап-проект, относят:

- Problem/Solution Fit - подтверждение проблемы и того, что предлагаемый продукт ее решит (формирование идеи);

- Minimum Viable Product (MVP) - разработка рабочего прототипа с минимальным функционалом для решения проблемы; выпуск альфа-версии и тестирование ее недочетов (становление проекта);
- Product/Market Fit - закрытая бета-версия продукта, подтверждение, что у продукта есть рынок сбыта (раннее развитие проекта);
- Scale - открытая бета-версия и выпуск потребительской версии продукта, масштабирование бизнес-модели (расширение проекта);
- Maturity - переход от стартапа к бизнесу (зрелость).

Перед запуском идеи в работу ее автор (группа авторов) должны прежде всего определиться с тем, что будет со стартапом после того, как будут привлечены инвестиции для воплощения идеи и завершения развития бизнес-модели. Exit Stage или выход стартапа - это заключительный этап развития. Есть несколько его вариантов завершения стартап-проекта:

- переход в форму традиционного бизнеса;
- продажа доли инвесторов;
- запуск на фондовом рынке.

Данная информация особенно важна для потенциальных инвесторов, чтобы понимать риски и рентабельность вложений. Инвесторы, получив прибыль от успешной реализации проекта, либо сохраняют свою долю в нем, чтобы иметь пассивный доход в будущем, либо продают ее. То, как будет делиться полученная прибыль, обсуждается заранее.

Таблица 1. Источники финансирования развития Startup проектов

№ п/п	Источник финансирования
1	Личные средства стартаперов, разработчиков проекта
2	Привлечение средств физических лиц (родственники, друзья, коллеги)
3	Кредиты банков
4	Частные инвесторы (меценаты и юридические лица)
5	Государственные средства, в том числе: <ul style="list-style-type: none"> ▪ субсидии на открытие бизнеса или государственная финансовая поддержка

№ п/п	Источник финансирования
	<ul style="list-style-type: none"> ▪ помощь структур, направленная на оказание юридической, организационной и других видов помощи (бизнес-инкубаторы, технопарки и др.) ▪ региональные программы поддержки предприятий малого бизнеса и стартап-проектов
6	Венчурные фонды
7	Краудфандинг
8	Гранты

3. Стартапы в IT-индустрии

Информационные технологии давно стали частью бизнеса и жизни людей, уже на наших глазах происходит глубокая цифровая трансформация. Внедряется новое поколение технологических решений, многие из которых связаны между собой: например, обработка «больших данных», использование искусственного интеллекта, распределенные вычисления и хранилища данных. Перспективные цифровые решения внедряет как вновь создаваемый бизнес (стартапы), так и бизнес с многолетней историей [3]. Стартап в любой сфере деятельности - это риск, в том числе и в IT-индустрии, где также нет никаких гарантий успешности проекта, ибо конкуренция в этой сфере не дает времени на то, чтобы расслабиться. Решая вопрос об организации успешного, по расчетам стартаперов, нового проекта в IT-сфере, уже как минимум нужно:

- владеть инновационной /новой технологической идеей;
- иметь представление о том, как вести проект, развивать его и конкурировать с себе подобными в данной области деятельности.

Базовая основа стартап-проекта для создания чего-то нового в IT - это активный спрос. Это значит, что продукт должен быть нужным и востребованным. Но фундаментом проекта всегда является новая технология или инновационная технологическая идея. Чем более она оригинальна и уникальна, тем успешнее будет ваше дело. Большинство стартапов проваливаются именно из-за неудачной

первоначальной идеи. Так происходит, когда команда придумывает ненужный продукт или не может сделать его привлекательным для рынка.

Для стартапов в IT-бизнесе сегодня характерны два основных момента:

- Положительный - политика современных инвесторов по отношению к IT-стартапам и их идеям положительная. Большинство инвесторов стараются вложить капитал по максимуму, боясь пропустить новый Apple или Google, которые в свое время завоевали первенство в IT-технологиях.
- Отрицательный - процент успешности IT-стартапов довольно низкий, хотя если он становится успешным, то приносит значительный доход.

Тем не менее, результаты коммерческих стартап-проектов по внедрению новейших информационно-технических технологий, в их числе и IT-технологий, сегодня являются привлекательными не только для пользователей, но и для инвесторов, обеспечивая их новыми возможностями эффективного вложения финансовых средств или способом максимально эффективного развития основного бизнеса [4].

Для молодых и амбициозных IT-компаний Startup - это один из наиболее перспективных и доступных способов продвижения своих проектов и привлечения сторонних источников финансирования для дальнейшего развития. Однако важно понимать, что даже гениальная идея - это еще не стартап. Пока она не начала привлекать аудиторию и приносить деньги - то инвестора, скорее всего, не убедить. Без работающего прототипа продукта идти к инвестору практически не имеет смысла. Инвесторы - люди, которые отвечают за свои вложения репутацией и карьерой. И они не видят смысла в таком огромном риске: вкладывать в проект и команду, которые еще не доказали свою состоятельность.

Начало продвижения своей идеи для доведения ее до уровня перспективного стартапа для новичка и молодой инициативной группы - это образовательные программы, бизнес-инкубаторы, преакселераторы, хакатоны, конкурсы, тематические выставки различного уровня и др. Они позволяют найти команду единомышленников, получить обратную связь, набраться опыта и сконвертировать «голую» идею в перспективный коммерческий проект. Также на уровне региональных программ развития малого и среднего бизнеса и продвижения стартапов предоставляются неплохие возможности начинающим стартаперам для получения первоначального опыта и поддержки в развитии идей. Все это является «начальной» школой, которая на старте будущего проекта может дать практический опыт в продвижении идеи,

возможность оценить силы свои и будущих конкурентов, посмотреть на себя «со стороны» и получить реальную оценку актуальности и востребованности проекта в сравнении с подобными.

Несколько советов молодым и начинающим стартаперам на основе рекомендаций опытных экспертов (В. Баймурзин) в части принятия решения о целесообразности начала работы над инновационной (технологической) идеей в рамках стартап-проекта:

- Просчитайте показатель коммерческой привлекательности вашей IT-идеи, используя имеющиеся формы расчетов экономической оценки IT-проектов. При положительном результате показателя коммерческой привлекательности проекта перейдите к следующему этапу.
- Проведите юридическую экспертизу идеи стартапа-проекта сразу у нескольких независимых друг от друга юристов. Это даст возможность узнать, есть ли у проекта законодательные препятствия, не ограничивает ли он чьи-либо права, реально ли собрать юридические документы, необходимые для работы.
- Взгляните на проект со стороны. Изучите мнение об идее проекта от нескольких человек, к мнению которых вы готовы прислушаться. Презентуйте этим людям идею стартапа и определитесь с их мнением о его рисках, перспективах и сложностях. Выпишите ключевые тезисы, обобщите нюансы, которые вы пропустили и не учли.
- Прислушайтесь к внутреннему голосу и попробуйте честно себе ответить: готовы ли вы заниматься проблемным стартапом? Пугает ли вас отсутствие заработка? Есть ли желание бросить и не начинать? Решение за вами.

В стартапах всегда всё сложнее, чем кажется на первый взгляд, и не все стартап-проекты становятся востребованными и успешными. Поэтому опытные стартаперы советуют довести идею до приближённого к конечному виду, протестировать ее в необходимых условиях и только тогда, когда имея фундамент для презентации проекта, заниматься исключительно только им [6]. Когда же у стартап-проекта появились первые операционные признаки существования (клиенты, договоры, платежи), настало время официально его зарегистрировать. Для чего участникам проекта необходимо:

- Определиться в количестве учредителей и распределить доли между участниками проекта.

- Определиться с бизнес-моделью - определить отрасль деятельности по статистике и организационно-правовую форму собственности проекта Startup (Самозанятый, Индивидуальный предприниматель (ИП), Общество с ограниченной ответственностью (ООО), Акционерное общество (АО).

И соответственно пройти процедуру регистрации ИП или юридического лица. Юридическая регистрация проекта является началом официальной работы по прототипированию продукта, привлечению клиентов и получению оплат и, конечно, это уже показатель наработки определенного «портфеля» для выхода на солидного инвестора и дальнейшего продвижения проекта.

Сегодня современная маркетинговая технология продвижения и продаж в интернете не оставили в стороне сферу Star-tup. В последнее время стала использоваться система продвижения Star-tup проектов и в интернете на так называемых «биржах» стартапов [5]. Биржа стартапов формируется на базе интернет-площадки, где частные и юридические лица размещают свои идеи, проекты, а инвесторы могут вкладывать средства в развитие этих проектов (Napartner, Startupnetwork, Planeta, Kickstarter, Indiegogo и др.). Биржи стартапов - это как магазин по инвестициям в новые проекты. Платформы (биржи) отличаются друг от друга направлениями деятельности, схемами получения прибыли.

И в заключение. Сейчас, когда внедряются новые форматы технологических решений развития экономики и общества, молодое поколение обусловлено быть нацеленным на генерацию идей в области IT. Сегодня - это самая развивающаяся ниша предпринимательской деятельности, где реально продвинуться профессионально целеустремленным со здоровыми амбициями молодым людям, где есть возможность «стартовать с нуля» с инновационной идеей, и создать что-то свое новое, успешное. Но какой бы не была перспективной эта идея на первый взгляд, важно устоять на «ногах» на этапах ее становления.

Литература

1. Стартап. <https://ru.wikipedia.org/wiki/Стартап>.
2. Стартапы в IT. <https://www.sviaz-expo.ru/ru/articles/2016/startapy>.
3. 20 идей для IT-стартапа. <https://viafuture.ru/katalog-idej/idei-dlya-it-startapa>.
4. Как вложить деньги в стартап через Биржи Стартапов. <https://equity.today/kak-vlozhit-dengi-v-startap-birzhi-startapov.html>.

5. Что такое стартап и как развить его в успешный бизнес. <https://trends.rbc.ru/trends/innovation/60ab6bc69a7947711da48a97>.
6. Как проверить идею IT-стартапа: инструкция для будущих программистов - фаундеров.
https://skillbox.ru/media/code/kak_proverit_ideyu_it_startapa_instruktsiya_dlya_budushchikh_programmistov_faunderov.

Выходные данные статьи

Першин Д.С. Стартапы // Корпоративные информационные системы. – 2018. – №2 – С. 1-9. – URL: <https://corpinfosys.ru/archive/issue-2/124-2018-2-startup>.

Об авторе



Першин Дмитрий Сергеевич - инженер по технической поддержке корпоративных информационных систем. Принимал участие в проектах по развертыванию и поддержке информационных систем в крупных ИТ-компаниях как государственных, так и коммерческих. Имеет более чем 10-и летний опыт работы с ITSM системами и системами виртуализации. Адрес контактной электронной почты: mail@corpinfosys.ru.

Стратегия реализации в проектах имплементации корпоративных информационных систем

Прядильников Егор Вячеславович

Аннотация: в статье содержится описание стратегии реализации ERP-систем. Рассматривается соглашение по наименованию технических объектов, задающее маску имен всех новых разрабатываемых и настраиваемых сущностей информационной системы. Анализируются требования из регламента разработки по вынесению константных переменных в отдельную подпрограмму, обязательному ведению на селекционном экране программы параметров, характеризующих оргуровни компании, а также проверке полномочий и ограничению выборки данных на их основе.

Введение

Внедрение современных корпоративных информационных систем класса ERP/ERP2, представляющих сегодня по существу преднастроенные коробочные программные продукты, сильно отличается от имплементации решений, разрабатываемых в рамках проекта «с нуля». В ERP-системах реализация пользовательских требований может осуществляться как путем конфигурирования, так и разработки. Обычно порядка 30% требований покрываются за счет доработки ERP-системы, остальная часть требует программной доработки.

После успешного продуктивного запуска ERP-системы, работа над ней не останавливается. Законодательные требования и рыночная обстановка диктуют новые требования, которые также должны быть реализованы в информационной системе. Поэтому поток программных изменений ERP-системы имеет место быть на протяжении всего ее жизненного цикла. Чем больше доработок осуществляется в корпоративной информационной системе, тем актуальнее становится вопрос качества программных разработок.

В отличие от гибких методологий внедрения, классические каскадные методы требуют изначального проектирования программы, что отражается в функциональной спецификации на разработку. Предполагаемые настройки же описываются в документах конфигурации. Оба документа являются ключевыми результатами фазы проектирования. Для обеспечения высокого качества реализации настроек и разработок согласно упомянутым документам необходимо единообразно подходить к вопросам конфигурирования и программирования, что обычно описывается в

стратегии реализации. В рамках данной статьи мы остановимся на ключевых моментах подготовки такой стратегии.

Цели и задачи

Цель статьи заключается в рассмотрении стратегий реализации программных продуктов в проектах имплементации ERP-систем. Это позволит реализовать программное решение с высоким качеством и вовремя перейти в режим продуктивной эксплуатации, что столь значимо для заказчика. Достижение цели потребует решения следующих задач:

- рассмотрение соглашения по наименованию;
- обзор правил разработки программных продуктов;
- подготовка стратегии реализации.

1. Принципы наименования технических объектов

В процесс разработки ERP-системы формируется программный код, описывающий ту или иную функцию системы, объединенный в программу. Каждая программная разработка характеризуется уникальным наименованием. Однако это не единственный технический объект, генерируемый при разработке. Создаются новые объекты таблиц баз данных, фоновых задач, кодов транзакций, элементов данных, процедур и др., каждый из которых дифференцируется названием. Почему мы делаем такой акцент на названии, позвольте объяснить. Любой вендор регулярно высылает пакеты обновлений своих продуктов, выполняющие изменения программ и прочих системных объектов. Поиск программ для обновления ведется по названию. Поэтому, если вы изменили стандартную программу ERP-комплекта, то пакет обновлений может ее заменить, тем самым вы потеряете свои правки. Если мы говорим о системе SAP ERP, то в ней все клиентские программы могут начинаться только с символов X, Y или Z, чтобы избежать подобных недоразумений (рис. 1). Компании, в которых доработки поставлены на конвейер, пошли дальше и предложили правила (маски) наименования всех объектов ERP-системы. Допустим, первый символ характеризует клиентскую разработку, следующие несколько символов отводятся под код страны, далее задается функциональная область, тип разработки и т.д. Свод правил назвали соглашением по наименованию объектов (Naming convention), который преимущественно готовится на стороне заказчика силами программистов [1].

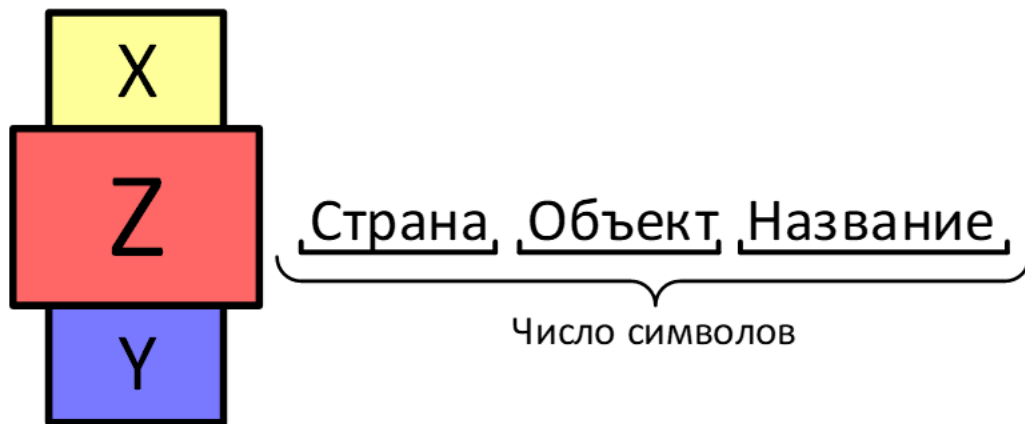


Рис.1. Пример принципа наименования технических объектов

Поэтому, если в ERP-проекте готовятся документы спецификаций на разработку или настроек, то необходимо заблаговременно выяснить наличие правил наименования, в противном случае все придется переделывать и ретестировать.

2. Регламенты качества ПО и реализация программного кода

Помимо соглашения о наименовании объектов могут присутствовать регламенты качества разрабатываемого программного обеспечения [2]. Здесь речь идет об обязательной логике, которая должна быть реализована в приложении, в частности:

- отдельное хранение и ведение константных переменных, используемых в программах;
- обязательность указания организационных уровней на селекционном экране программы;
- контроль полномочий и ограничение данных на их основе.

Реализация программного кода часто требует указания константных переменных. Одним из самых распространённых требований является запрет на указание конкретного значения в теле программы. Для выполнения этого правила, константные переменные выносятся в отдельную подпрограмму или настроечную таблицу, а в самой программе делается лишь ссылка. Кроме того, на одну и ту же константу могут ссылаться различные программы. Тем самым централизация ведения константных переменных исключает необходимость изменения каждой из программ в случае обновления значения константы. На рисунке 2 показан пример настроечной таблицы из системы SAP ERP, позволяющей вести константные переменные и их значения.

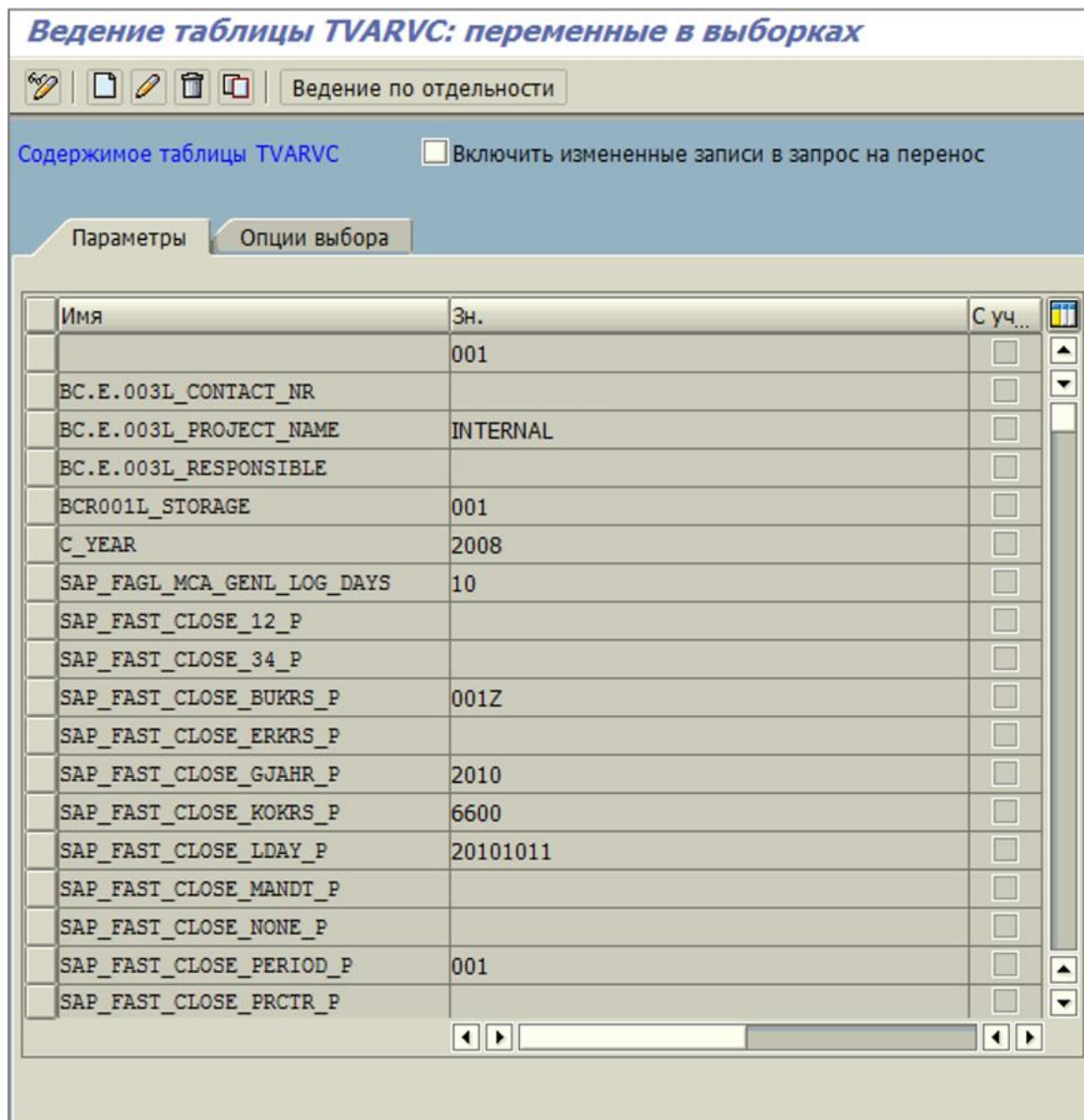


Рис.2. Пример настроечной таблицы для хранения константных значений

3. Организационные уровни разрабатываемой программы

Следующим распространенным правилом, предъявляемым к программным разработкам и их описанию в документах спецификаций и настроек, является обязательное указание организационных уровней на селекционном экране разрабатываемой программы. Под селекционным экраном подразумевается первая экранная форма приложения при ее запуске. ERP-системы являются транзакционными, т.е. ориентированными на обработку большого массива данных, поэтому, чтобы максимально ограничить выборку, применяется селекционный экран (рис. 3), где

пользователь может задать критерии фильтрации. Помимо сокращения объема выбираемых данных, организационные уровни помогают выстраивать проверку ролей и полномочий пользователей. Это становится возможным и обязательным, так как сущность оргуровня является обязательным атрибутом технических ролей системы ERP.

Просмотр оцененного блокированного при поступлении материала

Ограничения выбора из базы данных

Материал		По		→
Завод	1000	По		→
Балансовая единица	1000	По		→
Поставщик		По		→
Группа закупок		По		→
Закупочная организация		По		→
Документ закупки		По		→
Позиция		По		→

Выбор заказов на поставку

История заказа на поставку с приемкой у поставщика

Дата отпуска		По		→
Дата поставки		По		→

История заказа на поставку без приемки у поставщика

Дата поставки		По		→
Дата деблокирования		По		→

Рис.3. Пример указания оргуровней на селекционном экране программы

Если программа разрабатывается «с нуля», то ответственность за безопасность также ложится на плечи разработчика и функционального консультанта, кто поставляем ему спецификацию. Проверка полномочий дает или не дает возможность пользователю запустить на выполнение ту или иную программу из приложения. Однако ни о каком ограничении данных речи не идет. Поэтому выстраивание алгоритма проверки полномочий для запуска программы и его выравнивание с ограничением данных является обязательным требованием со стороны команды по контролю качества программных продуктов. В последующем разработанная программа подлежит переносу в подсистему контроля качества для проведения всевозможных испытаний. В зависимости от ERP-системы перенос настроек и разработок может вестись как в полуавтоматическом режиме, так и полностью ручном.

4. Содержание стратегии разработки

Давайте разберемся, что же должно войти в содержание стратегии разработки и настройки ERP-системы, согласно рассмотренному выше материалу:

- соглашение по наименованию технических объектов, в частности, требуется ли его создание «с нуля» силами исполнителя или соблюдение при наличии у заказчика;
- процедура контроля качества разрабатываемых программных продуктов, а именно: ведется или нет контроль качества на стороне заказчика.

Несмотря на критичность вопроса переноса разработок в смежные подсистемы, чаще всего в стратегию реализации он не входит и прорабатывается в ходе операционной деятельности. Два приведенных выше пункта сильно влияют на трудозатраты проектной команды, так как следование соглашению о наименовании и соблюдение регламентов разработок требует предварительную проверку внутри команды программистов со стороны исполнителя. Чем больше разработчиков, тем больше времени понадобится на проверку.

Заключение

Подведем итоги работы. Соглашение по наименованию технических объектов в проектах внедрения и развития ERP-систем определяет модель зрелости процессов разработки. При имплементации ERP-решения «с нуля» достаточно часто соглашение игнорируется или вовсе отсутствует, что частично можно понять из-за большого объема выполняемых проектных работ. Однако, если речь заходит о проектах развития информационных систем, проводимых намного позже базового ERP-внедрения, отсутствие соглашения свидетельствует о низкой культуре модификации программной системы. Что неминуемо приводит к беспорядку в системе, дублированию данных и сложностям при детективном анализе ошибок. Формирование стратегии реализации позволяет задать единое понимание правил выполнения настроек и ведения разработок, а также порядка контроля их качества. Максимальный эффект от использования стратегии реализации можно достичь лишь при ее применении совместно с концепциями тестирования и технической подготовки систем. О чем вы можете узнать из отдельных статей.

Литература

1. Степанов Д.Ю. Анализ, проектирование и разработка корпоративных информационных систем: уровень приложений / МГТУ МИРЭА. - М., 2017. - URL: [12-erp/52-erp-8-applicationlevel](https://corpinfosys.ru/12-erp/52-erp-8-applicationlevel).
2. Dmitry Yu. Stepanov, Mikhail N. Statsenko. Analysis of functional requirements relevant for developing corporate information systems // Естественные и технические науки. - 2021. - vol.160, №9. - p.98-103. - URL: <https://stepanovd.com/science/article/113-2021-4-requirements>.

Выходные данные статьи

Прядильников Е.В. Стратегия реализации в проектах имплементации корпоративных информационных систем // Корпоративные информационные системы. - 2018. - №2 - С. 10-16. - URL: <https://corpinfosys.ru/archive/issue-2/130-2018-2-developmentstrategy>.

Об авторе



Прядильников Егор Вячеславович - эксперт в области разработки программных решений для корпоративных информационных систем с использованием среды АВАР. Принимал участие в проектах по реализации отчетов, интерфейсов, программ обработки данных, формуляров и прочих расширений различной сложности. Опыт разработки программных приложений более 15 лет.

Адрес контактной электронной почты: mail@corpinfosys.ru.

Стратегия управления изменениями в проектах имплементации ERP/ERP2-систем

Степанов Дмитрий Юрьевич

Аннотация: в статье описывается процесс управления изменениями применительно к проектам внедрения ERP-систем. Описываются порядок управления изменениями, включающий выработку решения по реагированию на внешнее воздействие, оценку параметров изменений, подготовку списка мероприятий для достижения целевых значений параметров, а также их выполнение. Введены параметры изменений, характеризующие технологии, процессы, людей, знания, товары, оргструктуру, подход к управлению предприятием и корпоративную культуру. Отмечено, что в проектах имплементации ERP-систем преимущественно рассматриваются только параметры бизнес-процессов, технологий и численности персонала. Сформулирована стратегия управления изменениями, заданная выбором параметров изменений.

Введение

Естественный ход развития любой компании вне зависимости от ее размера и отраслевой специфики подразумевает применение технологических новинок. Новые технологии заставляют нас иначе смотреть на регулярные бизнес-процессы, а также перестраивать их с учетом последних тенденций на рынке. За примерами далеко ходить не нужно, возьмите, хотя бы Индустрию 4.0, вводящую технологии блокчейн, больших данных, роботизации, виртуальной реальности и интернета вещей в ключевые хозяйственные операции предприятия. Использование любых современных технологий объединяет следующее: все они приводят к изменениям в организации.

Объективно говоря, изменения мало кто воспринимает положительно, по крайней мере, сначала. Воспринимая изменения отрицанием и страхом на начальном этапе, мы в итоге вынуждены принять их. Что, собственно говоря, подтверждено статистикой: лишь 30% опрошенных поддерживают изменения, все оставшиеся – нейтральны или выступают против. Не исключением являются проекты по внедрению ERP-систем, подразумевающие масштабную цифровую трансформацию бизнеса.

Реализация корпоративных информационных систем затрагивает множество аспектов деятельности компании, как то: люди, процессы, знания и др. Своевременное

оценивание изменений, предложение вариантов реагирования на них, а также имплементация решений позволяют планомерно вовлекать большее число сторонников, управлять их участием и получать неоценимую поддержку и помощь. Выработка правильной стратегии управления изменениями позволяет добиться ощутимых положительных результатов, в том числе в ERP-проектах.

Цель и задачи

Целью работы видится в рассмотрении стратегий по управлению изменениями в проектах внедрения ERP-систем. Что позволяет правильно реагировать на сопротивление сотрудников заказчика и доставлять проект реализации информационной системы точно в срок. Выполнение цели потребует проработки следующих задач:

- обзор параметров изменений организации;
- анализ параметров изменений в ERP-проектах;
- подготовка стратегии по управлению изменениями.

1. Процесс управления изменениями

Процесс управления изменениями можно кратко охарактеризовать как знание способов адаптации организации в условиях изменяющегося окружения и действия внешних сил, а также их внедрение, что позволяет закрепить необходимые свойства на значительный период времени и обеспечить тем самым большую конкурентоспособность организации [1]. Особенность любых изменений состоит в том, что поначалу к ним возникает сильное внутреннее сопротивление со стороны сотрудников компании. Обычно, подобная ситуация решается за счет максимального информирования и прямого общения с людьми, что позволяет усилить их участие и вовлеченность в изменениях, а также приводит к постепенной поддержке самих изменений. Существует семь болевых точек, возникающих в компании в процессе изменения, назовем их параметрами изменения (рис. 1).

Практически любое внешнее воздействие меняет устоявшийся порядок работы компании, то есть появляются новые, изменяются и упраздняются существующие бизнес-процессы. Модификация любого бизнес-процесса затрагивает используемые на предприятии технологии, в частности программные обеспечения. Формулируются новые требования к информационной системе, для покрытия которых необходима доработка существующих или внедрение новых программных решений. Последнее в совокупности с внешним воздействием, являющимся причиной изменений, может

привести к увеличению/уменьшению численного состава компании. В идеале, автоматизация подразумевает сокращение ручных операций и количества сотрудников, но это далеко не всегда так.

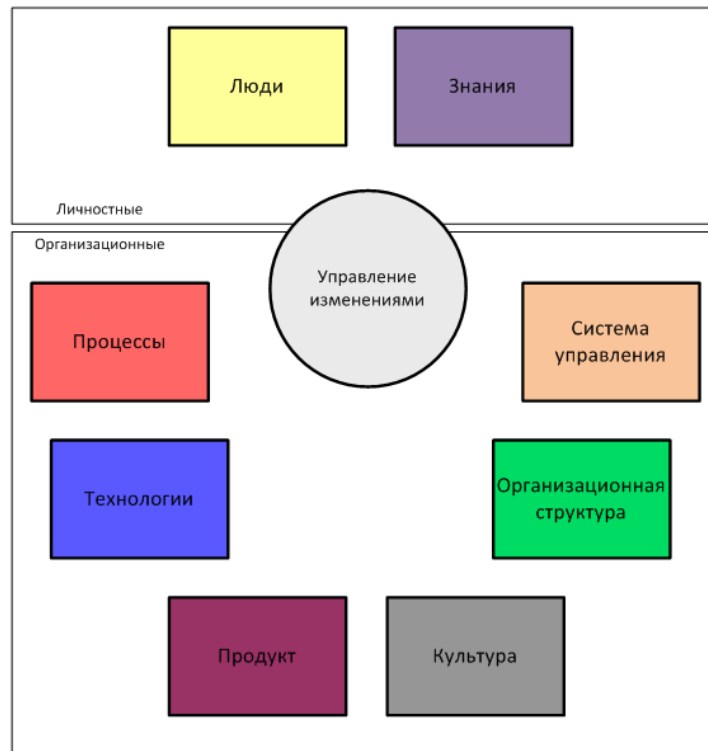


Рис.1. Параметры изменения организации

Обновления процессов, их автоматизация, а также изменение численного состава приводят к образованию дефицита знаний: сотрудникам не понятно, как себя вести для выполнения регулярных бизнес-процессов. Более того, не всегда очевидно, а какой итоговый продукт или услугу компания в ходе изменений предлагает конечному покупателю. Сокращение, а в некоторых случаях увеличение количества сотрудников неминуемо приводит к перераспределению человеческих ресурсов, что находит свое отражение в организационной структуре предприятия. В зависимости от характера изменений может наблюдаться смена как стратегии управления организацией, так и в целом ее корпоративной культуры, задающей миссию, ценности и принципы всего предприятия [2]. Порядок управления изменениями выглядит следующим образом:

- возникновение внешнего воздействия на организацию;
- предложение решения по реагированию на воздействие;
- оценка возможности имплементации решения в компании;

- выявление параметров предприятия, подлежащих изменению путем сравнения их текущих и ожидаемых целевых значений;
- определение мероприятий, позволяющих обеспечить достижение целевых значений параметров изменения;
- оценка реализуемости мероприятий;
- планирование и выполнение мероприятий в случае положительной оценки.

Так изменение технологий требует обновления оборудования, модификации программного обеспечения и адаптации пользовательских инструкций по работе с компьютерными приложениями. Изменение бизнес-процессов и численного состава приводят к необходимости обновления должностных инструкций и пересмотру трудовых договоров. Если поменялся уровень знаний, то сотрудников нужно дообучить. Для принципиально нового продукта определяется позиционирование на рынке, формулируется маркетинговый план по продвижению и задается политика ценообразования. Организационная структура, создаваемая по линейной, функциональной, дивизионной или матричной схемам также может претерпевать изменения, что отражается в соответствующем положении. Обновление в стратегии управления предприятием требует кардинального пересмотра принципов планирования, выполнения и контроля бизнес-процессов, что подразумевает введение новых регламентов работы. И, наконец, положение по корпоративной культуре, описывающее убеждения, правила и нормы поведения в компании, также подлежит изменению, что должно быть доведено до каждого сотрудника. Это далеко не весь список мероприятий, относящийся к параметрам изменений, финальный перечень инициатив зависит от характера внешнего воздействия [1].

Если решением по реагированию на внешнее воздействие служит имплементация ERP-системы, то число рассматриваемых и оцениваемых параметров обычно сокращается до трех: процессы, технологии и люди, остальные, ограниченные знаниями и организационной структурой, являются производными от первых трех и подлежат изменению в любом случае (рис. 2). Действительно, изменение в технологиях всегда приводит к обновлению бизнес-процессов. При этом оба вида изменений диктуют необходимость обучения сотрудников, так как уровень их знаний требует пополнения. Более того изменения в процессах и технологиях преимущественно всегда влияют на численность персонала. Последний параметр является оцениваемым, так как количество сотрудников может, как уменьшится, так и увеличится, однако любые его колебания отразятся на организационной структуре компании.

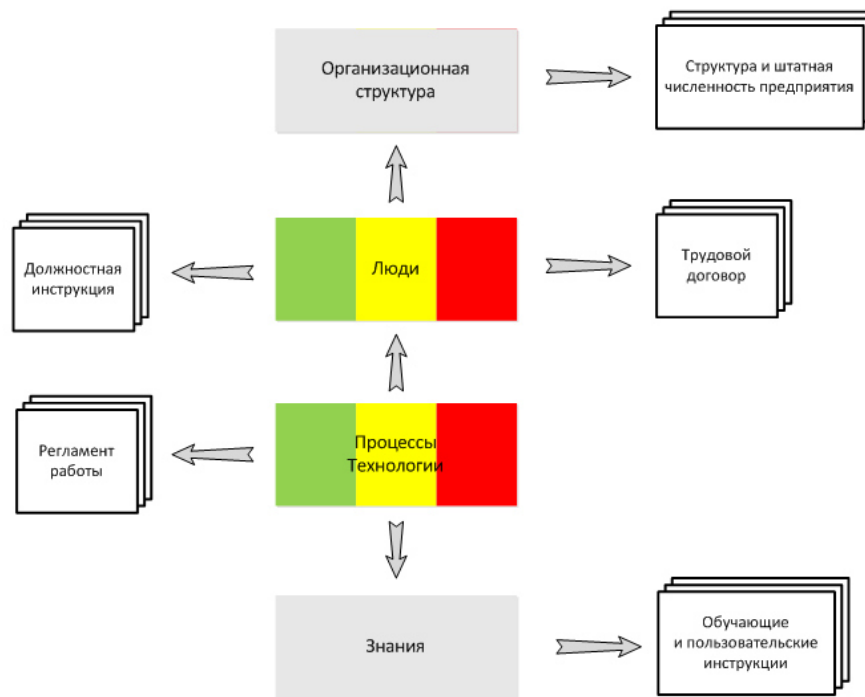


Рис.2. Мероприятия по результатам оценивания ERP-параметров

Тогда под стратегией управления изменениями в проектах внедрения ERP/ERP2-систем будет подразумеваться выбор набора параметров предприятия, подлежащих оцениванию и для которых в последующем будут определяться мероприятия по достижению целевых значений. В случае внедрения ERP/ERP2-систем ограничиваются рассмотрением таких параметров изменения, как: технологии, процессы и люди, а перечень мероприятий обычно включает:

- закупку необходимого оборудования и программного обеспечения;
- подготовку инструкций и обучение конечных пользователей;
- обновление должностных инструкций и трудовых договоров;
- изменение регламентов работы;
- проработка положений по организационной структуре и штатной численности.

Заключение

Мы с вами увидели, что изменения затрагивают практически все аспекты функционирования компании, начиная от миссии, заканчивая программными системами. Параметры изменений включают как личностные, так и организационные повестки. Первым шагом управления изменениями является задание параметров,

подлежащих отслеживанию. Для каждого параметра формулируется порядок оценивания степени его изменения, а также дальнейшие шаги, направленные как на подготовку предприятия к работе в изменившихся реалиях, так и информирование, вовлечение и мотивацию сотрудников.

Рассматривая проект внедрения ERP-системы как внешний фактор, влияющий на текущую организацию и порядок работы компании, описываются три наиболее значимых параметра изменения: численность человеческих ресурсов, технологии и бизнес-процессы. Для указанной тройки продемонстрированы последующие документы, требующие обновления и включающие пользовательские инструкции по работе с ERP-системой, трудовые договора и должностные инструкции, а также положения по структуре и штатной численности сотрудников. Стратегия по управлению изменениями, касающаяся внедрения ERP-систем, затрагивает вопрос выбора тех параметров предприятия, которые наиболее релевантны специфике проекта.

Литература

1. Кожевина О.В. Управление изменениями: учебное пособие. – М.: ИНФРА-М, 2012. – 285 с.
2. Степанов Д.Ю. Анализ, проектирование и разработка корпоративных информационных систем: уровень изменений / МГТУ МИРЭА. – М., 2017. – URL: <https://stepanovd.com/training/12-erp/53-erp-9-changelevel>.

Выходные данные статьи

Степанов Д.Ю. Стратегия управления изменениями в проектах имплементации ERP/ERP2-систем // Корпоративные информационные системы. – 2018. – №2 – С. 17-22. – URL: <https://corpinfosys.ru/archive/issue-2/138-2018-2-changestrategy>.

Об авторе



Степанов Дмитрий Юрьевич – кандидат технических наук, доцент МИРЭА, принимал участие более чем в 10 проектах внедрения корпоративных информационных систем на базе SAP, Microsoft и Sage. Специализируется на управлении материальными потоками, сбытом и системой документов. Автор более 25 статей, в том числе в «Логистика сегодня», «Проблемы экономики», «САТтер». Электронный адрес: mail@stepanovd.com.

Использование SQL-запросов и команд языков программирования для описания алгоритмов выборки данных в функциональных спецификациях на разработку ERP-систем

Петров Сергей Владимирович

Аннотация: в статье затрагивается вопрос описания алгоритмов выборки данных, содержащихся в функциональных спецификациях на разработку ERP-систем. Вводится модель зрелости задания алгоритмов, связанная с опытом технических консультантов, формирующих спецификации. Рассматриваются способы записывания логики селекции данных на основе краткого текста, SQL-запросов и комбинации структурированного языка запросов и выборочных команд из прикладных языков программирования ERP-систем. Показывается, чем ближе описание к текстовому, тем разработчику сложнее понять логику и наоборот.

Введение

Практически каждый проект внедрения корпоративных информационных систем требует их доработки. Для чего готовится документ функциональной спецификации на разработку, содержащий в себе описание исходного требования, верхнеуровневое понимание решения, а также технические детали реализации [1]. Документ готовится вне зависимости от применяемой методологии имплементации, не исключением являются Agile-подходы. В спецификации должен быть соблюден баланс бизнес и технических составляющих: пользователи должны увидеть свои исходные требования к системе и сформировать общую картину решения, в то время как разработчики - найти технические детали, достаточные для старта программирования решения.

Обычно эта задача решается за счет выделения в документе спецификации бизнес и технических частей, тем самым каждый из заинтересованных получает требуемую для работы информацию. К сожалению, нет какого-либо регламентированного подхода, описывающего структуру и содержание спецификации, поэтому в каждом проекте наполнение документа уникально. С технической точки зрения документ спецификации должен содержать описание всех необходимых атрибутов пользовательских экранов и алгоритмов их заполнения,

изменения и очистки. Правил по записи алгоритмов в спецификации так же нет, поэтому основное требование состоит в том, чтобы разработчику была понятна логика техрешения.

Чем опытнее функциональный консультант, тем детальнее он описывает алгоритмы выборки данных, приближая их к языку программирования ERP-системы. И, наоборот, новичок указывает минимум деталей в форме близкой к краткому тексту. Существует несколько подходов, как можно описать алгоритмы обработки данных. При этом они сильно зависят от зрелости и опытности технических специалистов, участвующих в ERP-проекте [2]. Давайте проанализируем несколько методов и дадим им оценку.

Цель и задачи

Цель статьи состоит в обзоре способов описания алгоритмов выборки данных в спецификациях на разработку ERP-систем. Чем качественнее написана функциональная спецификация, тем меньше алгоритмических ошибок ожидается при ее реализации. Достижение этой цели потребует выполнения следующих задач:

- рассмотрение способов описания алгоритмов;
- анализ SQL-запросов для использования в спецификациях на разработку;
- оценивание методов описания алгоритмов.

1. Алгоритмы в спецификациях на разработку ERP-систем

ERP-система является транзакционной, то есть ориентированной на обработку больших массивов данных. Преимущественно в системах подобного класса используются реляционные базы данных, таблицы в которых взаимосвязаны между собой по принципу «сущность-связь». Следовательно, вся информация, необходимая для пользователей хранится в таблицах баз данных и отображается в экранные формы программ по запросу. Подготовка функциональных спецификаций требует описания алгоритмов выборки данных, иными словами логики выбора информации из таблиц баз данных. Существует несколько способов, как можно описать подобные алгоритмы в спецификациях на разработку:

- ссылка на поле таблицы баз данных без указания деталей;
- применение SQL-запросов;
- использование SQL-запросов и прочих команд языков программирования.

Первый и самый распространенный метод описания алгоритмов заполнения полей в спецификации на разработку представляет собой явную ссылку на поле таблицы баз указания деталей, откуда изымается значение. Форма записи в этом случае принимает вид (1):

<Таблица>-<Поле> (1)

Таким способом пользуются новички, так как приведенное описание не содержит технических деталей алгоритмов: входные данные, ограничения и прочие параметры. Более того, если программная разработка является сложной, то достаточно непросто передать всю логику обработки только используя (1), приходится пользоваться кратким текстом для описания алгоритма, что порождает разночтение. Пример использования (1) дан в таблице 1. Как видно из таблицы, описание правил заполнения схоже по принципу с оператором присвоения в языках программирования: слева указывается переменная, а справа - передаваемая ей значения, с той лишь разницей, что переменная характеризует поле на экране программы.

Таблица 1. Пример описания логики заполнения поля на основе ссылки

№	Название поля	Описание поля	Правило	Алгоритм
1	BUDAT	Дата проводки/Posting date	=	MKPF-BUDAT
2	MATNR	Материал/Material	=	MSEG-MATNR
3	MENGE	Коли-во/Quantity	=	MSEG-MENGE

Следующим подходом для указания логики заполнения полей служит применение языка структурированных запросов SQL (Structured Query Language). Структура SQL единообразна и не зависит от языка программирования. Среда программирования может слегка менять синтаксис SQL, добавляя в него специфичный диалект, однако принципиальных различий он не дает. Команд обработки данных в языке SQL немного. Примеры самых частых из них приведены на рис. 1. В спецификациях на разработку в 90% случаев используется оператор SELECT,

все остальные применяются гораздо реже. Этот оператор позволяет осуществить выборку данных из таблицы при указании начальных ограничений (2):

*SELECT * FROM <Таблица> INTO <Временный Массив> WHERE <Условия>*, (2)

где INTO <Временный Массив> является опциональной специфичной командой языка программирования, позволяющей сохранять найденные значения во временную переменную. Обычно переменная именуется схожим с названием таблицы образом, чтобы не путать разработчика. Рассмотрим пример, данный в табл. 2. Из него легко заметить, что использование SQL-запросов значительно упрощает задание выборки. Однако в случае, если на экранной форме нужно выдать сразу несколько полей, значения которых хранятся в той же таблице баз данных, описание кажется слишком нагромождённым. Ощущается необходимость разовой записи SQL-запроса, а далее лишь ссылки на результаты выборки. Поэтому переходим к следующему, финальному способу описания.

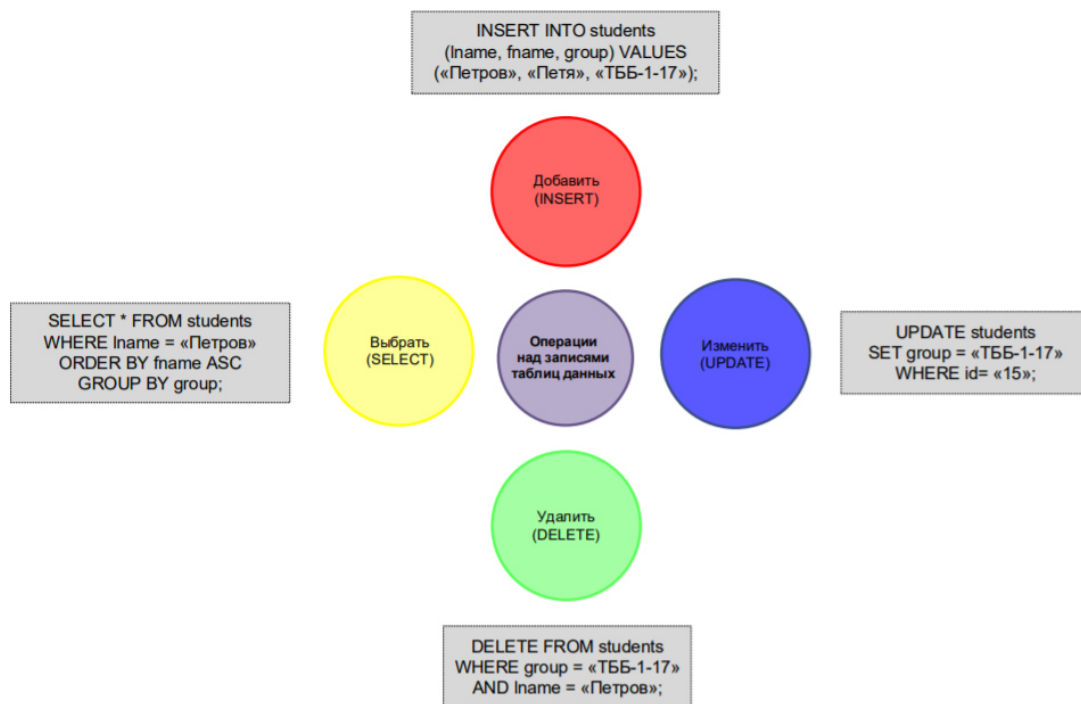


Рис.1. Типовые SQL-запросы и примеры их использования

Таблица 2. Пример описания алгоритма выборки данных на основе SQL-запросов

№	Название поля	Описание поля	Правило	Алгоритм
1	BUDAT	Дата проводки /Posting date	=	Select BUDAT from MKPF where MBLNR = «Номер документа материала» селекционного экрана and MJahr = «Год документа материала» селекционного экрана
2	MATNR	Материал/Material	=	Select MATNR from MSEG where MBLNR = «Номер документа материала» селекционного экрана and MJahr = «Год документа материала» селекционного экрана
3	MENGE	Коли-во/Quantity	=	Select MENGE from MSEG where MBLNR = «Номер документа материала» селекционного экрана and MJahr = «Год документа материала» селекционного экрана

Наконец, третий способ проектирования алгоритмов выборки данных подразумевает применение как SQL-запросов, так и части команд прикладных языков программирования. В отличие от предыдущего способа, SQL-запросы выносятся в отдельные спецификации описания, находящиеся много раньше логики заполнения полей. Здесь работает подход: сначала выбираются все данные во временные массивы, которые затем используются для заполнения полей экранных форм. Поля же, в свою очередь, ссылаются на них. Более того, в случае необходимости задания сложной логики расчета, применяются команды языков программирования. Самая применяемая из которых LOOP AT (3):

Loop at <Таблица/Временный Массив>, (3)

позволяющая выполнять циклическую обработку данных для всех записей массива. Команда, в частности, сильно помогает, когда данные объекта нормализованы и разнесены по нескольким таблицам, например, заголовка и позиций. Проанализируем пример из табл. 3. Обратите внимание, что алгоритмы выбора данных 1-2 описаны в начале секции, причем оба из них сохраняют найденные значения во временные массивы. Алгоритм 2 использует команду LOOP AT для циклического выбора данных из нормализованной таблицы, относящейся к уровню позиции. Для заполнения поля 1, находящегося на уровне заголовка, применяется значение из временного массива, найденного согласно алгоритму 1, в то время, как поля 2-3 ссылаются на данные, полученные на основе алгоритма 2. Тем самым исключается дублирование описания, как это было отмечено в прошлом методе. Данный способ применяют более продвинутые функциональные специалисты в виду сложности.

Таблица 3. Пример описания логики обработки данных на основе SQL-запросов и прочих команд из языков программирования

№	Название поля	Описание поля	Правило	Алгоритм
Алгоритмы выборки				
A1		Select * from MKPF Into MKPF where MBLNR = «Номер документа материала» селекционного экрана and MJAHR = «Год документа материала» селекционного экрана		
A2		Loop at MKPF (шага A1) Select * from MSEG Into MSEG where MBLNR = MKPF-MBLNR and MJAHR = MKPF-MJAHR		
Заполнение полей				
1	BUDAT	Дата проводки/Posting date	=	MKPF-BUDAT (шаг A1)
2	MATNR	Материал/Material	=	MSEG-MATNR (шаг A2)
3	MENGE	Коли-во/Quantity	=	MSEG-MENGE (шаг A2)

Заключение

В виду того, что нет какого-либо общего правила по описанию алгоритмов в спецификациях на разработку для проектов внедрения ERP-систем, каждый функциональный консультант использует свой подход. Если отталкиваться от опытности специалистов, то можно выделить три модели зрелости задания логики обработки данных. Первая модель подразумевает использование краткого текста для описания алгоритмов или проставления ссылки на конкретные поля таблиц данных, где хранится требуемая информация. При этом входные параметры выборок, тонкости алгоритмов и скорость их работы обычно игнорируется. По существу разработчику говорится, что нужно, а вот способ реализации он определяет сам.

Вторая модель зрелости подразумевает применение SQL-запросов с целью записи шагов выборки данных. Этот способ весьма популярен, так как язык структурированных запросов имеет единую форму записи во всех странах и знаком каждому разработчику. Из множества доступных SQL-запросов в спецификациях преимущественно используется команда SELECT. Проблемы в данном случае остаются лишь в том, как записать логику, требующую сложную формулы расчета. Третья модель зрелости предполагает применение как SQL-запросов, так и команд языков программирования, например, LOOP для задания циклов обработки данных, что дает возможность записать даже самые незаурядные формулы. Однако стоит помнить, вне зависимости от модели зрелости, всегда остается возможность снабжения алгоритмов комментариями.

Литература

1. Степанов Д.Ю. Формирование универсальных требований к пользовательским программам при подготовке спецификации на АВАР-разработку // Актуальные проблемы современной науки. - 2014. - т.78, №4. - с.258-268. - URL: <https://stepanovd.com/science/26-article-2014-4-design>.
2. Грофф Д.Р., Вайнберг П.Н., Оппель Э.Д. SQL. Полное руководство. - М.: Вильямс, 2014. - 960 с.

Выходные данные статьи

Петров С.В. Использование SQL-запросов и команд языков программирования для описания алгоритмов выборки данных в функциональных спецификациях на разработку ERP-систем // Корпоративные информационные системы. – 2018. – №2 – С. 30-37. – URL: <https://corpinfosys.ru/archive/issue-2/140-2018-2-sqlspecifications>.

Об авторе



Петров Сергей Владимирович - эксперт по разработке программных решений в банковской, торговой и производственной сферах. Специализируется на языках программирования высокого уровня C++, Java и Transact SQL. Имеет более чем 10-летний опыт разработки приложений. Принимал участие в проектах разработки аналитических, экспертных, биотехнических и корпоративных систем. Электронный адрес: mail@corpinfosys.ru.