

# Подготовка функциональных спецификаций для разработки корпоративных информационных систем на примере SAP ERP (часть 1)

Степанов Дмитрий Юрьевич

**Аннотация:** в работе рассматривается процедура Fit/Gap-анализа и последующее оценивание разработок, описывается RICEF-классификация и принципы проектирования программ, дается структура и содержание функциональной спецификации на разработку на примере системы SAP ERP, вводятся типовые функциональные требования для отражения в документе спецификации.

Внедрение практически любой корпоративной информационной системы требует ее доработки для реализации как законодательных, так и специфических требований предприятия. Согласно [1], стандартный функционал КИС покрывает не более 30% бизнес-требований, все оставшиеся – реализуются разработками и донастройками системы. Ведение доработок ERP и ERP2-систем (ERP, Enterprise Resource Planning) – задача нетривиальная по причине того, что разрабатываемая программа должна успешно решать сформулированную бизнес-задачу, быть масштабируемой и расширяемой, а также не нарушать работу смежных модулей системы.

**Определение 1.** Корпоративная информационная система (КИС) – это расширяемая информационная система, предназначенная для комплексной автоматизации всех видов хозяйственной деятельности компаний, а также корпораций, требующих единого управления [2].

К сожалению, число литературных источников, посвященных проектированию и разработке подобных программ, не так велико, более того существует следующая крайность: либо повествование ведется исключительно для аудитории разработчиков, преимущественно описывая алгоритмы обработки данных, их оптимизацию и построение соответствующей структуры программы [3-5], либо теоретических проектировщиков – вводя всевозможные классы и типы систем и подпрограмм, банальные принципы и требования, не очевидные к реализации, что не дает ответа на вопрос, как правильно моделировать программу и отражать ее в задании на разработку. Конечно существуют различные ГОСТы в области информационных систем [6, 7], однако подобные документы преимущественно описывают постановку задачи и требования к результатам нежели содержательную часть решения. Именно

поэтому процесс проектирования программ весьма критичен и напрямую влияет на качество имплементации ERP/ERP2-систем.

Цель работы состоит в рассмотрении типовой структуры и содержания спецификации на доработку информационных систем, особенностей проектирования приложений и их отражения в документе спецификации, а также формировании универсальных требований к разрабатываемым программам. Для наглядности приводятся практические примеры из ERP-системы от компании SAP AG. Статья дополняет и расширяет содержание работ [8, 9], призвана восполнить пробелы ERP-аналитиков и консультантов при подготовке функциональной спецификации на разработку пользовательских программ. Реализация вышеуказанной цели потребует проработки следующих вопросов:

- Fit/Gap-анализ и оценивание разработок;
- RICEF классификация разработок и принципы проектирования программ;
- моделирование приложения и подготовка к написанию спецификации;
- структура и содержание спецификации на разработку;
- функциональные требования и их отражение в спецификации;
- обработка спецификации после ее реализации.

## **1. Fit/Gap-анализ и оценивание разработок**

### **1.1. Проведение Fit/Gap-анализа**

Анализ, проектирование, конфигурирование и доработка, а также тестирование и внедрение ERP-системы на предприятии потребуют от полу года до полутора лет. Именно поэтому ведется группирование ключевых событий, работ и задач на этапы, уровни и команды, о чем подробно рассказывается в статье [10]. Выделение уровней имплементации КИС соответствует классическому подходу к описанию бизнес-архитектуры предприятия, дополненному активностями по управлению изменениями и непосредственно проектом (рис.1).

На уровне изменений решаются задачи оценивания численности человеческого персонала, его обучения работе с ERP-системой, а также перехода к продуктивному использованию КИС как с технической, так и бизнес точек зрения. Уровень проекта обеспечивает планирование, контроль выполнения и корректировку отклонений в ходе реализации задач всех уровней [11]: процессы, данные, приложения, техника и изменения.

Потребность в доработке информационной системы изначально определяется на этапе анализа, в последующем описывается на фазе проектирования и реализуется в



процессе разработки. Очевидно, что разработка затрагивает самые содержательные уровни проекта внедрения ERP: процессы, данные, приложения и технику. В ходе выполнения процедуры Fit/Gap-анализа идентифицируются, описываются и приоритизируются бизнес-требования пользователей [12], которые согласно каскадной методологии внедрения КИС [13] ведутся в документе Матрицы отслеживания требований.

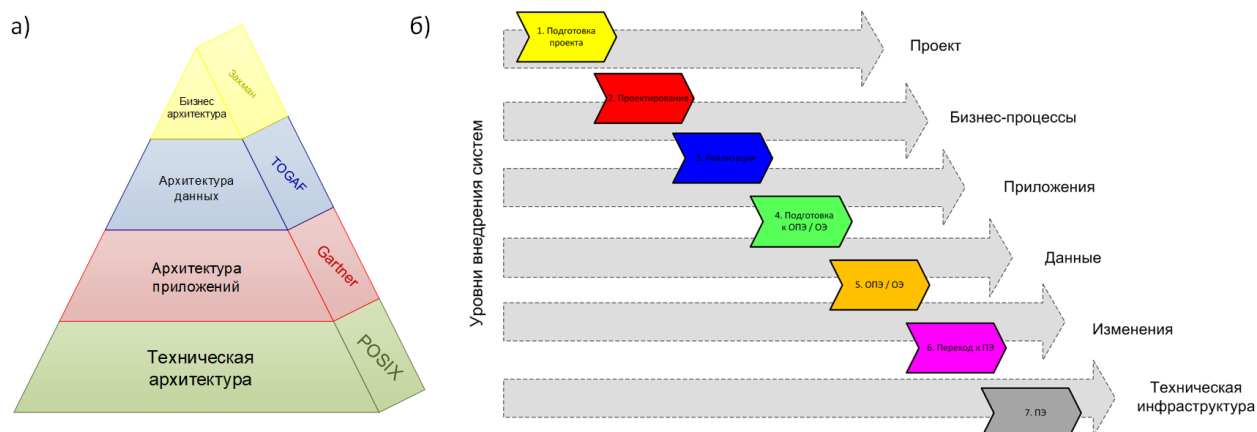


Рис. 1. Группирование работ и задач для: а) описания бизнес-архитектуры предприятия; б) проекта внедрения ERP-системы

**Определение 2.** Матрица отслеживания требований (*Requirement Traceability Matrix, RTM*) – это таблица, позволяющая отслеживать ход обработки бизнес, пользовательских и функционально-технических требований от момента их идентификации до реализации посредством конфигурирования и/или доработки корпоративной информационной системы.

Для каждого бизнес-требования относящегося к категории Gap в документе RTM указывается тип, сложность и процент ре-использования разработки, кроме того наименование подготавливаемой функциональной спецификации.

**Определение 3.** Функциональная спецификация на разработку (*Functional Specification*) – документ, описывающий технические детали разрабатываемой программы, позволяющей реализовать требования к информационной системе. Детали решения включает концептуальную схему программы, алгоритмы обработки данных и заполнения полей, структуру и взаимосвязь реализуемых таблиц баз данных.

Тип разработки определяется на основе RICEF-классификации всех пользовательских программ. Позже мы поговорим о данной классификации более подробно.

**Определение 4.** *RICEF* (сокращение от Report, Interface, Conversion, Enhancement и Form) – это классификация всех программных разработок на отчет, интерфейс, программу обработки данных, расширение и печатную форму [14].

Сложность программы чаще всего задается следующей шкалой:

- низкая;
- средняя;
- высокая;
- очень высокая.

Процент переиспользования характеризует величину ре-использования уже имеющейся программной разработки, обладающей функционалом схожим с требуемым. Введение тройки «тип – сложность – % ре-использования» позволяет унифицировать процесс оценивания трудозатрат всех видов разработки.

**Определение 5.** *Трудозатраты* – это количество дней, затрачиваемое на выполнение работы силами одного человеческого ресурса, выражается в единице измерения человеко-дни. Срок выполнения работы имеет следующую зависимость от трудозатрат:

*Срок выполнения работы = Трудозатраты / Количество человеческих ресурсов.*

Для каждой пары «тип – сложность» приводится экспертная оценка трудозатрат в разрезе таких задач, как:

- подготовка спецификации;
- разработка;
- проведения функционально-модульного тестирования,

значение которых может быть понижено при указании не нулевого %-переиспользования. Оценка ведется единожды и завершается не позже окончания фазы анализа. Назовем документ содержащий расчет трудозатрат для всех комбинаций типов и сложностей программ Оценщиком трудозатрат разработки. Пример фрагмента Оценщика и RTM даны на рис.2.

**Определение 6.** *Оценщик трудозатрат разработки (RICEF Estimator)* – это матрица, определяющая плановые трудозатраты на подготовку функциональной спецификации, разработку программы и ее функционально-модульное тестирование для каждого вида разработки RICEF и сложности.

а)

| Тип RICEF | Сложность     | Трудозатраты подготовки спецификации | Трудозатраты разработки | Трудозатраты тестирования |
|-----------|---------------|--------------------------------------|-------------------------|---------------------------|
| Report    | Очень высокая | 5                                    | 20                      | 4                         |
| Report    | Высокая       | 4                                    | 15                      | 3                         |
| Report    | Средняя       | 3,5                                  | 14                      | 2                         |
| Report    | Низкая        | 3                                    | 12                      | 2                         |

б)

| Требование   | Fit/Gap | Сложность | % переиспользования | RICEF | Трудозатраты подготовки спецификации | Трудозатраты разработки | Трудозатраты тестирования |
|--------------|---------|-----------|---------------------|-------|--------------------------------------|-------------------------|---------------------------|
| Требование 1 | Gap     | Средняя   | –                   | R     | 3,5                                  | 14                      | 2                         |
| Требование 2 | Gap     | Высокая   | 50                  | R     | 2                                    | 7,5                     | 1,5                       |
| Требование 3 | Fit     | Средняя   | –                   | –     | –                                    | –                       | –                         |

Рис. 2. Пример фрагмента: а) RICEF Estimator;  
б) RTM с указанием трудозатрат разработки на основе Оценщика

## 1.2. Оценивание RICEF-разработок

Использование Оценщика обеспечивает единый подход к определению плановых трудозатрат, необходимых для формирования спецификации на разработку. В случае, если разработка представляет собой совокупность нескольких программ, оценивание на основе RICEF Estimator проводится независимо для каждой из ее частей, а итоговые трудозатраты суммируются. Применение плановых значений трудозатрат соответствует PDCA-циклу (цикл Деминга), являющегося основой проектного менеджмента и постулирующего следующее: выполнению любой задачи предшествует шаг планирования, после чего ведется выполнение запланированных работ, далее контроль хода исполнения, а также корректировка возникших план-факт отклонений [11].

Индикативная оценка трудозатрат, необходимых для подготовки спецификации на разработку всевозможных видов программ дана на рис.3. Следуя данным рисунка, становится очевидным: формирование даже самой простой спецификации потребует значительных усилий в 4-7 человеко-дней. Указанная оценка должна быть принята во внимание еще на этапе планирования сроков. Игнорирование плановых трудозатрат, в частности их занижение, приведет к ожидаемым негативным последствиям: качество спецификации оставит желать лучшего, а сэкономленное время планомерно распределится на стадию разработки для устранения дефектов функционально-модульного тестирования.



| Вид разработки                   | Время подготовки |
|----------------------------------|------------------|
| Отчет                            | 3 – 5 чел.дн.    |
| Интерфейс                        | 5 – 7 чел.дн.    |
| Программа<br>обработки<br>данных | 5 – 7 чел.дн.    |
| Расширение                       | 1 – 5 чел.дн.    |
| Печатная<br>форма                | 3 – 7 чел.дн.    |
| Среднее                          | 4 – 6 чел.дн.    |

**Рис. 3.** Индикативная оценка трудозатрат подготовки спецификации  
(как часть документа *RICEF Estimator*)

Предварительно оценив время, необходимое для подготовки спецификации на разработку, обратимся к теории, для чего охарактеризуем виды разработок, а также введем базовые принципы, которым должна удовлетворять пользовательская программа на примере использования продукта SAP ERP.

## 2. RICEF-классификация разработок и принципы проектирования программ

### 2.1. Классификация разработок RICEF

Корпоративная информационная система состоит из большого числа программных разработок. RICEF-классификация позволяет отнести каждую из программ к определенному типу разработки, что накладывает ограничения как на структуру спецификации, так и время ее подготовки. Классификация RICEF расшифровывается следующим образом [14]:

- R – report или отчет, позволяющий отображать данные ERP-системы конечному пользователю в определенном формате без возможности изменения;
- I – interface или интерфейс, обеспечивает как получение данных в систему ERP из внешней подсистемы, так и обратную передачу;
- C – conversion или программа обработки, призванная выполнять операции по созданию, изменению и удалению данных ИС;

- F - form или печатная форма, схожа с разработкой вида отчет, но отображает данные в регламентированном законодательством порядке для целей последующей распечатки и визирования;
- E - enhancement или расширение, дополняющее работу программ указанных выше логикой, продиктованной бизнес потребностями.

Примеры программ согласно RICEF-классификации даны на рисунке ниже. На практике возможна ситуация, когда программная разработка состоит из множества взаимосвязанных подпрограмм. В этом случае классификации подлежит каждая подпрограмма, являющаяся частью сложного приложения.

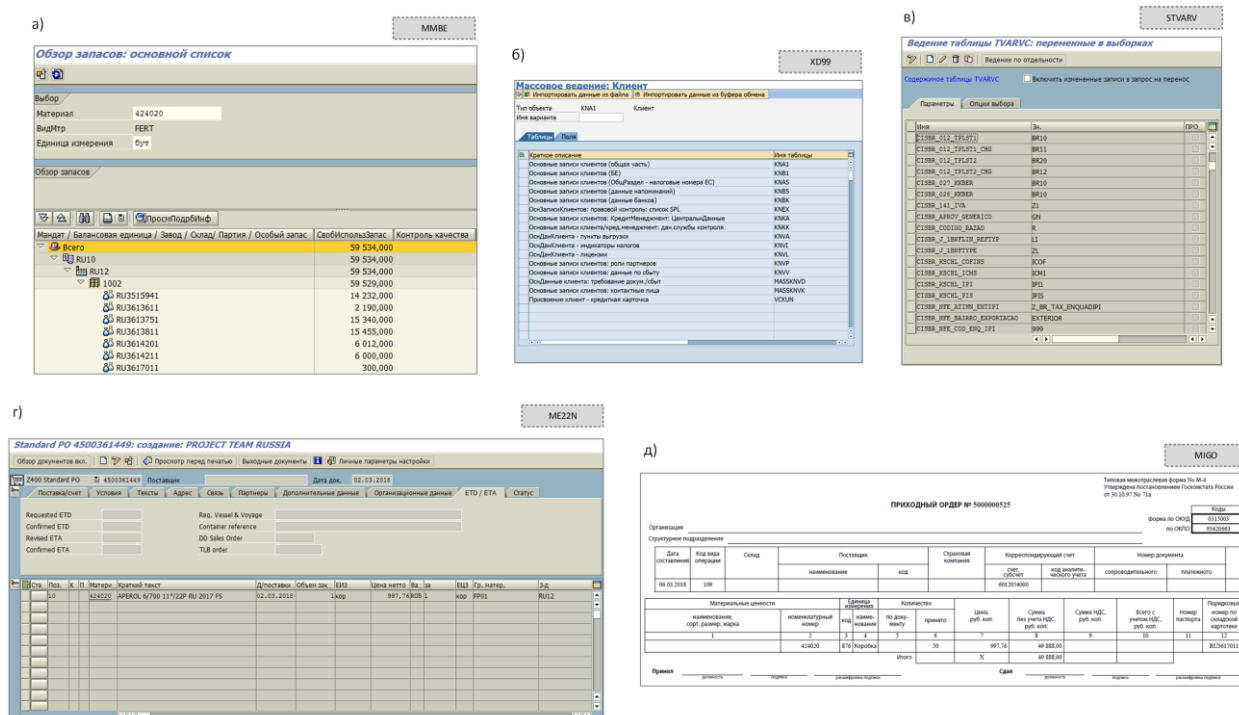


Рис. 4. Примеры программ в SAP ERP по классификации RICEF с указанием кодов транзакций: а) отчет; б) интерфейс; в) программа обработки данных; г) расширение; д) печатная форма

## 2.2. Принципы проектирования программ

Проектируемая RICEF-программа, в последующем отражаемая в спецификации на разработку, должна удовлетворять ряду правил. Например, быть масштабируемой в случае увеличения числа пользователей, гибкой к будущим изменениям, решать задачу оптимальным образом и прочее. Рассмотрим принципы разработки программ, заданные, но не ограниченные следующими предметными областями:

- теория управления системами;

- системный анализ;
- программирование.

### 2.2.1. Принципы теории управления

Среди принципов теории управления: программный, по обратной связи, по возмущению и комбинированный, ведению программных разработок наиболее релевантно правило обратной связи [15]. Применительно к ERP-системам данный принцип заключается в следующем: разрабатываемая программа должна обеспечивать максимальное информирование пользователя о ходе работы, что немаловажно в случае возникновения непредвиденных обстоятельств и ошибок. Рис.5 демонстрирует пример реализации данного принципа в SAP ERP: при попытке сохранить документ Поступления к Заказу на Закупку было выявлено, что не заполнены обязательные поля, о чем система сообщает пользователю в информативном сообщении об ошибке.

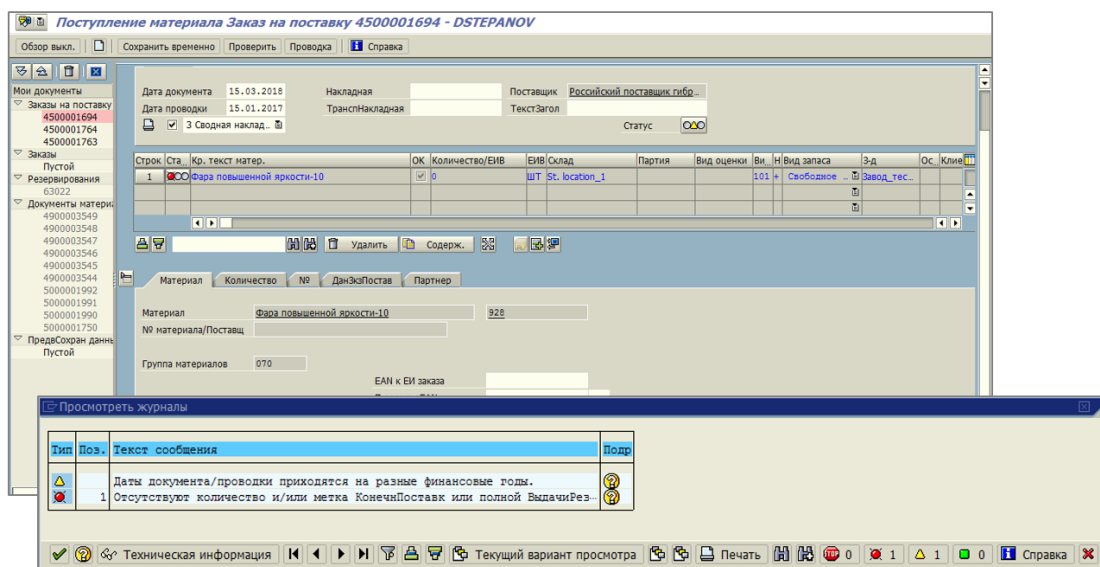


Рис. 5. Пример возникновения ошибки в SAP ERP при попытке сохранить Приход, не заполнив обязательные поля данных (транзакция MIGO)

### 2.2.2. Принципы системного анализа

В процесс разработки ERP-программ системный анализ привносит следующие принципы [16]:

- функциональности;
- развития;
- неопределенности.



Принцип функциональности подразумевает, что структура разрабатываемой программы КИС должна целиком и полностью определяться изначально поставленной задачей. Поэтому чрезмерно сложная, состоящая из большого числа кнопок, пунктов меню и подэкранов программа – неудачная попытка решения различных задач средствами одной программной разработки. Конечно, только в том случае, если программа не представляет собой автоматизированное рабочее место сотрудника, объединяющее множество подпрограмм, каждую из которых можно запустить независимо (рис.6).

а) **Выходные документы/движения материалов**

|                    |               |
|--------------------|---------------|
| Выходные документы |               |
| Вид ВыходДокумента | WE01 По       |
| Средство отправки  | По            |
| Сортировка         | 01            |
| Режим обработки    | 2             |
| Движения материала |               |
| ГодДокумМатериала  | 2016          |
| Документ материала | 4900003252 По |

б) **Поступление материала**

Параметры настройки Система Справка  
Значения по умолчанию  
Правила копирования/проводка переноса  
Прогноз  
Обзор вкл. Сохранить изменения Проверить проводка Справка

ВКв 4530151968 - Dmitry Stepanov

А01 Поступление мат. R01 Заказ на поставку ПМ ПоступлМатериала 101

Общ. Поставщик Paper Work

Дата документа 05.01.2020 Накладная Поставщик PESSONE - IT04  
Дата проводки 05.01.2020 ТранспНакладная ТекстЗагол

1 Отдельная накладн.

| Строк | Стат | Кр. текст матер.                | OK | Количество/ЕИВ | ЕИВ В | Склад        | Партия     | Вид оценки |
|-------|------|---------------------------------|----|----------------|-------|--------------|------------|------------|
| 1     | оо   | Вермут Мартіні Б'яно 15% 0,5*12 |    | 20             | CS    | FG Duty Paid | 0000231333 |            |

Материал Количество № ДанкзПостав. Партнер Партия Фрахт Ext Spec Ext Spec

Материал Вермут Мартіні Б'яно 15% 0,5\*12 4403001760

№ материала/Поставц

Группа материалов 0100

EAN к ЕИ заказа 5010677013154  
Проверка EAN

Рис. 6. Пример печати формуляра в SAP ERP через: а) независимую программу печати (транзакция MB90); б) программу регистрации прихода продукции с возможностью запуска печати через дополнительный пункт меню (транзакция MIGO)

С течением времени каждая программа претерпевает изменения, т.е. развивается. Например, из-за введения новых законодательных требований, увеличения числа пользователей системы и прочее. Принцип развития диктует необходимость проектирования ERP-программ с учетом данного обстоятельства. Именно поэтому в SAP ERP заданная программа успешно функционирует со всевозможными объектами и типами данных, а также на различных организационных уровнях (рис.7). Следуя

данному принципу, разработка отчета, который будет корректно работать, к примеру, только для одного завода, но не всех имеющихся, будет ошибочной.

Рис. 7. Пример отчета в SAP ERP, в котором можно указывать различные организационные уровни для просмотра складских запасов (транзакция MB52)

Рис. 8. Пример ошибки в SAP ERP при неправильном заполнении значения даты (транзакция MB51)

Неопределенность в работе программы должна быть сведена к минимуму. Однако никто не застрахован от неправильных данных, подаваемых на вход программы пользователем системы. Сложность заключается в том, что анализ того, почему программа выдала тот или иной ошибочный результат на основе неправильно

заполненных входных данных, займет гораздо больше времени нежели исключение возможности возникновения подобной ситуации. Поэтому «защита от дурака» требует организации всевозможных проверок для вводимых данных, что является содержанием принципа неопределенности (рис.8).

### 2.2.3. Принципы программирования

Обратимся к базовым правилам ведения программных разработок, среди которых можно выделить принципы [3-5]:

- модульности;
- функциональной избирательности;
- «по умолчанию»;
- надежности.

Ситуация, когда один и тот же программный код многократно повторяется в тексте приложения согласно принципу модульности обрабатывается следующим образом: повторяющийся участок описывается единожды в виде процедуры, далее, если по ходу работы требуется выполнить этот участок кода, осуществляется запуск процедуры по ссылке. Принцип применим и для подготовки спецификации на разработку, что, к сожалению, ухудшает читабельность документа, так как приходится постоянно «перепрыгивать» между страницами, однако значительно сокращает трудозатраты в случае частого изменения документа (рис.9).

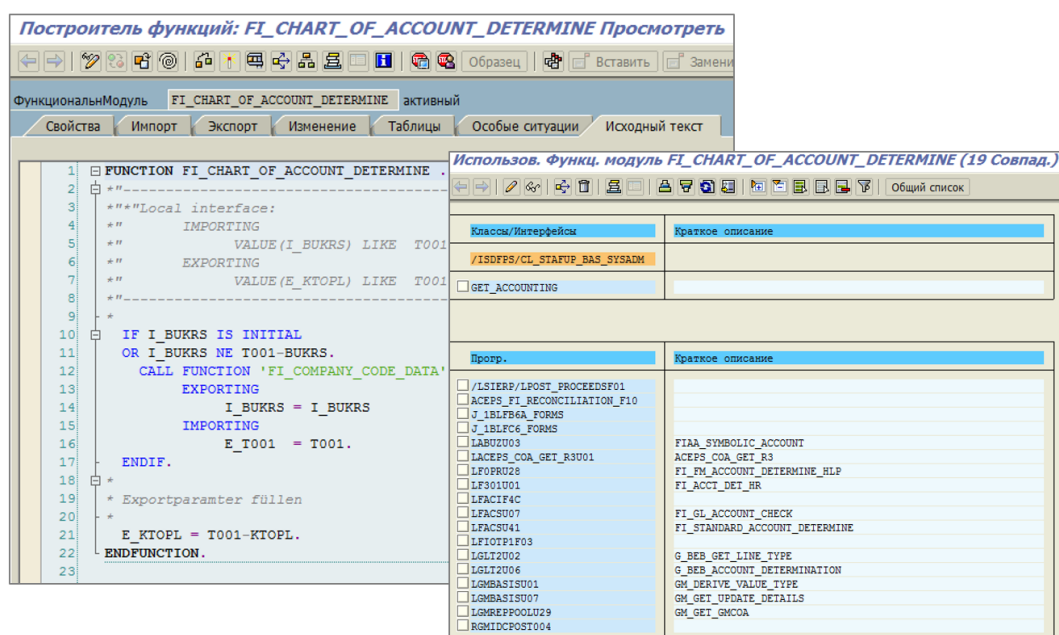


Рис. 9. Пример использования функционального модуля 'FI\_CHART\_OF\_ACCOUNT\_DETERMINE' различными программами SAP ERP (транзакция SE37)



Часто используемые программы должны находиться в оперативной памяти компьютера для возможности их быстрого запуска, что задает принцип функциональной избирательности. Примерами применения данного принципа в работе ERP-систем служат фоновые программы, выполняемые на регулярной основе вне поля зрения конечных пользователей, чаще всего в ночное время (рис.10).

| ИмяЗад                  | Списо | Докумен | Статус      | ДатаЗапуска | ВрЗапуска | Продолж. (сек.) | Задержка (сек.) |
|-------------------------|-------|---------|-------------|-------------|-----------|-----------------|-----------------|
| CODE_INSPECTOR_DELETION |       |         | выпол.      | 15.03.2018  | 01:00:34  | 11              | 34              |
| SAP_SLD_DATA_COLLECT    |       |         | деблокир-но | 15.03.2018  | 05:07:36  | 0               | 0               |
| SAP_SLD_DATA_COLLECT    |       |         | выпол.      | 15.03.2018  | 05:07:36  | 5               | 36              |
| *Объединение            |       |         |             |             |           | 16              | 70              |

Рис. 10. Пример монитора запланированных фоновых задач в SAP ERP (транзакция SM37)

Суть принципа «по умолчанию» однозначно определяется из его названия: все, что может уменьшить число ручных операций выполняемых пользователем, в частности, автоматическое заполнение данных в поля ввода программы, значительно облегчают ему жизнь. С точки зрения отражения требования в документе спецификации и сложности его разработки, казалось бы, мелочь, однако для конечных потребителей ERP-системы – это существенное сокращение трудозатрат (рис.11).

а) **Ведение собственных постоянных значений**

Пользователь: DSTEPANOV  
Изменил: DSTEPANOV 05.01.2020 17:12:27

Адрес: ПостЗначения / Параметры

| Ид. параметра SET/GET | Значение параметра | Краткое о |
|-----------------------|--------------------|-----------|
| LAG                   | 0150               | Склад     |
| WRK                   | RU01               | Завод     |

б) **Обзор запасов: БЕ/завод/склад/партия**

Ограничения выбора из базы данных

Материал: [ ]

Завод: RU01 По: [ ]

Склад: 0150 По: [ ]

Партия: [ ] По: [ ]

Выбор по видам запаса

☒ Особые запасы также выбрать

☒ Открытые запасы также выбрать

Рис. 11. Пример реализации принципа «по умолчанию» в SAP ERP за счет ручного указания пользовательских параметров (а) и их автозаполнение в поля отчета по складским запасам (б), используя транзакции SU01 и MMBE соответственно

Часто мы описываем и в дальнейшем реализуем программные разработки по созданию документов в ERP-системе, предполагая, что приложение будет запускаться довольно редко и на небольшом объеме данных. Исходя из этого, логика функционирования программы нередко оставляет желать лучшего. Допустим, как быть в случае, если программа отработала некорректно и созданные данные требуется аннулировать? Правило надежности говорит о том, что любая программа, создающая или изменяющая данные в КИС, должна обеспечивать возможность их последующего нахождения для целей корректировки или удаления. Именно поэтому в системе SAP ERP большая часть программ разработана и разделена на операции: создание, изменение, удаление и отображение в виде списка (рис.12).

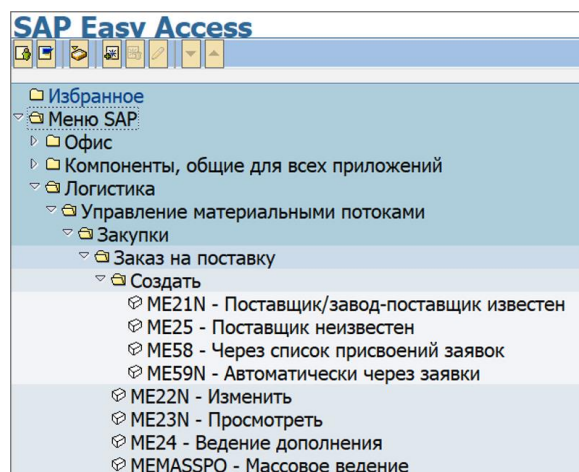


Рис. 12. Пример пользовательского меню SAP ERP, содержащего программы по созданию, изменению и просмотру Заказов на Закупку (транзакции ME21N, ME22N и ME23N)

### 3. Моделирование приложения и подготовка к написанию спецификации

Нередко применение принципов разработки приложений приносит все новые и новые требования к реализации. В первую очередь давайте разберемся с тем, что же такое требование.

**Определение 7.** *Требование* – это настойчивая просьба выполнить что-либо, в контексте внедрения ERP-систем категоризируется на бизнес, пользовательские и функционально-технические. Первые две категории требований часто определяются законодательными изменениями и наиболее трудозатратными бизнес-операциями.

Имплементация ERP-систем, ориентированных на средний и малый бизнес, преимущественно ведется на основе водопадной методологии внедрения информационных систем [13]. Как дополнение к данной методологии часто применяется V-модель разработки через тестирование (рис.13). Особенностью модели



является то, что она определяет и устанавливает четкое соответствие между видами требований и испытаний. Так вводятся бизнес, пользовательские и функциональные требования.

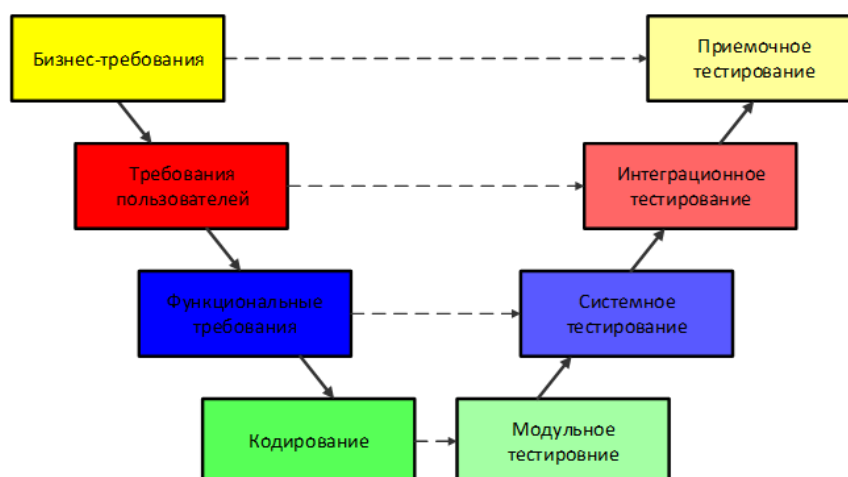


Рис. 13. V-модель разработки через тестирование

Первые два вида требований задаются бизнес-пользователями и носят общий и достаточно верхнеуровневый характер. Детали таких требований преимущественно отсутствуют либо могут по-разному трактоваться. Именно поэтому они подлежат детальной проработки и уточнению для получения четко сформулированных потребностей, реализацию которых легко проверить по результатам завершения разработки. Третий вид требований – функциональные, относящиеся к программной реализации и необходимые консультанту для проработки технических деталей, без которых решение будет неработоспособно.

Существует ряд способов идентификации требований, описание которых дано в [12]. Особо хочется отметить методы демонстрации системы заказчику и прототипирования, наиболее часто используемые на практике. Суть демонстрации (или Workshop, рабочая встреча) состоит в следующем: заказчику показывается существующая программная система в демо-режиме или на презентационных слайдах, после чего ведется уточнение и детализация требований. Прототипирование применяется, когда бизнес потребности не могут быть явно сформулированы, в этом случае создается предварительный рабочий макет программы, на основе которого совместными усилиями пользователей и специалистов определяются требования.

### 3.1. Проектирование структуры и логики работы программы

Идентифицированные пользовательские требования, относящиеся к категории Gap, а также соответствующие им функциональные потребности, отражаются в спецификации на разработку. В будущем данные требования будут покрыты



программной разработкой в ERP-системе. Для этого, руководствуясь PDCA-принципом и предварительно оценив трудозатраты, ведется проектирование предварительной концепции решения как в текстовом, так и графическом видах.

Текстовое описание предполагает отражение наиболее критичных моментов работы программы, в то время как графическое – задание ее логической структуры.

В отличие от привычных всем офисных программ (например, MS Word и Excel, Lotus Notes и др.), ERP-системы оперируют большими данными. Поэтому отображение информации осуществляется преимущественно на экранные формы схожие с электронными таблицами, что удобно для целей последующего анализа. Ранее в статье [8] была введена трехуровневая структура описания программ, суть которой сводится к выделению в программной разработке экранов: селекционного, выбранных и обработанных данных. Используя данную структуру, осуществляется графическое моделирование программы, для чего визуализируются ее экраны и задаются логические взаимосвязи между ними (рис.14).

**Определение 8.** Трехуровневая структура описания программы – это способ моделирования структуры RICEF-программы, состоящий из описания селекционного экрана, экранов выбранных и обработанных данных, а также логики перехода между ними.

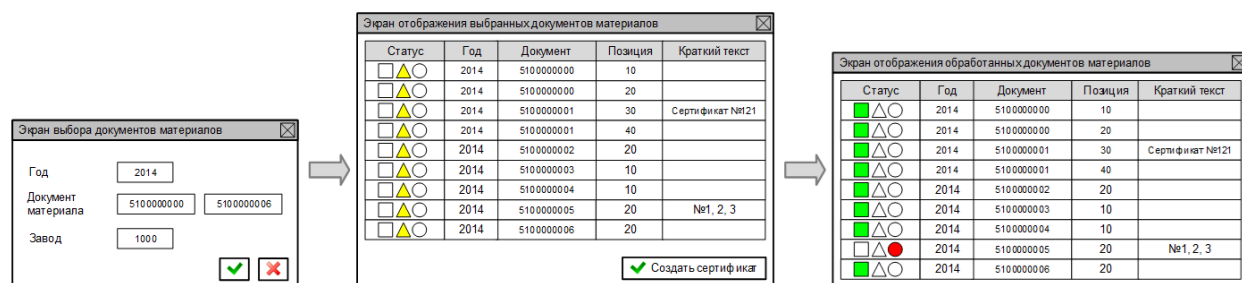


Рис. 14. Пример моделирования программы на основе трехуровневой структуры описания

Проектный опыт подтверждает, структура программы не ограничивается лишь тремя экранами, в частности, когда речь заходит о сложных разработках, содержащих большое число подэкранов, уровней и кнопок. Обобщим трехуровневую структуру описания программ, введя дополнительные пользовательские экраны.

**Определение 9.** Обобщенная структура описания программы – это расширение трехуровневой структуры описания программы, позволяющее выполнять моделирование не только селекционного экрана и экранов выбранных/обработанных данных, но и множества прочих подэкранов, запускаемых различными функциями разрабатываемого приложения.

| Пример разработки                        | Дополнительный экран 1 | Селекционный экран | Экран выбранных данных | Экран обработанных данных | Дополнительный экран 2 | Дополнительный экран 3 |
|------------------------------------------|------------------------|--------------------|------------------------|---------------------------|------------------------|------------------------|
| Интерфейс для загрузки из внешнего файла | ✓                      | ✓                  | ✓                      | ✓                         |                        |                        |
| Программа обработки с 2-мя кнопками*     |                        | ✓                  | ✓                      | ✓                         | ✓                      |                        |
| Программа обработки с 3-мя кнопками*     |                        | ✓                  | ✓                      | ✓                         | ✓                      | ✓                      |

\* - каждая из кнопок запускает отображение дополнительного экрана

Рис. 15. Применимость обобщенной структуры для описания сложных I и C-программ

Рис.15 демонстрирует применимость обобщенной структуры для моделирования программных разработок, состоящих из большого числа пользовательских экранов. Проектирование целесообразно вести, когда количество экранов больше или равно трем, в противном случае непросто понять логику работы приложения. Пример моделирования сложной программы, состоящей из шести экранов дан на рисунке ниже.

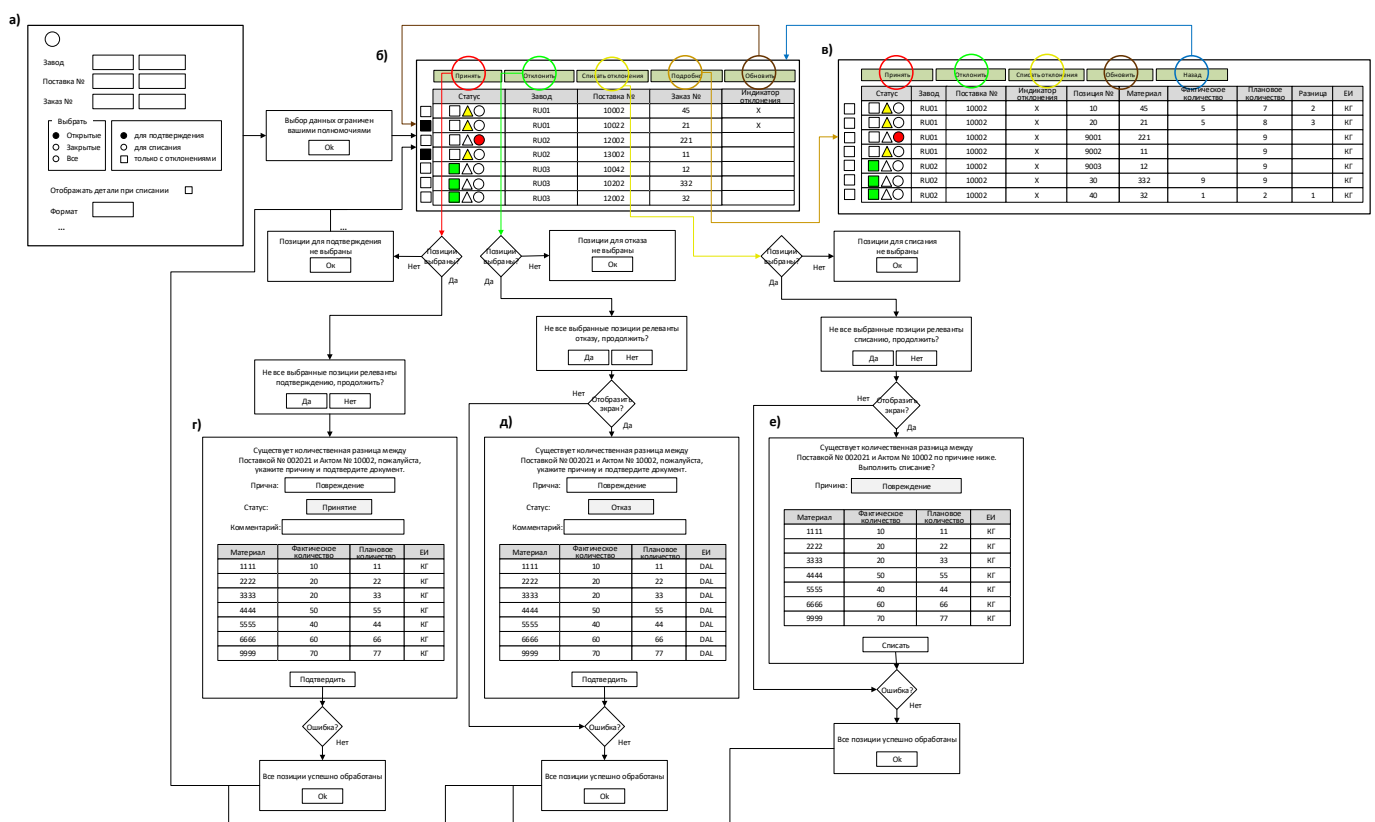


Рис. 16. Моделирование сложной программы на основе обобщенной структуры: а) селекционный экран; б) экран выбранных данных; в) экран обработанных данных; г-е) вспомогательные функциональные подэкраны

### 3.2. Моделирование экранов, объектов и таблиц данных

Моделирование структуры программы помимо демонстрации логики взаимодействия экранов косвенно позволяет решать еще одну задачу – оценивать объем работ на проработку документа спецификации. Ведь каждый введенный пользовательский экран потребует описания полей данных и алгоритма их заполнения. Например, если программа описывается трехуровневой структурой, то в спецификации потребуется задать:

- поля селекционного экрана;
- поля экрана выбранных данных;
- алгоритм заполнения полей экрана выбранных данных, используя поля селекционного экрана;
- поля экрана обработанных данных;
- алгоритм заполнения полей экрана обработанных данных, используя поля селекционного экрана и экрана выбранных данных.

Описание содержательной части разработки в функциональной спецификации начинается с селекционного экрана. Каждое поле селекционного экрана помимо названия, типа и размерности данных будет содержать признаки:

- вида (параметр, диапазон значений, индикатор, радио-кнопка);
- обязательности заполнения;
- значения по умолчанию.

a)

| Селекционный экран                                 |                    |                                                           |            |             |                |                          |                       |            |
|----------------------------------------------------|--------------------|-----------------------------------------------------------|------------|-------------|----------------|--------------------------|-----------------------|------------|
| №                                                  | Наименование поля  | Категория (Params, Select-Options, RadioButton, CheckBox) | Тип данных | Размерность | Элемент данных | Обязательность для ввода | Значение по умолчанию | Примечание |
| Группа полей «Выбор заблокированных счетов-фактур» |                    |                                                           |            |             |                |                          |                       |            |
| 1                                                  | Балансовая единица | Select-Options                                            | CHAR       | 4           | RBKP-BUKRS     | X                        | RU10                  |            |
| 2                                                  | Документ счета     | Select-Options                                            | CHAR       | 10          | RBKP-BELNR     |                          |                       |            |
| 3                                                  | Финансовый год     | Select-Options                                            | NUMC       | 4           | RBKP-GJAHR     | X                        |                       |            |
| 4                                                  | Поставщик          | Select-Options                                            | CHAR       | 10          | RBKP-LIFRE     |                          |                       |            |
| 5                                                  | Дата проводки      | Select-Options                                            | DATS       | 8           | RBKP-BUDAT     |                          |                       |            |
| 6                                                  | Срок оплаты        | Select-Options                                            | DATS       | 8           | RBKP-ZFBDT     |                          |                       |            |
| 7                                                  | Группа закупок     | Select-Options                                            | CHAR       | 3           | EKKO-BKGRP     |                          |                       |            |
| 8                                                  | Пользователь       | Select-Options                                            | CHAR       | 12          | RBKP-USNAM     |                          |                       |            |

b)

Делблокировать заблокированные счета-фактуры

Рис. 17. Пример селекционного экрана: а) описание в спецификации на разработку;  
б) реализация экрана в SAP ERP на примере транзакции MRBR



Отличие вида поля «параметр» и «диапазон значений» заключается в том, что первый позволяет вводить только одно единственное значение, в то время как второй – множество. Пример описания селекционного экрана и его возможная реализация в системе SAP ERP дан на рис.17.

При описании полей экранов выбранных и обработанных данных указываются все те же название, тип и размерность данных, кроме того задаются функции обработки, такие как: фильтрация, сортировка, суммирование, подсуммирование и др. (рис.18).

а) **Экран выбранных данных**

| № | Наименование поля     | Тип данных | Размерность | Элемент данных | Доступность | Примечание |
|---|-----------------------|------------|-------------|----------------|-------------|------------|
| 1 | Номер документа счета | CHAR       | 10          | RBKP-BELNR     | Отображено  |            |
| 2 | Финансовый год        | NUMC       | 4           | RBKP-GJAHR     | Отображено  |            |
| 3 | Валюта                | CUKY       | 5           | RBKP-WAERS     | Отображено  |            |
| 4 | Дата проводки         | DATS       | 8           | RBKP-BUDAT     | Отображено  |            |
| 5 | Балансовая единица    | CHAR       | 4           | RBKP-BUKRS     | Отображено  |            |
| 6 | Выставитель счета     | CHAR       | 10          | RBKP-LIFRE     | Отображено  |            |
| 7 | Документ закупки      | CHAR       | 10          | RSEG-EBELN     | Отображено  |            |
| 8 | Деблокировать         |            |             |                |             | Кнопка     |

б) **Деблокировать заблокированные счета-фактуры**

| № документа счета | ФинансГод | Влт | В... | Д/проводки | БЕ   | ВыставСч   | Базовая дата | Позиц. | ДокумЗакупки |
|-------------------|-----------|-----|------|------------|------|------------|--------------|--------|--------------|
| 5105728632        | 2016      | RUB | КС   | 29.11.2016 | RU10 | 1000037736 | 15.11.2016   | 1      | 4500298450   |
| 5105680155        | 2014      | RUB | РЕ   | 15.12.2014 | RU10 | 1000020466 | 04.12.2014   | 1      | 4500143755   |
| 5105656518        | 2017      | EUR | КС   | 31.05.2017 | RU10 | 1000031444 | 24.03.2017   | 1      | 4500323507   |
| 5105670226        | 2017      | EUR | КС   | 30.06.2017 | RU10 | 1000035489 | 28.06.2017   | 3      | 4500323510   |
| 5105642918        | 2017      | RUB | РЗ   | 28.04.2017 | RU10 | GIE40      | 19.04.2017   | 1      | 4500318816   |
| 5105703088        | 2016      | USD | КС   | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 1      | 4500286281   |
| 5105703088        | 2016      | USD | КС   | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 2      | 4500286281   |
| 5105703088        | 2016      | USD | КС   | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 3      | 4500286281   |
| 5105703088        | 2016      | USD | КС   | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 4      | 4500286281   |
| 5105703088        | 2016      | USD | КС   | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 5      | 4500286281   |
| 5105703088        | 2016      | USD | КС   | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 6      | 4500286281   |
| 5105684443        | 2017      | GBP | РЕ   | 01.12.2017 | RU10 | 1000036264 | 01.12.2017   | 1      | 4500360981   |

Рис. 18. Пример экрана выбранных данных: а) описание в спецификации на разработку; б) реализация экрана в SAP ERP на примере транзакции MRBR

Определив поля селекционного экрана и экрана выбранных данных, переходят к заполнению последнего. Алгоритмы заполнения полей можно описать следующими способами:

- графически в виде блок-схемы алгоритма [17];
- в текстовом виде на основе SQL-запросов на русском или английском языках [18];
- произвольным текстом.

Описание алгоритма в виде блок-схемы достаточно трудоемко, поэтому осуществляется только для задания самых важных и сложных моментов работы программы. Использование языка структурированных запросов к данным (SQL,

Structured Query Language) позволяет единообразно описывать алгоритмы, что немаловажно для международных проектов внедрения КИС. И, наконец, произвольное текстовое описание применяется достаточно редко и чаще всего новичками.

Как было сказано ранее, большая часть программ в ERP-системах представляются в табличном виде. Каждая таблица состоит из столбцов, заданных названиями полей, и строк, определяющих значение каждого поля. Базовый принцип заполнения полей экрана выбранных данных сводится к следующему (рис.19):

- определяется главная таблица баз данных, обеспечивающая информацией большее число полей;
- осуществляется выборка данных из главной таблицы;
- находятся прочие таблицы для заполнения всех оставшихся полей, связанные с главной;
- ведется выборка данных из прочих таблиц.

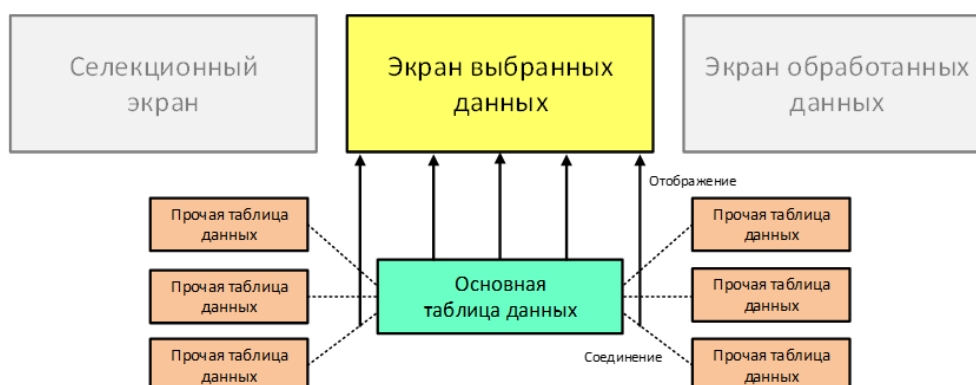


Рис. 19. Принцип заполнения полей экрана выбранных данных

С одной стороны заполнение полей экрана выбранных данных осуществляется на основе входных параметров селекционного экрана, что представляет собой некое сопоставление; с другой – определяются главные и прочие таблицы данных, используя алгоритмы выборки. Объединяя первое со вторым, получаем процедуру по описанию логики заполнения второго экрана программы. Более того, поля экранов могут заполняться статическими значениями, хранимыми в таблице баз данных, и динамическими, рассчитанными по определенной формуле на основе статических.

Подробнее рассмотрим пример на рис.20. Описание логики заполнения полей экрана выбранных данных можно разбить на три части:

- алгоритмы выборки данных из главной и прочих таблиц, на основе селекционных ограничений;



- сопоставление полей экрана выбранных данных и селекционного, если данные передаются без изменений; в противном случае задание источника данных из главной или прочих таблиц, или алгоритма динамической обработки;
- алгоритм удаления данных из массива найденных.

| Экран выбранных данных                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                       |            |             |         |            |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|------------|-------------|---------|------------|
| №                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Наименование поля     | Тип данных | Размерность | Правило | Алгоритм   |
| <b>1. Выбрать данные фактур согласно ограничениям селекционного экрана (основная таблица)</b><br>Select * from RBKP where<br>RBKP-BELNR = «Номер документа счета» селекционного экрана (если поле заполнено)<br>RBKP-GJAHR = «Финансовый год» селекционного экрана (если поле заполнено)<br>RBKP-WAERS = «Валюта» селекционного экрана (если поле заполнено)<br>RBKP-BUDAT = «Дата проводки» селекционного экрана (если поле заполнено)<br>RBKP-BUKRS = «Балансовая единица» селекционного экрана (если поле заполнено)<br>RBKP-LIFRE = «Выставитель счета» селекционного экрана (если поле заполнено) |                       |            |             |         |            |
| <b>2. Для каждой фактуры подобрать данные позиции</b><br>Loop at RBKP<br>Select EBELN from RSEG where<br>RSEG-BELNR = RBKP-BELNR<br>RSEG-GJAHR = RBKP-GJAHR<br>RSEG-EBELN = «Документ закупки» селекционного экрана (если поле заполнено)                                                                                                                                                                                                                                                                                                                                                              |                       |            |             |         |            |
| <b>3. Для каждой позиции фактуры подобрать группу закупки из заказа на закупку</b><br>Loop at RSEG<br>Select BKGRP from EKKO where<br>EKKO-EBELN = RSEG-EBELN                                                                                                                                                                                                                                                                                                                                                                                                                                          |                       |            |             |         |            |
| 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Номер документа счета | CHAR       | 10          | =       | RBKP-BELNR |
| 2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Финансовый год        | NUMC       | 4           | =       | RBKP-GJAHR |
| 3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Валюта                | CUKY       | 5           | =       | RBKP-WAERS |
| 4                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Дата проводки         | DATS       | 8           | =       | RBKP-BUDAT |
| 5                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Балансовая единица    | CHAR       | 4           | =       | RBKP-BUKRS |
| 6                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Выставитель счета     | CHAR       | 10          | =       | RBKP-LIFRE |
| 7                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Документ закупки      | CHAR       | 10          | =       | RSEG-EBELN |
| 8                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Группа закупок        | CHAR       | 3           | =       | EKKO-BKGRP |
| <b>4. Исключить позицию из обработки, если группа закупки не равна значению селекционного экрана</b><br>If EKKO-BKGRP <> «Группа закупок» селекционного экрана (если поле заполнено), then<br>Exclude item from consideration                                                                                                                                                                                                                                                                                                                                                                          |                       |            |             |         |            |

Рис. 20. Пример описания алгоритма заполнения полей экрана выбранных данных в спецификации на разработку

Если программа относится к категории «С – программа обработки данных», экран выбранных данных вероятнее всего будет содержать кнопку, изменяющую информацию системы. Функционирование кнопки описывается схожим образом, для чего задаются:

- предпосылки;
- логика выборки и обработки данных;
- выходные результаты.

Нажатие кнопки «Деблокировать» в экране выбранных/обработанных данных приводит к следующему:

- предварительно одна или несколько позиций входящих фактур должны быть выделены;
- после нажатия кнопки выполняется деблокирование фактуры (устанавливается значение RBKP-DFRGKZ = "");
- в случае успешного деблокирования, позиция окрашивается зеленым цветом в экране обработанных данных.

Рис. 21. Пример описания функциональной кнопки для транзакции MRBR



Пример описания кнопки дан на рис.21. Из рисунка видно, что процедура, отвечающая за изменение, применяется к позициям экрана выбранных данных, имеющих статические или динамические значения. Полученные выходные результаты будут в последующем отображены в экране обработанных данных.

Задание полей и алгоритмов заполнения экрана обработанных данных ведется аналогично экрану выбранных данных с тем лишь отличием, что выборка осуществляется с использованием информации, полученной при срабатывании функциональных кнопок (рис.22).

| № | Экран обработанных данных |            |             | Правило заполнения | Экран выбранных данных |            |             | Примечание                                                                                |
|---|---------------------------|------------|-------------|--------------------|------------------------|------------|-------------|-------------------------------------------------------------------------------------------|
|   | Наименование поля         | Тип данных | Размерность |                    | Наименование поля      | Тип данных | Размерность |                                                                                           |
| 1 | Номер документа счета     | CHAR       | 10          | =                  | Номер документа счета  | CHAR       | 10          | Успешно обработанная позиция при нажатии кнопки «Деблокировать» выделяется зеленым цветом |
| 2 | Финансовый год            | NUMC       | 4           | =                  | Финансовый год         | NUMC       | 4           |                                                                                           |
| 3 | Валюта                    | CUKY       | 5           | =                  | Валюта                 | CUKY       | 5           |                                                                                           |
| 4 | Дата проводки             | DATS       | 8           | =                  | Дата проводки          | DATS       | 8           |                                                                                           |
| 5 | Балансовая единица        | CHAR       | 4           | =                  | Балансовая единица     | CHAR       | 4           |                                                                                           |
| 6 | Выставитель счета         | CHAR       | 10          | =                  | Выставитель счета      | CHAR       | 10          |                                                                                           |
| 7 | Документ закупки          | CHAR       | 10          | =                  | Документ закупки       | CHAR       | 10          |                                                                                           |
| 8 | Группа закупок            | CHAR       | 3           | =                  | Группа закупок         | CHAR       | 3           |                                                                                           |

б) **Деблокировать заблокированные счета-фактуры**

| № документа счета | ФинансГод | Влт | В. | Д/проводки | БЕ   | ВыставСч   | Базовая дата | Позиц. | ДокумЗакупки | Поз. | Материал | ЗаводГрз | Количест. | ЕИЗ | РазнРаз. | ОтклД | К. | К. | Ц. | К. | Р. | С. | В. |
|-------------------|-----------|-----|----|------------|------|------------|--------------|--------|--------------|------|----------|----------|-----------|-----|----------|-------|----|----|----|----|----|----|----|
| 5105728632        | 2016      | RUB | KC | 29.11.2016 | RU10 | 1000037736 | 15.11.2016   | 1      | 4500298450   | 10   |          | RU10 506 | 1,000     | шт. | 0        | 0     |    |    |    |    |    |    |    |
| 5105680155        | 2014      | RUB | RE | 15.12.2014 | RU10 | 1000020466 | 04.12.2014   | 1      | 4500143755   | 10   |          | RU10 506 | 1,000     | шт. | 0        | 0     |    |    |    |    |    |    |    |
| 5105656518        | 2017      | EUR | KC | 31.05.2017 | RU10 | 1000031444 | 24.03.2017   | 1      | 4500323507   | 10   | 708558   | RU11 780 | 700,000   | шт. | 0        | 0     |    |    |    |    |    |    |    |
| 5105670226        | 2017      | EUR | KC | 30.06.2017 | RU10 | 1000035489 | 28.06.2017   | 3      | 4500323510   | 50   | 710508   | RU11 795 | 694,000   | шт. | 0        | 0     |    |    |    |    |    |    |    |
| 5105642918        | 2017      | RUB | R3 | 28.04.2017 | RU10 | GIE40      | 19.04.2017   | 1      | 4500318816   | 10   | 422546   | RU11 351 | 2 048,000 | кор | 0        | 0     |    |    |    |    |    |    |    |
| 5105703088        | 2016      | USD | KC | 30.09.2016 | RU10 | 1000037468 | 09.09.2016   | 1      | 4500286281   | 10   | 423771   | RU11 351 | 1,000     | кор | 0        | 0     |    |    |    |    |    |    |    |

Рис. 22. Пример экрана обработанных данных: а) описание правила заполнения в спецификации на разработку; б) реализация экрана в SAP ERP на примере транзакции MRBR

Моделирование сложных программ потребует больших усилий, в частности, если речь идет о создании программной С-разработки с нуля, возникнут дополнительные шаги проектирования и ведения таблиц баз данных, для чего необходимо:

- определить классы данных;
- определить таблицы для классов данных;
- нормализовать таблицы минимум до третьей нормальной формы (3НФ);
- исключить из нормализации поля, позволяющие формировать минимальную отчетность.

Разделение этапов проектирования классов и таблиц баз данных сделано неслучайно. Дело в том, что на начальных шагах моделирования детальная информация о составе данных может отсутствовать, более того с течением времени она изменится. Нормализовывать таблицы нужно тоже с умом: несмотря на то, что 3НФ постулирует вынесение не ключевых атрибутов, зависящих от других не ключевых

полей в отдельные таблицы [18], на практике данный принцип часто игнорируют в угоду получения минимальной информативной отчетности напрямую из таблиц.

### **3.3. Способы поиска таблиц баз данных в SAP ERP**

Продумав логику взаимодействия экранов программы, предварительное содержание каждого из экранов, а также структуру проектируемых баз данных, процесс подготовки к написанию спецификации завершается определением стандартных таблиц данных ERP-системы, из которых в последующем будет выбираться вся необходимая информация.

Применительно к системе SAP ERP доступны следующие способы выявления стандартных таблиц баз данных [19]:

- нажатие кнопки F1 в поле и последующее чтение технической информации;
- применение пакетов разработок, группирующих таблицы данных для каждой из предметных областей (транзакция SE80);
- использование трассировщика SQL-запросов, обращающихся к базам данных через стандартные команды SELECT, DELETE и UPDATE (транзакция ST21);
- использование функциональных модулей, содержащих в себе последовательность SQL-запросов и выдающих необходимый конечный результат (транзакция SE37),

кроме того вне зависимости от прикладной информационной системы большая часть таблиц баз данных, а также их ER-диаграммы (Entity Relationship, диаграммы взаимосвязи таблиц) доступны в сети интернет.

### **Литература**

1. Калянов Г.Н. Консалтинг: от бизнес-стратегии к корпоративной информационно-управляющей системе. – М.: Горячая линия – Телеком, 2014. – 210 с.
2. О'Лири Д. ERP системы. Современное планирование и управление ресурсами предприятия. Выбор, внедрение и эксплуатация / Пер. с англ. Водянова Ю.И. – М.: Вершина, 2004. – 272 с.
3. Кнут Д. Искусство программирования, том 1. Основные алгоритмы. / Пер. с англ. под ред. Козаченко Ю. – М.: Вильямс, 2006. – 720 с.
4. Архангельский А.Я. Программирование в C++ Builder. – М.: Бином, 2010. – 1508 с.
5. Котеров И.В., Симдянов И.В. PHP7. – СПб.: БХВ-Петербург, 2017. – 1088 с.



6. ГОСТ Р 54869-2011. Проектный менеджмент. Требования к управлению проектом. – М.: Стандартинформ, 2011. – 10 с.
7. Zandhuis A., Stellingwerf R. ISO 21500. Guidance on Project Management. A pocket guide. – NL.: Van Haren Publishing, 2013. – 148 p.
8. 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: <http://stepanovd.com/science/26-article-2014-4-design>.
9. Анализ, проектирование и разработка корпоративных информационных систем: уровень приложений [Электронный ресурс] // Официальный сайт Дмитрия Степанова – Режим доступа: <https://stepanovd.com/training/12-erp/52-erp-8-applicationlevel> (дата обращения 01.09.2019).
10. Степанов Д.Ю. Анализ, проектирование и разработка корпоративных информационных систем: теория и практика // Российский технологический журнал. – 2015. – т.8, №3. – с.227-238. – URL: <http://stepanovd.com/science/31-article-2015-2-erpthpr>.
11. ANSI/PMI 99-001-2014. A Guide to the Project Management Body of Knowledge (PMBOK Guide). – Pennsylvan.: Project Management Institute, 2013. – 589 p.
12. Степанов Д.Ю. Проблемы внедрения корпоративных информационных систем: уровень приложений // Менеджмент сегодня. – 2015. – т.87, №3. – с.180-191. – URL: <http://stepanovd.com/science/30-article-2015-1-erpappl>.
13. Гвоздева Т.В., Баллод Б.А. Проектирование информационных систем: учебное пособие. – Ростов н/Д.: Феникс, 2009. – 508 с.
14. Кречмер Р., Вейс В. Разработка приложений SAP R/3 на языке АВАР/4. – М.: Лори. 1998.
15. Ким Д.П. Теория автоматического управления: линейные системы. – М.: ФИЗМАТЛИТ, 2003. – 288 с.
16. Антонов А.В. Системный анализ. – М.: Высшая школа, 2008. – 456 с.
17. Заботина Н.Н. Проектирование информационных систем: учебное пособие. – М.: ИНФРА-М, 2014. – 330 с.
18. Грофф Д.Р., Вайнберг П.Н., Оппель Э.Д. SQL. Полное руководство. – М.: Вильямс, 2014. – 960 с.

19. Brand H. SAP R/3 Implementation With ASAP: The Official SAP Guide. – NJ.: Sybex Inc., 1999. – 591 p.
20. Лодон Дж., Лодон К. Управление информационными системами. / Пер. с англ. под ред. Трутнева Д.Р. – СПб.: Питер, 2005. – 912 с.

### Выходные данные статьи

Степанов Д.Ю. Подготовка функциональных спецификаций для разработки корпоративных информационных систем на примере SAP ERP (часть 1). – 2019. – №3(7). – С. 29-52. – URL: <https://corpinfosys.ru/archive/issue-7/66-2019-7-functionalspec>

### 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», «САТер». Электронный адрес: [mail@stepanovd.com](mailto:mail@stepanovd.com)