

## 5. Суровая программная инженерия. Методологии, модели, фреймворки, стандарты

### 5.1. Унифицированный процесс: рабочий инструмент или недостижимый идеал

**Унифицированный процесс (Unified Process, UP)** - тяжеловесная методология на основе итеративной модели жизненного цикла, использовании ООП как методологии и UML как рабочего инструмента и средства документирования [5-6]. Rational Rose - фреймворк компании IBM для поддержки вариации этой методологии - **RUP** (Rational Unified Process) [5-2,5-4]. Трудно не заметить почти полного совпадения технологических процессов UP с соответствующими документами SWEBOOK хотя бы по названию. Это вполне логично: методология, охватывающая все стороны и все виды деятельности в жизненном цикле системы, объединяет их в потоки естественным образом. Поэтому UP может служить недостижимым идеалом любой другой методологии. Если описанная в нем деятельность в другой методологии отсутствует, это означает:

- она не нужна в проекте данного масштаба или спецификации;
- она реализуется неформально и не документируется.

Нет смысла воспроизводить здесь подробное содержание UP. Все нижеследующее – конспективное изложение UP, отдельные технологические процессы жизненного цикла рассматриваются в последующих главах. Здесь же дается:

- собственно представление о методологии;
- терминология жизненного цикла. Она довольно объемна, в разных методологиях используются различные вариации терминов. Здесь термины UP выделены жирным шрифтом, синонимы, оригиналы терминов на английском и варианты их перевода - курсивом.

#### Структура UP

Базовые элементы UP, из которых состоит процесс:

- **роли** – исполнение участником процесса ряда взаимосвязанных деятельности обычно в рамках одного, реже – в рамках нескольких технологических процессов;
- **деятельности (activity)** – элементарные работы;
- **артефакты** – рабочие продукты процесса: модели, документы, компоненты программного кода в исходном и скомпилированном виде, библиотеки, конфигурационные файлы, сборки, релизы и т.п.;
- **технологические процессы (дисциплины)** – совокупность связанных между собой деятельностей, имеющих общую цель, результаты и артефакты;
- **фазы (этапы)** - самый крупный временной отрезок жизненного цикла, в результате которого достигается уникальное качественное состояние разработки;
- **итерации** – временной отрезок процесса, завершающийся созданием согласованного набора артефактов. Каждый этап состоит из последовательности итераций.

Фазы жизненного цикла и результаты (рис.5-1):

- **исследование** (в оригинале **inception** - начало) завершается принятием решения о разработке проекта, основным результатом является документ *видения проекта*. Все деятельности направлены на получение информации, необходимой для

принятия такого решения на основании полученных оценок характеристик будущего проекта;

- **развитие** (в оригинале **elaboration** – разработка, развитие, уточнение, в оригинале перевода *уточнение плана*). Представляет собой «теоретическую» проработку будущей системы во всех ее аспектах. Завершается созданием *архитектурного прототипа*;
- **построение** (в оригинале **construction** - построение) – непосредственная реализация системы в действующих артефактах, завершается созданием бета-версии продукта;
- **развертывание** (в оригинале **transition** – перемещение, переход) – перенос, установка системы, распространение и сопровождение.

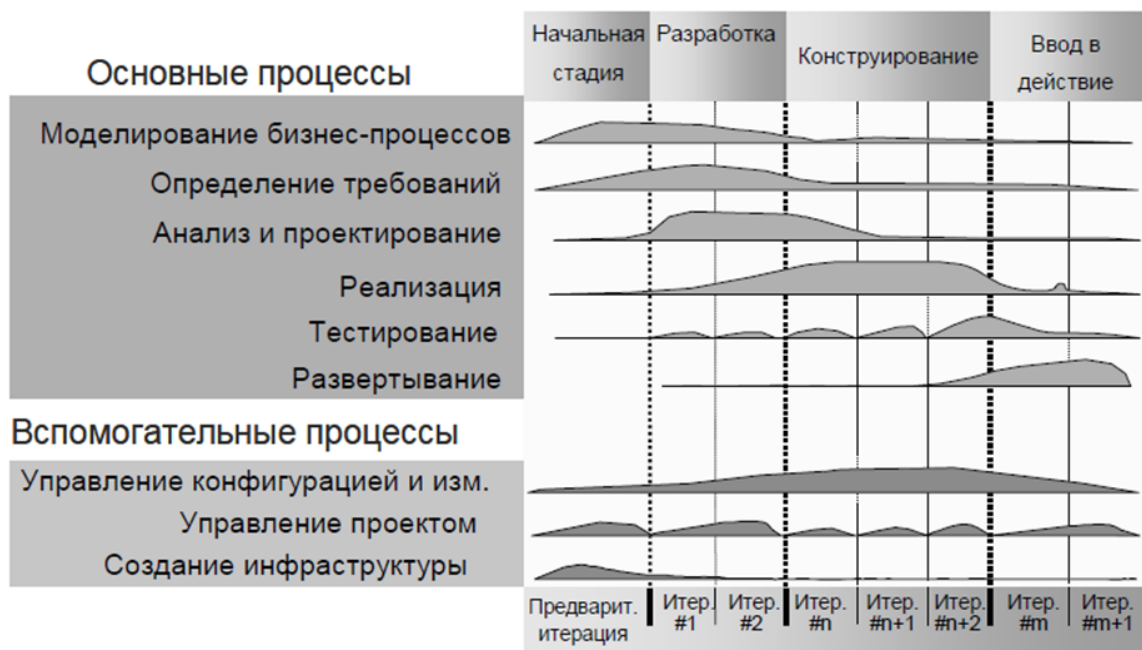


Рис.5-1. Этапы и технологические процессы в UP.

Технологические процессы жизненного цикла:

- **моделирование производства** - *бизнес-аналитика*;
- **управление требованиями** - *системная аналитика*;
- **анализ и проектирование** - *функциональное, архитектурное и детальное проектирование*;
- **реализация** - *конструирование, кодирование (кодинг)*;
- **тестирование**;
- **развертывание** - *распространение, установка, обучение*;

Вспомогательные процессы:

- **управление проектом** - *организация, менеджмент*;
- **управление конфигурацией** – *сборка системы из различных версий артефактов, управлением изменениями, сопровождение*;
- **управление средой разработки**.

Основные артефакты UP изображены на рис.5-2.

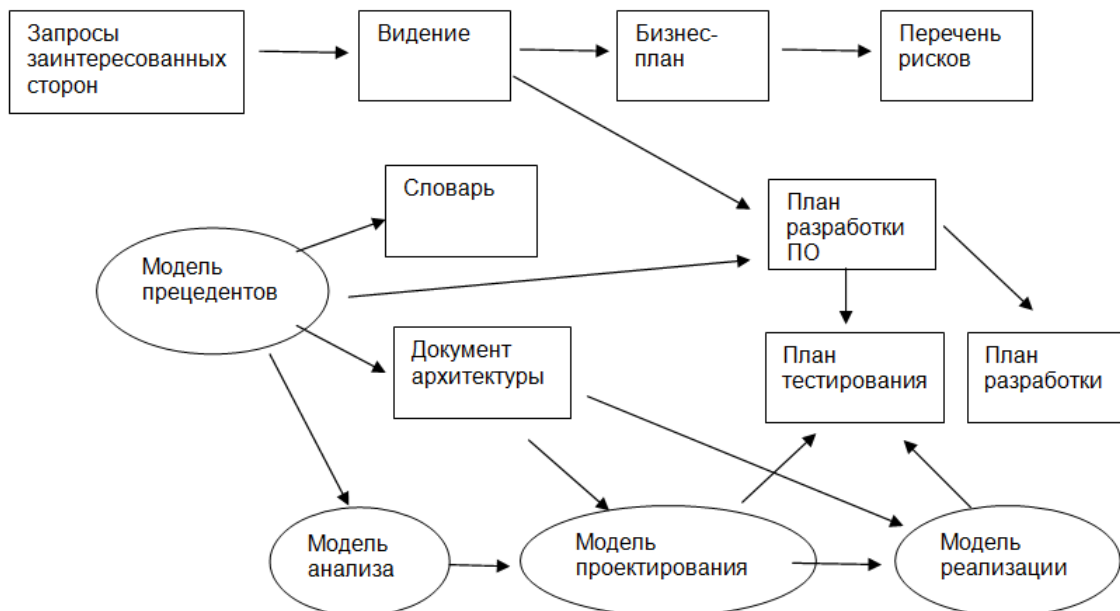


Рис.5-2. Основные артефакты UP

### Фаза исследования проекта

Цель фазы исследования – сбор и анализ информации, необходимой для старта разработки, оценка характеристик будущей системы.

Результаты фазы исследования:

- область применимости, граничные условия – для документа *видение проекта*;
- основные и критические прецеденты системы;
- план возможной архитектуры;
- предварительная оценка стоимости;
- предварительная оценка рисков.

Деятельности фазы исследования:

- формулировка **области действия проекта, рамки проекта** – важнейшие требования, ограничения, среда исполнения;
- разработка **бизнес-плана** - оценка рисков, кадровое обеспечение, цена, график работ, рентабельность;
- разработка возможной **архитектуры**.

Артефакты фазы исследования (рис.5-3):

- документ видения;
- обзор модели прецедентов;
- начальный глоссарий проекта;
- начальный бизнес-план;
- начальная оценка рисков;
- план проекта – фазы и итерации.

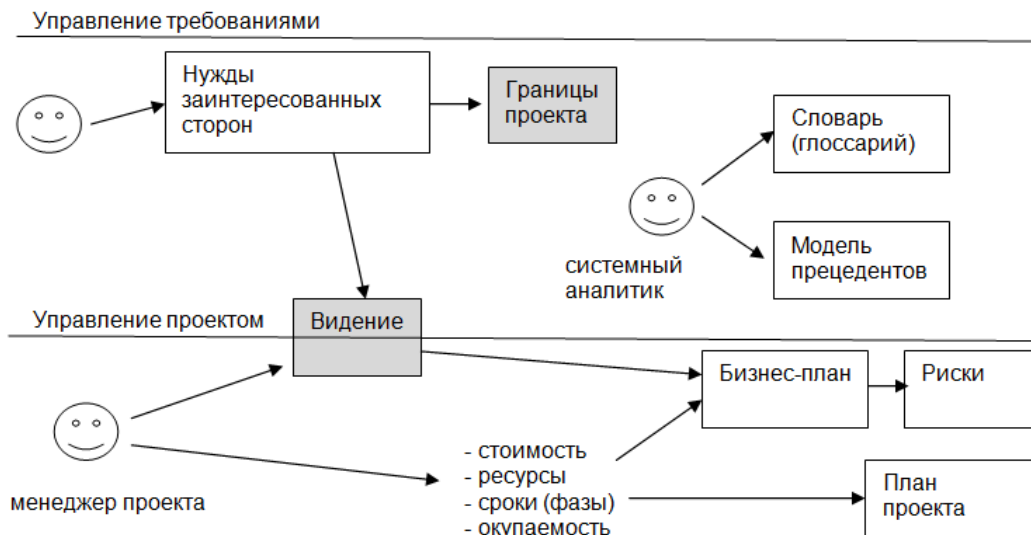


Рис.5-3. Артефакты фазы исследования

### Фаза развития

Цель фазы развития – получение полного и согласованного набора описательных и исполнимых артефактов, необходимых для реализации системы. Проработка и принятие окончательных решений относительно фаз построения и развертывания.

Результаты фазы развития:

- базовая линия архитектуры;
- базовая линия видения;
- базовая линия точного плана построения;
- обоснование, что базовая архитектура позволяет получить требуемое видение за разумную цену, сроки и с разумными рисками;
- выполнимый архитектурный прототип.

Деятельности:

- детальная проработка видения;
- разработка критических прецедентов;
- проработка технологического процесса среды разработки;
- подробная разработка архитектуры и выбор компонентов. Создание/покупка/повторное использование компонентов, график фазы построения.

Артефакты:

- модели прецедентов, охватывающая 80% функционала;
- дополнительные спецификации на нефункциональные требования;
- описание архитектуры;
- выполнимый архитектурный прототип;
- скорректированный перечень рисков и бизнес-план;
- план проекта, итерации и критерии оценки итераций;

Все деятельности по созданию артефактов системы в UP подробно расписаны. Например, так выглядит в каждой итерации процесс создания архитектурного прототипа:

- *управление проектом*. Разработка плана итерации, оценка архитектурных рисков;
- *управление требованиями*. Выбор прецедентов и сценариев, направляющих разработку архитектуры. На основании этого обновление плана итерации и перечня рисков;
- *управление требованиями*. Создание прототипа пользовательского интерфейса;
- *анализ и проектирование*. Выделение очевидных классов, разбиение системы на подсистемы и подробное рассмотрение прецедентов. Артефакт: документ архитектуры;
- *анализ и проектирование*. Уточнение и гомогенизация классов, определение архитектурно значимых классов;
- *анализ и проектирование*. Рассмотрение низкоуровневого разбиения на пакеты;
- *анализ и проектирование*. Согласование среды реализации, принятие решений относительно ключевых сценариев, определение формальных интерфейсов классов;
- *анализ и проектирование*. Рассмотрение параллелизма и распределения архитектуры;
- *реализация*. Модель реализации и ее физическое представление, планирование интеграции, планирование интегральных и системных испытаний, реализация классов и интеграция, интеграция реализованных подсистем;
- *тестирование*. Оценка выполнимой архитектуры.

#### Фаза построения

Фаза построения в содержательном плане является наиболее тривиальной. Она включает в себя модульное конструирование и тестирование, сборку проекта, интеграционное и системное тестирование. Содержание завершающей итерации фазы построения выглядит так:

- *управление проектом*. Планирование итерации;
- *реализация*. Планирование интеграции на системном уровне;
- *тестирование*. Планирование и проектирование тестирования на системном уровне;
- *анализ и проектирование*. Уточнение реализации прецедентов;
- *тестирование*. Планирование и проектирование интегральных тестов на подсистемном и системном уровнях;
- *реализация*. Разработка кодов и тестирование блоков;
- *реализация*. Планирование и реализация блочных испытаний;
- *реализация*. Тестирование блоков в подсистеме;
- *реализация и тестирование*. Интеграция подсистемы;
- *реализация*. Тестирование подсистемы;
- *реализация*. Выпуск подсистемы;
- *реализация*. Интеграция подсистемы;
- *тестирование*. Тестирование интеграции, тестирование системы.

## Фаза развертывания

Деятельности фазы развертывания:

- бета-тестирование;
- параллельное функционирование с существующей системой и ее замещение;
- конвертирование данных, миграция данных;
- подготовка пользователей и персонала;
- маркетинг, распространение и продажа.

**Замечание по теме.** Как следует из описания фаз UP, существует четкий *водораздел* между фазами развития и реализации, между *теоретическим* проектированием в документах и моделях совместно с прототипированием на фазе развития и фактической *залиткой проекта кодом* в фазе реализации. Тем самым предполагается, что результаты фазы развития безупречны. Если вдруг в фазах реализации или развертывания возникнут непреодолимые препятствия, то откат к предыдущей фазе технологически невозможен. Здесь методология обнаруживает общие черты с каскадной.

Далее рассмотрим содержание технологических процессов UP.

## Технологический процесс моделирования производства

Используемый далее синоним названия процесса - **анализ предметной области**. Основная цель – разработка модели технологического процесса, подвергаемого автоматизации. Используются те же средства, что и при управлении требованиями, только модель составляется для действующего процесса. Деятельности в технологическом процессе подпадают под общую термин *бизнес-аналитика*.

Артефакты:

- **документ видения производства;**
- **модель производственных прецедентов** – описание бизнес-процессов предметной области;
- **модель объектов производства** - *диаграмма классов предметной области;*
- глоссарий производства;
- оценка целевой организации;
- **правила производства** - бизнес-правила;
- дополнительная производственная спецификация.

Сценарии, по которым осуществляется моделирование, зависят от того, как будет меняться схема технологического процесса в результате автоматизации. Возможны следующие виды моделей:

- **организационная схема.** Если схема технологический процесс производства не меняется, то создается описание текущего производства: процессы и требования, которая переносится *один к одному* в процесс управления требованиями;
- **моделирование предметной области.** Разрабатывается статическая модель данных предметной области без описания процессов, входит далее в модель объектов производства;
- **новое производство;**
- **реорганизация;**
- **общая модель производства** – усреднение моделей производства нескольких организаций при разработке общего программного проекта;

- **одно производство для нескольких систем** – части одного производства рассматриваются как основа для нескольких программных проектов.

## Технологический процесс управления требованиями

### Цели процесса:

- единое представление функционала системы;
- определение границ системы;
- основа для планирования содержания итераций, оценки стоимости и сроков исполнения;
- определение **пользовательского интерфейса** системы;

Требования в UP классифицируются как **функциональные** требования и прочие свойства. Нефункциональные требования:

- практичность (*Usability*);
- надежность (*Reliability*);
- производительность (*Performance*);
- возможность поддержки (*Supportability*) – требования к поддержке системы после ее выпуска, а также требования к среде проектирования;
- безопасность.

### Характеристики требований:

- приоритеты: обязательные, желательные, возможные, перспективные;
- полезность: критическое, важное, полезное;
- решаемость: проблематичное, выполнимое, тривиальное;
- трудоемкость: высокая, средняя, малая.

### Типы требований по источнику возникновения:

- запросы заинтересованных сторон;
- свойства – требования абстрактного вида;
- программные требования – требования к поведению отдельных компонент.
- прецеденты – единицы функционала.

### Артефакты управления требованиями:

- глоссарий - словарь;
- архив прецедентов;
- прототип пользовательского интерфейса.

В технологическом процессе UP выделяет следующие деятельности: анализ проблемы, понимание нужд заинтересованных сторон, определение системы на основе набора свойств, управление масштабом (границами системы), уточнение определения системы, разработка прецедентов и пользовательского интерфейса.

## Технологический процесс анализа и проектирования

В технологическом процессе создается **модель анализа** – классы анализа и подсистемы, построенные на основе модели предметной области, модели прецедентов и функциональных требований к системе. Далее она эволюционирует в **модель проектирования**, согласованную нефункциональными требованиями, средой

реализации и архитектурой. Также сюда включаются **артефакты систем реального времени и архитектуры** – шаблоны проектирования, протоколы, сигналы, события.

Деятельности и артефакты технологического процесса:

- определение потенциальной архитектуры на основе:
  - эскиза архитектуры;
  - модели классов анализа с учетом архитектурно-значимых прецедентов;
  - взаимодействия прецедентов и классов анализа, представленных с помощью диаграмм устойчивости.
- уточнение архитектуры;
- анализ поведения;
- проектирование компонентов;
- проектирование компонентов реального времени;
- проектирование баз данных.

### Технологический процесс конструирования

Состоит из множества итераций, на каждой из которых интегрируются компоненты, реализованные и прошедшие блочное тестирование. Технологический процесс сборки осуществляется по цепочке: компоненты – интеграция – подсистема реализации – интеграция – система. UP вводит для этого процесса несколько терминов.

**Конструкция** – функциональная версия системы или ее части, демонстрирующая сокращенный вариант возможностей системы. Синоним: *сборка* – вариант интеграции компонентов в подсистему реализации, *сырой* вариант действующей системы на текущий момент.

**Поэлементная интеграция** – интеграция на каждом этапе одной новой или измененной компоненты, **поэтапная** – одновременная интеграция несколько новых или измененных компонент.

**Прототип** – работоспособный, но функционально неполный вариант системы или ее части, предназначенный для оценки, анализа и экспериментирования:

- **поведенческий** – моделирование поведения системы, например, образец интерфейса пользователя, эмулятор для анализа API сервера при отсутствии полной документации;
- **структурный** – заготовки компонент проектируемой системы, которые эволюционируют в итоговый проект;
- **пробный** – временный, используемый для текущих нужд;
- **эволюционный** – в дальнейшем развиваемый как часть проекта.

Артефакты процесса конструирования:

- компонент;
- подсистема реализации (сборка);
- план проведения интеграции.

### Технологический процесс тестирования

Цели процесса тестирования:

- проверка взаимодействия компонентов;
- проверка правильности интеграции;

- проверка точности реализации всех требований;
- выявление дефектов и их устранение до фазы развертывания.

Классификация тестов:

- тестируемый параметр качества:
  - надежность - отсутствие обнаруженных программных ошибок;
  - функциональные возможности – соответствие прецедентам и ожидаемому поведению;
  - производительность.
- этапы тестирования:
  - блочное тестирование - модульное тестирование при конструировании компонента;
  - интегральное тестирование составных компонент или подсистем;
  - системное тестирование;
  - приемочное тестирование перед развертыванием.
- типы теста:
  - аттестационный – оценка производительности в сравнении с прототипом или в виде некоторой меры;
  - конфигурационный – оценка работоспособности системы в различных конфигурациях;
  - функциональные испытания – тестирование прецедентов;
  - установочные испытания – проверка установки в различных конфигурациях с различными ресурсными ограничениями;
  - тестирование целостности – устойчивость к ошибкам времени исполнения;
  - испытание под нагрузкой – функционирование при изменении числа пользователей, транзакций, соединений - *рабочей нагрузки*;
  - стресс-тестирование - испытание в жестком режиме, тестирование при ограниченных ресурсах или высокой рабочей нагрузке;
  - эксплуатационные испытания – производительность в различных конфигурациях при постоянной рабочей нагрузке.

**Регрессионное тестирование** – выполнение всех ранее проведенных тестов на очередной версии после внесенных изменений.

Артефакты модели тестирования:

- **контрольная задача** – набор тестовых данных, условий выполнения и ожидаемый результат;
- **методика испытаний** – описание набора контрольных задач, их настройки, выполнения и оценки результатов;
- **сценарий испытаний** – компонента автоматизации методики испытаний;
- **классы и компоненты испытаний** – программные компоненты системы тестирования;
- **описание программных средств тестирования** в виде диаграмм последовательностей или диаграмм взаимодействия.

Артефакты тестирования:

- модель тестирования;

- план тестирования;
- результаты тестирования и данные, собранные в процессе тестирования;
- **модель рабочей нагрузки** для эксплуатационных испытаний;
- **дефекты** – запросы на внесение изменений для процесса управления конфигурацией;
- **классы и пакеты тестов, подсистемы и компоненты тестирования.**

### Технологический процесс развертывания

Деятельности процесса развертывания:

- бета-тестирование в целевой операционной среде;
- оформление ПО;
- распространение;
- установка;
- обучение конечных пользователей, продавцов;
- замещение существующего ПО, конвертирование БД.

Типы развертывания:

- развертывание заказных систем;
- развертывание из архива с помощью программы установки;
- загрузка из Интернет;

Артефакты развертывания:

- исполняемое ПО;
- артефакты установки – сценарии, инструментальные средства, файлы, руководства;
- материалы поддержки – руководство пользователя и по эксплуатации;
- обучающие материалы.

Технологический процесс развертывания включает в себя деятельности:

- планирование распространения;
- разработка материалов поддержки;
- тестирование продукта в среде разработчика;
- создание версии;
- бета-тестирование версии;
- тестирование продукта в целевой системе;
- оформление продукта;
- обеспечение доступа к узлу загрузки.

## Технологический процесс управление конфигурациями

Цель процесса управления конфигурацией – обеспечение целостности проекта в условиях его непрерывного развития и изменения. Учитывает три фактора изменений в виде трехмерной модели - **куб управления конфигурациями**.

**Управление конфигурацией** рассчитано на три уровня артефактов конфигурации – компоненты, подсистемы, система. Каждый артефакт существует в виде дерева версии. Основные задачи: отслеживание зависимости между артефактами, обеспечение целостности системы артефактов и правильности сборки конфигурации из различных версий;

**Управление запросами на внесение изменений** поддерживает технологическую цепочку внесения изменения в проект, включающую несколько технологических процессов: *управление требованиями, анализ и проектирование, конструирование, тестирование*. Включает в себя деятельности по классификации изменений, отслеживанию состояния, накоплению связанных данных и артефактов.

**Управление состоянием и изменениями** затрагивает также процесс *управления проектом* – сбор и анализ метрических характеристик процесса внесения изменений: прогресс проекта относительно изменений, количество изменений в различных состояниях, *возраст* изменений, распределение изменений по серьезности, приоритету, влиянию на архитектуру, причинам появления.

Артефакты процесса управления конфигурацией:

- план управления конфигурацией;
- запросы на внесение изменений;
- модель реализации;
- метрики и отчеты.

## Технологический процесс управления проектом

Деятельности процесса управления проектом:

- планирование жизненного цикла и последовательности итераций;
- управление рисками;
- наблюдение за прогрессом проекта, сбор метрик и их оценка;
- подбор команды;
- стратегическое планирование: крупнозернистый план фаз - продолжительность основных фаз и итераций и их результаты;
- оперативное планирование текущей и следующей итераций с использованием сетевых графиков, диаграмм Ганта и т.п..

В процесс управления проектом не входят управление персоналом, наем, обучение тренировка, управление контрактами, управление ресурсами. Они являются частью общего менеджмента и не рассматриваются как проектные деятельности.