

Модели итеративных процессов



- Модели итеративных процессов обладают большей гибкостью и способностью работать в меняющемся окружении.
- Итеративный процесс предполагает, что разные виды деятельности не привязаны намертво к определенным фазам разработки, а выполняются по мере необходимости, иногда повторяются, до тех пор, пока не будет получен нужный результат.

Итеративная инкрементальная модель



Итеративная инкрементная модель или модель запланированного усовершенствования продукта использует разработку прототипов (выпусков) для последовательной реализации групп требований.

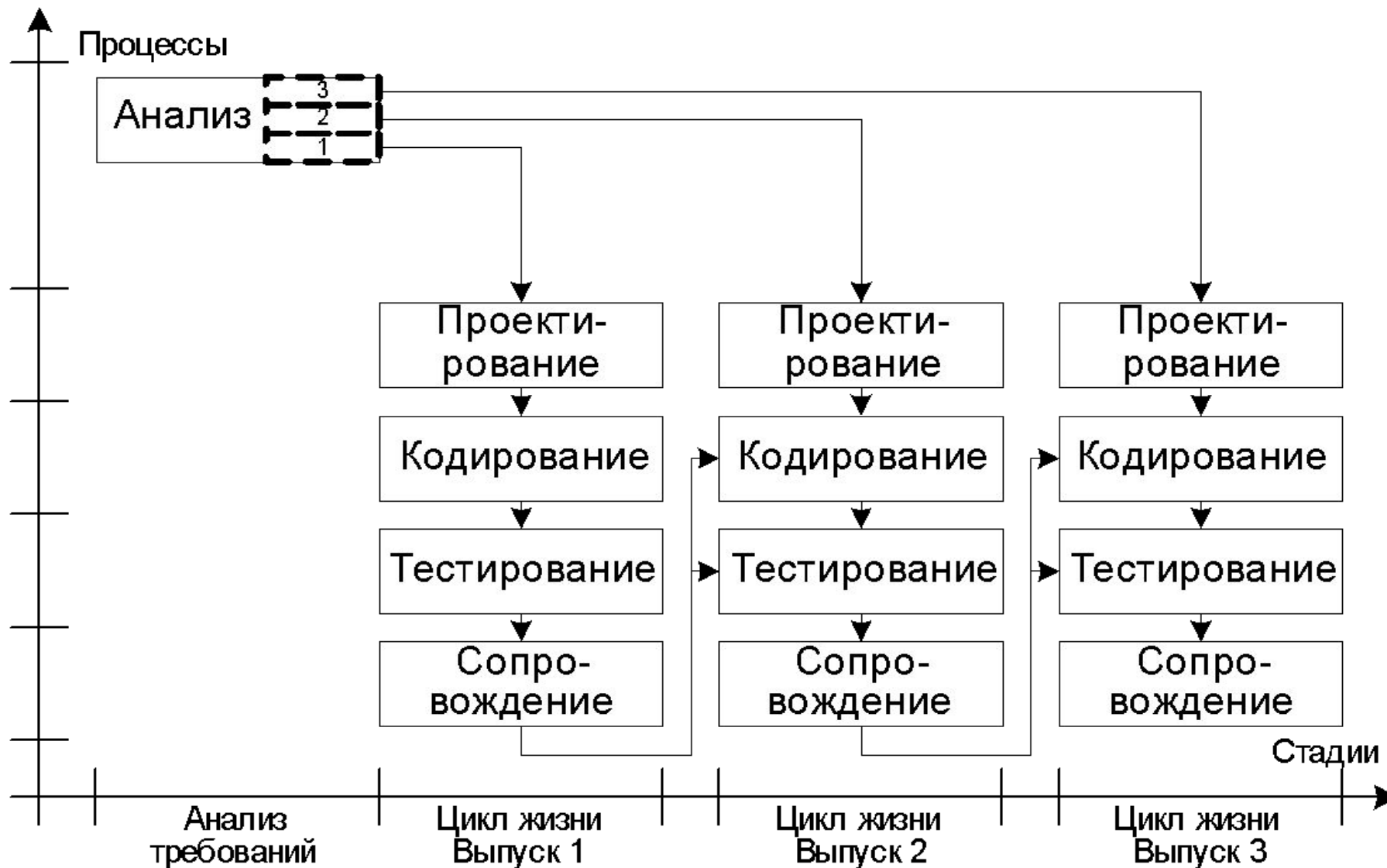
Принцип модели заключается в предварительном выделении требований и разработке прототипов, по функциональности всё более приближающихся к продукту.

Итеративная инкрементальная модель

Первый прототип-выпуск основывается на наиболее понятной группе требований, в последующие реализации добавляются всё новые группы требований, пока не будет закончено создание продукта.

Для каждого прототипа выполняются необходимые процессы, причём анализ требований и проектирование архитектуры выполняются одновременно, а остальные процессы – индивидуально для каждого прототипа.

Итеративная инкрементальная модель



Итеративная инкрементальная модель

Рассматриваемая модель в явном виде включает в своё название два принципа: итеративность и инкрементность разработки.

Итеративность означает разбиение ЖЦ на последовательность итераций, каждая из которых напоминает мини-проект. Цель каждой итерации – разработка прототипа, результатом последней итерации является продукт.

Инкрементность означает разработку продукта путём постепенного учёта требований к системе. Фактически это также приводит к разработке прототипов, причём последний (часто лишь по срокам) прототип считается продуктом.

Rational Unified Process (RUP)

RUP является довольно сложной, детально проработанной итеративной моделью жизненного цикла.

Rational Unified Process является также продуктом, предоставляемым IBM Rational.

Некоторые из средств **IBM Rational** благодаря возможности их расширения оказываются применимыми в качестве инструментария для других подходов. В частности, подход **MSF** долгое время основывался на этом наборе до разработки собственного набора средств.

Rational Unified Process (RUP)



RUP создан на основе лучших практик, позволяющих создавать успешные проекты.

Лучшая практика – практика разработки, проверенная на многих проектах и позволяющая вместе с другими практиками повысить эффективность и успех проекта.

RUP использует следующие практики:

1. Итеративная разработка;
2. Управление требованиями;
3. Использование компонентной архитектуры;
4. Визуальное моделирование;
5. Проверка качества;
6. Контроль изменений.

Практики RUP

Итеративная разработка заключается в создании требуемой системы итерациями, что обеспечивает:

- снижение воздействия серьёзных рисков на ранних этапах проекта, пока это ещё можно сделать с минимальными затратами;
- возможность организовать устойчивую обратную связь с будущими конечными пользователями с целью создания системы, реально отвечающей их потребностям;
- концентрация усилий на наиболее важных направлениях проекта;
- непрерывное итеративное тестирование продукта, позволяющее оценить успешность всего проекта в целом; раннее обнаружение несогласованностей между требованиями, дизайном и реализацией;
- равномерное распределение ресурсов проекта;
- реальная оценка текущего состояния проекта.

Практики RUP

Управление требованиями включает в себя выявление, организацию и документирование требований к системе и подразумевает:

- организованный подход к управлению требованиями;
- взаимодействие участников проекта на базе выявленных и утверждённых требований;
- ранжирование требований по приоритету, фильтрация их по необходимым параметрам и выявление зависимости между ними;
- объективная оценка состояния проекта на основе реализованной функциональности и текущей производительности системы;
- раннее обнаружение различных несоответствий и расхождений;
- использование инструментальных средств для организации более эффективного процесса управления требованиями.

Практики RUP

Использование компонентной архитектуры даёт возможность повысить эффективность процесса разработки за счёт:

- повышения гибкости архитектуры создаваемой системы;
- чёткого определения изменений, требующихся при доработке системы;
- наличия множества готовых коммерческих компонентов, которые построены на основе промышленных спецификаций, что облегчает реализацию и позволяет экономить проектные ресурсы;
- упрощения задач по управлению конфигурацией продукта;
- использования средств визуального моделирования, опирающихся на компонентный подход.

Практики RUP

Визуальное моделирование – это способ представления проблемы с помощью моделей, построенных вокруг идей из исследуемого мира. Оно позволяет:

- однозначно описать функциональное поведение разрабатываемой системы;
- определить архитектурно значимые компоненты системы и сосредоточиться на их реализации;
- обеспечить построение гибкой и надёжной архитектуры и системы в целом;
- исключить из рассмотрения второстепенные детали, не влияющие на решение задачи;
- выявить на ранних стадиях проекта ошибки проектирования и несогласованность в реализации отдельных компонент системы.

Практики RUP

Проверка качества реализуется с помощью тестирования сценариев.

Непрерывная проверка качества обеспечивает следующие выгоды:

- производит оценку состояния проекта по объективным показателям;
- позволяет на ранних стадиях проекта обнаружить несоответствия в требованиях, дизайне и реализации;
- акцентирует внимание на тех сторонах работы системы, которые имеют наибольшую важность и повышенный риск;
- дефекты выявляются на ранних этапах, снижая затраты на их устранение;
- автоматизированное тестирование обеспечивает снижение влияния «человеческого фактора» и повторяемость результатов.

Практики RUP

Контроль изменений включает в себя управление запросами на изменение, управление конфигурацией и управление измерением и обеспечивает:

- контроль состояния проекта в целом и отдельных задач на основании статусов запросов на изменение;
- хранение историй изменений по каждому запросу на изменение;
- актуальную информацию по загрузке участников проекта;
- возможность оценки текущего состояния на основании тенденций по сокращению / увеличению новых запросов на изменение, вновь обнаруженным ошибкам, средним срокам выполнения запросов и т.п.;
- учёт трудозатрат участников проекта по выполняемым изменениям;
- упрощение коммуникаций между участниками проекта: необходимые данные об изменениях всегда доступны и актуальны.

Принципы разработки RUP



RUP основан на наборе из 6 ключевых принципов бизнес-управляемой разработки, сокращённо называемый ABCDEF:

1. Адаптация процесса;
2. Баланс приоритетов заинтересованных лиц;
3. Сотрудничество между командами;
4. Демонстрация результата итерационно;
5. Эволюция (рост) уровня абстракции;
6. Фокусировка непрерывно на качестве.

Эти принципы были определены Пером Кроллом и Уолкером Ройсом.

Rational Unified Process (RUP)



RUP делит жизненный цикл ПО на 4 основные фазы, в рамках каждой из которых возможно проведение нескольких итераций.

Модели итеративных процессов . RUP

Фаза начала проекта (Inception)

Основная цель— достичь компромисса между всеми заинтересованными лицами относительно задач проекта. На этой фазе определяются основные цели проекта, руководитель проекта и бюджет проекта, основные средства его выполнения — технологии, инструменты, ключевой персонал, а также происходит, возможно, апробация выбранных технологий с целью подтверждения возможности достичь целей с их помощью, составляются предварительные планы проекта.

Модели итеративных процессов . RUP

Фаза проработки/уточнения (Elaboration)

Основная цель этой фазы — на базе основных, наиболее существенных требований разработать стабильную базовую архитектуру продукта, которая позволяет решать поставленные перед системой задачи и в дальнейшем используется как основа разработки системы.

Модели итеративных процессов . RUP

Фаза построения (Construction)

Основная цель этой фазы — детальное прояснение требований и разработка системы, удовлетворяющей им, на основе спроектированной ранее архитектуры.

Фаза передачи/внедрения (Transition)

Цель этой фазы — сделать систему полностью доступной конечным пользователям. Здесь происходит окончательное развертывание системы в ее рабочей среде, подгонка мелких деталей под нужды пользователей.

Модели итеративных процессов . RUP



Кроме того, RUP определяет следующие виды деятельности, которые в разных комбинациях и с разной интенсивностью, выполняются на разных фазах.

Модели итеративных процессов . RUP

- **Моделирование предметной области (бизнес-моделирование, Business Modeling)**

Цели — понять бизнес-контекст, в котором должна будет работать система (и убедиться, что все заинтересованные лица понимают его одинаково), понять возможные проблемы, оценить возможные их решения и их последствия для бизнеса организации, в которой будет работать система. Результат - модель в виде набора диаграмм классов (объектов предметной области) и деятельностей (представляющий бизнес-операции и бизнес-процессы).

Модели итеративных процессов . RUP

- **Определение требований (Requirements)**

Цели — понять, что должна делать система, и убедиться во взаимопонимании по этому поводу между заинтересованными лицами, определить границы системы и основу для планирования проекта и оценок затрат ресурсов в нем.

Требования принято фиксировать в виде модели *вариантов использования (use cases)*.

Модели итеративных процессов . RUP

- **Анализ и проектирование (Analysis and Design)**

Цели — выработать архитектуру системы на основе требований, убедиться, что данная архитектура может быть основой работающей системы в контексте ее будущего использования.

В результате проектирования должна появиться проектная модель, включающая диаграммы классов системы, диаграммы ее компонентов, диаграммы взаимодействий между объектами в ходе реализации вариантов использования, диаграммы состояний для отдельных объектов, и диаграммы развертывания.

Модели итеративных процессов . RUP

- **Реализация (Implementation)**

Цели — определить структуру исходного кода системы, разработать код ее компонентов и протестировать их, интегрировать систему в работающее целое

- **Тестирование (Test)**

Цели — найти и описать дефекты системы (проявления ее недостаточного качества), оценить ее качество в целом, оценить выполнение гипотез, лежащих в основе проектирования, оценить степень соответствия системы требованиям

Модели итеративных процессов . RUP

- **Развертывание (Deployment)**

Цели — развернуть систему в ее рабочем окружении и оценить ее работоспособность

- **Управление конфигурациями и изменениями (Configuration and Change Management)**

Цели — определение элементов, подлежащих хранению и правил построения из них согласованных конфигураций, поддержание целостности текущего состояния системы, проверка согласованности вносимых изменений

Модели итеративных процессов . RUP

- **Управление проектом (Project Management)**

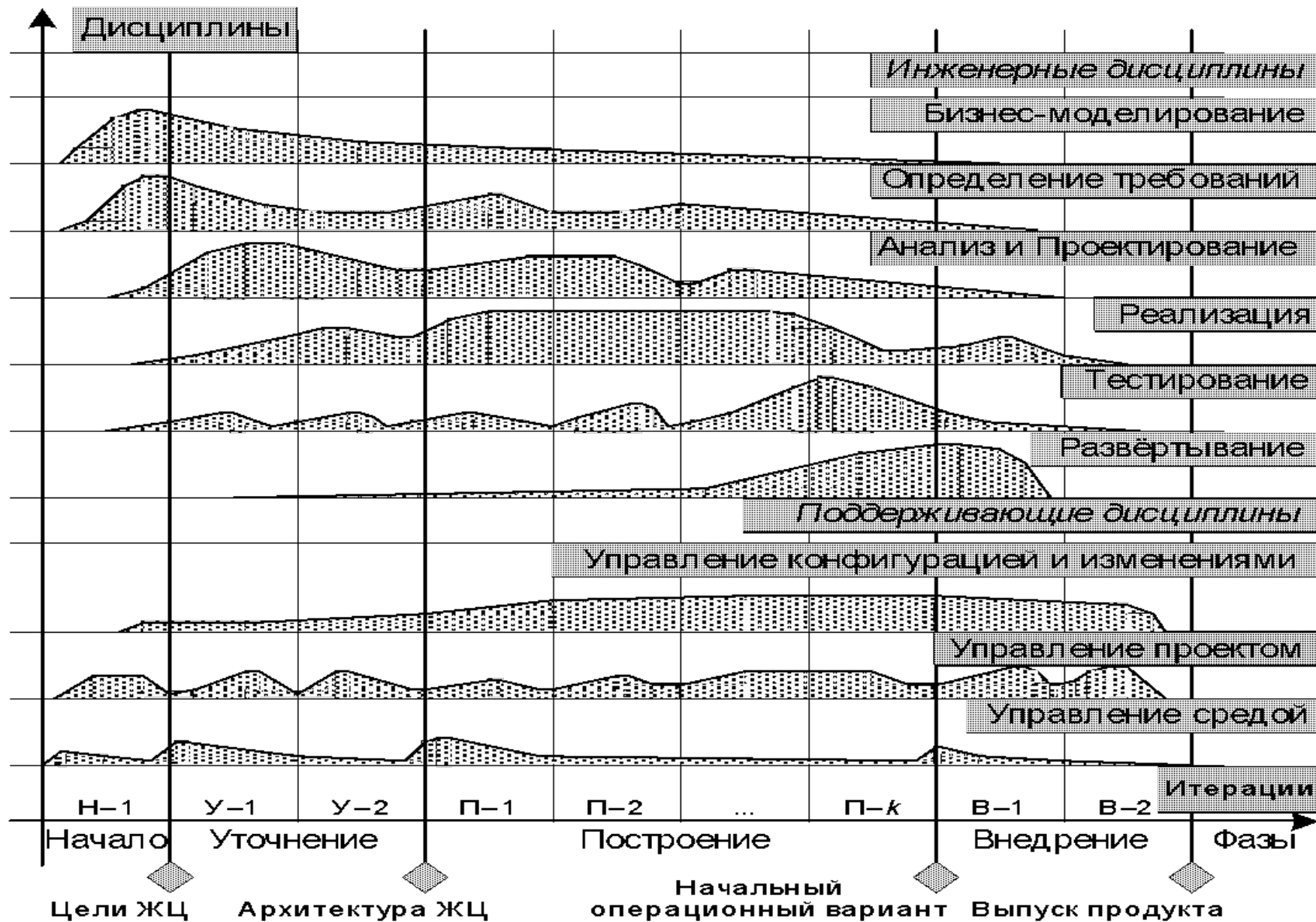
Цели — планирование, управление персоналом, обеспечения связей с другими заинтересованными лицами, управление рисками, отслеживание текущего состояния проекта

- **Управление средой проекта (Environment)**

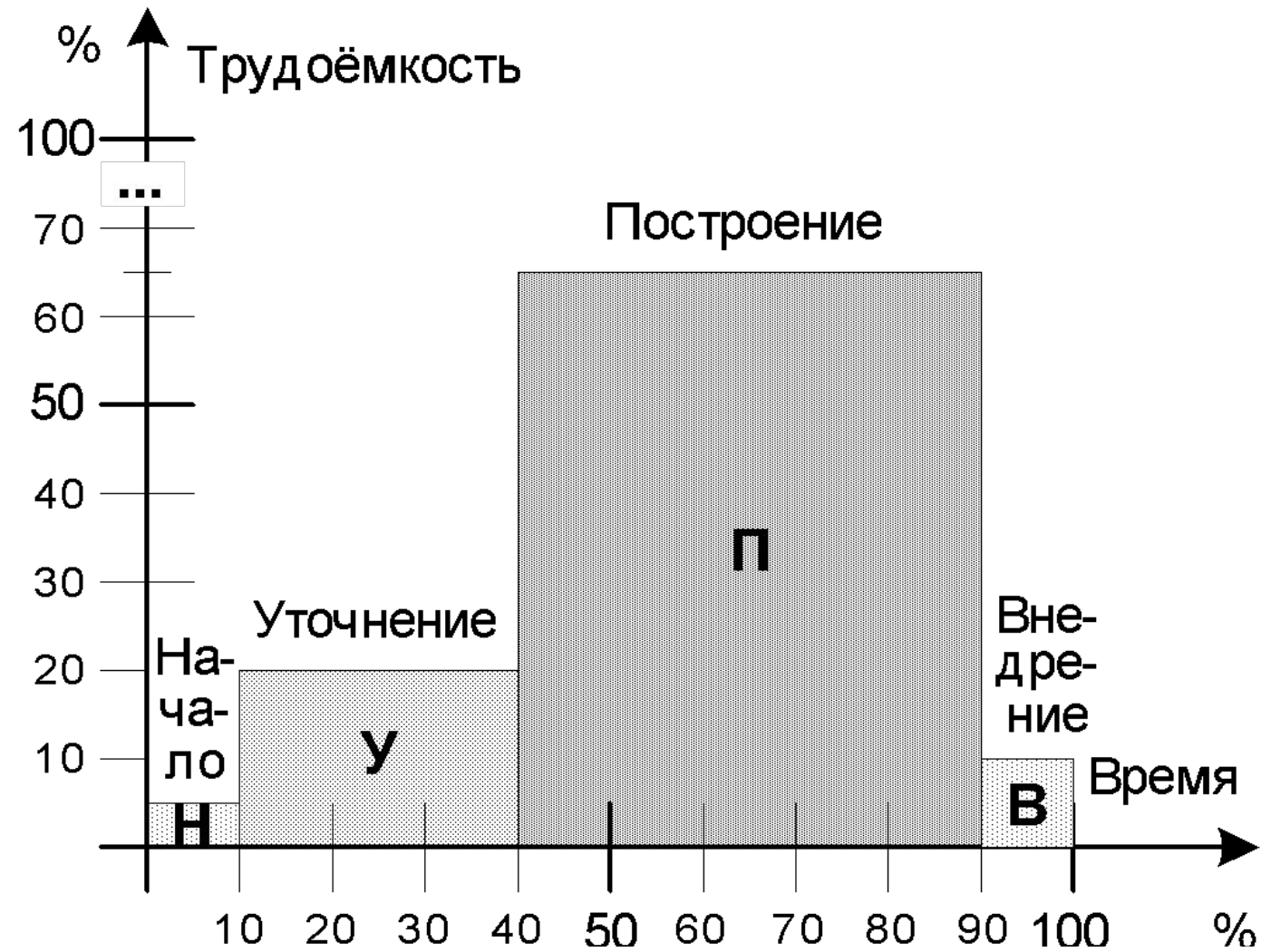
Цели — подстройка процесса под конкретный проект, выбор и смена технологий и инструментов, используемых в проекте

Первые пять деятельности считаются рабочими, остальные — поддерживающими.

Модели итеративных процессов . RUP



Трудоёмкость и затраты времени на фазы

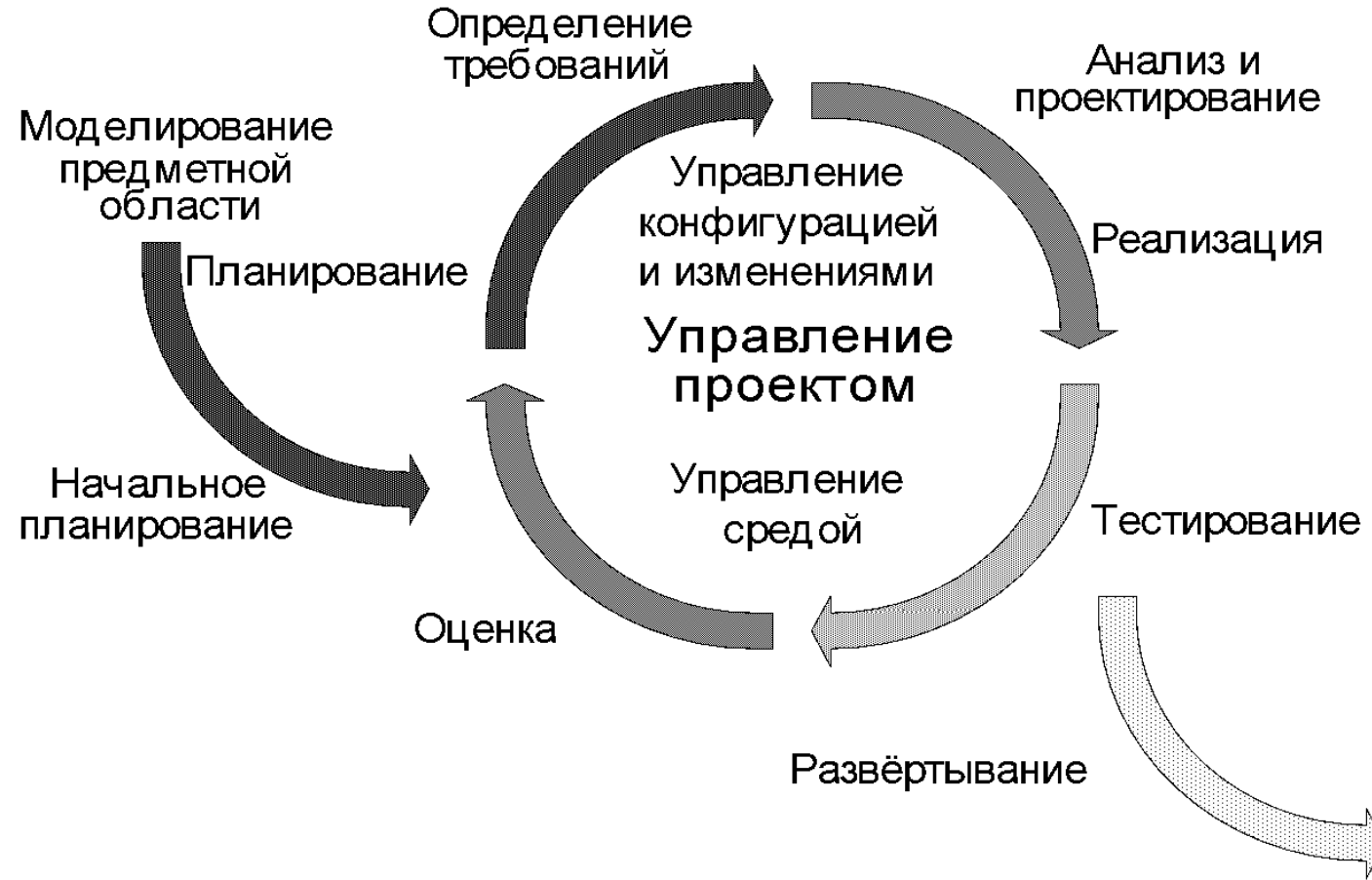


Нагрузка основных дисциплин RUP

Нагрузка основных дисциплин RUP распределяется по фазам следующим образом:

- в фазе Начала – Бизнес-моделирование и Определение требований,
- в фазе Уточнения – Анализ и проектирование и Реализация,
- в фазе Построения – Реализация и Тестирование,
- в фазе Внедрения – Развёртывание.

Итеративность разработки



В каждой итерации выполняются все дисциплины RUP, но с разной интенсивностью, зависящей от фазы, и в определённой взаимосвязи.