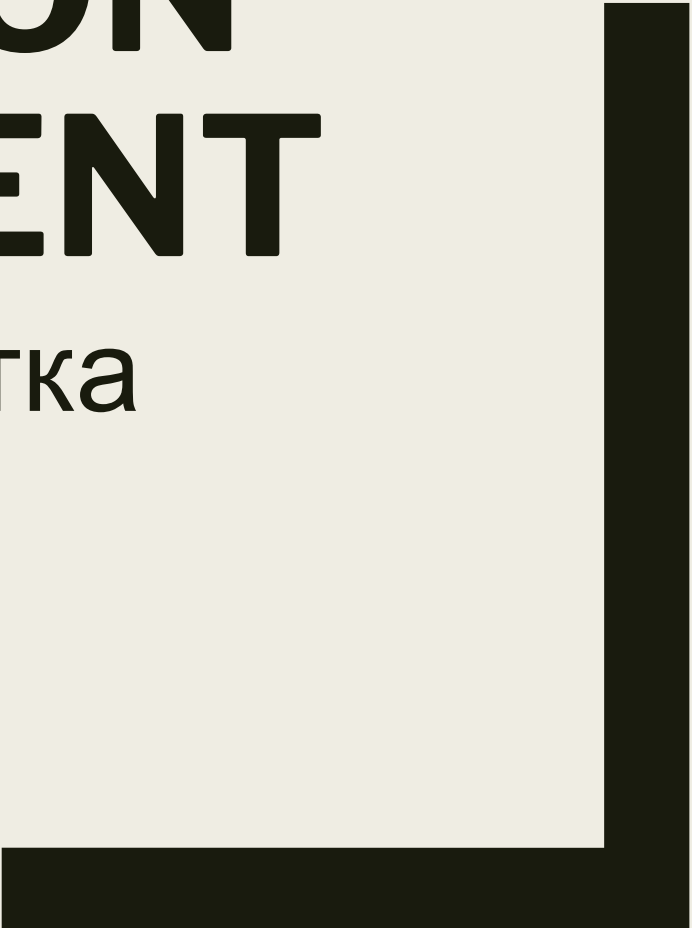


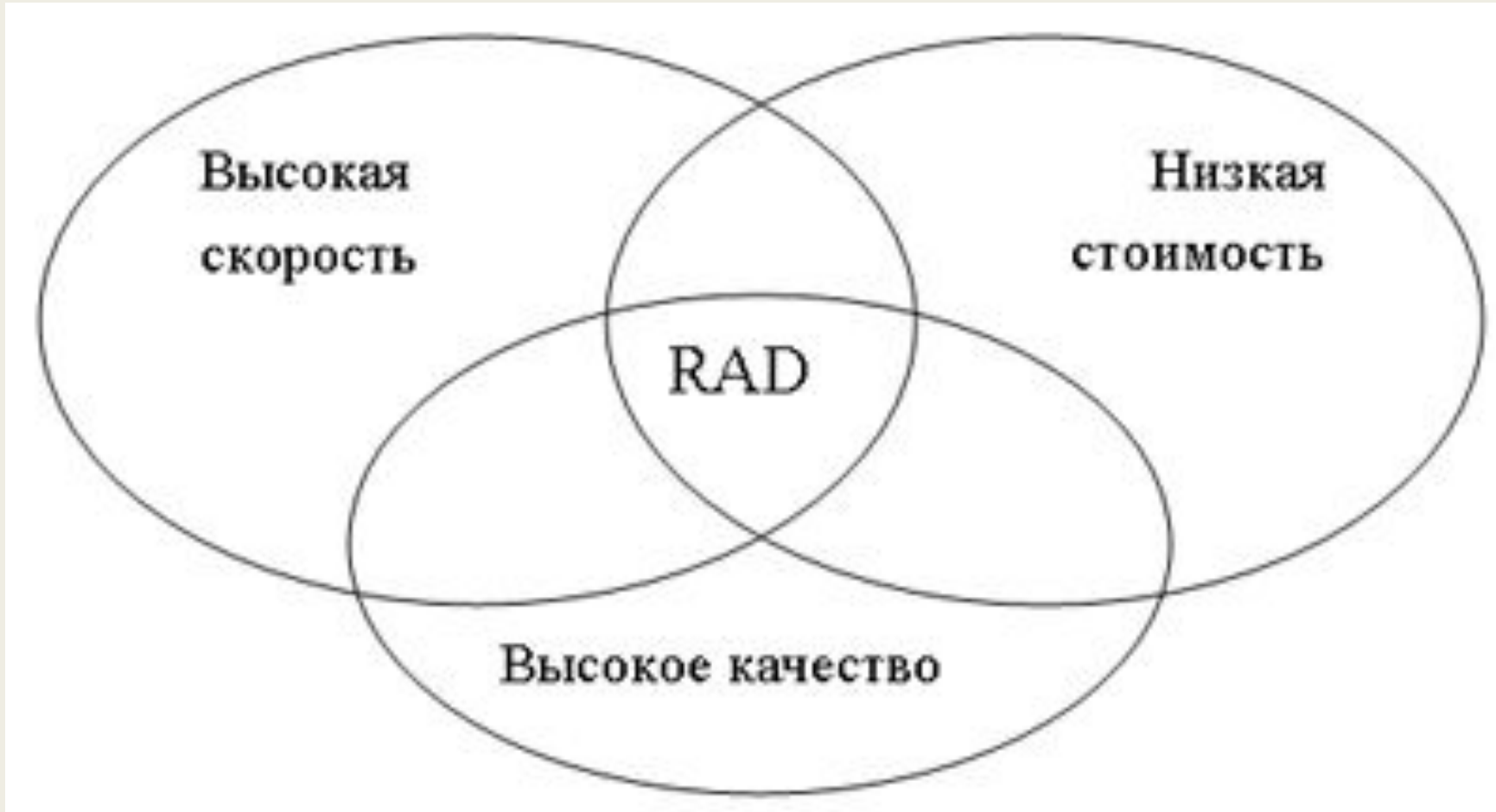


RAPID APPLICATION DEVELOPMENT

(Быстрая разработка
приложений)



Rapid Application Development (RAD) – это жизненный цикл процесса проектирования, созданный для достижения более высоких скорости разработки и качества ПО, чем это возможно при традиционном подходе к проектированию.



RAD предполагает, что разработка ПО осуществляется небольшой командой разработчиков за срок порядка трех-четырёх месяцев путем использования инкрементного прототипирования с применением инструментальных средств визуального моделирования и разработки. Технология RAD предусматривает активное привлечение заказчика уже на ранних стадиях – обследование организации, выработка требований к системе. Причины популярности RAD вытекают из тех преимуществ, которые обеспечивает эта технология.

Наиболее существенными из них являются:

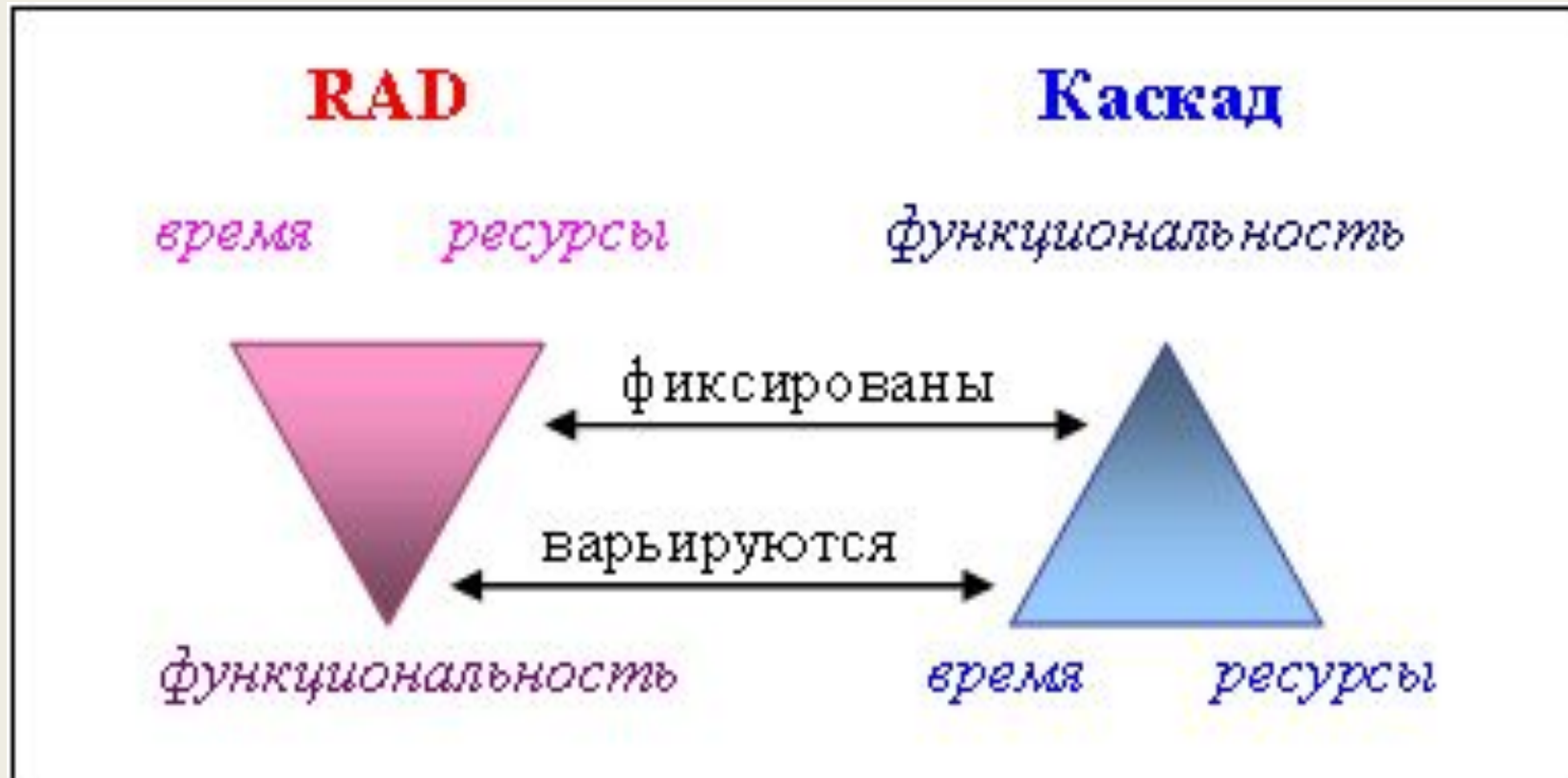
- высокая скорость разработки;
- низкая стоимость;
- высокое качество.

Когда применяется

Применение технологии **RAD** целесообразно, когда:

- требуется выполнение проекта в сжатые сроки (90 дней). Быстрое выполнение проекта позволяет создать систему, отвечающую требованиям сегодняшнего дня. Если система проектируется долго, то весьма высока вероятность, что за это время существенно изменятся фундаментальные положения, регламентирующие деятельность организации, то есть, система морально устареет еще до завершения ее проектирования.
- нечетко определены требования к ПО. В большинстве случаев заказчик весьма приблизительно представляет себе работу будущего программного продукта и не может четко сформулировать все требования к ПО. Требования могут быть вообще не определены к началу проекта либо могут изменяться по ходу его выполнения.
- проект выполняется в условиях ограниченности бюджета. Разработка ведется небольшими RAD группами в короткие сроки, что обеспечивает минимум трудозатрат и позволяет вписаться в бюджетные ограничения.
- интерфейс пользователя (GUI) есть главный фактор. Нет смысла заставлять пользователя рисовать картинки. RAD технология дает возможность продемонстрировать интерфейс в прототипе, причем достаточно скоро после начала проекта.
- проект большой, но поддается разделению на более мелкие функциональные компоненты. Если предполагаемая система велика, необходимо, чтобы ее можно было разбить на мелкие части, каждая из которых обладает четкой функциональностью. Они могут выпускаться последовательно или параллельно (в последнем случае привлекается несколько RAD групп).
- ПО не обладает большой вычислительной сложностью.

RAD-технология не является универсальной, то есть ее применение целесообразно не всегда. Например, в проектах, где требования к программному продукту четко определены и не должны меняться, вовлечение заказчика в процесс разработки не требуется и более эффективной может быть иерархическая разработка (каскадный метод). То же касается проектов, ПО, сложность которых определяется необходимостью реализации сложных алгоритмов, а роль и объем пользовательского интерфейса невелик.



Принципы организации

RAD
Главная идея RAD-технологии состоит в том, чтобы как можно быстрее донести до заказчика результаты разработки, пусть и не в полном виде. Например, реализация только пользовательского интерфейса и предъявление его заказчику позволяет уже на ранней стадии разработки получить замечания по экранным и отчетным формам и внести необходимые коррективы. В этом случае значительно возрастает вероятность успеха проекта, то есть возникает уверенность в том, что конечный продукт будет делать именно то, что ожидает заказчик.

Основные принципы RAD можно сформулировать следующим образом:

- Работа ведется группами. Типичный состав группы - руководитель, аналитик, два программиста, технический писатель. Если проект сложный, то для него может быть выделено несколько RAD-групп. Разработка проекта выполняется в условиях тесного взаимодействия между разработчиками и Заказчиком.
- Разработка базируется на моделях. Моделирование позволяет оценить проект и выполнить его декомпозицию на составные части, каждая из которых может разрабатываться отдельной RAD-группой.
- Итерационное прототипирование. Разработка системы и предъявление ее заказчику осуществляется в виде последовательности развиваемых прототипов. Любой из прототипов реализует определенную часть функциональности, требуемой от конечного продукта. При этом каждый последующий прототип включает всю функциональность, реализованную в предыдущем прототипе, с добавлением новой. Число прототипов определяется на основе учета разных параметров – размера проекта, анализа рисков, пожеланий заказчика и т. д. Традиционно для проектов ПО средней сложности разрабатываются три прототипа. Первый содержит весь пользовательский интерфейс с нулевой функциональностью. Он дает возможность собрать замечания заказчика и после их устранения утвердить у него экранные и отчетные формы. Вторым прототипом содержит реализованную на 70-80% функциональность системы, третий – полностью реализованную функциональность. Основаниями для очередной итерации являются:
 - Замечания заказчика. Привлечение заказчика и конечного пользователя к оценке выходных результатов прототипа с эффективной обратной связью с командой разработчиков является гарантией того, что созданная система будет делать то, что требуется заказчику. Если замечания носят характер исправлений, они учитываются в следующем прототипе, если же изменяются требования, то выполняется переоценка проекта и корректируются сроки и стоимость проекта.
 - Детализация. Выполняется программирование нереализованной части системы в соответствии с составленным планом.
 - Анализ результатов программирования. Исправляются ошибки, повышается эффективность программирования и т. д.

- RAD группа всегда работает только над одним прототипом. Это обеспечивает единство целей, лучшую наблюдаемость и управляемость процессом разработки, что в итоге повышает качество конечного продукта. Соответственно используемые инструментальные средства должны обеспечивать групповую разработку и конфигурационное управление проектом.
- Если проект сложный, то для него может быть выделено несколько RAD групп. Большие системы разбиваются на подсистемы. Каждая подсистема разрабатывается независимой группой. Ключ успеха – правильное разбиение системы на подсистемы. Команды должны использовать общие стандарты. Обязательно финальное тестирование полной системы.
- Обязательное использование инструментальных средств, автоматизирующих процесс разработки, и методик их использования, следствием чего является сокращение сроков разработки и повышение качества конечного продукта.

Фазы разработки

Планирование — совокупность требований, полученных при системном планировании и анализе процедуры разработки жизненного цикла (SDLC). На этом этапе пользователи, менеджеры и IT-специалисты обсуждают задачи проекта, его объём, системные требования, а также сложности, которые могут возникнуть при разработке. Фаза завершается согласованием ключевых моментов с RAD-группой и получением от руководителей проекта разрешения на продолжение.

Модель быстрой разработки приложений (RAD)

Пользовательское проектирование — на протяжении данного этапа пользователи, взаимодействуя с системными аналитиками, разрабатывают модели и прототипы, которые включают в себя все необходимые системные функции. Для перевода пользовательских прототипов в рабочие модели RAD-группа обычно использует технику объединенной разработки приложений (JAD) и CASE-инструменты. Пользовательское проектирование оказывается длительным интерактивным процессом, который позволяет пользователям понять, изменить и в конечном счёте выбрать рабочую модель, отвечающую их требованиям.

Конструирование — этап, в котором основная задача заключается в разработке программ и приложений. Аналогична стадии «реализация» в SDLC. В RAD, однако, пользователи продолжают принимать участие и по-прежнему могут предлагать изменения или улучшения в виде разработанных ими докладов. В их задачи входит программирование и разработка приложений, написание кода, интеграция модулей и системное тестирование.

Переключение — включает в себя операции по конверсии данных, тестирование, переход на новую систему и тренировку пользователей. По своим задачам напоминает финальную стадию SDLC. Сравнивая с традиционными методами разработки ПО, весь процесс оказывается сжатым по времени. Как результат, новая система оказывается быстрее построенной, доставленной до заказчика и установленной на рабочих местах.

Что обеспечивает RAD

Технология RAD обеспечивает

ТЕХНОЛОГИЯ

- быстроту продвижения программного продукта на рынок;
- интерфейс, устраивающий пользователя;
- легкую адаптируемость проекта к изменяющимся требованиям;
- простоту развития функциональности системы.