

Лекция 5

Технология разработки ПО

**О.В. Федорова, доцент каф.
ПМИ**

Технологический подход

- Конкретная технология (технологический подход) содержит в себе определённый набор процессов, а также используемых в них знаний, методов и средств.
- В узком смысле технология представляет собой определённый технологический подход. **Технологии опираются на понятие жизненного цикла.**

Жизненный цикл программного обеспечения (ЖЦ ПО)

– весь период его разработки и эксплуатации, начиная с момента возникновения замысла (идеи) и заканчивая прекращением всех видов его использования.

В общем случае ЖЦ определяется *моделью* и описывается в форме технологии разработки – *технологического подхода*.

Модель ЖЦ – структура, определяющая последовательность выполнения процессов и их взаимосвязь на протяжении ЖЦ. Упоминание ЖЦ обычно подразумевает указание конкретной модели ЖЦ.

Технология разработки ПО

Технология разработки ПО (технологический подход) – это определённая совокупность процессов, включающих их детальное содержание и распределение по стадиям, а также ролевую ответственность участников проекта на всех стадиях выбранной модели ЖЦ ПО.

Технология часто определяет и саму модель.

Обычно она основывается на методиках выбранной методологии, а также рекомендует практики, что позволяет максимально эффективно воспользоваться этой технологией и её моделью ЖЦ.

Технологию удобно характеризовать в двух измерениях – **вертикальном (процессы)** и **горизонтальном (стадии)**.

Действие

Связующим понятием между процессами и стадиями является «действие».

Действие (тж. работа, вид деятельности) – часть деятельности по проекту, выполняемая отдельным исполнителем или группой исполнителей.

Фактически процессы и стадии представляют собой определённые наборы действий: **по признаку преобразования данных действия объединяются в процессы, а по временному признаку и/или получаемому результату – в стадии.**

Процесс

- *Процесс – совокупность взаимосвязанных действий проекта, преобразующих некоторые входные данные в выходные.*
- Взаимосвязь действий заключается в их последовательности, завершённой с точки зрения содержания, временной и логической очередности.
Процессы состоят из набора действий, а каждое действие – из набора задач.
Дальнейшая детализация приводит к рассмотрению отдельных операций. Задача (тж. задание) – планируемый элемент действия: задача определяется/ в плане проекта и её могут быть назначены ресурсы для выполнения.
- Таким образом, иерархия понятий, связанных с процессом, выглядит следующим образом:
Процессы → Действия → Задачи → Операции.

Дисциплина

В некоторых подходах вместо понятия «процесс» используют понятие «дисциплина».

Дисциплина – процесс, рассматриваемый вместе с соответствующими ему артефактами и ролями.

С точки зрения управления дисциплина представляет собой поток работ (букв. рабочий поток), связанный с рабочим продуктом – артефактом, производимым участником в некоторой заданной роли.

Кроме того, в ряде работ вводится также понятие «процедура».

Процедура – пошаговое описание направления задач для выполнения и завершения конкретного действия.

В этом случае **описание процесса представляет собой документированное определение действий, формализованных в виде процедур.**

Стадия

Стадия – группа действий проекта, ограниченная некоторыми временными рамками и часто заканчивающаяся выпуском произведённого результата, определяемого заданными требованиями.

Стадии выделяются исходя из соображений разумного и рационального управления проектом.

Стадии часто состоят из этапов (тж. шаг), которые обычно имеют итерационный характер и поэтому представляются в виде итераций.

В ряде подходов **стадии объединяют в более крупные временные рамки – фазы**, в этом случае сами стадии имеют итерационный характер.

Таким образом, иерархия понятий, связанных со стадией, выглядит следующим образом:

Фазы → Стадии → Этапы.

Получаем следующее описание измерений технологии

- **Вертикальное измерение отражает статические аспекты процессов и оперирует такими понятиями, как *процессы, действия, задачи и операции.***
- **Горизонтальное измерение представляет время, отражает динамические аспекты процессов и оперирует такими понятиями, как *фазы, стадии и этапы.***

Методика и практика

Методика (букв. техника) – совокупность конкретизированных методов разработки в рамках заданной методологии, применяемая в одном или нескольких соответствующих этой совокупности процессах ЖЦ.

Практика – это определённая рекомендация по выполнению действий, для которых результаты проверяемы, но не передаваемы как материал для работы других процессов. Последовательность практик и последовательность действий внутри практики не задана.

Практики не привязаны к проекту (при этом говорят, что у них нет «экземпляра»).

Таким образом, **технология определяется спецификой комбинации процессов и стадий, ориентированной на разные классы ПО и особенности участников проекта и дополненной методиками и практиками.**

Управление разработкой

Управление разработкой заключается в организации проекта с учётом заданных ограничений.

Система ограничений определяется совокупностью приоритетов и должна учитывать требования заинтересованных лиц.

К основным ограничениям относят:

- содержание (проекта),
- время (выполнения),
- стоимость (проекта).

Следует отметить и следующие ограничения:

- ресурсы (людские и финансовые),
- качество (приемлемое для проекта),
- эффективность (проекта).

В результате при учёте ограничений получается некоторый многоугольник или многогранник ограничений.

На практике обычно 3 ограничения оказываются фиксированными. При этом говорят о треугольнике ограничений.

Проблема управления

- Выбор технологического подхода, связанного с совокупностью процессов и стадий, сочетанием методик и практик, а также с уровнем формализации разработки, позволяет урегулировать треугольник ограничений.
- *Поэтому проблема управления заключается в поиске баланса в рамках системы ограничений проекта.*

Эффективное управление разработкой ПО

Эффективное управление разработкой ПО требует определённых формализаций выполняемых работ и их результатов. *Произведённый результат* – определённый реальный результат, произведённый проектом.

Различают следующие произведённые результаты:

- *внутренний* – для использования проектной командой,
- и *внешний* – для использования другими заинтересованными лицами.

Артефакт

Формализацией произведённого результата является некоторый артефакт.

Примеры: модели и документы (планы, спецификации, диаграммы, код и т.п.).

Артефакт – формальный произведённый результат, создаваемый, изменяемый или используемый в нём при выполнении проекта.

Артефакт определяет область ответственности: за его создание отвечают определённые исполнители или их группы.

В рамках отдельной задачи (и таймбокса) таким результатом является *рабочий продукт*, но в ряде подходов это синоним артефакта.

Базовая линия

Для формализации внесения изменений в произведённый результат используется понятие «базовая линия».

Базовая линия – официально принятый вариант произведённого результата, обозначенный и зафиксированный в конкретный момент времени ЖЦ. Изменения, вносимые в базовую линию, должны быть предварительно утверждены, т.е. должны пройти через специальное формализованное действие. *Для проекта в целом базовая линия обычно переводится как базовый план – исходный план проекта с утверждёнными изменениями.*

Для анализа проекта используются контрольные точки и вехи.

Контрольная точка – событие в ЖЦ для проверки выполненной и оценки оставшейся работы по проекту.

Моментами времени для контрольных точек часто выступают границы стадий.

Веха – событие ЖЦ для обозначения завершения произведённого результата или их набора. Вехи часто используются в качестве контрольных точек.

Итерация

Вследствие итерационного выполнения работ получаемые результаты постепенно улучшаются до целевых результатов, диктуемых заданными требованиями, которые также могут изменяться.

Итерация – ограниченная во времени повторяемая часть проекта. Итерацией может выступать весь цикл разработки или его часть, что определяется их длительностью и моделью ЖЦ. В этом случае итерация цикла разработки называется макроитерацией или просто *циклом*, итерация стадии – просто *итерацией*, а итерация этапа – микроитерацией.

Для управляемого выполнения отдельных задач используются таймбоксы.

Таймбокс (букв. временной ящик) – небольшой промежуток времени для выполнения конкретной задачи и производства её результатов.

Таймбокс характеризуется жёстко заданными временными рамками. Рамки нужно соблюсти, даже если придётся выдать не вполне завершённые результаты. Иначе задача считается проваленной и приходится или отказаться от неё, или запланировать её повторно.

Таким образом, описание измерений технологии корректируется следующим образом:

Вертикальное измерение оперирует также такими понятиями, как:

- произведённые результаты (артефакты, рабочие продукты)
- и исполнители (роли и ответственности).

Горизонтальное измерение оперирует также такими понятиями, как:

- контрольные точки и вехи,
- итерации
- таймбоксы.

Однако эти понятия можно рассматривать и как характеристику нового, ***третьего измерения***. Это измерение оказывается связанным с ***формализацией, необходимой для управления проектом***.

Существует два основных набора технологических процессов.

Классический набор – совокупность основных процессов, сложившихся исторически в результате практического опыта разработки ПО.

Классический набор включает 9 процессов:

1. Исследование;
2. Управление;
3. Анализ;
4. Проектирование;
5. Кодирование;
6. Тестирование;
7. Ввод в действие;
8. Сопровождение;
9. Снятие с эксплуатации.

Процессы классического набора фактически являются подмножеством стандартного, выступая там как процессы или действия процессов.

Стандартный набор – совокупность процессов из **ISO/IEC 12207:1999** «Информационная технология – Процессы жизненного цикла ПО».

Стандартный набор включает **3 группы процессов:**

- основные,
- вспомогательные,
- организационные процессы.

Существует два основных вида формирования технологических стадий

- *Попроцессное формирование стадий –* выделение стадий, отражающих названия процессов (или отдельных действий), большая часть времени которых проходит в данной стадии.
- *Пофазное формирование стадий –* объединение стадий в фазы, отражающие крупные временные промежутки.

Попроцессное формирование стадий

Попроцессное формирование стадий обычно используют для классических процессов (или их под- или надмножества).

В подходах с этой классификацией обычно выделяют

9 стадий:

1. Исследование идеи;
2. Планирование;
3. Анализ требований;
4. Проектирование;
5. Кодирование;
6. Тестирование и отладка;
7. Ввод в действие;
8. Эксплуатация и сопровождение;
9. Снятие с эксплуатации.

Пофазное формирование стадий

Пофазное формирование стадий обычно используют для стандартных процессов (или их под- или надмножества).

В большинстве подходов с этой классификацией выделяют **4 основные фазы:**

1. Начало;
2. Середина;
3. Кульминация;
4. Переход.

В ряде подходов выделяют **2 дополнительные фазы:**

5. Работа;
6. Окончание.

Характеристики выполняемых проектов

Классификация подходов тесно связана с характеристиками выполняемых проектов.

- По каждому признаку классификации проектов можно выделить множество проектов, для которых будут ***указаны только граничные значения.***
- ***По масштабу***, определяющему количество исполнителей и протяжённость (время выполнения) проекта, выделяют ***5 категорий проектов*** (табл.5.1).

5 категорий проектов

Категория	Число исполнителей	Протяжённость проекта
мелкий	от 1 до 3	от 1 часа до 2 месяцев
малый	от 3 до 10	от 2 до 6 месяцев
средний	от 10 до 30	от 6 месяцев до 1 года
крупный	от 30 до 100	от 1 года до 2 лет
гигантский	от 100 до 300 и более	от 2 до 6 лет и более

В настоящее время выделяют два класса подходов

В настоящее время выделяют два класса подходов.

Строгие (тж. *тяжёлые, жёсткие*) подходы ориентированы в основном на применение в средних, крупных и гигантских проектах с фиксированным объёмом работ. Поэтому **основное требование к таким проектам – предсказуемость.**

Гибкие (тж. *лёгкие, живые*) подходы ориентированы в основном на применение в мелких, малых или средних проектах в случае неясных или изменяющихся требований к системе. Поэтому **основное требование к таким проектам – непосредственное участие заказчика в проекте.** Для большинства гибких подходов важным является **требование адаптируемости.**

Внутри классов подходов принято условно выделять группы подходов с рядом принципов, общих для этих подходов.

К классу строгих подходов относят:

- каскадные,
- каркасные,
- генетические,
- формальные подходы.

к классу гибких:

- эволюционные
- адаптивные подходы.