

В. В. Шилов



ВВЕДЕНИЕ В ПРОГРАММНУЮ ИНЖЕНЕРИЮ

Команда разработчиков

Москва, 27 апреля 2017 года

Управление программным проектом включает решение трех основных задач:

1. Подбор команды и управление командой.
2. Выбор процесса.
3. Выбор инструментальных средств.

Для успеха проекта одинаково важны все три задачи, но едва ли не ключевую роль играют правильный **подбор членов команды** и **управление командой**.



Сандро Боттичелли. Суд Париса



Успех проекта во многом зависит от того, удастся ли состав участников проекта преобразовать в команду единомышленников!



Команда должна быть:



Три аспекта управления командой:

1. Ролевая модель команды.
2. Модель организации команды.
3. Общение в команде.

1. Ролевая модель команды.

Состав команды определяется:

- опытом и уровнем коллектива,
- особенностями проекта,
- применяемыми технологиями.

1. Ролевая модель команды.

“Классический” вариант состава команды включает:

- Менеджер проекта.
- Проектировщик.
- Разработчик.
- Тестировщик.
- Инженер по качеству.
- Технический писатель.
- Технолог разработки ПО.



1. Ролевая модель команды.

Менеджер проекта – главное действующее лицо, обладающее знаниями и навыками, необходимыми для успешного управления проектом.

Основные функции:

- Подбор и управление кадрами.
- Подготовка и исполнение плана проекта.
- Руководство командой.
- Обеспечение коммуникации между подразделениями.
- Обеспечение готовности продукта.

1. Ролевая модель команды.



Руководство:
два результата



1. Ролевая модель команды.

Проектировщик – функция проектирования архитектуры высокого уровня и контроля ее выполнения.

Основные функции:

- Анализ требований.
- Разработка архитектуры и основных интерфейсов.
- Участие в планировании проекта.
- Контроль выполнения проекта.
- Участие в подборе кадров.

В небольших командах функция распределяется между менеджером и разработчиками. В больших проектах это может быть целый отдел.

1. Ролевая модель команды.



От замысла к воплощению...



1. Ролевая модель команды.

Разработчик – роль, ответственная за непосредственное создание конечного продукта.

Основные функции:

- Программирование (кодирование).
- Контроль архитектурных и технических спецификаций продукта.
- Подбор инструментов разработки, метрик и стандартов; контроль их использования.
- Диагностика и разрешение технических проблем.
- Контроль за работой разработчиков документации, тестировщиков, технологов.
- Мониторинг состояния продукта (ведение списка ошибок).

1. Ролевая модель команды.

Тестировщик – роль, ответственная за удовлетворение функциональных и нефункциональных требований к продукту.

Основные функции:

- *Составление плана тестирования.* Является одним из элементов проекта, составляется до начала реализации (разработки) проекта. Время, отводимое в плане на тестирование, может быть сопоставимо с временем разработки.
- *Контроль выполнения плана.* Важнейшая функция контроля – поддержка целостности базы данных зарегистрированных ошибок. В БД регистрируется: кто, когда и где обнаружил, описание ошибки, описание состояния среды; статус ошибки: приоритет, кто разрешает; состояние ошибки: висит, в разработке, разрешена, проблемы. БД должна быть доступна всем, т.к. в тестировании принимают участие все члены команды.

1. Ролевая модель команды.

- *Разработка тестов.* Самая трудоемкая часть в работе тестировщика. Тестирование должно обеспечить полную проверку функциональности при всех режимах работы продукта.
- *Автоматизация тестирования.* Включает автоматизацию составления тестов, автоматизацию пропуска тестов и автоматизацию обработки результатов тестирования. Иногда в команду вводится новый участник – инженер по автоматизации.
- *Выбор инструментов, метрик, стандартов для организации процесса тестирования.*
- *Организация бета-тестирования.* Тестирование почти готового продукта внешними тестерами (пользователями). Эта процедура должна быть продумана и организована при разработке коробочного продукта.

1. Ролевая модель команды.



Тестирование?

Инженер по качеству.

Сегодня обычно рассматриваются три аспекта (уровня) качества:

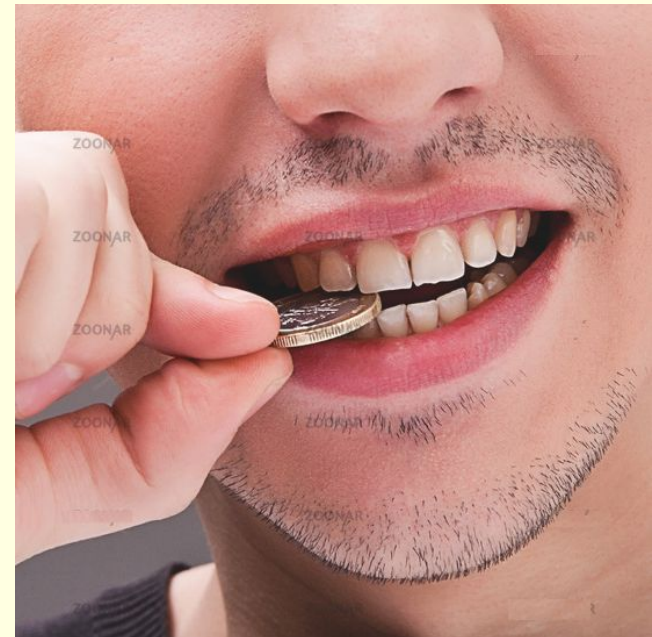
- Качество конечного продукта (обеспечивается тестированием).
- Качество процесса разработки (для повышения качества продукта надо повысить качество процесса разработки).
- Качество организации работ (для повышения качества процесса надо повысить качество организации работ).

Иногда функции инженера по качеству возлагаются на тестировщика. На самом деле его функции шире – это два следующих уровня качества.

1. Ролевая модель команды.

Основные функции:

- Составление плана качества. Он включает все мероприятия по повышению качества (на всех уровнях). Имеет долгосрочный характер. План тестирования – его оперативная составляющая.
- Описание (формализация) процессов. При описании вводятся метрики процесса, влияющие на качество продукта.
 - Оценка процессов включает регистрацию хода выполнения процессов и оценку значений установленных метрик процессов. Выявление “слабых” мест, выработка рекомендаций по улучшению процессов.
 - Улучшение процессов – их переопределение, автоматизация части работ, обучение персонала.



1. Ролевая модель команды.

Технический писатель или разработчик пользовательской (и иной) документации как части программного продукта.

Основные функции:

- Разработка плана документирования. Он включает: состав, сроки подготовки и порядок тестирования документов.
- Выбор и разработка стандартов и шаблонов подготовки документов.
- Выбор средств автоматизации документирования.
- Разработка документации.
- Организация тестирования документации.
- Участие в тестировании продукта.

Технический писатель все время работает с готовыми версиями продукта и, выступая как бы от имени пользователя, обнаруживает все недочеты и несоответствия.

1. Ролевая модель команды.



Но... без документации нельзя!



Технолог разработки ПО.

Основные функции:

- Поддержка модели ЖЦ – создание служб и структур по поддержке работоспособности принятой модели ЖЦ ПО. Участвуют все, но контроль возложен на технолога.
- Создание и сопровождение среды сборки продукта. На завершающих этапах разработки сборка проводится достаточно часто (иногда ежедневно). Среда сборки должна быть подготовлена заранее, сборка должна проводиться быстро и без сбоев. С учетом сборки версий это не простая задача.
- Создание и сопровождение процедуры установки с тем, чтобы каждая сборка устанавливалась автоматически с учетом версии и конфигураций сред.
- Управление исходными текстами – сопровождение и администрирование системы управления версиями исходных текстов.

1. Ролевая модель команды.

Это были основные функциональные роли в команде (ролевая модель команды).

Выделенные позиции не обязательно представлены конкретными людьми!

В небольших командах роли могут совмещаться.

В больших могут выделяться группы или отделы (отдел проектирования, отдел тестирования, отдел контроля качества, отдел подготовки документации и др.).

Ролевые модели команд могут быть самыми разнообразными!

Как организовать работу команды?

- Команды из 8 человек и команды из 400 человек?
- Есть ли различия и в чем они состоят?
- Надо ли организовывать работу по жесткой технологии или надо предоставить свободу действий?
- Можно ли найти методологию (технологию) выполнения проекта, гарантированно обеспечивающую успех?

Анализ опыта выполнения проектов показывает:

- Практически любую методологию можно с успехом применять в каком-нибудь проекте.
- Любая методология может привести к провалу проекта.

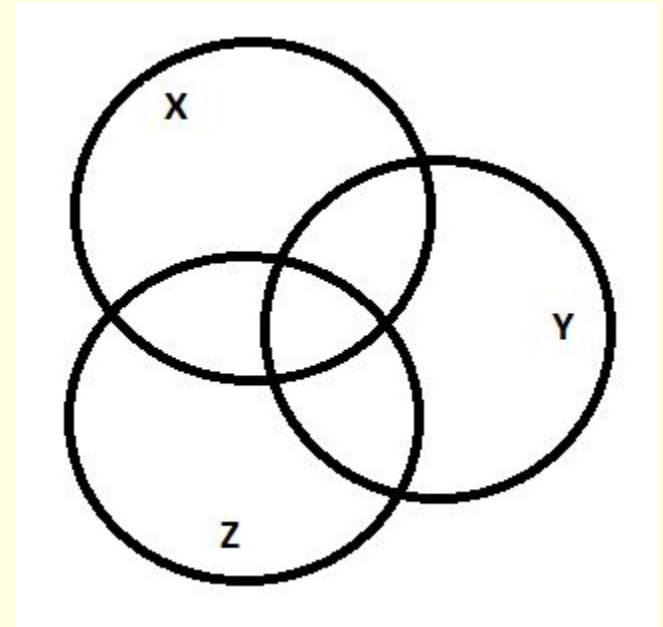
Причина этого – в людях:

- Различаются по характеру, темпераменту, активности, целям.
- Различаются по типу: индивидуалисты – члены команды; генераторы идей – исполнители; ответственные – безответственные.

2. Модель организации команды.

Три основные модели управления командой:

- Административная модель (теория X).
- Модель хаоса (теория Y).
- Открытая архитектура (теория Z).



2. Модель организации команды.

Административная модель.

Характерные черты:

Властная пирамида – решения принимаются сверху-вниз.

Четкое распределение ролей и обязанностей.

Четкое распределение ответственности.

Следование инструкциям, процедурам, технологиям.

Роль менеджера: планирование, контроль, принятие основных решений.

Преимущества модели: ясность, простота, прогнозируемость.

Недостатки модели: административная система стремится к самосохранению (стабильности), плохо восприимчива к изменению ситуации – новые типы проектов, применение новых технологий, оперативная реакция на изменение рынка. В ней плохо уживаются индивидуалисты и генераторы идей.

2. Модель организации команды.

Модель хаоса.

Характерные черты:

Отсутствие явно выраженных признаков власти.

Менеджер ставит задачу, обеспечивает ресурсами, не мешает и следит, чтобы не мешали другие.

Отсутствие инструкций и регламентированных процедур.

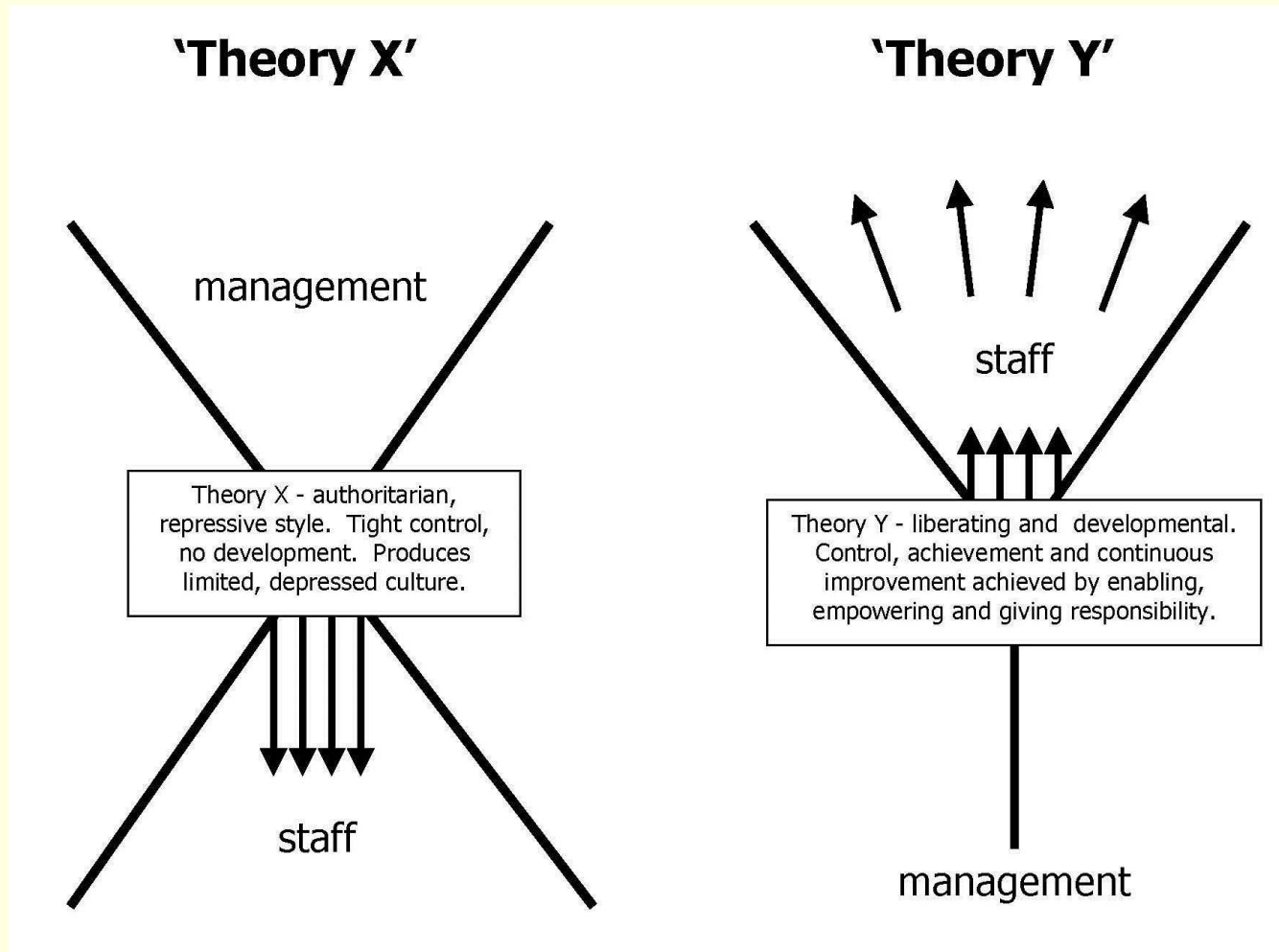
Индивидуальная инициатива – решение по проблеме принимается там, где проблема обнаружена.

Основа процесса – “дружеская соревновательность”.

Преимущества модели: творческая инициатива участников ничем не связана и потенциал участников в полной мере раскрывается.

Недостатки модели: при определенных условиях команда прорыва может стать командой провала.

2. Модель организации команды.



2. Модель организации команды.



Дуглас МакГрегор

(Douglas McGregor, 1906-1964) – американский социальный психолог.

Автор Теории X (Theory X) и Теории Y (Theory Y), в которых подвел под факторы мотивации рациональную и приемлемую основу.

Douglas McGregor.

Human Side of Enterprise // Management Review. 1957. № 11. Pp. 41-49.

2. Модель организации команды.

Открытая архитектура. Основана на модели Z.

Характерные черты:

Адаптация к условиям работы – если делаем независимые модули, то расходимся и делаем, если нужна архитектура базы данных, то собираемся и обсуждаем идеи.

Коллективное обсуждение проблем, выработка консенсуса и принятие решения, обязательного для всех.

Распределенная ответственность – отвечают все, кто обсуждал, выработывал, принимал решение.

Динамика состава рабочих групп в зависимости от текущих задач.

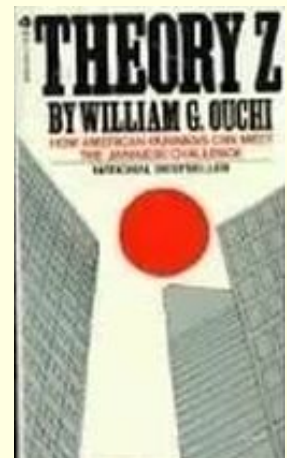
Отсутствие специализации – участники меняются ролями и функциями и могут при необходимости заменить друг друга.

Роль менеджера – активное (но не руководящее) участие в процессе, контроль конструктивности обсуждений, обеспечение возможности активного участия всех.

2. Модель организации команды.

Открытая архитектура является более гибкой, адаптируемой, настраиваемой на ситуацию. Позволяет проявить себя всем членам команды – в ней могут уживаться и индивидуалисты и коллективисты. Коллективное обсуждение высказанных идей позволяет оставлять только прагматичные идеи.

Уильям Оучи
(William G. Ouchi, р. 1943) –
американский профессор,
автор Теории Z (Theory Z).



**Theory Z: How American Business Can Meet the Japanese Challenge.
Perseus Publishing, 1981.**

Коммуникации.

Принятие решений – компромисс и консенсус.

Компромисс – соглашение, достигнутое посредством взаимных уступок. Это среднее решение, которое может оказаться (и, как правило, оказывается) хуже каждого из вариантов. Достигается путем взаимных уступок (мы согласимся с вашим вариантом интерфейса, если вы согласитесь с нашей организацией базы данных). Может быть принят большинством путем голосования.

Консенсус (коллективное мнение) – общее для конкретной группы мнение. Оптимальное решение, сочетающее лучшее из предложенных вариантов. Достигается путем обсуждения, анализа и генерации новых идей. Принимается общим согласием (все согласны, что найдено лучшее решение).

**СПАСИБО
ЗА ВНИМАНИЕ!**