

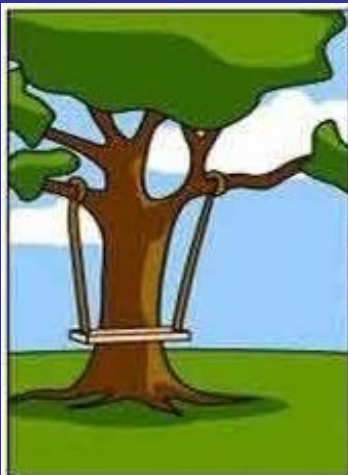
Стратегический менеджмент при разработке программного обеспечения

Михаил Елашкин
Elashkin Research

Программные проекты: как это часто бывает



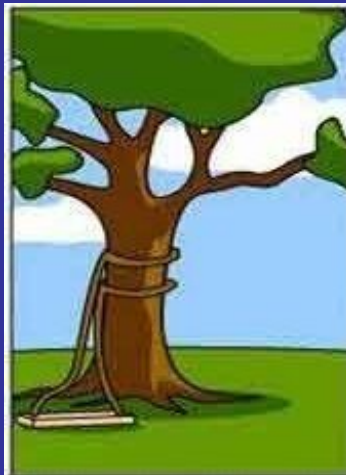
Так объяснил
заказчик



Так понял менеджер
проекта



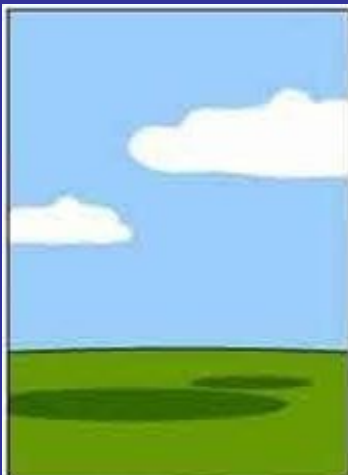
Так описал аналитик



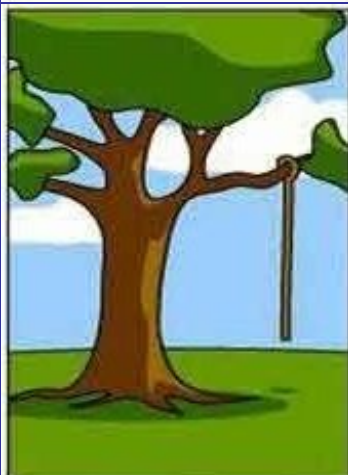
Так реализовал
программист



Так презентовал
проект менеджер



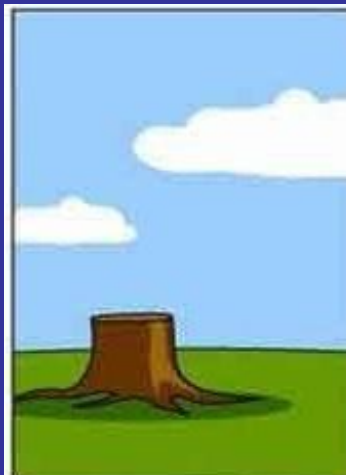
Такой оказалась
документация



Таким оказался
продукт



Такими оказались
затраты



Такой оказалась
работоспособность

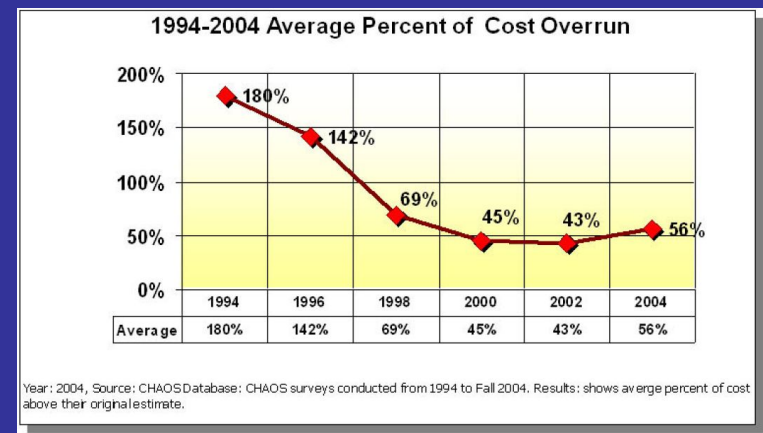
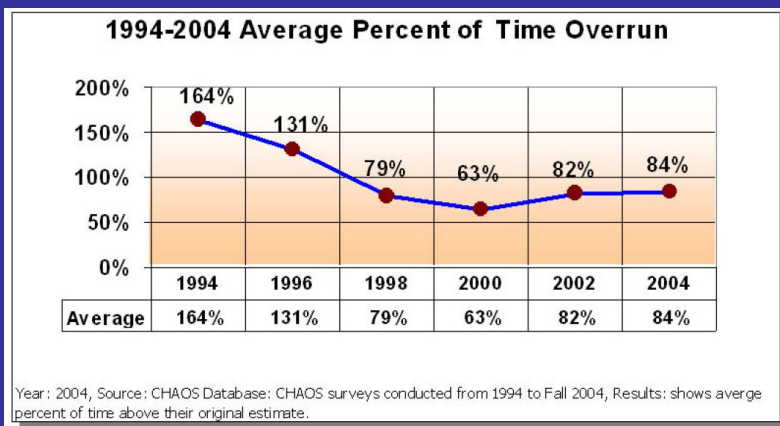
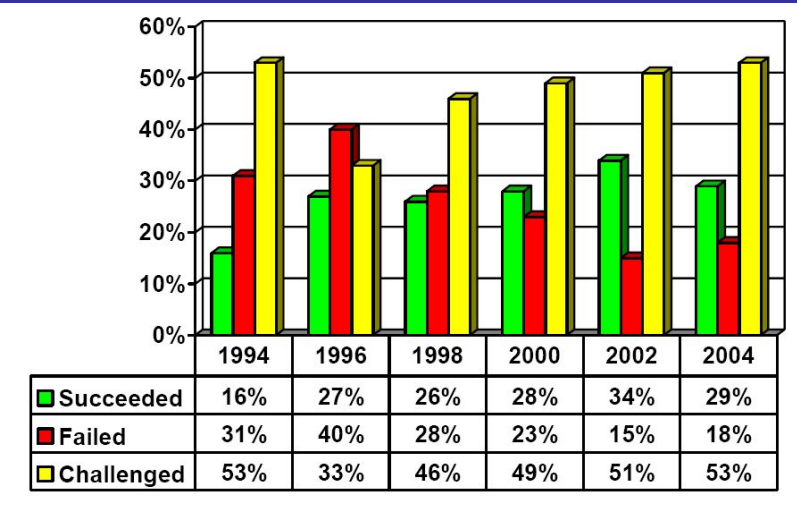


Чего хотел заказчик
на самом деле ...

Проблемы разработки

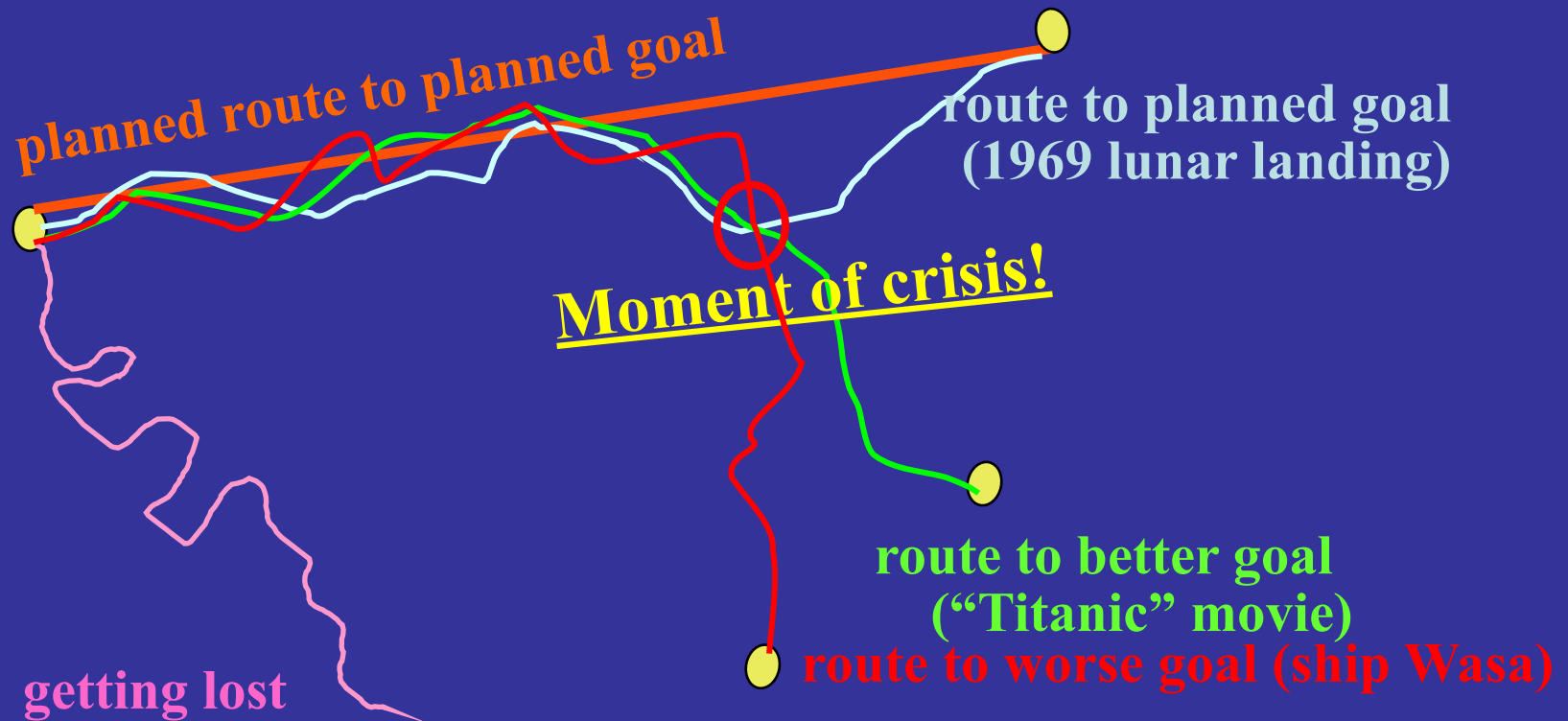
- Масштабы решаемых бизнес-задач (значимость ИТ для бизнеса)
- Сложность современных прикладных систем
 - Архитектурная
 - Технологическая
 - Прикладная (бизнес-функциональность)
- Качество создаваемых приложений и его оценка
- Скорость реализации проектов (продуктивность)
- Изменяющиеся требования/приоритеты
- Взаимодействие между всеми “сторонами” ИТ-проектов
 - Между конечными пользователями и “айтишниками”
 - Между бизнес-аналитиками и разработчиками
 - Между разработчиками и тестировщиками
 - ...

“Хаос” ИТ-проектов (2004)





Часто дела идут не по плану.
Как узнать, что правильно, а что нет?



Топ 10 причин успешности проектов

1. User Involvement
2. Executive Management Support
3. Clear Business Objectives
4. Experienced Project Manager
5. Minimizing Scope and Requirements
6. Iterative and Agile Process
7. Skilled Resources
8. Formal Methodology
9. Financial Management
10. Standard Tools and Infrastructure

Какие бывают ИТ-проекты

- Создание новых систем
- Интеграция существующих систем
- Настройка и адаптация “готовых” систем

Какие бывают ИТ-проекты

- Создание новых систем
- Интеграция существующих систем
- Настройка и адаптация “готовых” систем

Является ли поддержка и эксплуатация системы самостоятельным проектом?

Рамки проектов (constraints)

- Содержание/функциональность (scope)
- Сроки (schedule)
- Качество (quality)
- Бюджет (budget)

Конфликты/компромиссы/риски в рамках проектов

Компромиссы
(compromise)

Заказчик (бизнес)

- Содержание (scope)
- Сроки (schedule)
- Качество (quality)
- Бюджет (budget)

Подрядчик (ИТ)

- Приоритеты (priorities)
- Время (time)
- Продуктивность
- Стоимость (cost)

Риски (risks)

Роль ограничений в проектах



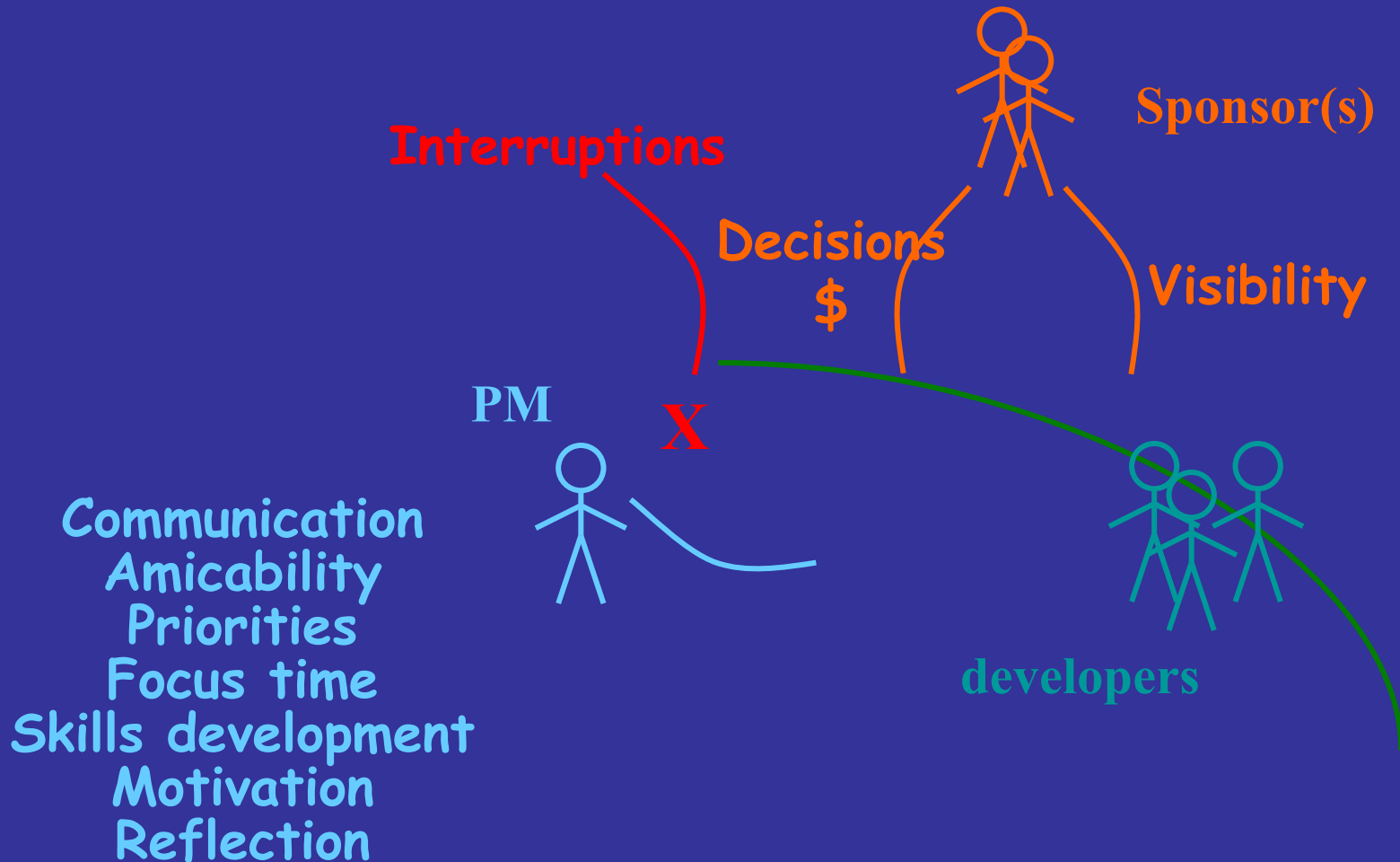
Потребности заказчика

- Соответствие ожиданиям
 - Функциональные требования
 - Удобство использования
 - Качество
 - Производительность
- Гарантированность достижения результата
 - Бюджет
 - Сроки
 - Ресурсы
- Сохранение инвестиций
 - Интеграция существующих приложений
 - Использование существующей культуры и навыков

Инструменты подрядчика

- Сохранение инвестиций
 - Интеграция и повторное использование компонентов существующих систем
 - Использование полученных навыков
- Эволюция вместо революции
 - Модульное наращивание функциональности
 - Прозрачность интеграции <существующих и новых> систем
- Унификация как инструмент снижения издержек
 - Форматов обмена информацией
 - Инструментальных средств и связующего ПО
- Повышение эффективности
 - Улучшение процесса(-ов) разработки ПО

Роль проектного менеджера: находить поддержку, мотивировать команду и блокировать проблемы

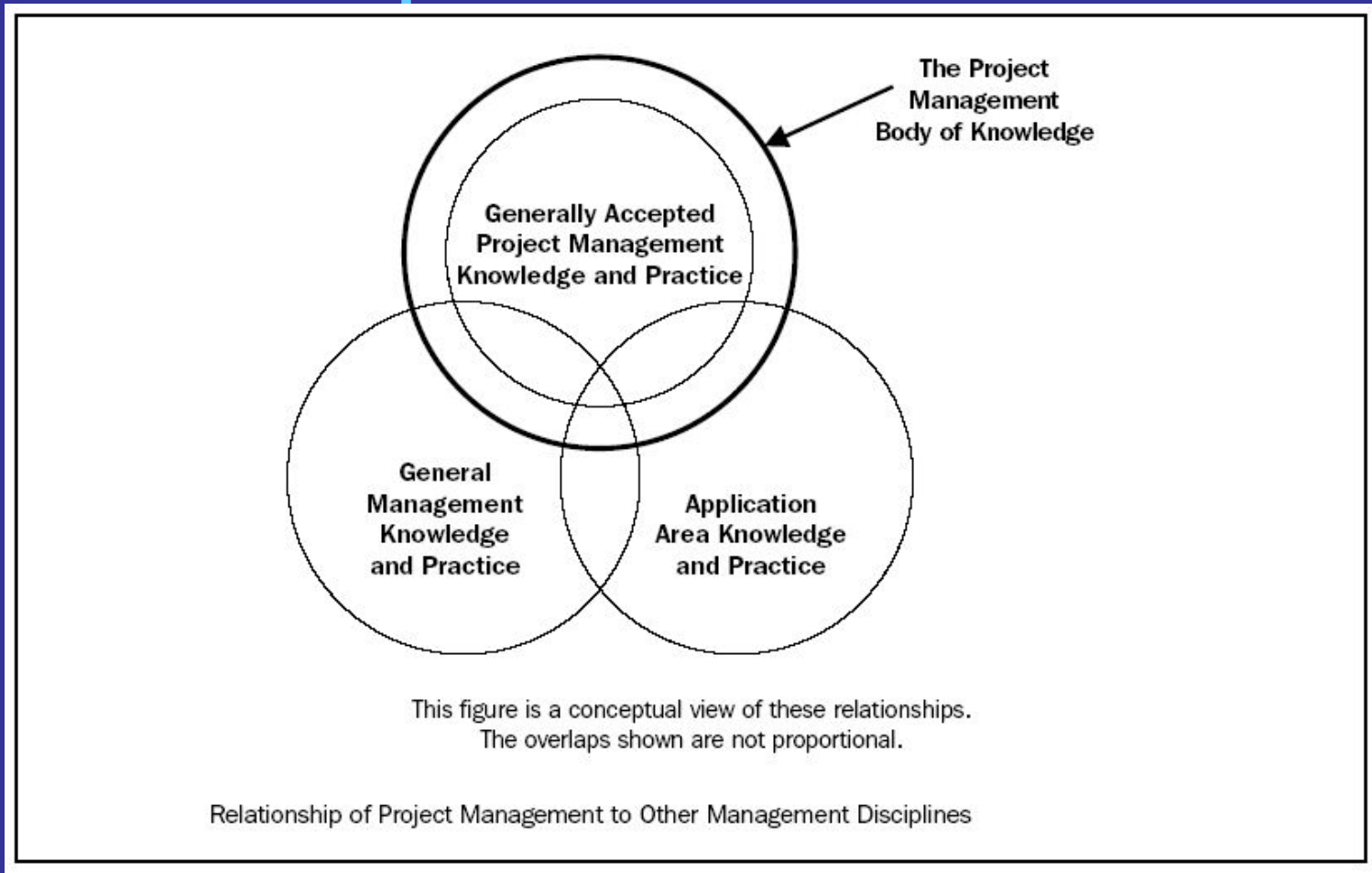


Управление проектами

Проект

- Проект
 - временное предприятие для создания уникального продукта или услуги
- Управление проектом (project management)
 - приложение знаний, умений/навыков (skills), инструментов и техник/практик к проектной деятельности (activities) для удовлетворения требований к результату проекта

Дисциплина управления проектами



Дисциплина управления проектами

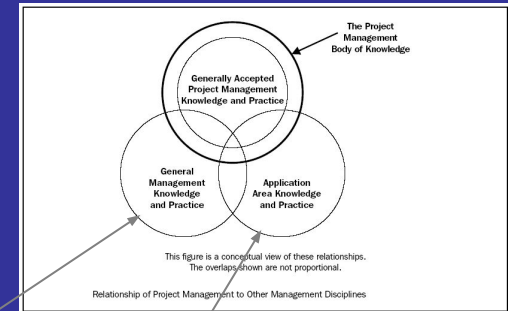
1.4 RELATIONSHIP TO OTHER MANAGEMENT DISCIPLINES

Much of the knowledge needed to manage projects is unique to project management (e.g., critical path analysis and work breakdown structures). However, the PMBOK® does overlap other management disciplines, as illustrated in **Figure 1-2**.

General management encompasses planning, organizing, staffing, executing, and controlling the operations of an ongoing enterprise. General management also includes supporting disciplines such as law, strategic planning, logistics, and human resources management. The PMBOK® overlaps or modifies general management in many areas—organizational behavior, financial forecasting, and planning techniques, to name just a few. Section 2.4 provides a more detailed discussion of general management.

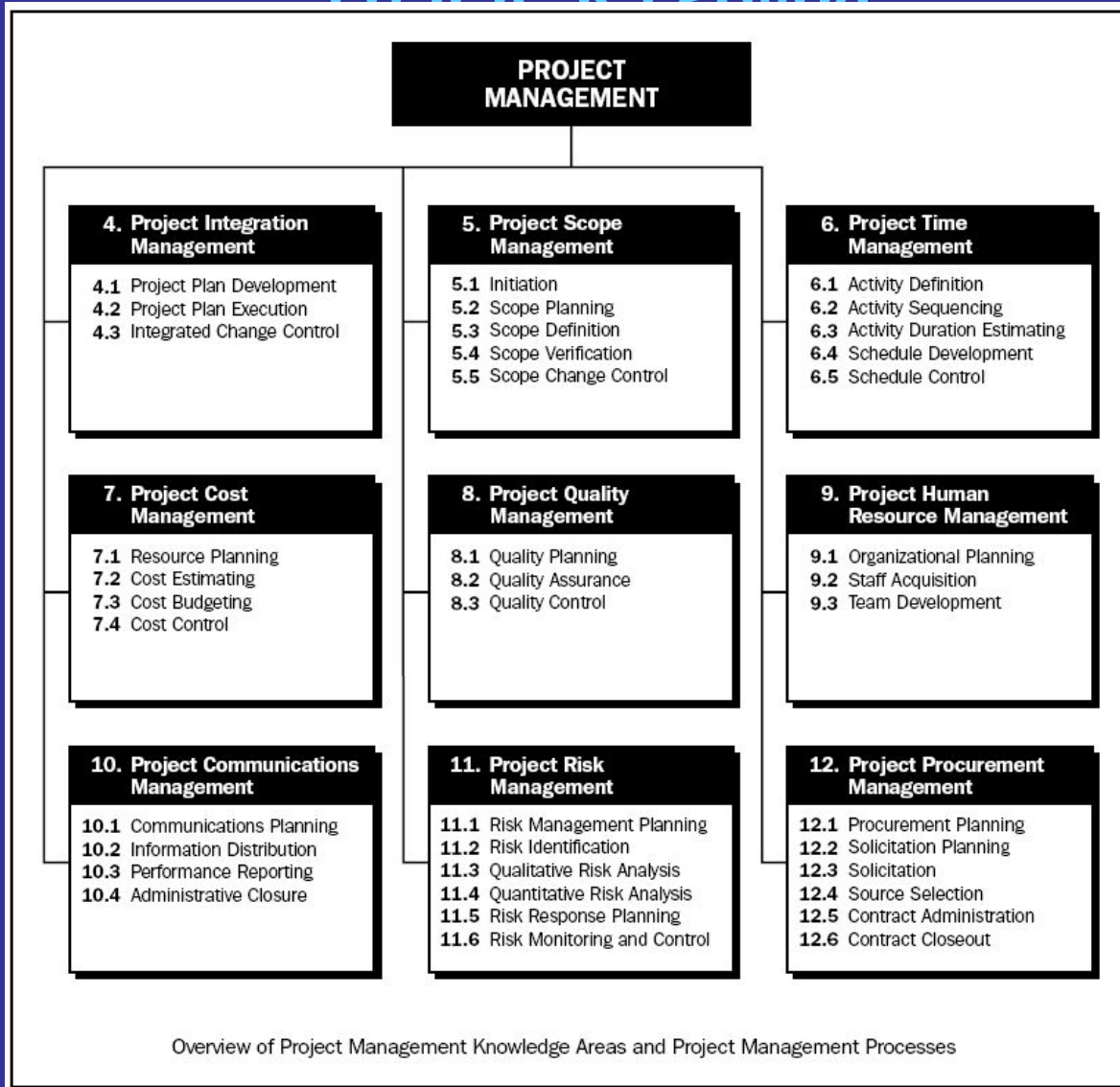
Application areas are categories of projects that have common elements significant in such projects, but are not needed or present in all projects. Application areas are usually defined in terms of:

- Functional departments and supporting disciplines, such as legal, production and inventory management, marketing, logistics and personnel.
- Technical elements, such as software development, pharmaceuticals, water and sanitation engineering, or construction engineering.
- Management specializations, such as government contracting, community development, or new product development.
- Industry groups, such as automotive, chemicals, agriculture, or financial services.



A Guide to the
Project Management Body of Knowledge
(PMBOK® Guide)
2000 Edition

Дисциплина управления проектами



Процесс

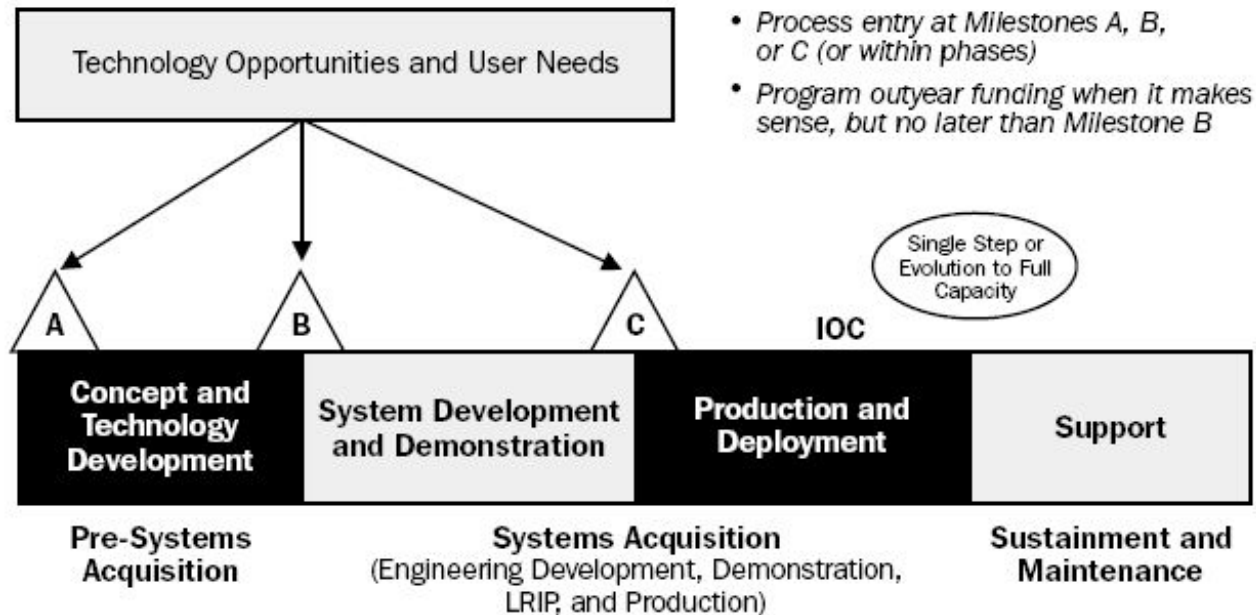
Процесс определяет:

- Кто?
- Что?
- Когда (в какой последовательности)
делает для достижения определенной
цели

*КРА

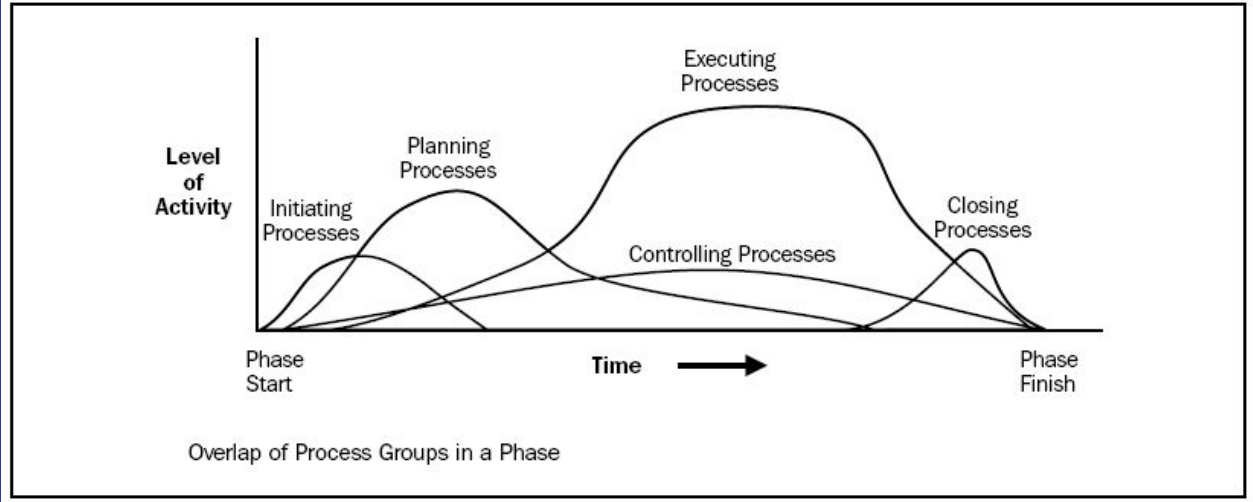
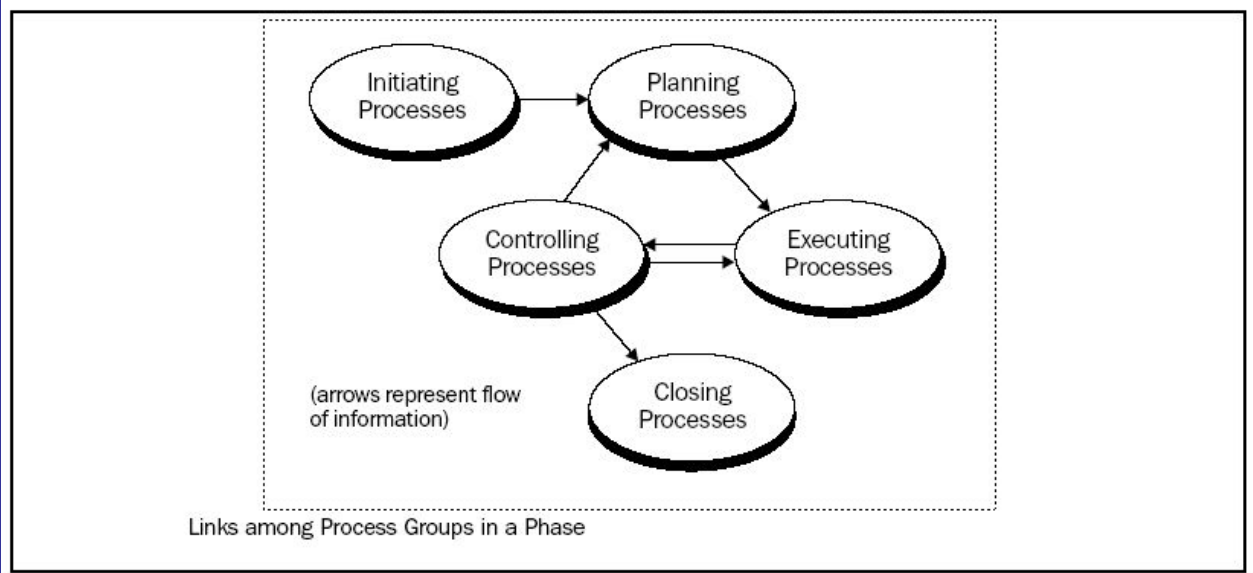
- “Область компетенции”
<Knowledge/Key> <Process/Practice>
Area
 - описывает знания и практики в виде группы взаимосвязанных процессов, необходимых для решения определенного комплекса задач

Жизненный цикл проектов (life cycle)

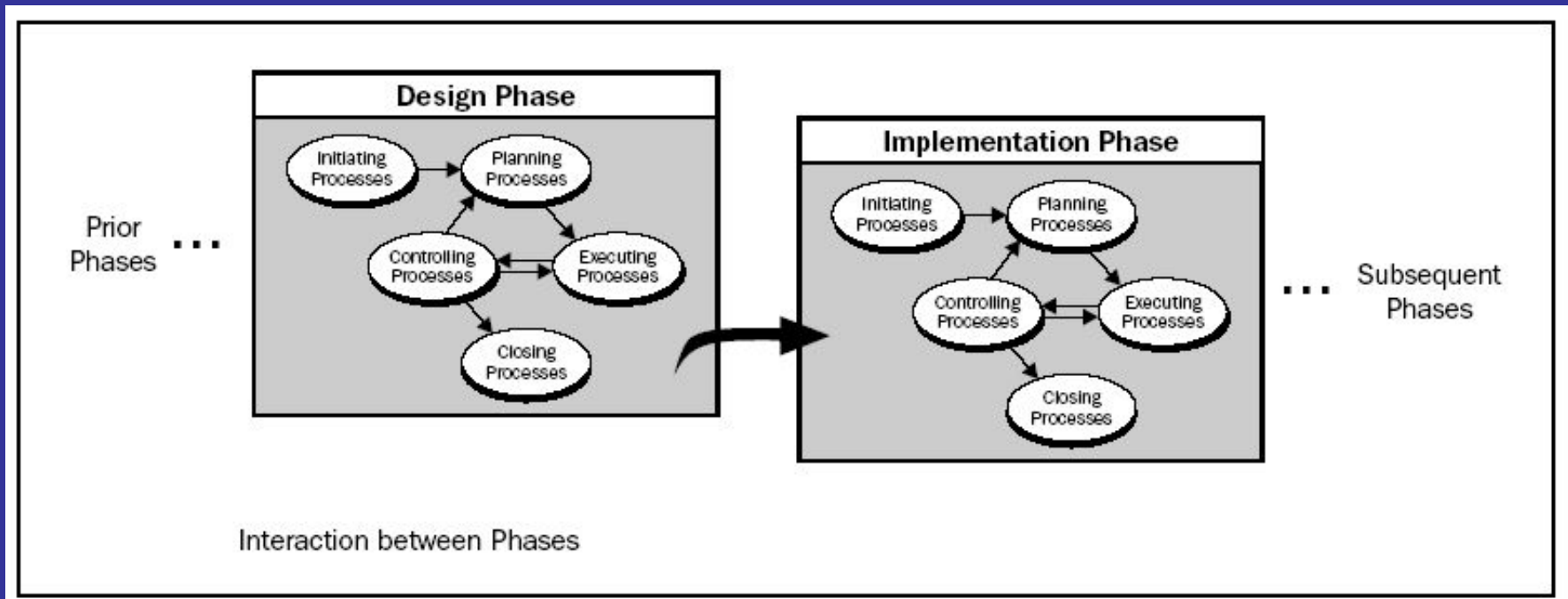


Representative Life Cycle for Defense Acquisition, per US DODI 5000.2
(Final Coordination Draft, April 2000)

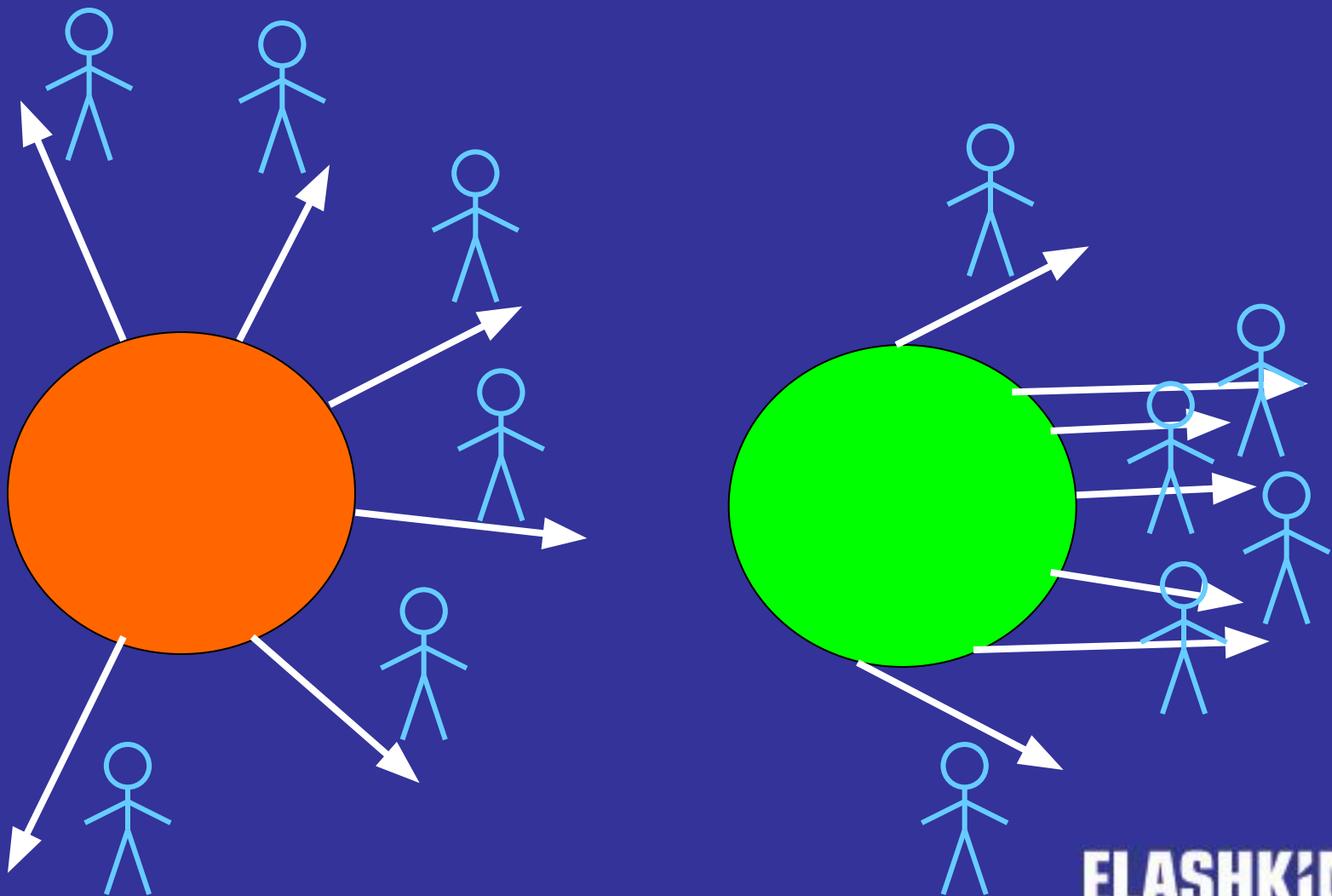
Процессы



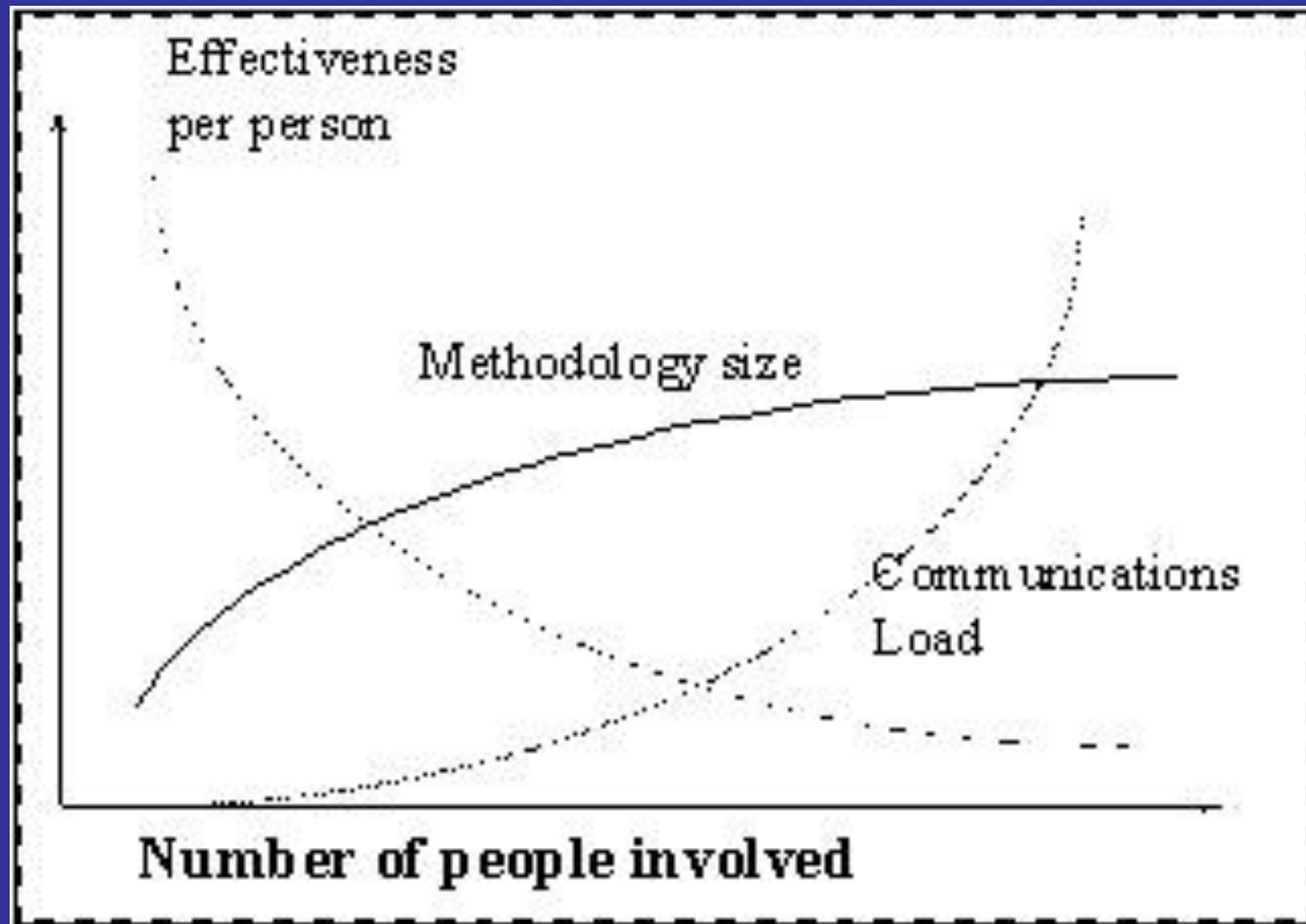
Процессы и фазы проекта



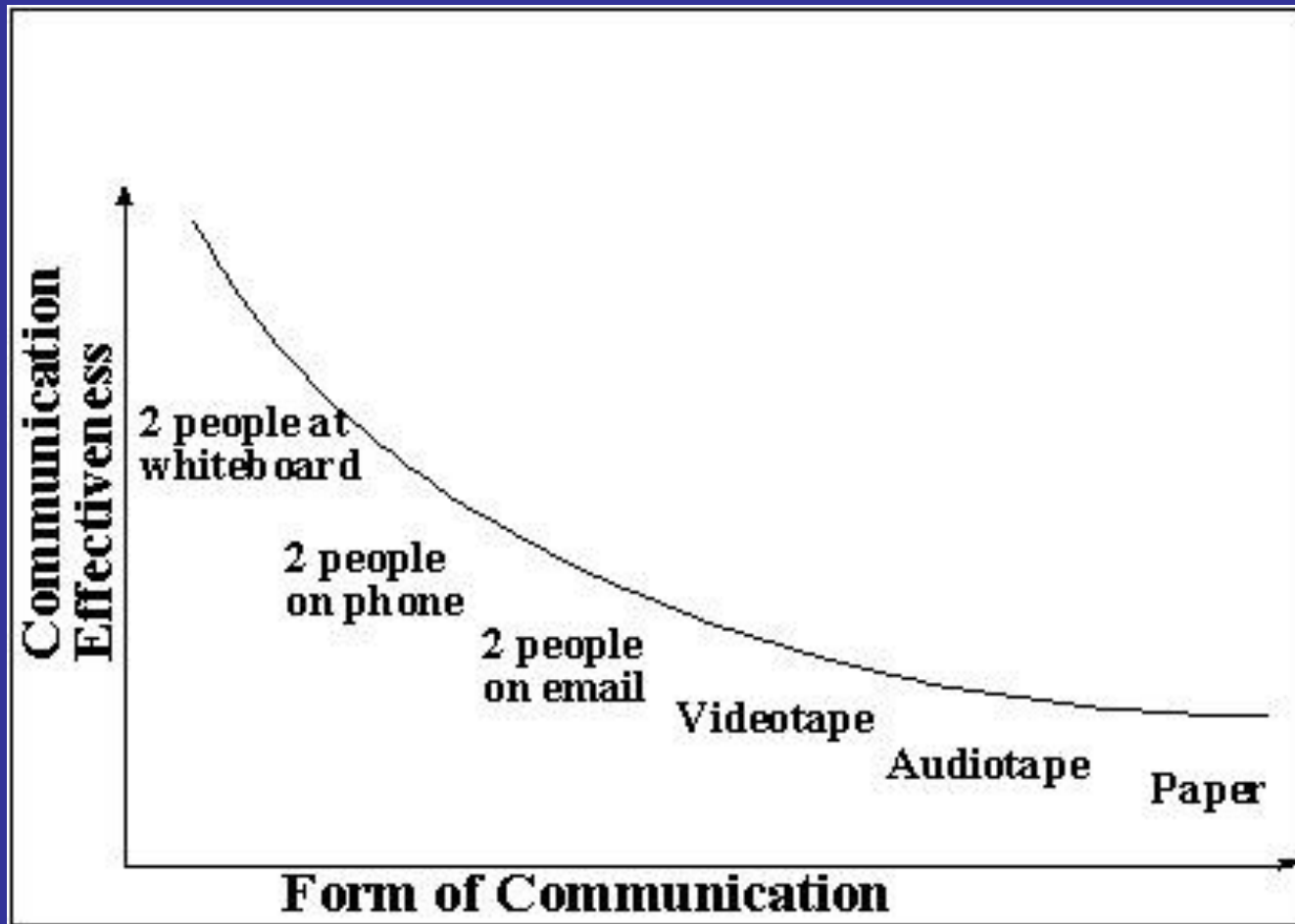
Правильная команда



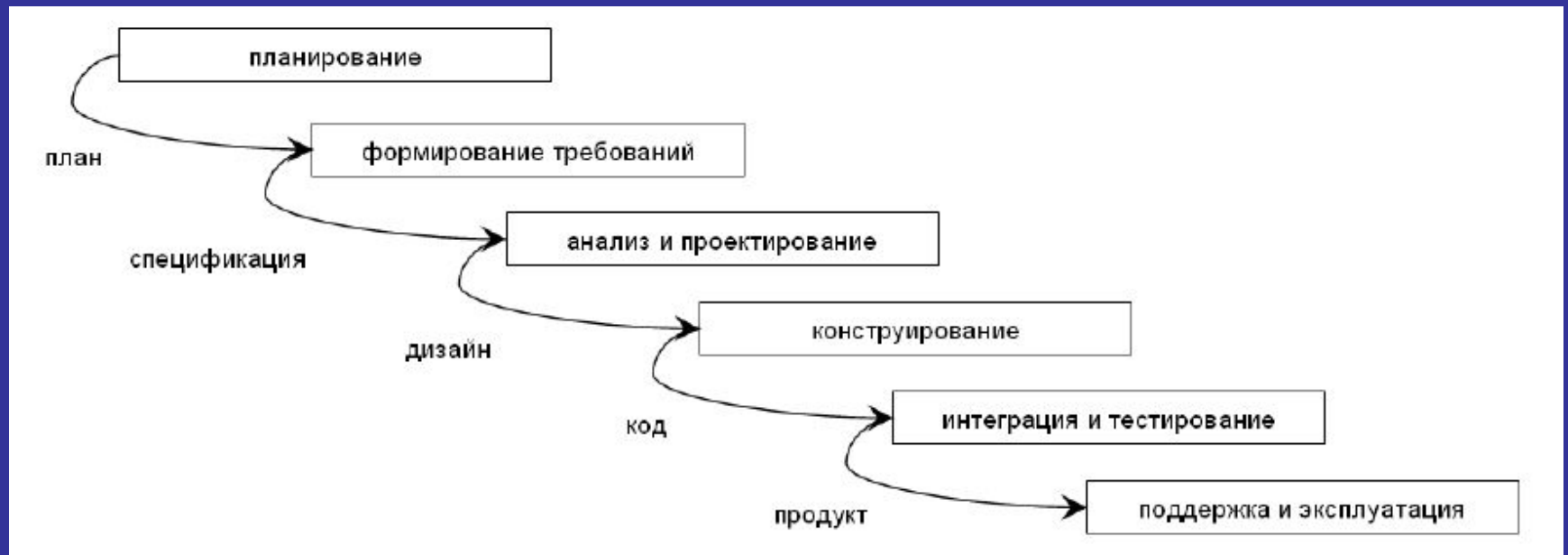
Когда применять методики управления проектами?



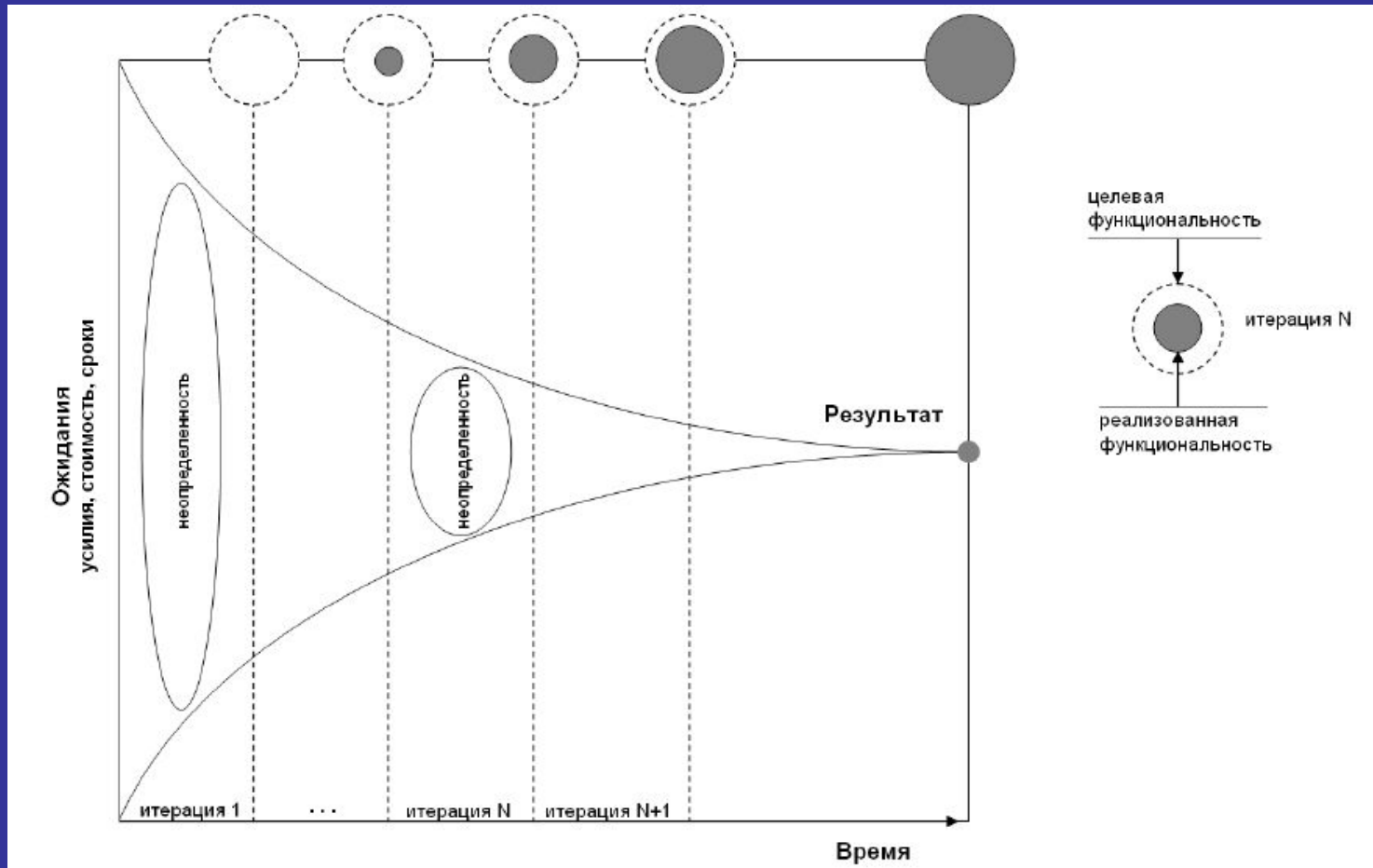
Эффективность коммуникаций



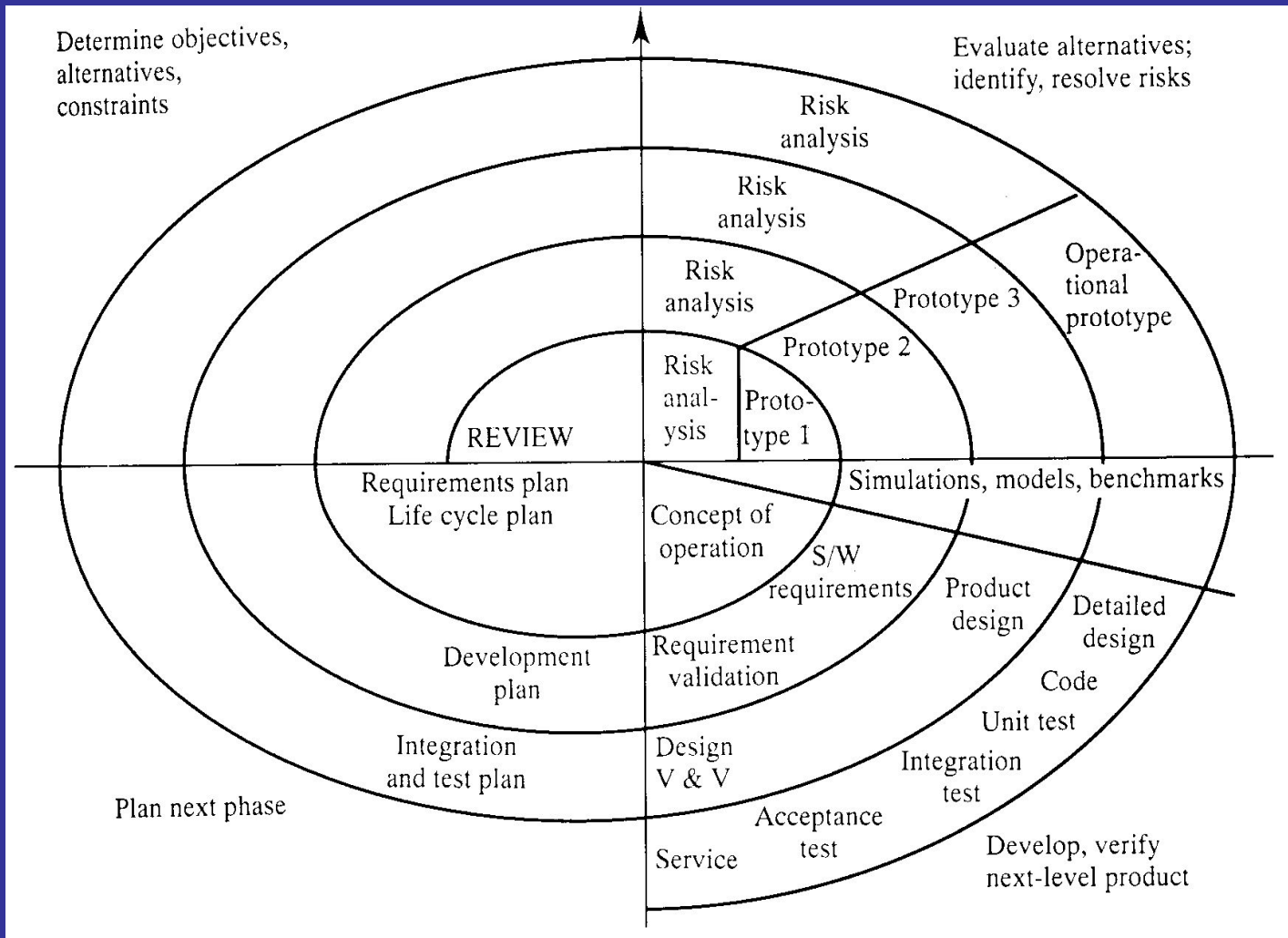
Каскадная модель (водопад)



Инкрементальная модель



Эволюционная модель



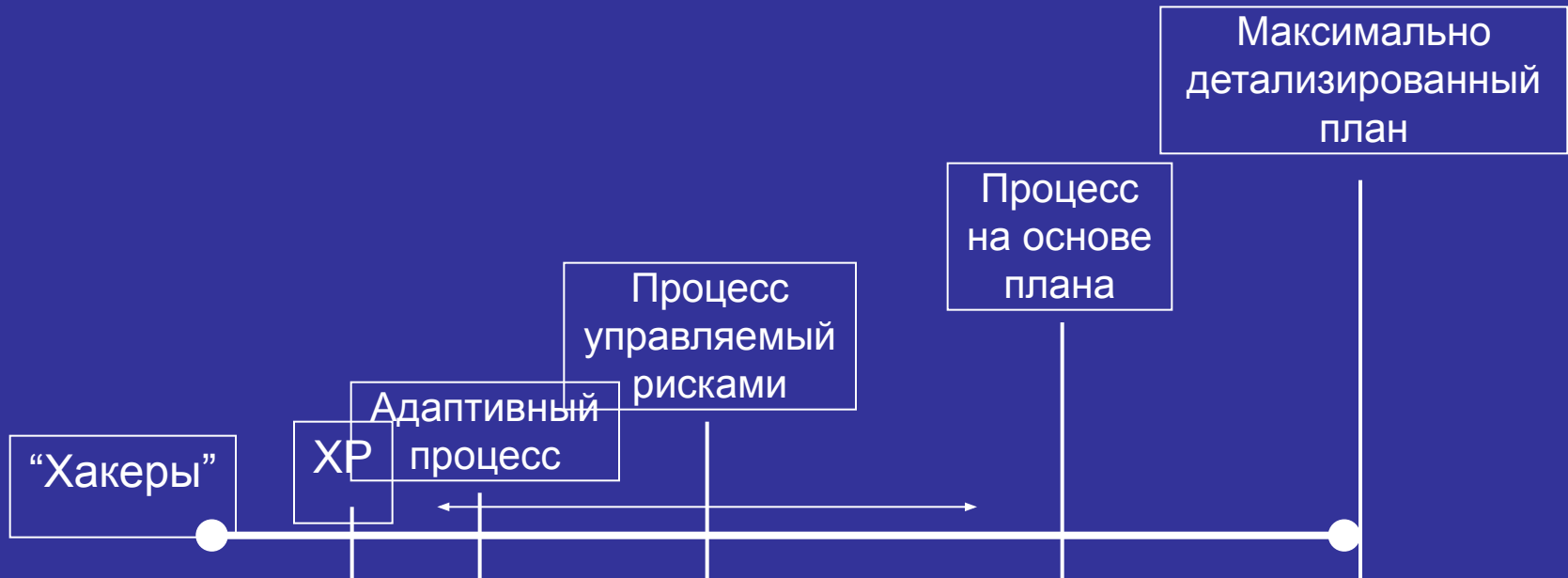
Варианты моделей процессов ЖЦ

Модель процесса	Все ли требования определены?	Создаются ли промежуточные сборки?	Доступна ли “промежуточная” версия продукта?	Соотношение между сборками (build)
Каскадная (Waterfall)	Да	Нет	Нет	Build N=Build 1
Инкрементальная (Incremental)	Да	Да	Может быть	Build N=Build (N-1) + новые возможности
Эволюционная (Evolutionary)	Нет	Да	Да	Build N=Build (N-1) + уточненная спецификация
Гибридная (Hybrid = I+E)	Большинство	Да	Да	Build N=Build (N-1) + новые возможности + уточненная спецификация

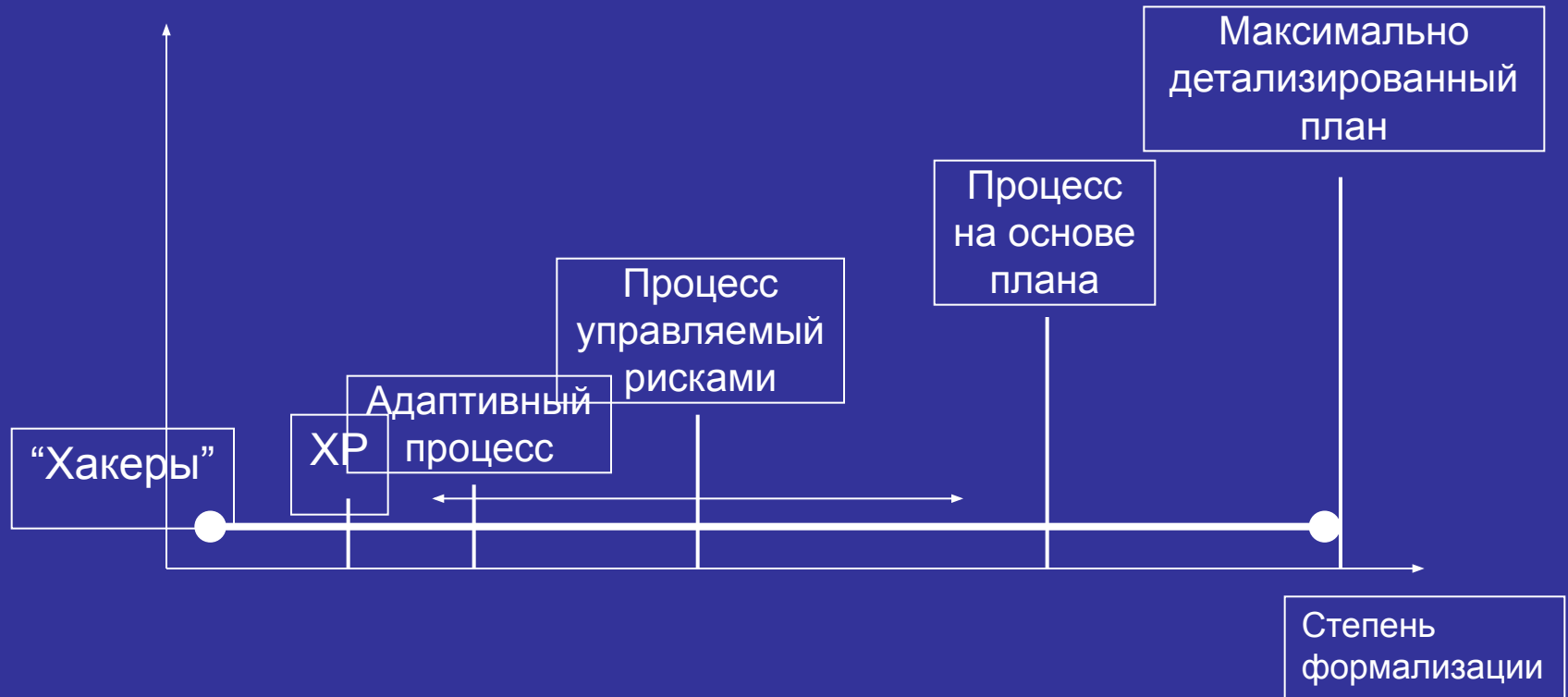
Факторы выбора

Ограничения	Возможность	Риск
Требования недостаточно ясны	E	W, I
Система слишком велика, чтобы реализовать целиком “на одном дыхании”	I, E	W
Все возможности (характеристики, функциональность) должны быть доступны “сразу”	W	I, E
На первом этапе может быть доступна только часть возможностей	I, E	W
Вывод из эксплуатации старой системы может быть произведен постепенно	I, E	W
Возможно/ожидается скорое изменение требований	E	W, I
Возможно/ожидается скорое изменение технологий	E	W, I
Неопределенность в последующем (позапном) финансировании	W	I, E

Различная степень формализации планирования



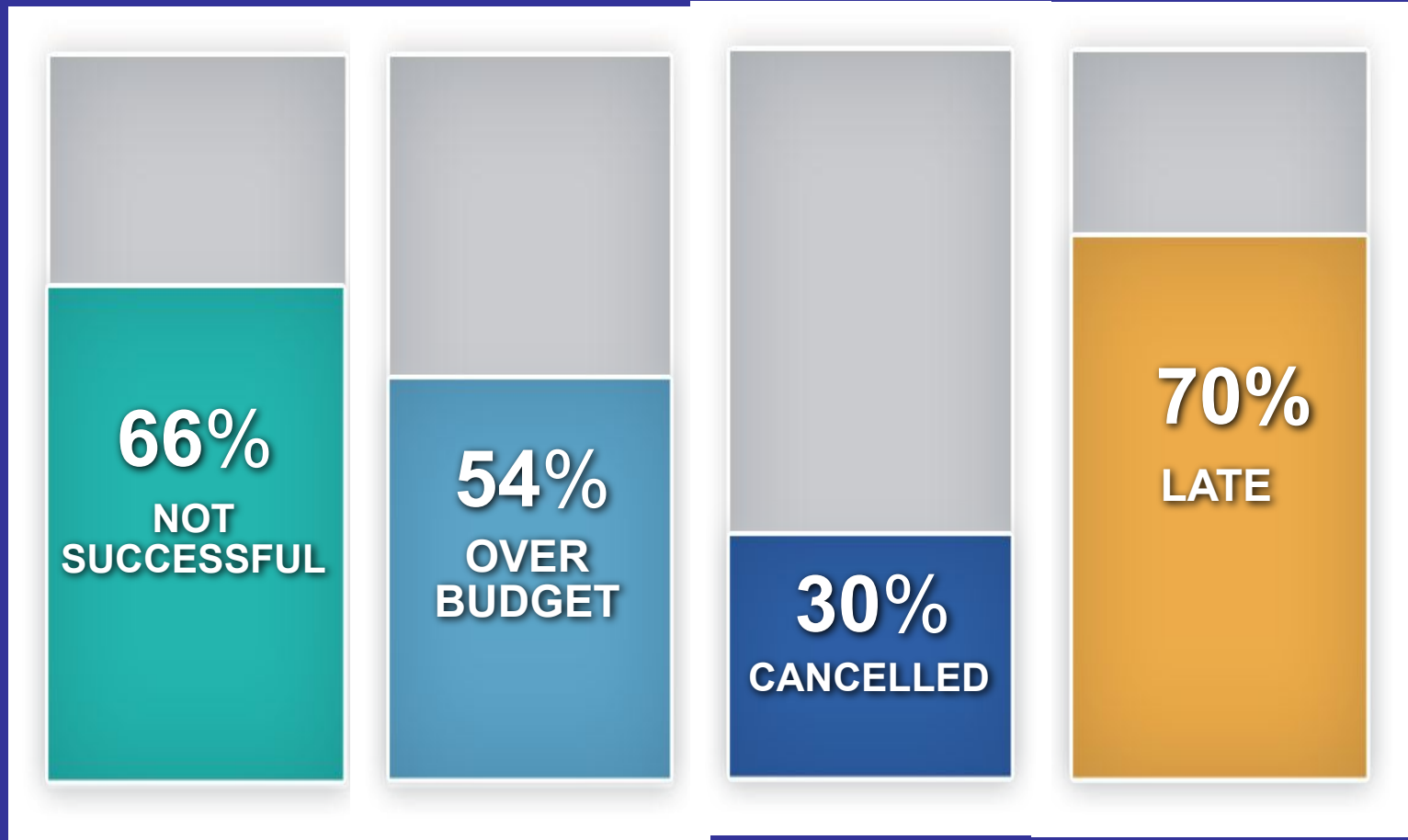
Различная степень формализации планирования



Различие в подходах к процессам

Agile: “План по малу”	Формальный процесс: “Вал по плану”
“Гнезда” разработки	Место расположения разработчиков значения не имеет
Постоянно присутствующий заказчик	Степень вовлеченности заказчика варьируется на этапе разработки
Доверие персонам	Доверие документам
Быстрое изменение требований; готовность корректировать “на ходу”	Стабильные заранее известные требования
Архитектура соответствует “текущему моменту”	Архитектура разрабатывается с учетом существующих и возможных требований
Рефакторинг лёгок	Рефакторинг тяжёл
Малые команды и проекты	Большие команды и проекты
Быстрая отдача – минимизация времени между релизами	Максимальное соответствие плану, в том числе по “уровню качества”

The Problem: Project Results are Poor

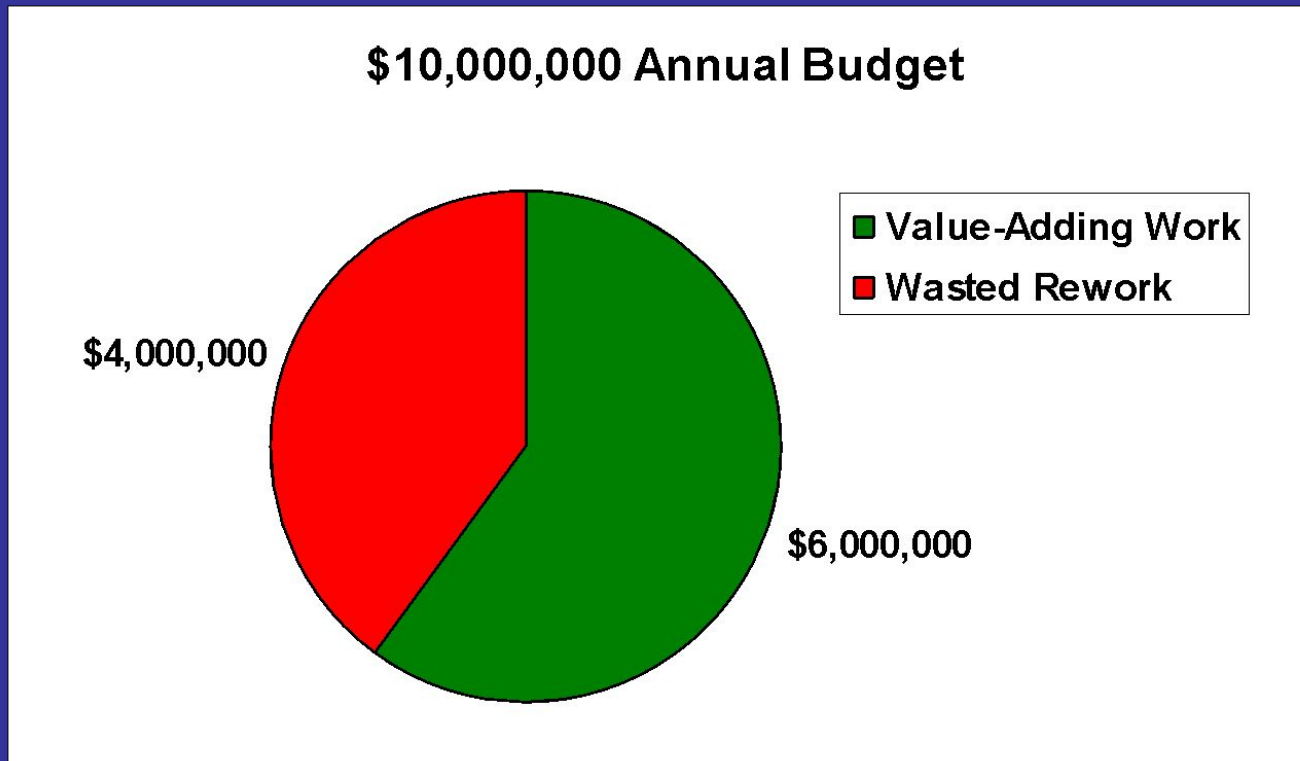


Source: THE STANDISH GROUP 2003

Rework Costs are High Even for Successful Projects!

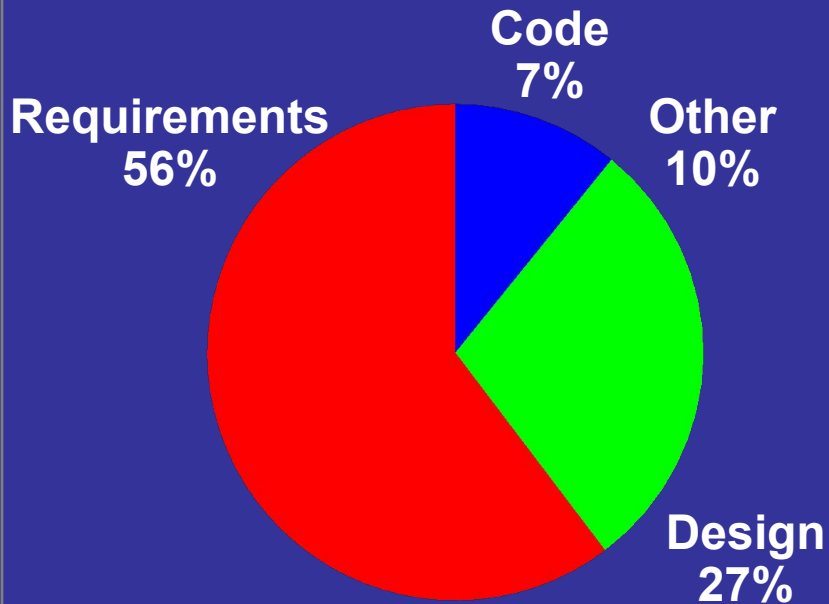


Application development organizations typically spend about 40% of their development effort on rework



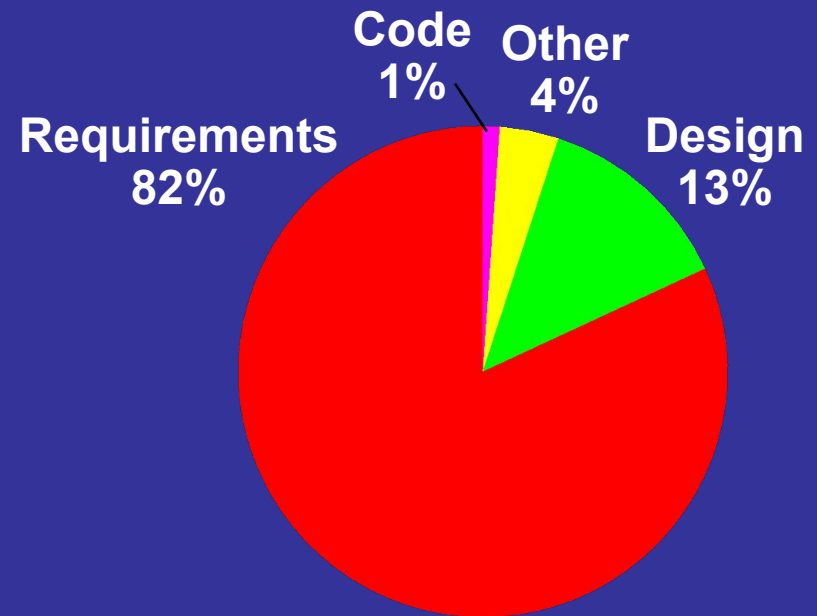
Poor Requirements is the Principal Cause

Distribution of Defects



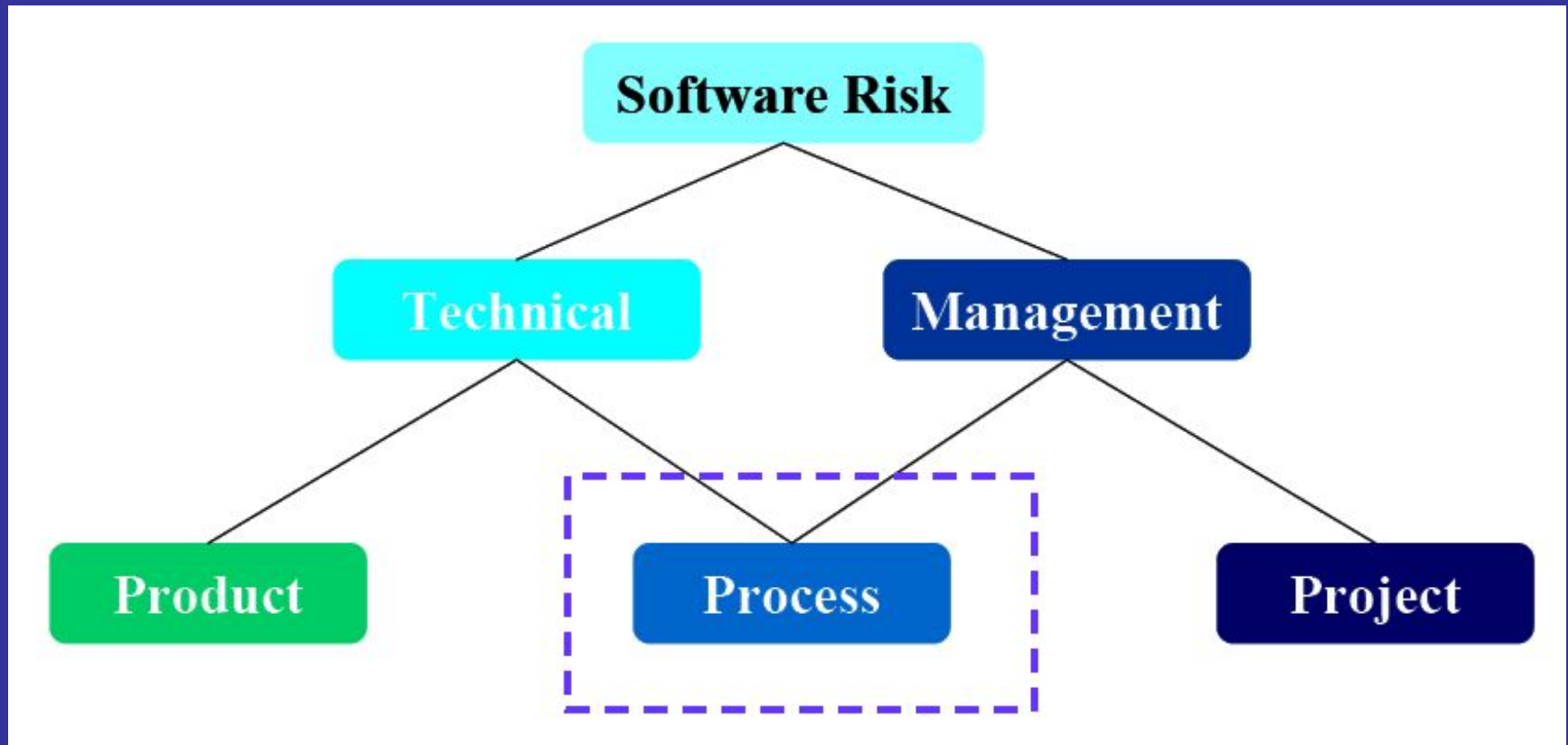
Source: James Martin

Distribution of Effort to Repair Defects



Source: Dean Leffingwell

Риски в области ПО (Software Risks)



Возможные атрибуты рисков

- Категория риска
 - Описание риска
 - Признак(и) появления проблемы
 - Близость риска (ожидаемое время наступления проблемы)
 - План предотвращения риска
 - Статус риска
 - ...
-
- Подверженность риску = вероятность * воздействие
 $RE = L^2$
(Risk Exposure = Likelihood * Loss = Probability * Impact)

Образ мыслей MSFv4

- MSF – это не просто набор рекомендаций, MSF – это образ мыслей!
- MSF стремится к созданию культуры, которая помогает успешно выполнять проекты
- Образ мыслей – это набор ценностей, которые определяют, как мы интерпретируем ситуации и реагируем на них
- Образ мыслей помогает членам команды принимать решения, приоритезировать работу, представлять свои роли в команде и взаимодействовать с другими участниками проекта

Основные принципы MSFv4

- Взаимодействие с партнерами
- Поощрение открытого общения
- Общее видение проекта
- Качество – это ежедневная работа каждого сотрудника
- Оставайтесь гибкими, адаптируйтесь к изменениям
- Внедрение проекта должно стать привычкой
- Постоянная демонстрация прогресса для заказчика

Состав MSFv4

- Рекомендованные процессы создания ИТ-проектов
- Структура итераций
- Определение рабочих элементов, создаваемых в ИТ-проектах
- Стандартные рабочие элементы и критерии их создания/завершения
- Роли членов команды / Группы безопасности
- Шаблоны документов (Excel, Word)
- Шаблоны Microsoft Project
- Отчеты
- Портал проекта / Шаблон сайта SharePoint

Скорость или предсказуемость?

MSF Agile

- “Эволюция и адаптация”
- Идеально для условий конкуренции
- Опора на людей
- Планируй по мере продвижения

MSF Formal

- “Планирование и оптимизация”
- Идеально для устойчивых условий
- Опора на процессы
- Планируй заранее

Потоки работ в MSF Agile

- Формулировка целей и задач проекта
- Создание сценариев
- Создание требований по качеству обслуживания
- Планирование итераций
- Создание архитектуры решения
- Реализация задачи по разработке
- Построение продукта
- Тестирование сценария
- Тестирования требования по качеству обслуживания
- Исправление ошибок
- Заккрытие ошибок
- Выпуск продукта
- Управление проектом

The End.