

Программная инженерия

Лекция
3

Лекция 3 Рабочий продукт, проект, дисциплина

Обязательства Цели

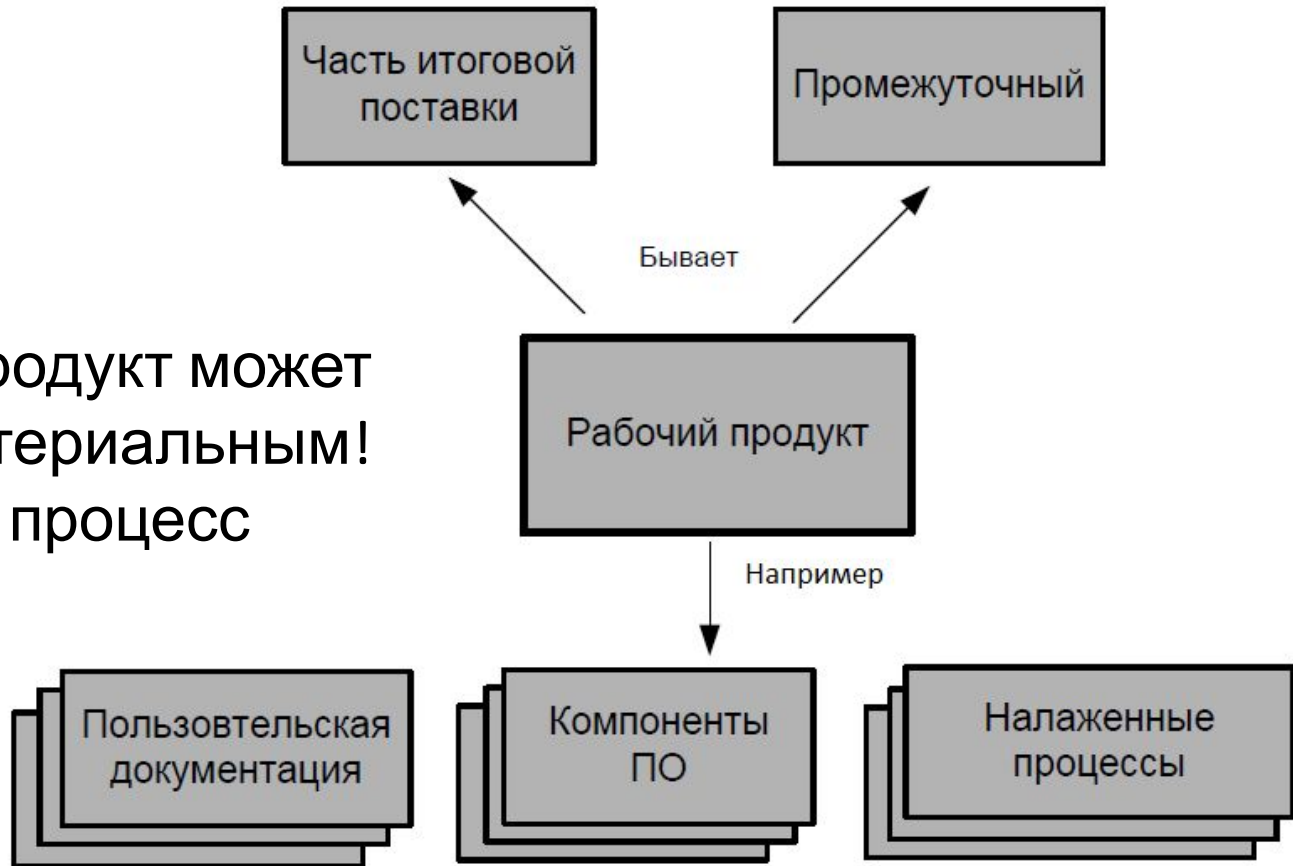
1. Различие между управлением программными проектами и инженерными проектами
1. Задачи руководителя программного проекта
2. Значение и роль этапа планирования проекта среди всех этапов разработки ПО
3. Графические способы представления графиков работ
4. Представление о рисках

Читать по Лекции 3

Книга	Страницы
<i>Д.В.Кознов ,и др. Введение в программную инженерию</i>	<i>Лекция 3: Стр. 14-19</i>
Иан Соммервилл Инженерия программного обеспечения	<i>Глава 4: Стр. 81 - 100</i>

Рабочий

Рабочий продукт (working product) – любой артефакт (т.е. предмет, изделие, что-то сделанное человеком) произведенный в процессе разработки ПО и **отчуждаемый от его автора**, например, файл или набор файлов, документы, составные части продукта, сервисы, процессы, специфика:



Рабочий продукт может быть нематериальным! Например, процесс отладки

Рабочий

продукт

Методологии проектирования и разработки ПО включают в себя рабочие продукты

Например, методология RUP фирмы Rational Rose включает:

Диаграммы использования (USE CASE Diagram)

Диаграммы классов (Class Diagram)

Диаграммы действий (Activity Diagram) и другие.

Не любой промежуточный результат является рабочим продуктом!

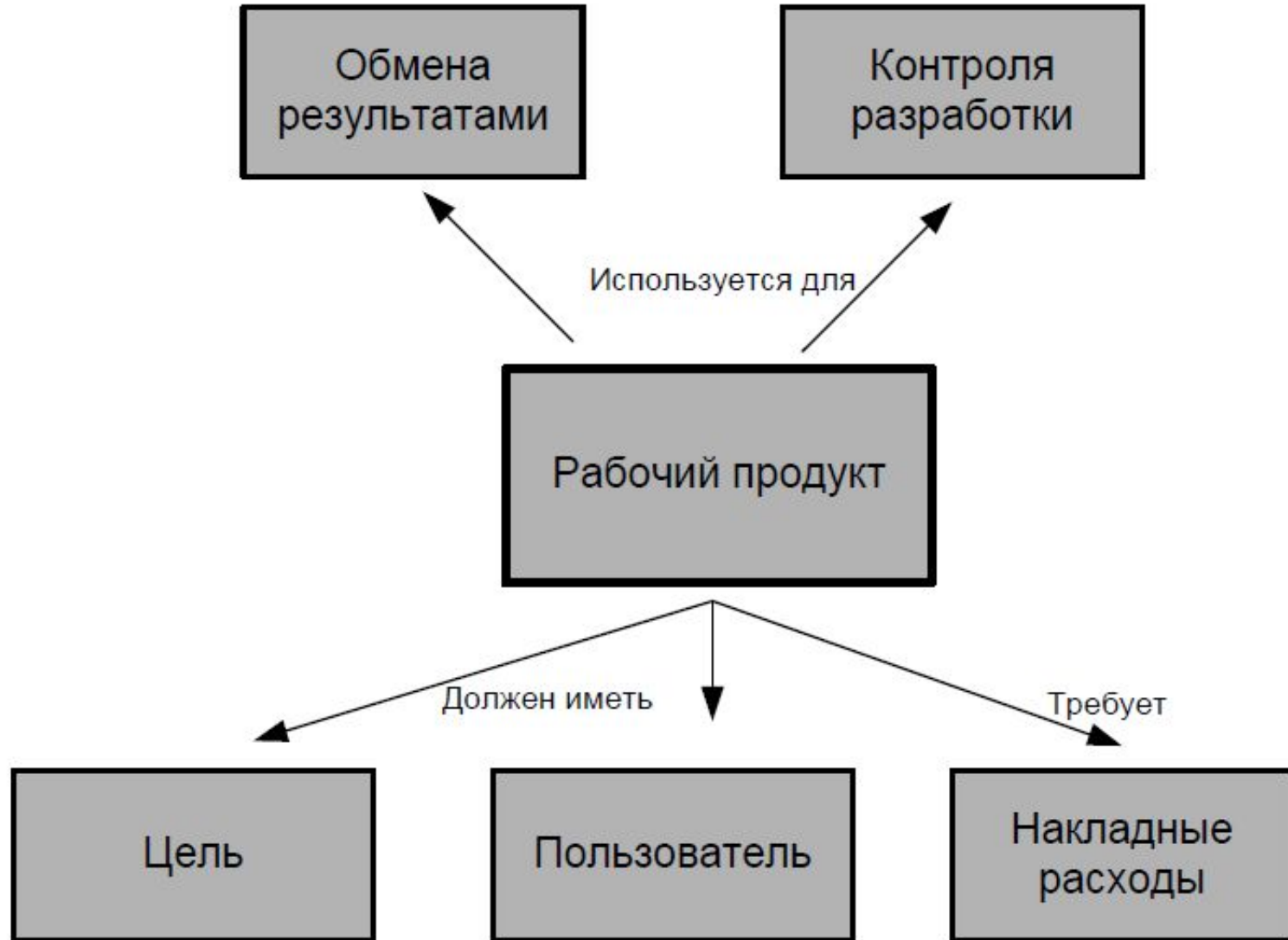
Чтобы промежуточный результат стал рабочим продуктом, требуются дополнительные трудовые затраты на его

«отчуждение», т.е. отделение его от автора-разработчика (дополнительное тестирование, аккуратная документация, перевод

документации на иностранный язык и т.п.).

Т.о. , чтобы минимизировать накладные расходы на создание рабочего продукта, необходимо иметь ясную цель его создания

Рабочий продукт



Рабочий продукт



Дисциплина

обязательства

В основе разделения обязанностей в бизнесе и производстве лежит определенная деловая этика,

частью которой является дисциплина
Обязательства:

- даются добровольно;
- не даются легко (не обдуманно)
- формулируются в виде: **что, кем, сроки**
- формулируются открыто и публично

Кроме того:

- ответственная сторона стремится выполнить обязательства, даже если нужна помощь
- до наступления deadline, как только становится очевидно, что работа не может быть закончена в срок,

Дисциплина обязательств

Дисциплине обязательств уделяется много внимания в рамках методик MSF (Microsoft Solutions Framework) , поскольку там в модели команды нет лидера, начальника.

Эта дисциплина реализована также в Scrum (*Scrum - толкотня; схватка вокруг мяча (в регби)*).

Scrum-команда имеет много свобод, и в силу этого – большую ответственность. Регламентируются также правила действий, когда обязательства не могут быть выполнены такой командой

Проек

Существует много творческих, новых задач, где, быть может, в будущем и удастся создать конвейеры, но в данный момент для их решения требуется существенная концентрация сил и энергии людей, неожиданные решения, а также удача и легкая рука. Это и есть область проектов

Прое́кт – это уникальная (в отличие от традиционного пооперационного стиля промышленного производства) деятельность, имеющая начало и конец во времени, направленная на достижение определённого результата (цели), создание определённого, уникального продукта или услуги, при заданных ограничениях по ресурсами и срокам, а также требованиям к качеству и допустимому уровню риска.

В частности, разработка программного обеспечения, является, преимущественно, проектной областью.

Проек

Т

Промышленные проекты – большое количество участников, сложность самих работ.

Задача в управлении такими проектами – это все охватить,

все проконтролировать, ничего не забыть, все свести воедино, добиться согласованного движения.

Творческие проекты – это новые идеи, новый сервис, абсолютно новый программный продукт, какого еще не было на рынке.

Проекты по разработке программного обеспечения находятся между двумя этими полюсами, занимая в этом

пространстве различное положение.

Проек

Управление проектами (project management) – область деятельности, в ходе которой, в рамках определенных проектов, определяются и достигаются четкие цели при нахождении компромисса между объемом работ, ресурсами (такими как время, деньги, труд, материалы, энергия, пространство и др.), временем, качеством и рисками.

Важные аспекты управления проектами:

- **Stakeholders** – это люди/стороны, которые не участвуют непосредственно в проекте, но влияют на него и или заинтересованы в его результатах
- **Project scope** – это границы проекта

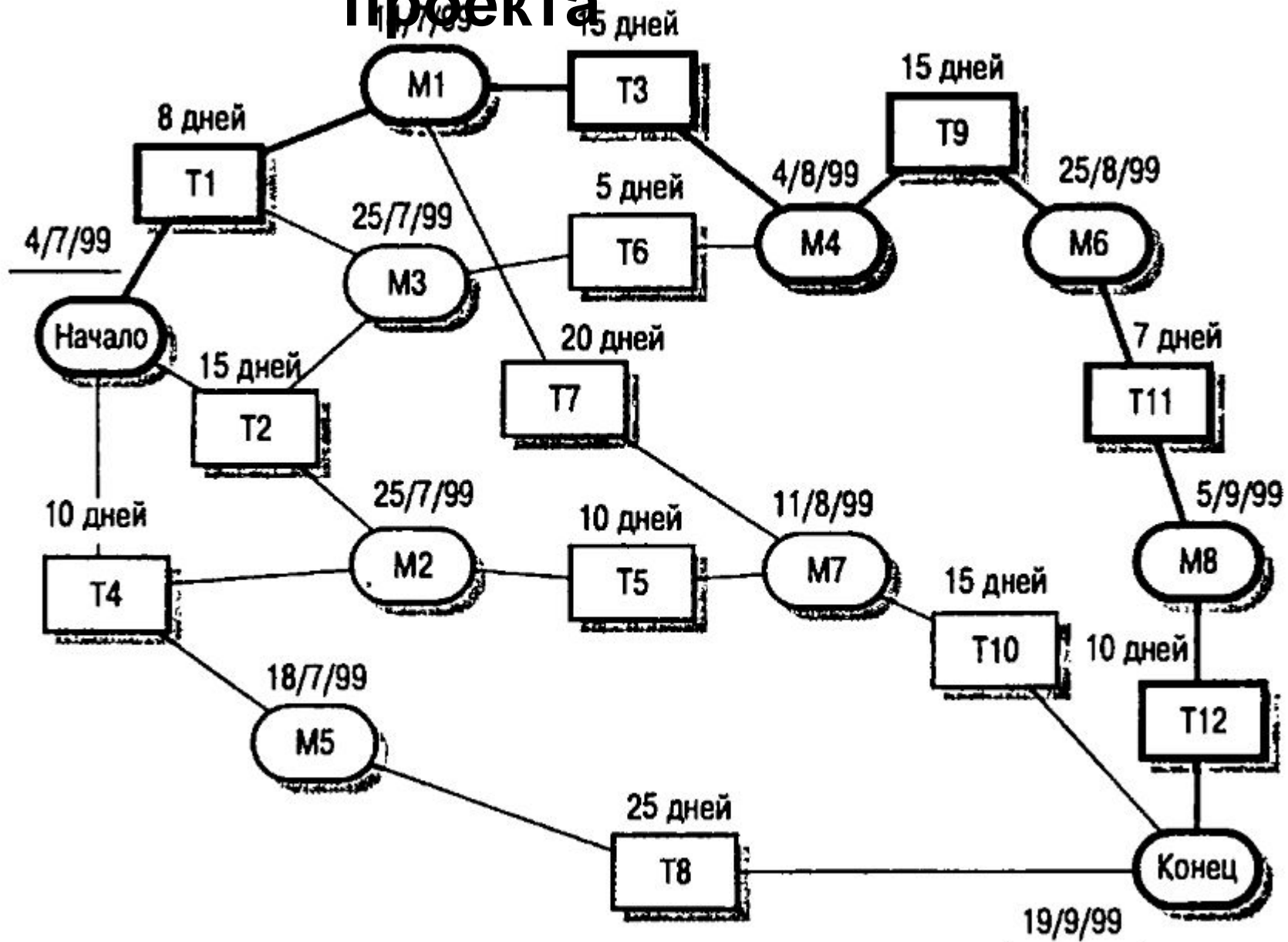
Компромиссы – важнейший аспект управления

Управление

При разработке **проектами** проектов, следуя MSF 3.1, важны следующие области управления.

Область управления проектами	Описание
Планирование и мониторинг проекта, контроль за изменениями в проекте (Project planning / Tracking / Change Control)	Интеграция и синхронизация планов проекта; организация процедур и систем управления и мониторинга проектных изменений
Управление рамками проекта (Scope Management)	Определение и распределение объема работы(рамок проекта); управление компромиссными решениями в проекте
Управление календарным графиком проекта (Schedule Management)	Составление календарного графика исходя из оценок трудозатрат, упорядочивание

Сетевой график проекта



Управление проектами

Область управления проектами

Описание

Управление стоимостью (Cost Management)

Оценки стоимости исходя из оценок временных затрат; отчетность о ходе проекта и его анализ; анализ затратных рисков (value analysis);

Управление персоналом (Staff Resource Management)

Планирование ресурсов; формирование проектной команды; разрешение конфликтов; планирование и управление подготовкой

Управление коммуникацией (Communications Management)

Коммуникационное планирование (между проектной группой, заказчиком/спонсором, потребителями/пользователями), отчеты

Управление рисками (Risk Management)

Организация процесса управления рисками в команде

Управление персоналом

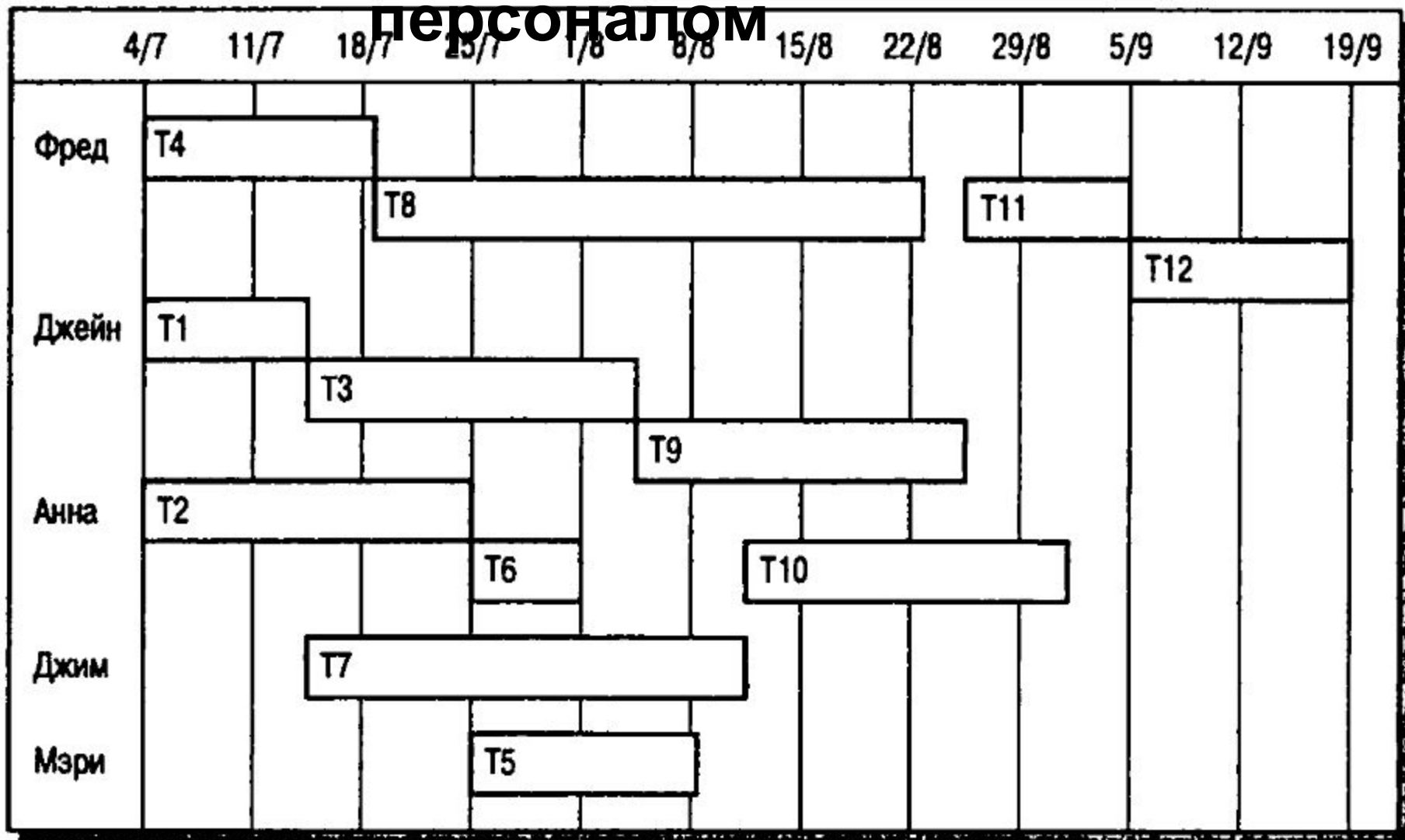


Рис. 4.5. Временная диаграмма распределения работников по этапам

Управление проектами

Область управления проектами	Описание
Управление рисками (Risk Management)	Организация процесса управления рисками в команде
Управление снабжением (Procurement)	Анализ цен поставщиков услуг и/или аппаратного/программного обеспечения; подготовка документов об инициировании предложений (requests for proposals – RFPs), выбор поставщиков и субподрядчиков; составление контрактов и переговоры об их условиях; договоры; заказы на поставку и платежные требования
Управление качеством (Quality Management)	Планирование качества, определение применяемых стандартов,

Управление

рисками

Возможные риски программных проектов

Риск	Тип риска	Описание риска
Текучесть разработчиков	Риск для проекта	Опытные разработчики покидают проект до его завершения
Изменение в управлении организацией	Риск для проекта	Организация меняет свои приоритеты в управлении проектом
Неготовность аппаратных средств	Риск для проекта	Аппаратные средства, которые необходимы для проекта, не поступили вовремя или не готовы к эксплуатации
Изменение требований	Риск для проекта и для разрабатываемого продукта	Появление большого количества непредвиденных изменений в требованиях, предъявляемых к разрабатываемому ПО
Задержка в разработке спецификации	Риск для проекта и для разрабатываемого продукта	Спецификации основных интерфейсов подсистем не поступили к разработчикам в соответствии с графиком работ

Управление рисками

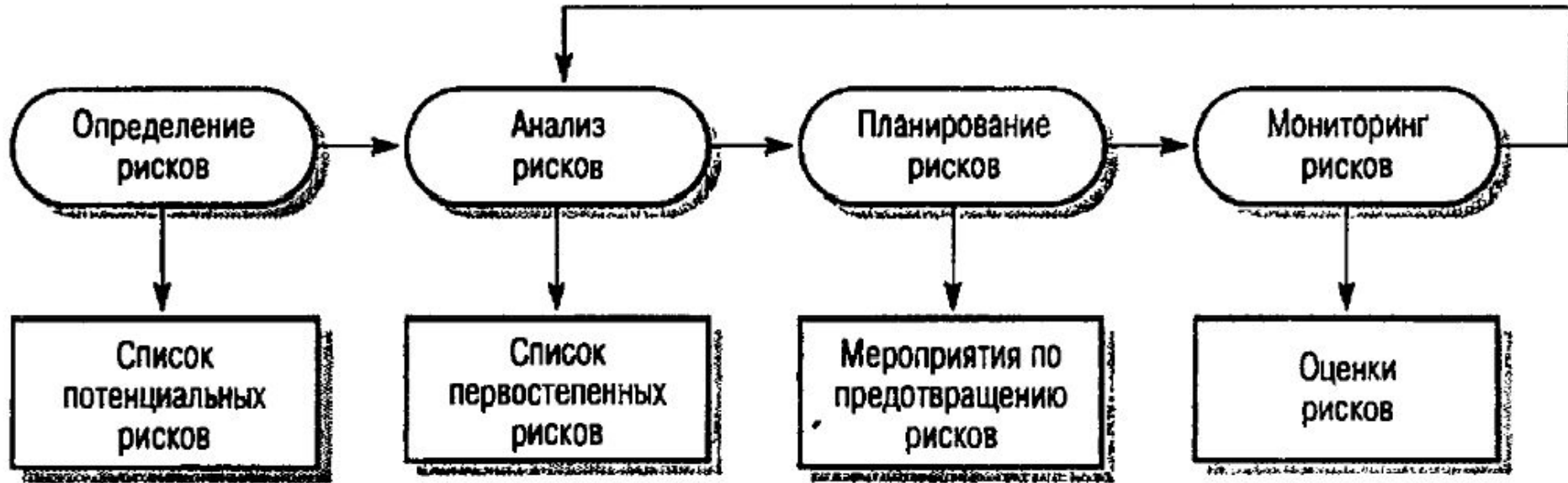


Рис. 4.6. Процесс управления рисками

Категория рисков	Примеры рисков
Технологические риски	База данных, которая используется в программной системе, не обеспечивает обработку ожидаемого объема транзакций.
	Программные компоненты, которые используются повторно, имеют дефекты, ограничивающие их функциональные возможности
Риски, связанные с персоналом	Невозможно подобрать работников с требуемым профессиональным уровнем.
	Ведущий разработчик заболел в самое критическое время.
	Невозможно организовать необходимое обучение персонала
Организационные риски	В организации, выполняющей разработку ПО, произошла реорганизация, в результате чего изменились приоритеты в управлении проектом.
	Финансовые затруднения в организации привели к уменьшению бюджета проекта
Инструментальные риски	Программный код, генерируемый CASE-средствами, не эффективен.
	CASE-средства невозможно интегрировать с другими средствами поддержки проекта
Риски, связанные с системными требованиями	Изменения требований приводят к значительным повторным работам по проектированию системы.