

Экономика программной инженерии

Клименков С.В.
2015-2016 уч. год
v.1.0.3 от 12.03.2016

- Липаев В.В. Экономика производства программных продуктов. Издание второе - М.: СИНТЕГ, 2011. - 358 с.
- Брукс Ф. Мифический человеко-месяц или как создаются программные системы: Пер. с англ. – СПб.: Символ-Плюс, 2006
- Борисова Л.Г. «Организационный дизайн и современные концепции управления»
- Боэм Б. Инженерное проектирование программного обеспечения: Пер. с англ. – М.: Мир, 1985
- Boehm V.W. et al. Software cost estimation with COCOMO II. Prentice Hall PTR. New Jersey. 2000.
- С. Архипенков. Лекции по управлению программными проектами http://citforum.ru/SE/project/arkhipenkov_lectures/

1

Современный IT бизнес в России по разработке ПО

Типы компаний связанных с разработкой ПО в РФ

- RND центры крупных вендоров.
- Стартапы и небольшие частные компании
- Software подразделения системных интеграторов
- Оборонные заводы и институты связанные с обороной
- Компании, предоставляющие Web-услуги
- Бизнес-ПО, ориентированное на управление и учет; автоматизация банков
- «Железячники»



Research & Development центры

- В СПб Oracle, EMC, Intel, AMD, Veam
- 100-500 разработчиков
- Множество проектных команд
- Западная «бюрократия» с характерным способом ведения бизнеса
- Возможность участвовать в разработке мейнстрим технологий
- Высокие зарплаты
- Отсутствие карьерного роста по вертикали
- Возможен «диагональный» рост

Стартапы и небольшие частные компании

- Появляются и прогорают
- 3-5 человек
- Базируются на оригинальной «идее»
- Почти семья → «Санта-Барбара», «Друзья»
- Возможность резкого роста, если идея сработает (маловероятно)
- Карьерный рост вместе с компанией
- Зарплата нестабильна

Подразделения системных интеграторов



- Достаточно крупные и обслуживают существующих клиентов
- 3-100 (до 1000) сотрудников
- Условно-стабильные ЗП, зависящие от общего положения компании, политики.
- Крайности от «Раздолбайство» до «Формализм»

- Крупные, неповоротливые, зависящие от государства (госзаказ)
- Относительно невысокая ЗП
- Советская (СССР) «бюрократия» с характерным подходом к работе
- Возможность интересной работы
«Но все ж мы делаем ракеты!» (С) Визбор.
- Вертикальный карьерный рост



Компании, предоставляющие Web-услуги

- Развитие стартапа, который «вдруг» стал приносить стабильный доход
- Средний размер
- Разная прикладная область (м. б. интересно, а может и нет)
- Достойные ЗП
- Современные технологии
- Неопределенный рост

- 1С, OEBS, SAP
- «Мрак и Копоть» (С) — сложная предметная область и бизнес-процессы
- Стабильная работа «навсегда»
- Большое количество сотрудников
- Очень маленькие и очень большие заработные платы
- Необходимо быть не только программистом но и бухгалтером, психологом,

- Работа на крупные компании РЖД, Газпром...
- Размер компаний от небольших до средних
- Куча работы с железом, микроконтроллерами, паяльными станциями
- Необходимы инвестиции: закупка дорогого оборудования и ПО
- Интересная работа для «гиков»
- Карьерный рост не определен



Экономические особенности разработки «чистого» ПО

- Нет больших закупок расходных материалов (бумага не в счет)
- Относительно невысокие вложения в оборудование? (зависит от...)
- Расходы на аренду помещения
- Расходы на зарплату
- Тщательный подбор кадров
 - Производительность/сотрудник отличается на порядки
 - программисты — люди со странностями ;)
- Сложно нормировать рабочий день

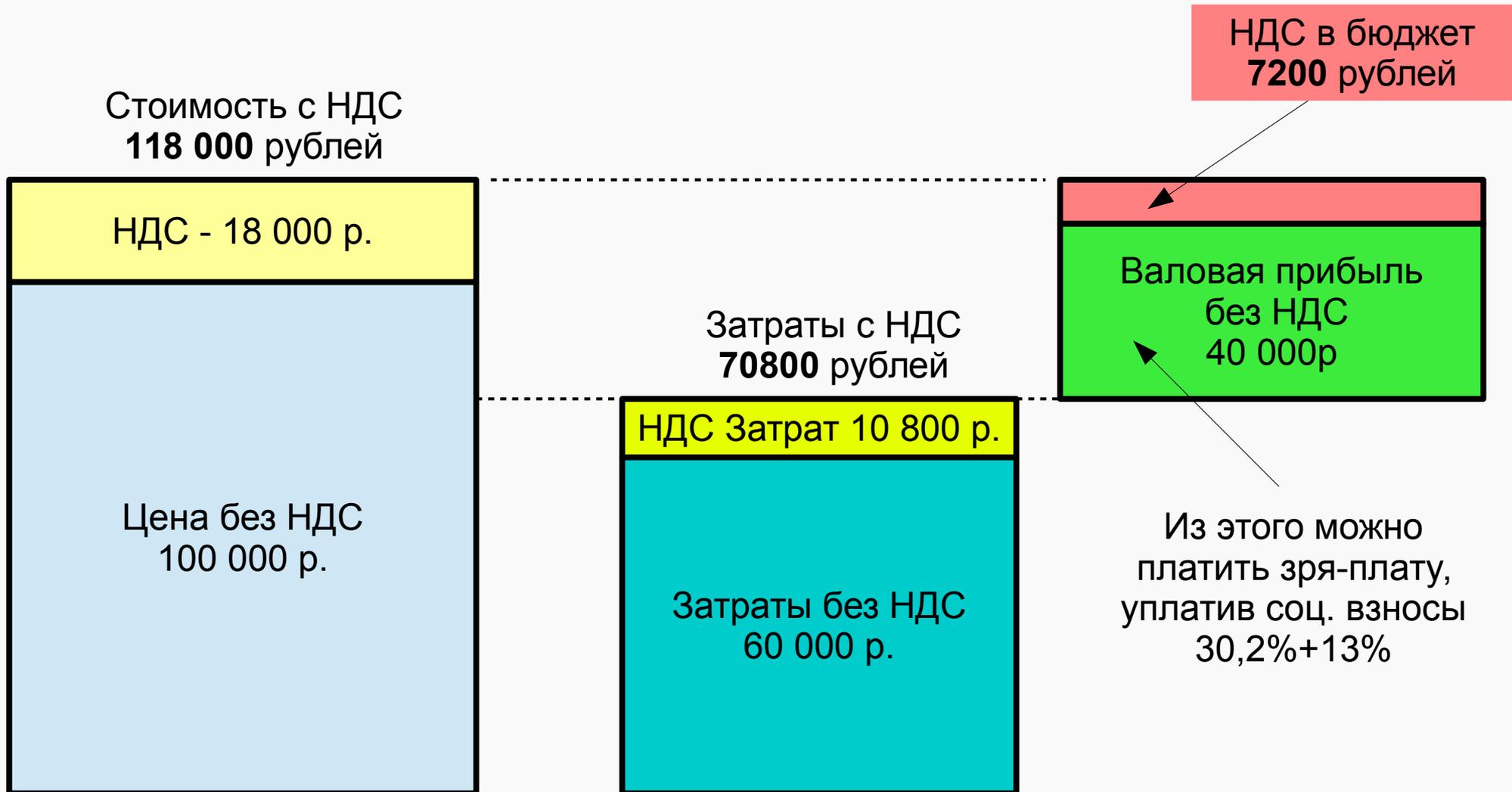
- Налоговый кодекс РФ
- Достаточно бюрократичен, запутан и неоднозначно трактуется налоговыми органами
- Ошибка, неточность -> штраф
- Разные системы налогообложения
- Подвержен сильным изменениям
- Для эффективной работы необходимы профессиональные юрист и бухгалтер



Общая система налогообложения 2015

- Налог на добавленную стоимость (18%)
 - Учитывает расходы (уменьшение НДС)
- Взносы в ПФРФ, ФССРФ, ФОМС
 - Общие 22%, 2,9%, 5,1% (>711 000 руб. 10%,0%,5,1%)
 - Разработка ПО, БД, инкубаторы (8%, 2%, 4%)
 - «Сколково» (14%, 0%, 0%)
 - 0,2% страхование от несчастного случая (разное для разных видов деятельности)
- Подоходный налог 13%
- Если не выплачивать зарплату, то необходимо заплатить налог на прибыль 20%
- Другие налоги, куча **ОТЧЕТОВ!**

НДС 18%: Налог на добавленную стоимость





Упрощенная система налогообложения 2015

- Разные модели (Доходы, Доходы-Расходы)
- Для разработки ПО — Доходы
- **Нет НДС!**
 - Невыгодно, если контрагент на *другой* системе налогообложения
 - Выгодно НДС<->НДС, НЕТ_НДС<->НЕТ_НДС
- ПФРФ, ФССРФ, ФОМС — такие же.
 - НО! ИП «за себя» - на основе МРОТ:
 $5965 * 12 = 71580 \text{руб.} + 1\% > 300000 \text{руб.}$
- 6% ставка налога УСН
- Отчеты - минимизированы

Что все эти цифры и буквы значат для вашего кармана?

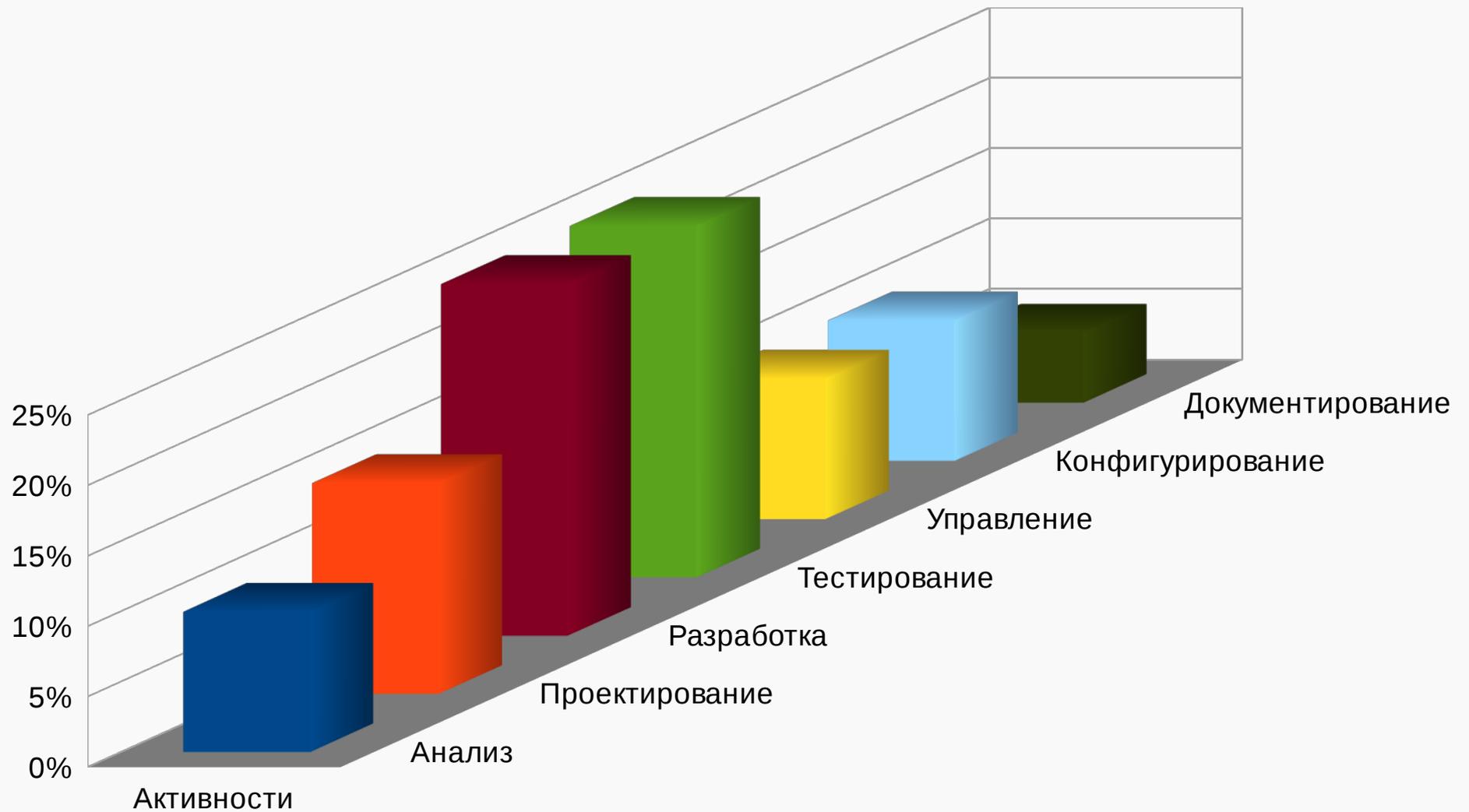
Клиент заплатил за разработку

ОСН	ИП на УСН*
118000 руб. / 18% НДС	* - 26+5,1% МРОТ в мес в ПФ
100000 руб. / 30,2% соц.	100000 руб. - 6% (УСН)
76804.91 руб. - 13%	94000 руб. 1% (>300000)
66820.27 руб. на руки.	94000 руб. на руки.
ИТОГО >51 тыс.руб. ~44%	Итого 6 тыс.руб. 6%

2

Оценка трудоемкости разработки программного обеспечения

Распределение затрат в проекте



Оценка трудоемкости — почему важно?

- Экономические причины:
 - Себестоимость
 - Планируемые затраты ~ Фактические затраты
 - Прибыль
- Маркетинговые причины:
 - Быстрая оценка входящих запросов
 - Репутация для следующих проектов
- Причины, связанные с персоналом:
 - Перегруженность
 - Недогруженность
 - Стимуляция

В чем оценивать трудоемкость ПО ?

- Строках кода KSLOC
 - Физические и логические
 - Хорошо подходит для языков ассемблерного типа
- Человеко-часах (месяцах, годах)
- Попугаях
 - Безразмерная сложность
- Функциональных точках

Оценка — вероятностное утверждение

- Невозможно дать точную оценку!
 - Сложно учесть все факторы
- Оффтопик 1:
 - В расчетах описывается плотностью вероятности
 - Центральная предельная теорема
- Оффтопик 2:
 - Сколько Россиян не пьют пива?
 - Температура плавления спирта?

- Менее формальные:
 - Наивный, на основе предыдущего опыта
 - Экспертная оценка
 - «Покер-планирование»
- Более формальные:
 - PERT (метод трех точек)
 - FPA IFPUG - метод функциональных точек (14 параметров, 150 стр.)
 - Use Case Points
 - Конструктивная модель стоимости COSOMO II (21 параметров, описание модели 90 стр.)
 - Методика Госкомтруда

- Проекты должны быть:
 - Однотипные
 - Построенные на известной технологии
- Оценка производится «главным программистом»
- Оценка обычно занижена, нужно умножить:
 - На 2
 - лучше на π .
 - Вспомним, что в проекте программирования всего 25% и умножим на 4.

Наивный: Пример

N	Функционал	Оценка, мин./ чел.час	Оценка, макс./ чел.час
		
21	Глобальный агрегатор всего и вся ("Я хочу...", "Хочу узнать..." и т.д.)	80	120
22	Интеграция с ЛК ("подключить услугу")	60	80
23	Базовый функционал (навигация + конфигуратор + список результатов)	20	24
24	Помощник по подбору (+ конфигуратор), интеграция с магазином ("положить в корзину", "оформить покупку")	15	30
		

- Разнообразные формальные методики
 - Можно применять элементы подходов из статического тестирования (см. 3 курс):
 - Сквозной контроль
 - Инспекции (Michael Fagan, IBM)
 - Wideband Delphi («Метод Дельфийских Оракулов»)
- Основная идея собирать формальные ответы экспертов
 - Анализ при помощи статистики
- Модерирование дискуссий

Wideband Delphi

- 1950-60 г. Olaf Helmer, Norman Dalkey, Nicholas Rescher
- Развитие методов мозгового штурма, без очных споров
- Анонимность, несколько туров, <20 экспертов
 - Между турами знакомство с анкетами других экспертов
- Составление подробных анкет для опроса
 - Обработка их статистически
- Назначение экспертам весовых коэф.

<http://phd-thesis.wikispaces.com/Detailed+summary+of+Delphi+literature>

<http://is.njit.edu/pubs/delphibook/index.html#toc>

- Используется в гибких методологиях
- Оценивает «User Story»
 - Необходим выбор «Эталонной истории»
- Участвует вся команда
- Несколько раундов
- Может занять значительное количество времени всей команды

Покер-планирование



PERT

- Program/Project Evaluation and Review Technique
- Booz Allen Hamilton, 1958 г.
- Проект ракеты UGM-27 «Polaris»
- Простой
- Реализован в M\$ Project
- Использует вероятностный подход
- Вход — список элементарных работ E_i

PERT: оценка — вероятностное утверждение



M_i — Наиболее вероятная оценка трудозатрат.

O_i — Оптимистичная оценка. Так бывает только в кино.

P_i — Пессимистическая оценка. Наступили на все возможные грабли.

PERT: формулы

- Оценка средней трудоемкости
 $E_i = (P_i + 4M_i + O_i)/6$
- Среднеквадратичное отклонение
 $CKO_i = (P_i - O_i)/6.$
- Общая оценка статистически независимых работ (вспомним ЦПТ!): $E = \sum E_i$
- Среднеквадратичное отклонение для оценки суммарной трудоемкости: $CKO = \sqrt{\sum CKO_i^2}$
- Суммарная трудоемкость проекта (с вероятностью 95%)
 $E_{95\%} = E + 2 * CKO.$

PERT: Пример

- Нужен ли здесь действительно пример?
-

FPA IFPUG - метод функциональных точек

- Alan Albrecht, 1979 г.
- Использует логическую модель программного продукта
 - Не зависит от платформы и технологии
- ISO/IEC 20926:2009 (IFPUG CPM 4.3)
- Единообразный подход ко всем проектам
- Множество расширений и видоизменений под конкретные технологии

Самостоятельно изучить

Use Case Points

- Gustav Karner, 1993
- Центрирована вокруг прецедентов использования
 - Use Case диаграмма с экторами и прецедентами использования
 - Use Case description document
- Опирается на Objectory process by Ivar Jacobson
 - Один из предшественников RUP
- Может быть использована в рамках *UP
 - RUP, AUP, OpenUP, EUP....
- Упрощен в отношении декомпозиции данных (нет DET, RET, ... и пр.)

$$UCP = UUCP \times TCF \times ECF \times PF$$

- UUCP — Unadjusted (невыравненные) Use Case Points
 - $UUCP = UUCW + UAW$
 - UUCW – невыравненный вес прецедента
 - UAW – невыравненный вес эктора
- TCF — Technical Complexity Factor — факторы технической сложности
- ECF — Environment Complexity Factor — факторы сложности окружения
- PF — Productivity Factor — учитывает прошлые проекты, необходимо калибровать.

UAW: определение веса экторов

$$UAW = \sum_{i=1}^3 AW_i \times N_i$$

Для подсчета умножаем Вес на количество (N) акторов данного типа

Сложность	Описание	Вес (AW _i)	N _i	*
Простая	Внешняя система с доступным извне API	1	0	0
Средняя	1. Внешняя система с заданным протоколом 2. Пользователь с командным интерфейсом	2	0	0
Высокая	Пользователь с графическим интерфейсом	3	5	15

=15

UUCW: определение веса прецедентов

$$UUCW = \sum_{i=1}^3 UCW_i \times N_i$$

Сложность	Описание	Вес (UCWi)	Ni	*
Простая	Интерфейс — простой (н-р CRUD); использует одну таблицу БД; успешный сценарий включает 3 или менее шагов (включая альтернативные сценарии); 5 или менее классов анализа	5	10	50
Средняя	Интерфейс — продвинутый; использует 2 или более таблиц БД; от 4-х до 7-ми шагов, от 5 до 10 классов анализа	10	25	250
Высокая	Сложный интерфейс; более 3 таблиц; более 7-ми прецедентов; более 10-ти аналитических классов	15	5	75

=375

ТСФ: определение веса технических факторов

TF	Описание	Вес W_i	Слж. F_i	*
T1	Распределенность системы	2	1	2
T2	Производительность	1	3	3
T3	Эффективность для пользователя	1	1	1
T4	Сложная внутренняя обработка	1	5	5
T5	Повторное использование кода	1	0	0
T6	Простота установки	0.5	0	0
T7	Простота использования	0.5	4	2
T8	Переносимость	2	0	0
T9	Простота изменений	1	3	3
T10	Многопоточность	1	0	0
T11	Дополнительные возможности безопасности	1	0	0
T12	Доступ к другим системам (н-р B2B)	1	3	3
T13	Необходимы тренажеры для пользователей	1	0	0

Слж. F_i - фактор субъективной сложности (от 1 до 5)

=19

ТСФ: определение веса технических факторов 2

$$TCF = C_1 + C_2 \sum_{i=1}^{13} W_i * F_i$$

- Формула ограничивает ТСФ от 0.6 до 1.3
 - $C_1 = 0.6$; $C_2 = 0.01$
- W_i — вес фактора
- В нашем случае $TCF = 0.6 + 0.01 * 19 = 0.79$
- F_i — фактор субъективной сложности выбирается группой разработчиков на основании своего опыта и восприятия сложности проекта
 - 1 - просто, 5 - сложно

ЕСФ: определение веса факторов окружения

TF	Описание	Вес W_i	влн. F_i	*
E1	Уверенное использование UML/RUP	1.5	4	6
E2	Кол-во работников на неполный рабочий день	-1	0	0
E3	Опытность аналитика	0.5	5	2.5
E4	Опыт работы с приложениями	0.5	0	0
E5	Опыт ОО разработки	1	5	5
E6	Мотивация	1	3	3
E7	Сложный язык разработки	-1	0	0
E8	Неизменность требований	2	1	2

=18.5

влн. F_i - фактор субъективного влияния:

- 1 — сильное негативное влияние
- 3 — среднее
- 5 — сильное позитивное влияние



ЕСФ: определение веса факторов окружения 2

$$ЕСФ = C_1 + C_2 \sum_{i=1}^{13} W_i * F_i$$

- Формула ограничивает ЕСФ от 0.425 до 1.4
 - $C_1 = 1.4$; $C_2 = -0.03$
- W_i — вес фактора
- В нашем случае $ЕСФ = 1.4 - 0.03 * 18.5 = 0.845$
- F_i — фактор субъективного влияния выбирается группой разработчиков на основании своего опыта и восприятия влияния фактор на проект

УСР: ИТОГО

- $УСР' = (375 + 15) * 0.79 * 0.845 = 260.3445$
 - 260 УСР..... и ЧЁ (WTF) ?
 - А где PF? А что такое калибровать?
- $УСР = УСР' * PF$

PF - (фактор продуктивности) - приводит УСР к размерным единицам

 - к человеко-часам
 - человеко-месяцам
 - человеко-тысячелетиям

UCP: Калибровка

- Данный проект занял 2100 часов
$$PF = UCP / UCP' = 2100 / 260 = 8.07$$
- Индустриальные данные говорят о разбросе данных от 5 до 30 %
 - Проекты с высокой ошибкой получили высокую ошибку и у группы экспертов
- PF зависит от масштаба проекта
 - Необходимо исследовать соответствие $UCP \sim KSLOC$

- Constructive Cost Model, Barry Boehm,
 - 1981 — первая модель
 - 1997 — текущая версия
- Большое количество формул, параметров и коэффициентов
- Самостоятельно ! Для знакомства:
http://citforum.ru/SE/project/arkhipenkov_lectures/13.shtml
- На контрольной будет!

3

Элементы организационного дизайна

Быть или не быть?

- Какая основная цель коммерческой организации?
- А организации - разработчика программ?
- Фриланс?





ИТМО ВТ

Действующие «лица» организации - разработчика ПО

- Генеральный директор — единоличный исполнительный орган
 - Совет директоров, учредители (акционеры)
- Главный бухгалтер (им может быть и ген. дир.)
 - Бухгалтерия (бумажки, отчеты)
- Отделы разработки ПО, тестирования, архитектуры, админы, хелп-деск, ... и пр.
- Продажи и маркетинг
- Кадры (Human Resources)
- Юридический отдел
- Безопасность и др.



Элементы организационного дизайна

- Организация — группа людей, деятельность которых сознательно координируется для достижение общих целей
- Стратегия — деловая концепция организации на заданную перспективу
- Разработка программных систем — обычный бизнес-процесс, подверженный всем экономическим законам

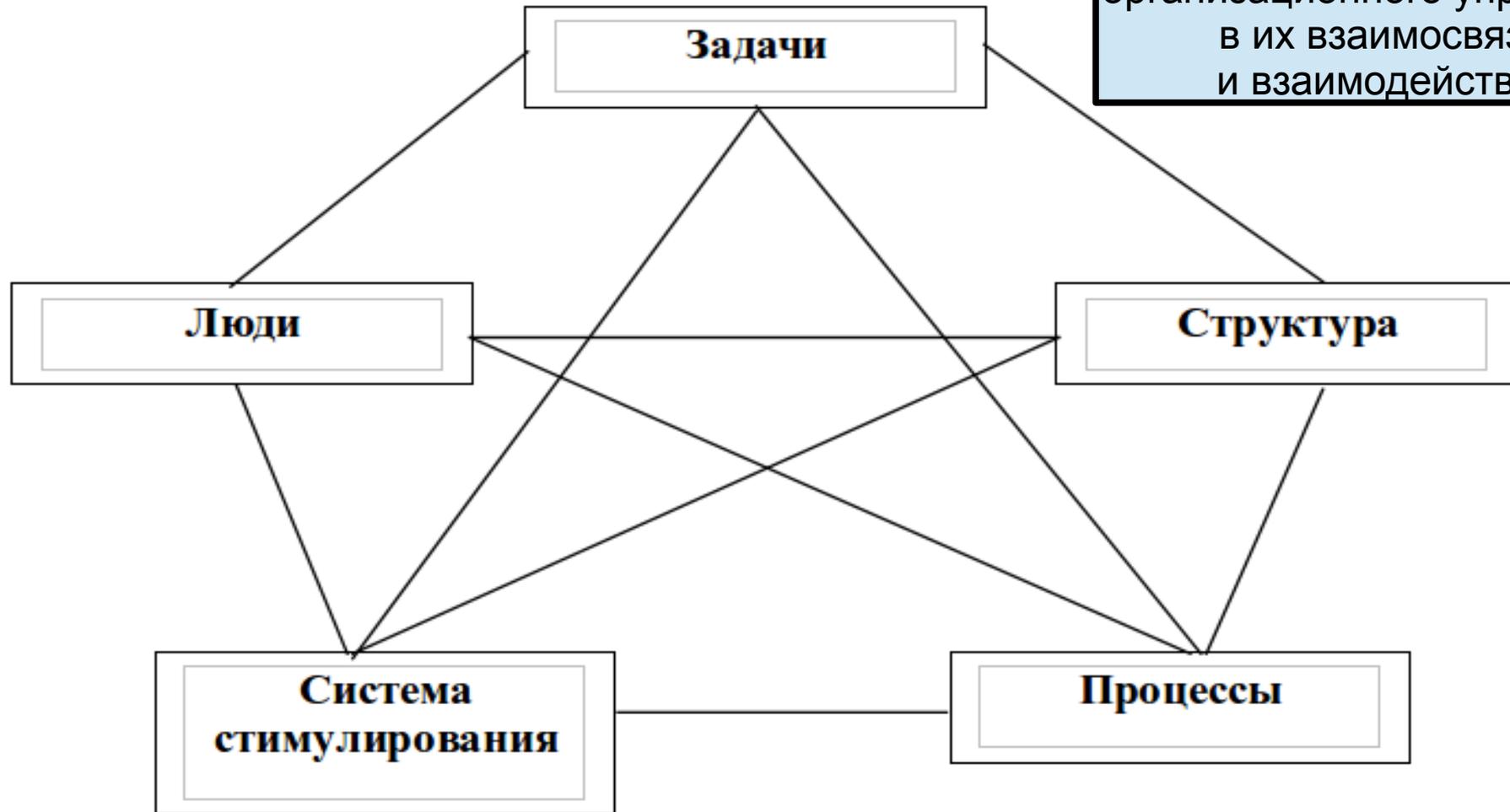


Вопросы при создании или изменении структуры организации

- Какие задачи стоят перед организацией?
- Кто конкретно и сколько людей отвечают за выполнение задач? Как их сгруппировать?
- Какие правила, процедуры, механизмы требуются для решения?
- Как заставить стимулировать людей работать?
- Какие люди требуются? (опыт, умения, квалификация)

Модель организационного дизайна

Организационный дизайн:
совокупность блоков
организационного управления
в их взаимосвязи
и взаимодействии



- Борисова Л.Г. «Организационный дизайн и современные концепции управления»

- Задачи/люди
- Задачи/стимулирование
- Неформальная организация/люди
- Люди/структура/задачи
- Задачи/структура/процессы/система стимулирования
- Люди/система стимулирования/задачи

Симптомы проблем организационного дизайна

- Конфликты (программисты, такие программисты)
- Слишком высокий уровень принятия решений
- Сложности координации
- Недостаток внимания важнейшим задачам организации
- Отсутствие четкости в определении круга обязанностей сотрудников

- Способность быстро разрабатывать и выводить на рынок новые продукты в условиях постоянно меняющихся требований потребителей;
- Способность улавливать новые возможности развития компании, проистекающие из изменений во внешней среде, и реализовывать эти возможности;
- Способность создавать стратегические и временные партнерства и эффективно управлять ими;
- Способность к реализации новых проектов с повышенным риском;
-



Типичные организационные структуры

- Простая
- Функциональная
- Дивизионная
- Гибридная (функциональная и дивизионная)
- Матричная

Простая структура (стартап)



- ✓ простой контроль
- ✓ быстрое принятие решений
- ✓ простые и неформальные мотивации, вознаграждения и контроля
- ✗ высокие требования к директору
- ✗ не способствует развитию менеджеров изнутри
- ✗ директор занят ежедневными проблемами а не стратегией
- ✗ препятствие к росту организации

Функциональная структура (н-р Web, «железячники»)



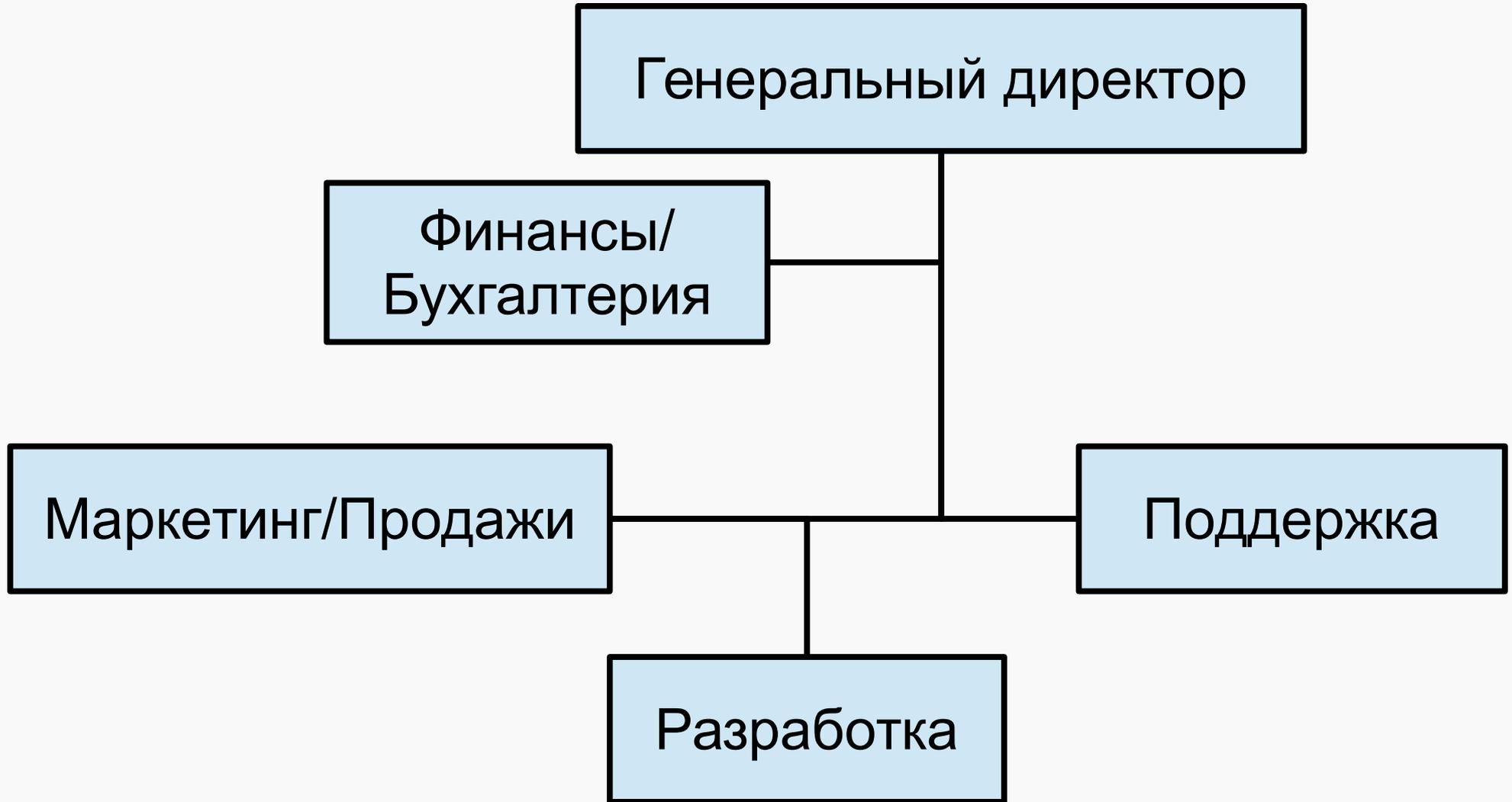
Генеральный директор

Финансы/
Бухгалтерия

Маркетинг/Продажи

Поддержка

Разработка

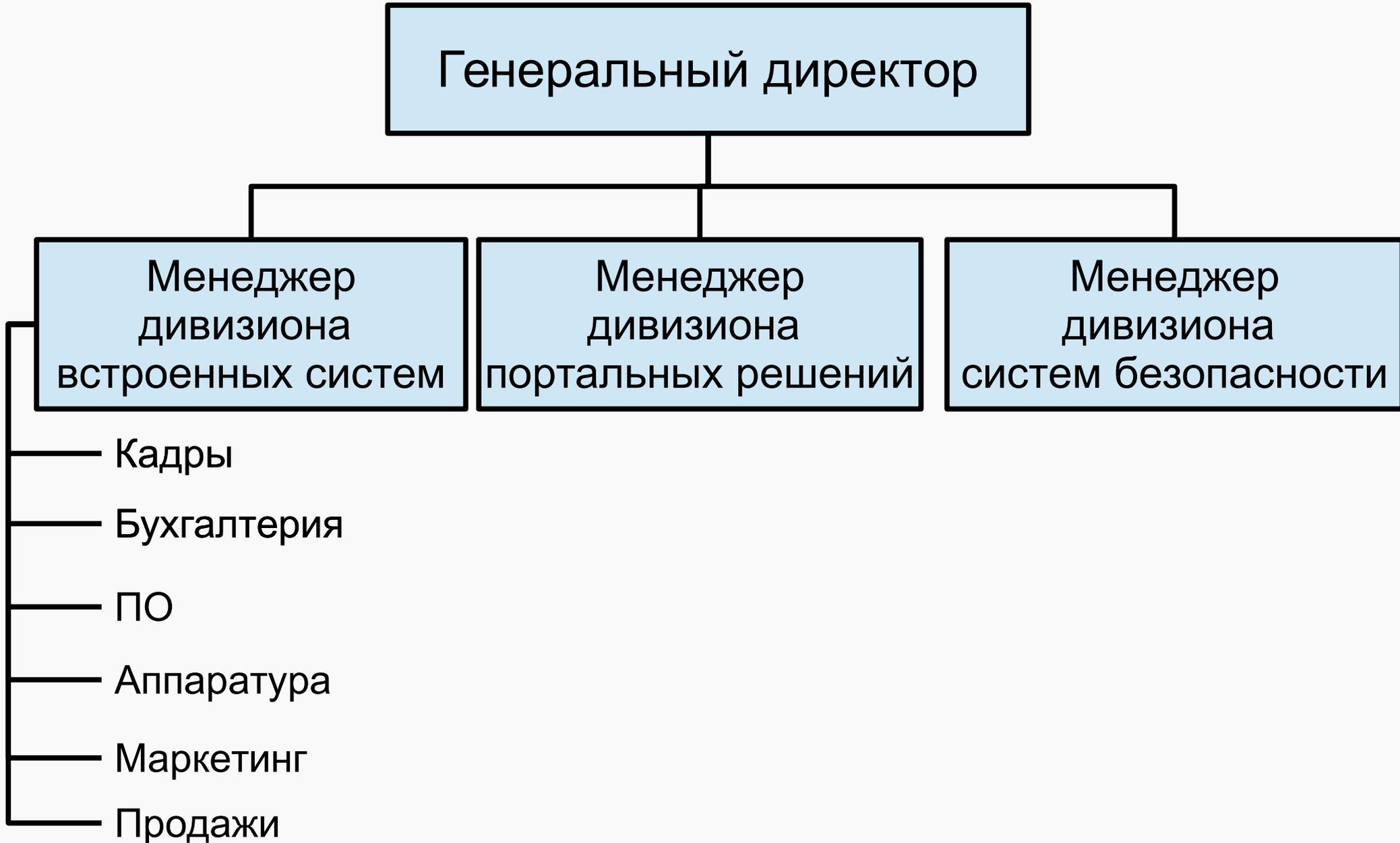


Функциональная структура: за и против



- ✓ Повышает эффективность за счет специализации
 - ✓ Опыт и профессионализм
 - ✓ Возможность делегирования оперативных решений
 - ✓ Контроль над стратегическими решениями
 - ✓ Рост профессионалов
 - ✓ Эффективное использование людей, ресурсов
- х Узкая специализация и почва для конфликтов
 - х Координация нескольких подразделений
 - х Конфликт между линейными и штабными ф-циями
 - х Ограничивает развитие менеджеров внутри компании

Дивизионная структура



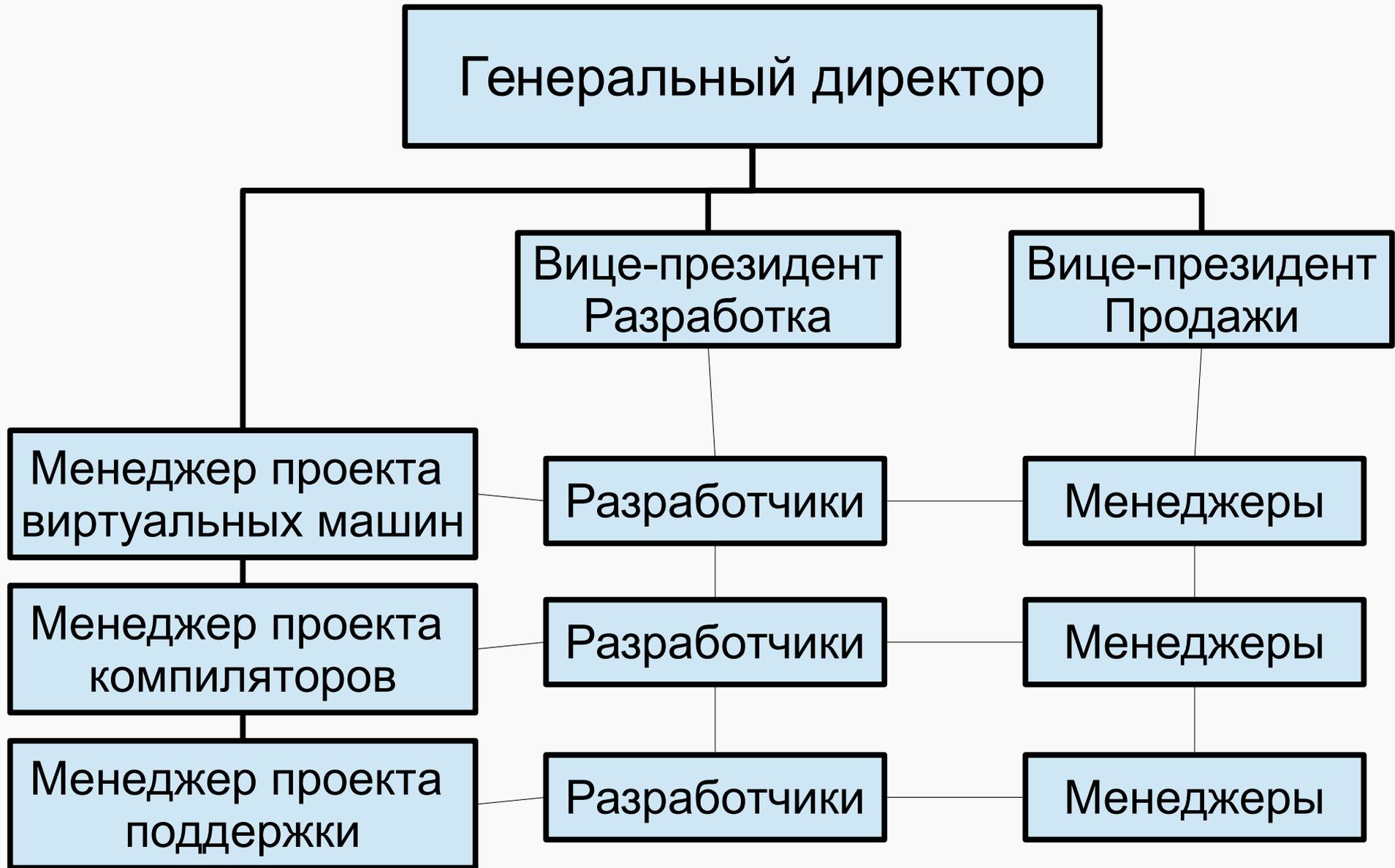


Дивизионная структура: за и против

- ✓ Реакция на требования рынка
- ✓ Высшие руководители посвящают больше времени на стратегическое планирование
- ✓ Ответственность и контроль руководителей дивизионов
- ✓ Сохраняет функциональную специализацию внутри подразделений
- ✓ Развитие стратегических менеджеров

- х Конкуренция за ресурсы между дивизионами
- х Степень полномочий менеджеров дивизионов
- х Несовместимость действий дивизионов
- х Проблема распределения накладных расходов между дивизионами

Матричная структура



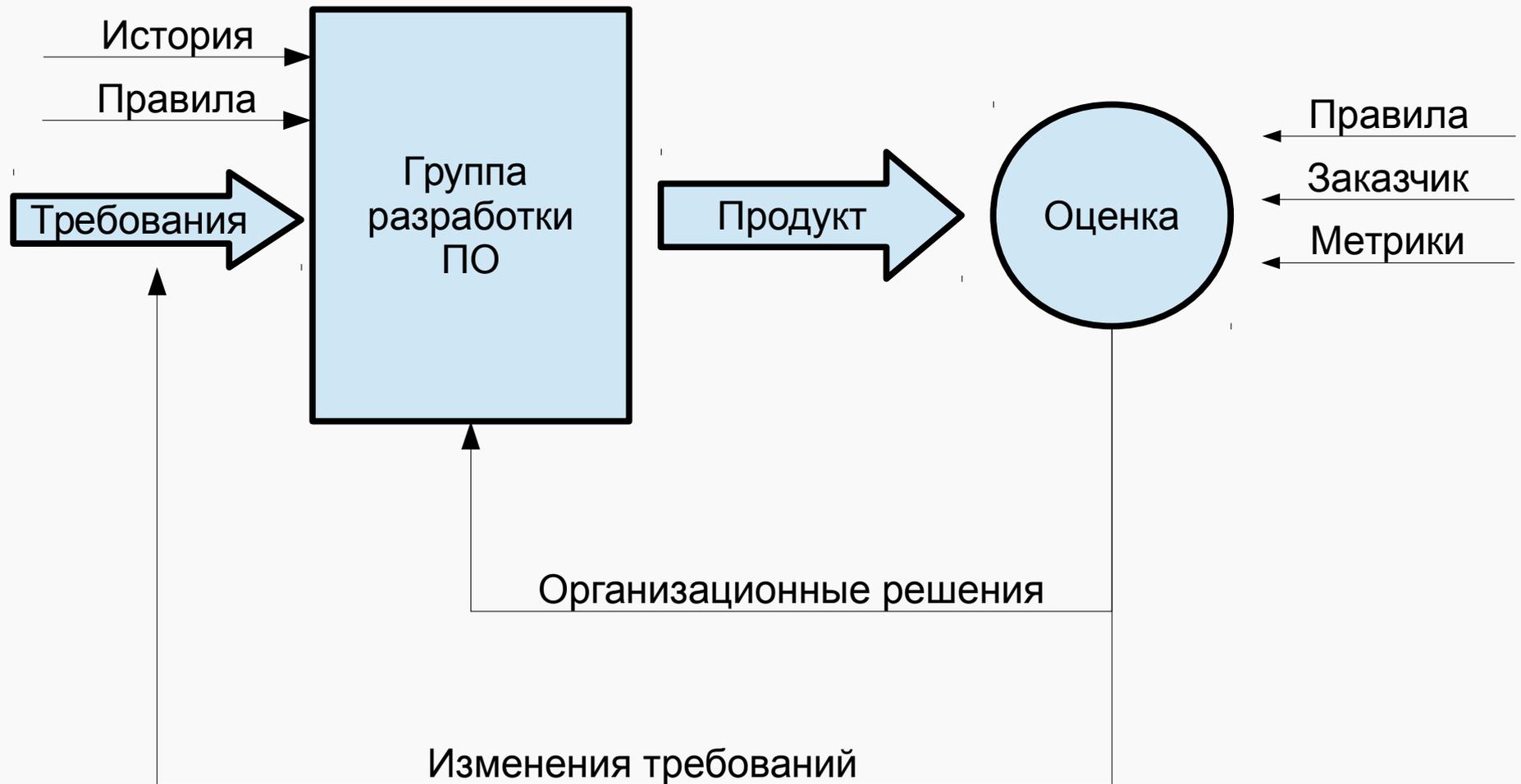
- ✓ Широкий спектр деятельности
- ✓ Создание эффективных команд
- ✓ Развитие стратегических менеджеров
- ✓ Эффективное и гибкое использование человеческих ресурсов
- ✓ Творчество и инновации
- ✓ Вовлечение менеджеров среднего звена в решение стратегических вопросов

- х Двойное подчинение
- х Горизонтальная и вертикальная координация

4

Контроль производства ПО

Модель управления разработкой ПО



Скажи, как ты будешь измерять мою эффективность, и я скажу, как я буду себя вести. Э. Гольдратт

Проблемы менеджеров разработки ПО

- Оплата труда и объективные данные о производительности работников
 - KSLOC?
 - Отработанные часы?
 - Реализованный функционал?
- Контроль сроков и бюджетов проекта
 - Диаграмма Ганта, сгорания, ...
- People менеджмент
- Другие

- Системы управления потоком работ
 - Задачи и ресурсы
 - Время выполнения
- Системы контроля за действиями работника
 - ПО на компьютерах работников
 - Формирует метрики автоматически
 - Персональные (<http://pomodorotechnique.com>)
- Системы плановой оценки сотрудников
- Анализ систем контроля версий

- Реализации на любой вкус:
 - Wrike, Megaplan,
- Развитые внутренние средства контроля
 - Всевозможные напоминания
 - Построение диаграмм
 - Интеграция с почтой
- Стоят ~немного~ денег
- В небольших командах и стартапах обычно не используются
 - Значительные усилия на ведение информации

Системы контроля потока задач: Пример интерфейса

Задачи История Дополнительно

⚠ Мои - 6 Свободные - 70 Созданные - 5 ⚠ Закрытые - 44 Отслеживаемые - 3 Изменённые - 96 Новая

Поиск

Фильтр задач

Мои задачи

Статус	#	Заявки на обслуживание	Компания	Создал	Исполнитель	Дата ▲	Описание
выполнена	2700				Смирнов Н. В.	26/06/2010 19:00	
открыта	3692			Цопа Е. А.	Цопа Е. А.	28/02/2010 20:57	
выполнена	174422			Цопа Е. А.	Цопа Е. А.	05/02/2012 01:15	
выполнена	174427			Цопа Е. А.	Цопа Е. А.	05/02/2012 01:17	
выполнена	174442			Цопа Е. А.	Цопа Е. А.	05/02/2012 04:16	
выполняется	174447			Цопа Е. А.	Цопа Е. А.	05/02/2012 04:18	
выполняется	174433			Цопа Е. А.	Цопа Е. А.	05/02/2012 01:18	
выполняется	173795			Цопа Е. А.	Цопа Е. А.	31/12/2012 23:59	
выполнена	173739			Цопа Е. А.	Цопа Е. А.	31/12/2012 23:59	
открыта	173729			Цопа Е. А.		29/02/2012 23:59	
выполнена	173006			Цопа Е. А.	Совинский В. М.	22/01/2012 01:46	
выполнена	173011			Цопа Е. А.	Совинский В. М.	22/01/2012 01:48	
выполняется	166970			Цопа Е. А.	Цопа Е. А.	30/06/2012 23:59	



Системы контроля действий работника

- Вызывают жесткое противостояние у существующих работников
- Нейтрально воспринимаются новыми
 - Подходит для стартапа
- Требуют установки ПО-агента на компьютер работника
- Фиксируют на основании шаблонов время работы пользователя в программах
- Производят оценку на продуктивное и непродуктивное время
- Для программистов могут быть использованы с оговорками

Системы контроля действий работника: crocotime.com



demo.crocotime.com/departments

<input type="checkbox"/>	Отдел	Общее продуктивное ▾	Общее непродуктивное	Количество прогулов	Количество опозданий
<input type="checkbox"/>	- Головной офис	3062 ч 72%	1208 ч 28%	0	356
<input type="checkbox"/>	+ Бухгалтерия	1247 ч 77%	369 ч 23%	0	141
<input type="checkbox"/>	+ Отдел логистики	957 ч 69%	438 ч 31%	0	115
<input type="checkbox"/>	+ IT-отдел	433 ч 68%	208 ч 32%	0	47
<input type="checkbox"/>	+ Отдел маркетинга	424 ч 69%	191 ч 31%	0	53
<input type="checkbox"/>	- Западный филиал	1419 ч 74%	488 ч 26%	0	145
<input type="checkbox"/>	+ Бухгалтерия	1283 ч 78%	371 ч 22%	0	142
<input type="checkbox"/>	+ Отдел продаж	135 ч 54%	116 ч 46%	0	3
<input type="checkbox"/>	- Южный филиал	849 ч 75%	281 ч 25%	0	76
<input type="checkbox"/>	+ Бухгалтерия	796 ч 80%	204 ч 20%	0	69
<input type="checkbox"/>	+ Отдел продаж	53 ч 23 мин 41%	76 ч 42 мин 59%	0	7
<input type="checkbox"/>	olenin	91 ч 13 мин 68%	42 ч 21 мин 32%	0	12
<input type="checkbox"/>	Администратор	88 ч 37 мин 74%	30 ч 34 мин 26%	0	11
	Итого	5511 ч 73%	2051 ч 27%	0	600

Показать

- Программы
- По дням
- Табель
- Экспорт

Действия

- [Добавить отдел](#)
- Переместить
- Удалить

Параметры

- Последние 30 дней
- с 11 марта
- по 9 апреля
- [Настроить колонки](#)

Группировка

- Список
- Иерархия

По программам

- [Все программы](#)

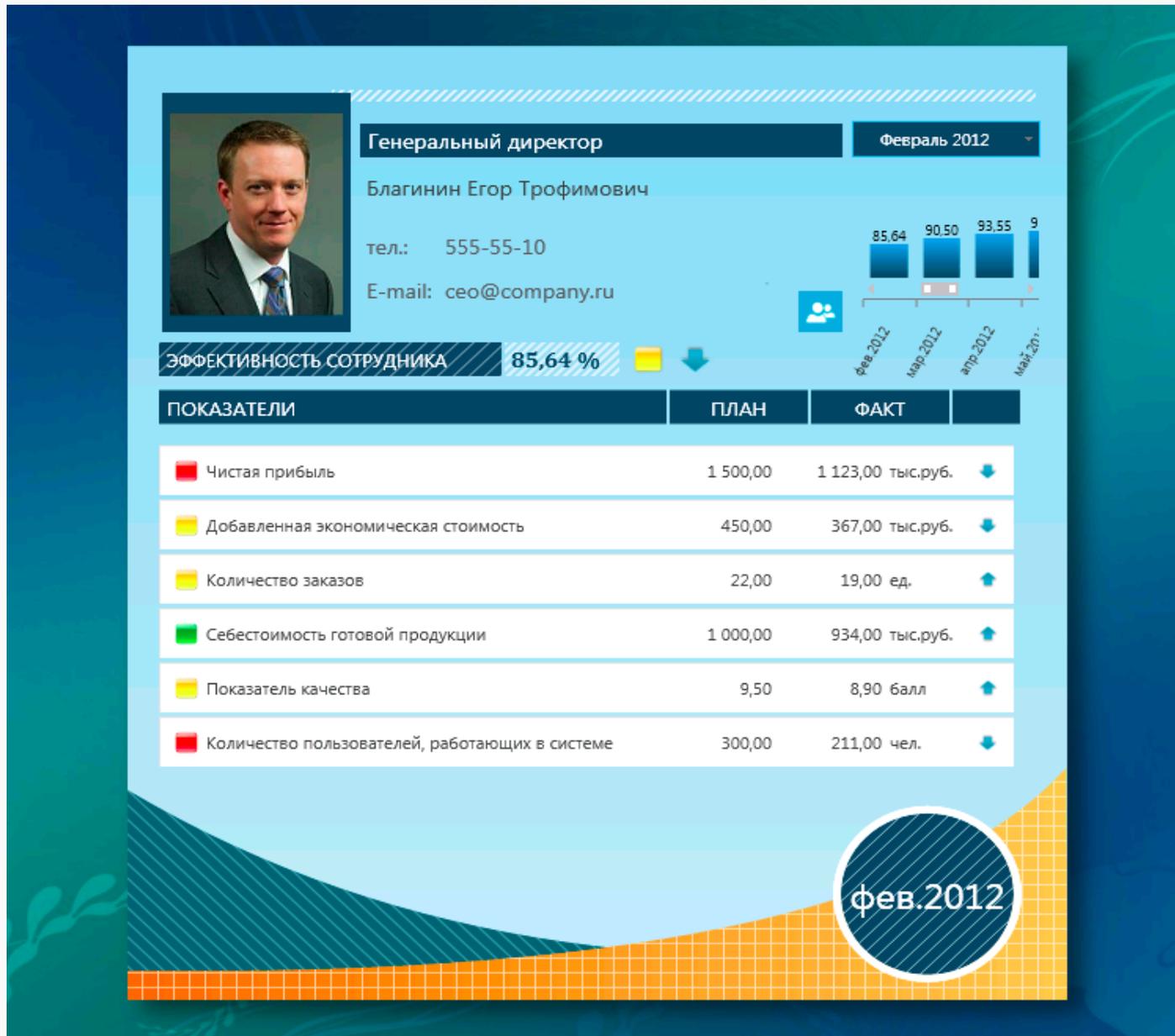




Системы плановой оценки сотрудников (используя KPI)

- Тарифная сетка / Grades
- KPI за период год/полугодие/месяц
- Ставит задачи работникам
 - Персональные
 - Локальные, уровня отдела или подразделения
 - Компании в целом
- Решение о премировании-депремировании
- Оценка производится:
 - Работником, начальником, коллегами, ... (360°)

Системы плановой оценки сотрудников (KPIsuite.com)



Анализ системы управления версиями

- В системах контроля версий есть «почти» вся необходимая информация:
 - Время, количество измененных строк, пользователь
 - Дополнить комментарий к коммиту идентификатором функционала, улучшения или дефекта
- <http://svnstat.sourceforge.net/>
 - Простая, можно легко доработать
- <http://programer.edicypages.com/>
 - Коммерческая

Диаграмма Ганта

- Разработал Генри Л. Гант в 1910 г.
- Связывает вместе и наглядно представляет
 - Задачи
 - Ресурсы
 - Время
- Не отражает значимости и ресурсоемкости работ
- Хорошо известна, более 9000 реализаций
 - Ganttter.com

Диаграмма Ганта: пример

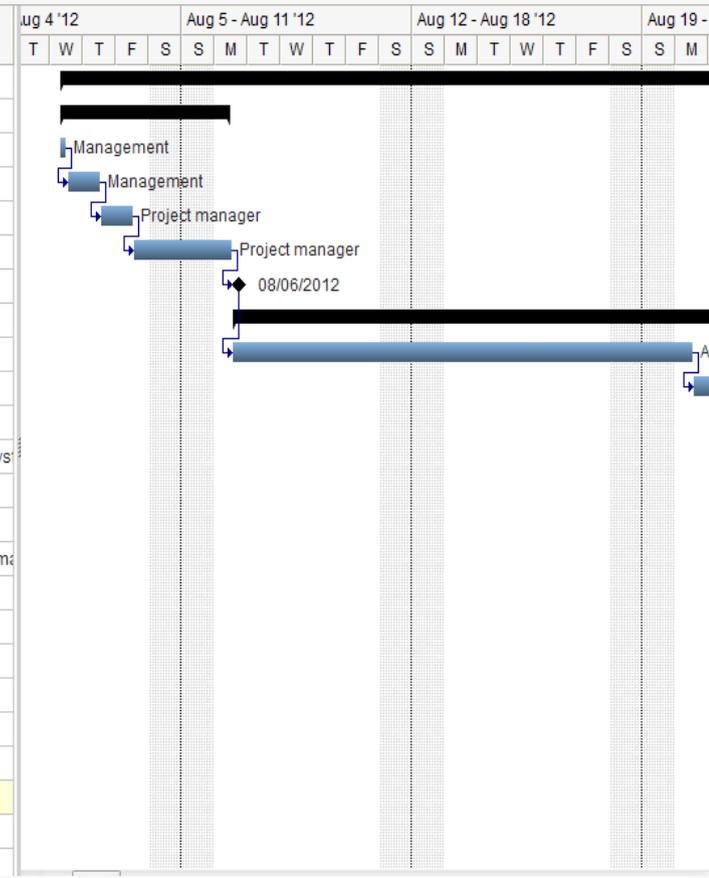
gantter Example

Save to Google Drive

Project Edit View Actions Help



ID	Name	Duration	Start	Finish	Predecessors	Resources
1	Sample Software Development Schedule	125d	08/01/2012	01/22/2013		
2	Scope	3.5d	08/01/2012	08/06/2012		
3	Determine project scope	4h	08/01/2012	08/01/2012		Management
4	Secure project sponsorship	1d	08/01/2012	08/02/2012	3	Management
5	Define preliminary resources	1d	08/02/2012	08/03/2012	4	Project manager
6	Secure core resources	1d	08/03/2012	08/06/2012	5	Project manager
7	Scope complete	0d	08/06/2012	08/06/2012	6	
8	Analysis/Software Requirements	28d	08/06/2012	09/13/2012		
9	Conduct needs analysis	10d	08/06/2012	08/20/2012	7	Analyst
10	Draft preliminary software specifications	3d	08/20/2012	08/23/2012	9	Analyst
11	Develop preliminary budget	2d	08/23/2012	08/27/2012	10	Project manager
12	Review software specifications/budget with team	4h	08/27/2012	08/27/2012	11	Project manager,Analyst
13	Incorporate feedback on software specifications	1d	08/28/2012	08/28/2012	12	Analyst
14	Develop delivery timeline	1d	08/29/2012	08/29/2012	13	Project manager
15	Obtain approvals to proceed (concept, timeline, budget)	4h	08/30/2012	08/30/2012	14	Management,Project ma
16	Secure required resources	10d	08/30/2012	09/13/2012	15	Project manager
17	Analysis complete	0d	09/13/2012	09/13/2012	16	
18	Design	23d	09/13/2012	10/16/2012		
19	Review preliminary software specifications	2d	09/13/2012	09/17/2012	17	Analyst
20	Develop functional specifications	21d	09/17/2012	10/16/2012		
21	Use Case 1	2d	09/17/2012	09/19/2012	19	Analyst
22	Use Case 2	3d	09/19/2012	09/24/2012	21	Analyst
23	Use Case 3	1d	09/24/2012	09/25/2012	22	Analyst
24	Use Case 4	1d	09/25/2012	09/26/2012	23	Analyst





ИТМО ВТ

К делу не относится: Scrum

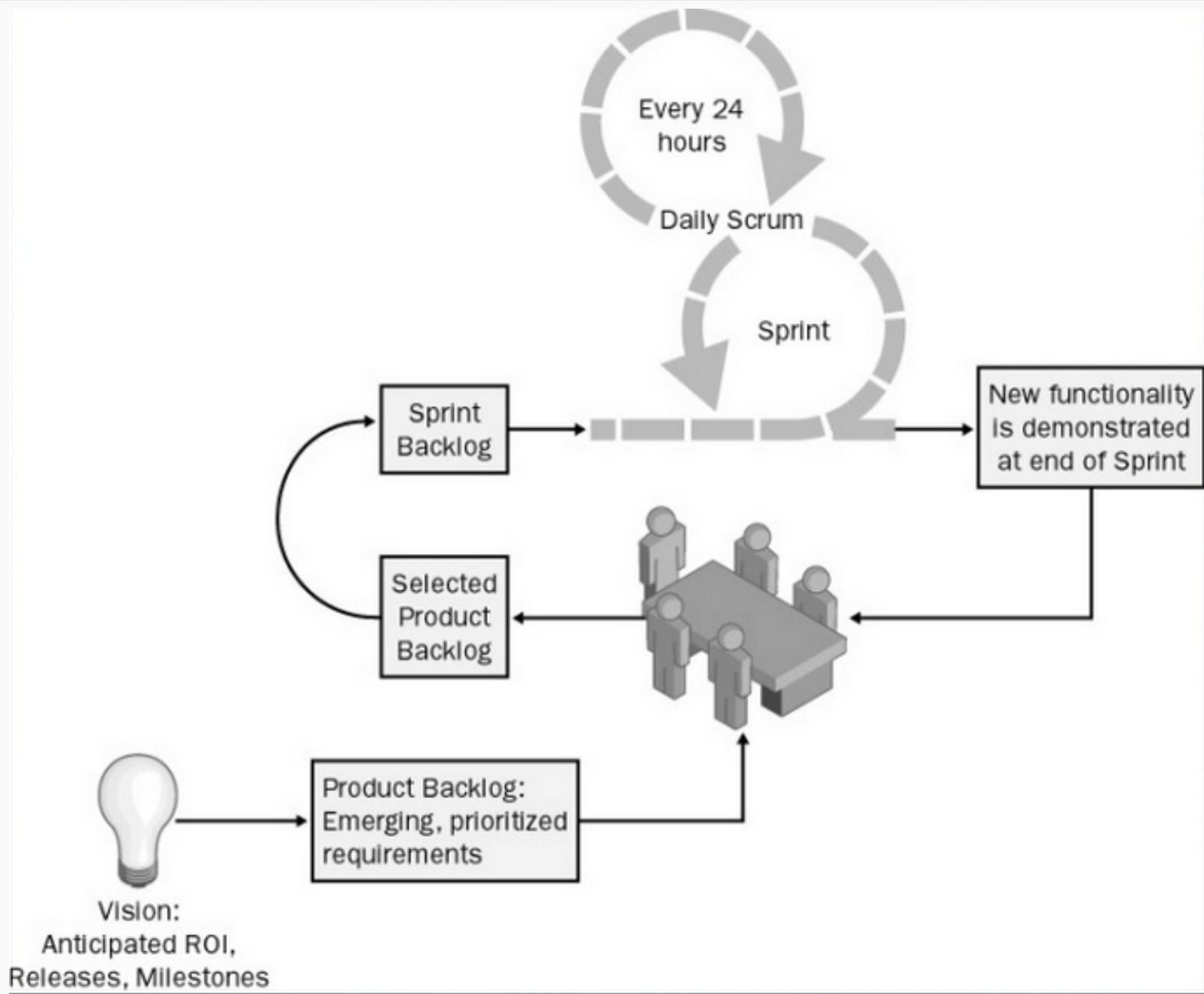
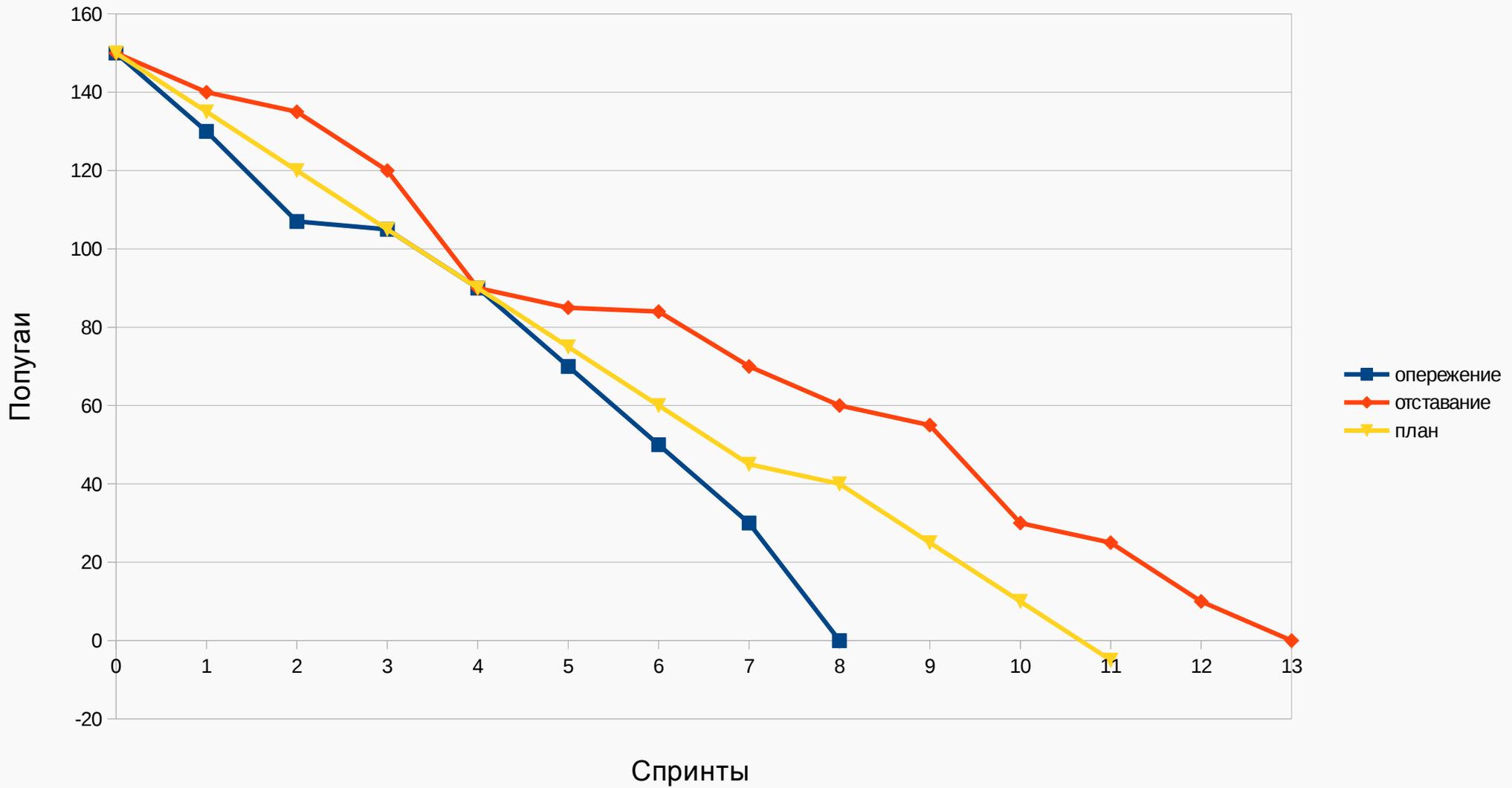


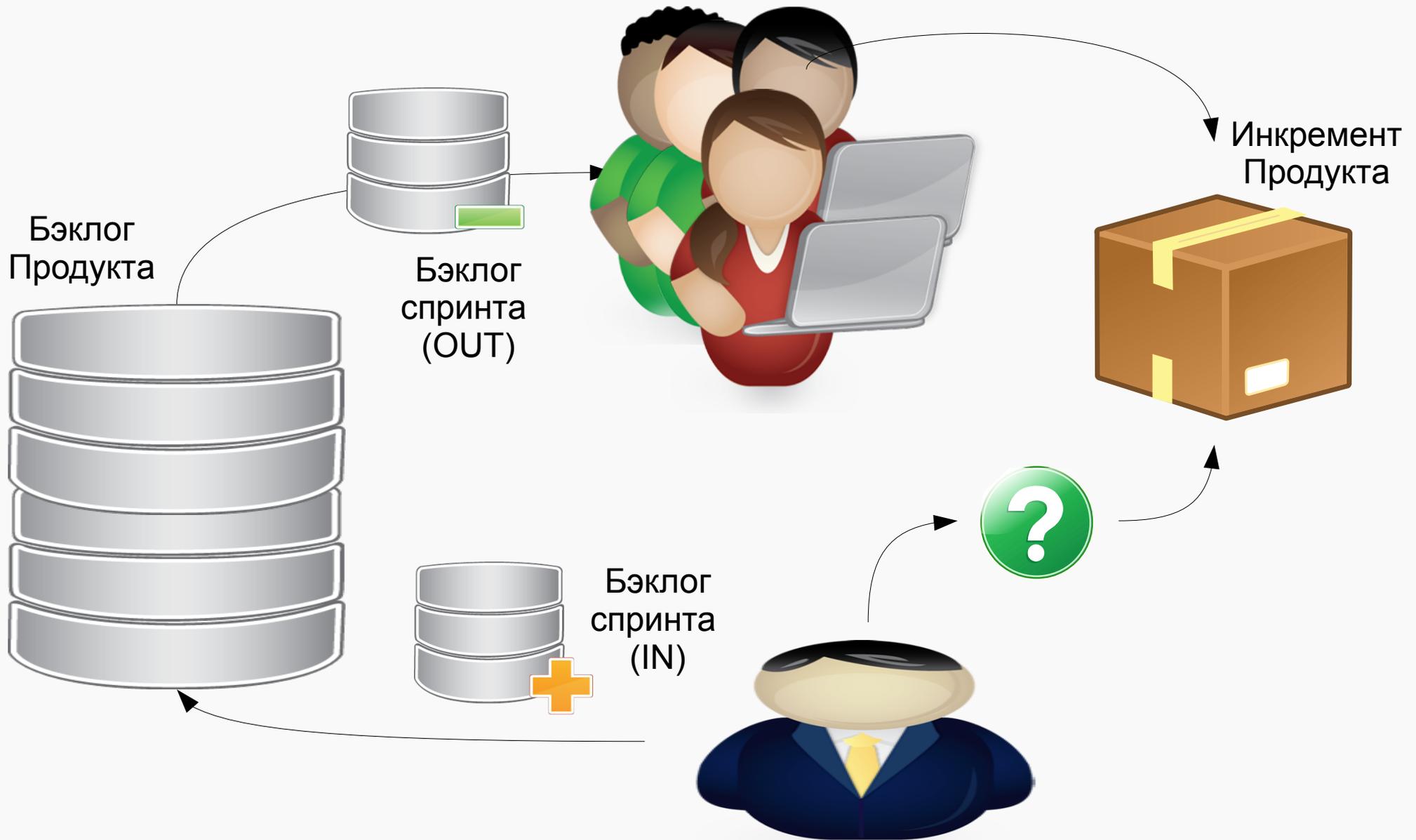
Диаграмма сгорания (Burndown)

- Используется в Scrum (и др. Agile-методологиях)
- Позволяет оценить отставание или опережение от идеального графика
- Для спринта и для полного продукта
- Должна учитывать все обстоятельства
 - Отпуска, болезни, праздники, форсмажоры
- Оперировать термином «производительность» команды

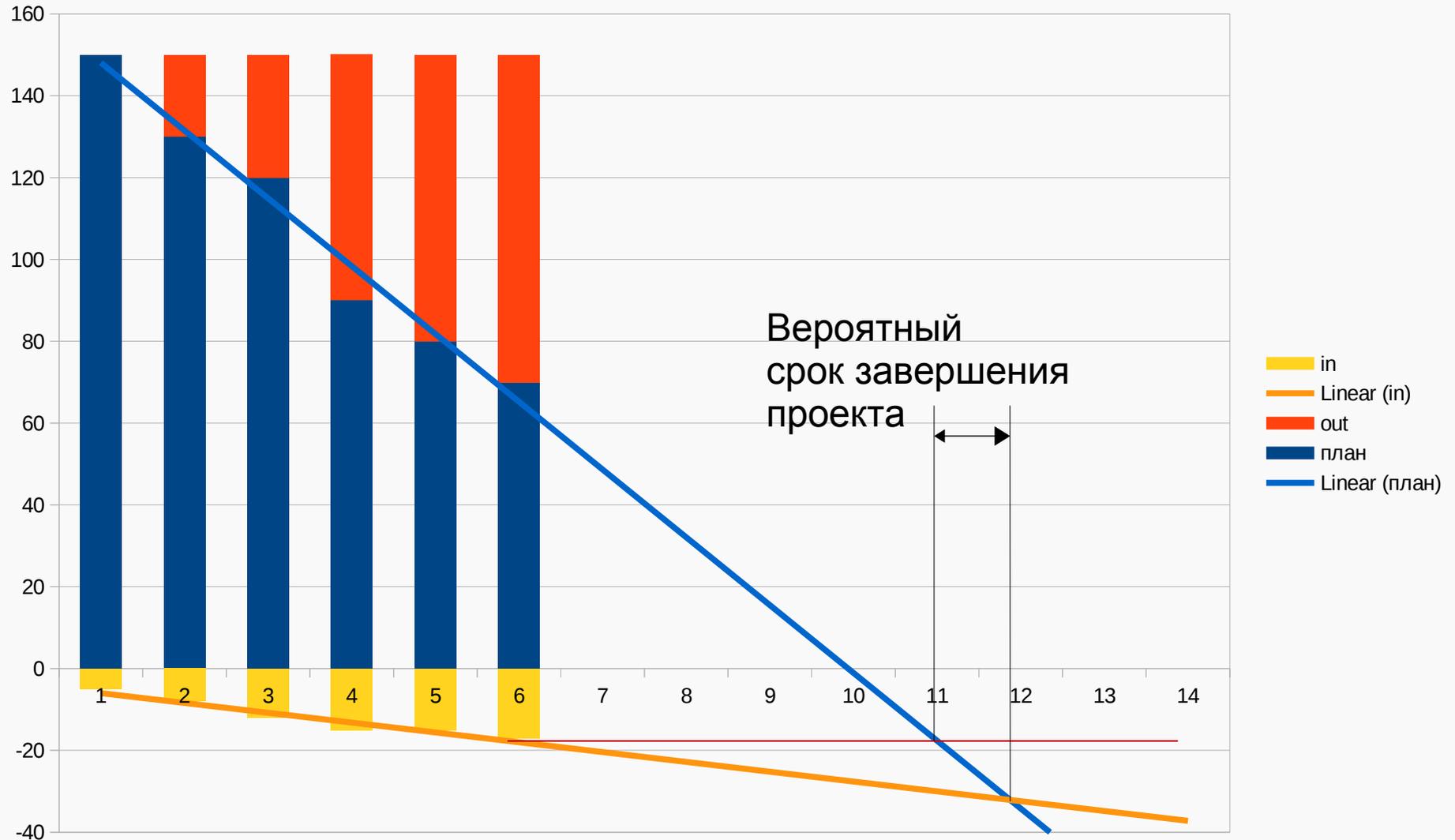
Диаграмма сгорания



Улучшенная диаграмма сгорания: учет изменений требований



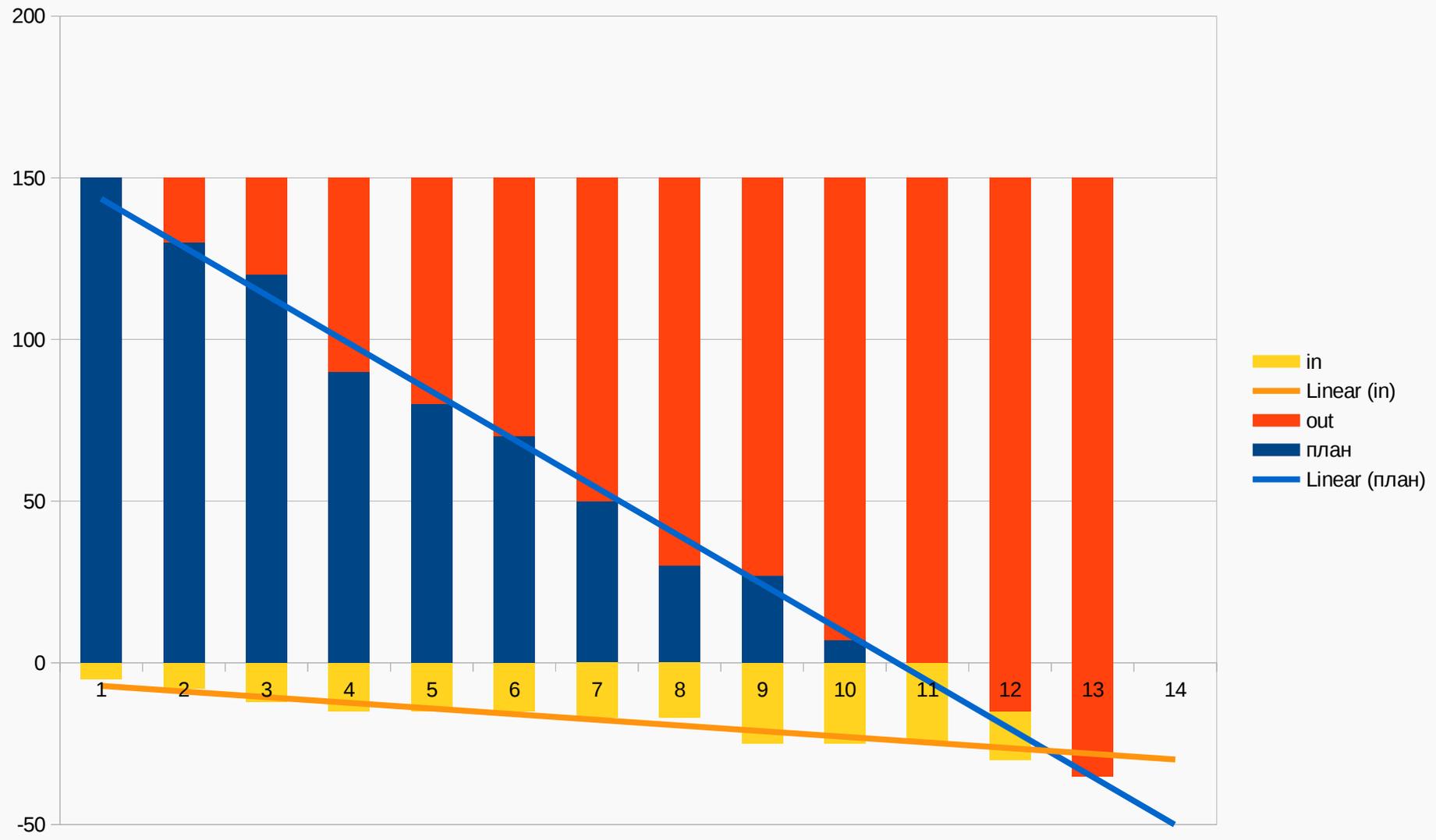
Улучшенная диаграмма сгорания



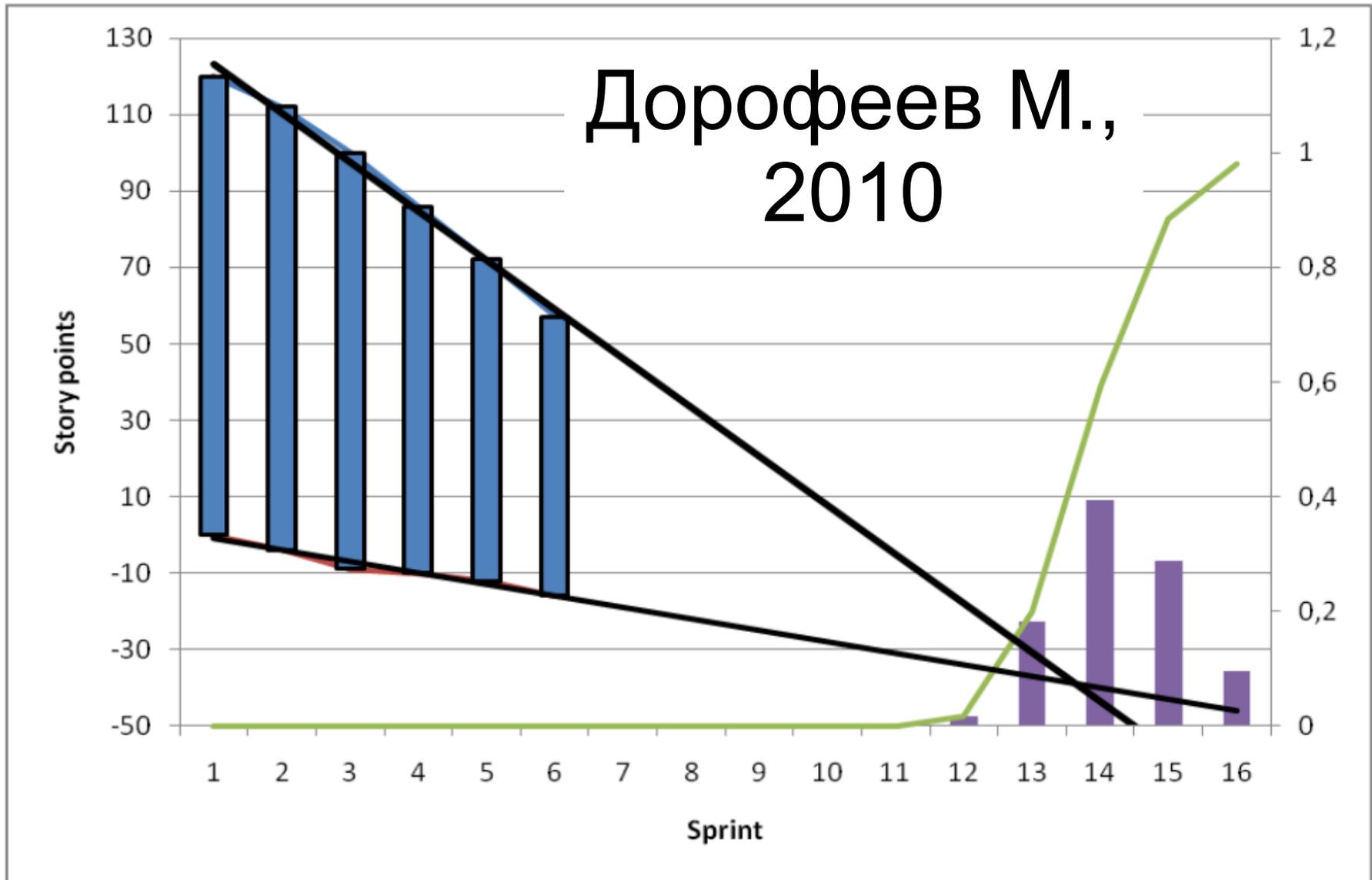


Улучшенная диаграмма сгорания

ИТМО ВТ



Улучшенная диаграмма сгорания + вероятностный подход



- Самое «тонкое» управление в принципе
- Классические методы:
 - «Армейский», «Кнута и пряника», «Разделяй и властвуй», метод «Черного Властелина» ...
- Пирамида Маслоу, 1954 г. «Мотивация и личность»
- Д. Макгрегор 1960 г. Теория X и теория Y
 - <http://www.grandars.ru/college/psihologiya/teoriya-makgregora.html>
- Другие:
 - Теории мотиваций, теория ожиданий, Герцберга, ...
- Для контрольной прочитать все теории из ссылки выше

Управление: Пирамида Маслоу

Критика:

- зависит от возраста
- не подтверждена эмпирически



- Тейлор Ф., 1920, Макгрегор, 1960 г.
- Большинство людей:
 - Не любят трудиться
 - Нет честолюбия, не заинтересованы в ответственности
 - Работают только за деньги и из-за угроз
 - Хотят защищенности и постоянства
- Хотя они могут быть:
 - талантливymi сотрудниками
- Стандартное решение: постоянный контроль, формализованное автократическое управление
- Данное поведение возможно корректировать

15%-20%

- Люди Y: (везде кавычки)
 - Удовольствие от работы! (в зависимости от условий труда)
 - Могут и берут на себя разумный уровень ответственности. Собственная значимость
 - «Ощущают путь» - меняют организации и ищут себя
 - Любят лидировать
 - Хорошо решают поставленные задачи
- Необходимо не давить, ставить задачи, интересные задачи, и помогать их реализовывать
- Если не обеспечивать их потребности, люди Y станут людьми X

Управление: X & Y

- Руководитель:
 - Работает на стыке двух теорий
 - Априори (в стартапе) считает людей высокомотивированными
 - В зависимости от обстоятельств поступает, как необходимо для X или Y
 - «кнут и пряник» =)
- Опыт и терпение. Терпение => опыт.

- У. Оучи, 1981 г.
- Люди Z:
 - В мотивах людей сочетаются социальные и биологические потребности
 - Работать в группе и групповые решения
 - Неформальный, но понятный контроль
 - Индивидуальная ответственность
 - Человек — основа коллектива
 - Ротация кадров
- «Современное руководство» - крупные компании