ScrumTrek
culture of process

Применение Лин в масштабах предприятия

Асхат Уразбаев
Agile Coach
ScrumTrek

Асхат Уразбаев

- ScrumTrek
  - Agile Coach
  - Управляющий партнер
- В прошлом
  - Программист, менеджер проектов, методолог

ScrumTrek
culture of process

Agile Russia

ЛИН тривиален

- Если отбросить «философию», все «реальные» практики ЛИН есть в Scrum, XP, Kanban

ЛИН нетривиален

- Большие проекты и продукты,
- Распределенные команды,
- Сложные взаимодействия,
- Синхронизация программы проектов,
- Работа всей организации
- Сложные ситуации, там где Agile напрямую не работает

Супер-быстрая команда

- Производительность команды вырастает
- Дизайнеры не успевают предоставлять интерфейсы
- Работают по несогласованным экранам
- Много переделок

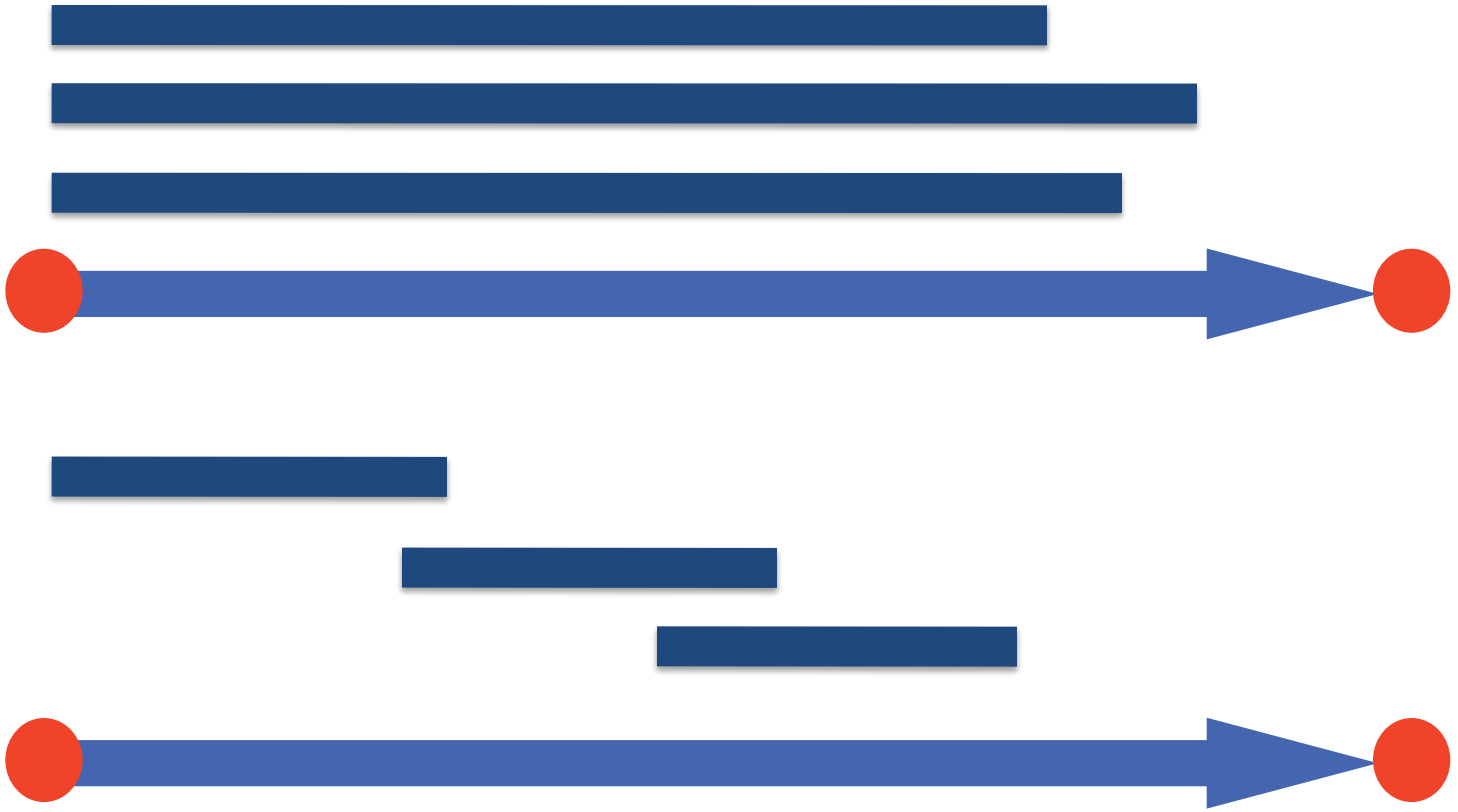
Вы опять сделали не то! Переделывайте

- Производительность команды вырастает
- Аналитики/заказчик не умеют качественно **продумывать** требования
- Обвиняют нижнего в иерархии ответственности
- Команда виновата

Классическое проектное управление и пул ресурсов

- Основная задача менеджера – выбить «ресурсы» на реализацию
- Ресурс «попилен» на проценты между несколькими проектами
- Проекты тянутся долго

Проектная разработка / командная разработка



Оптимизация всего процесса

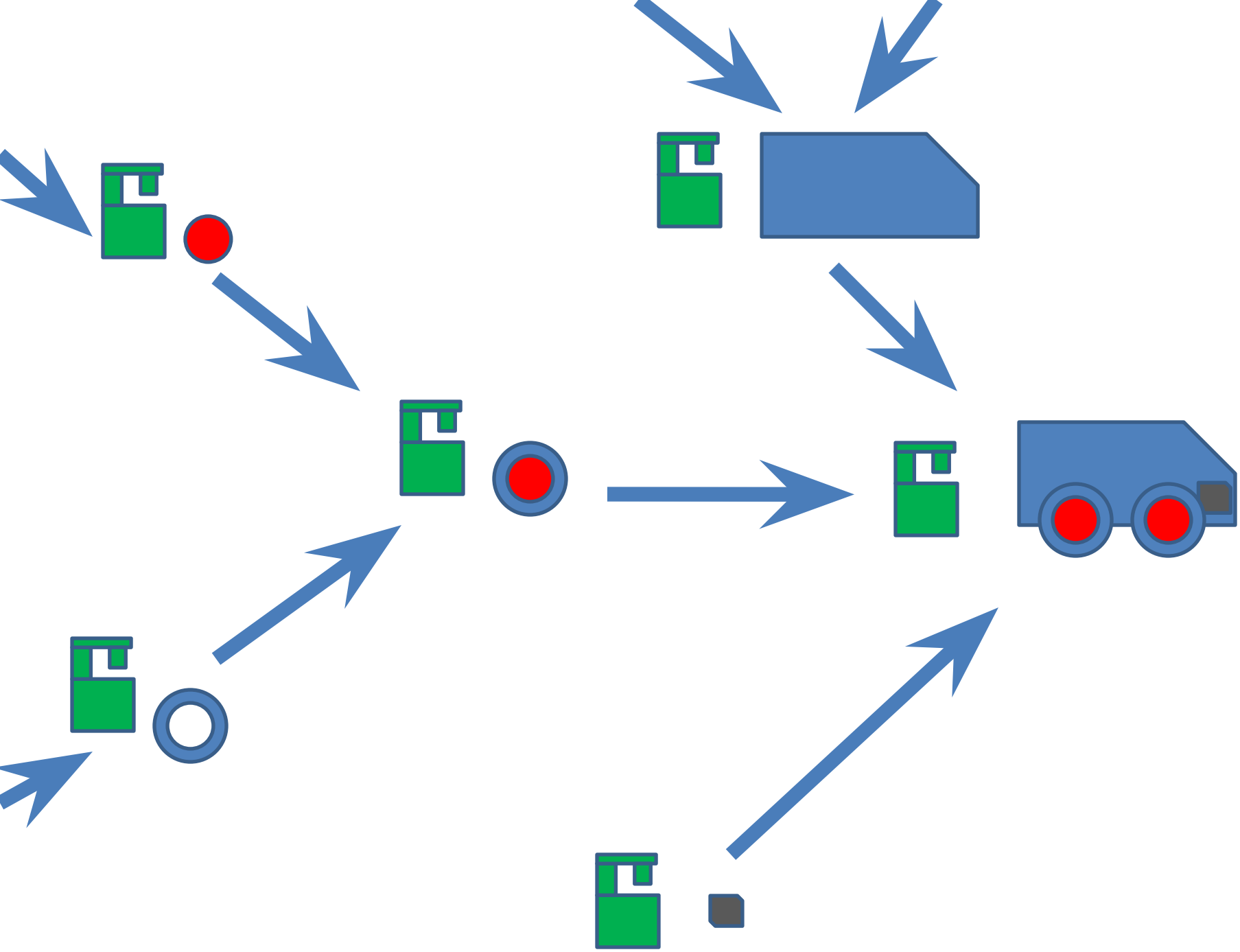
- Сисадмин не может задеплоить продукт
- Команда разработки ждет (6 человек)
- Бизнес-пользователи жду (50 человек)
- Конечные пользователи ждут (50000 человек)
- Компания теряет деньги

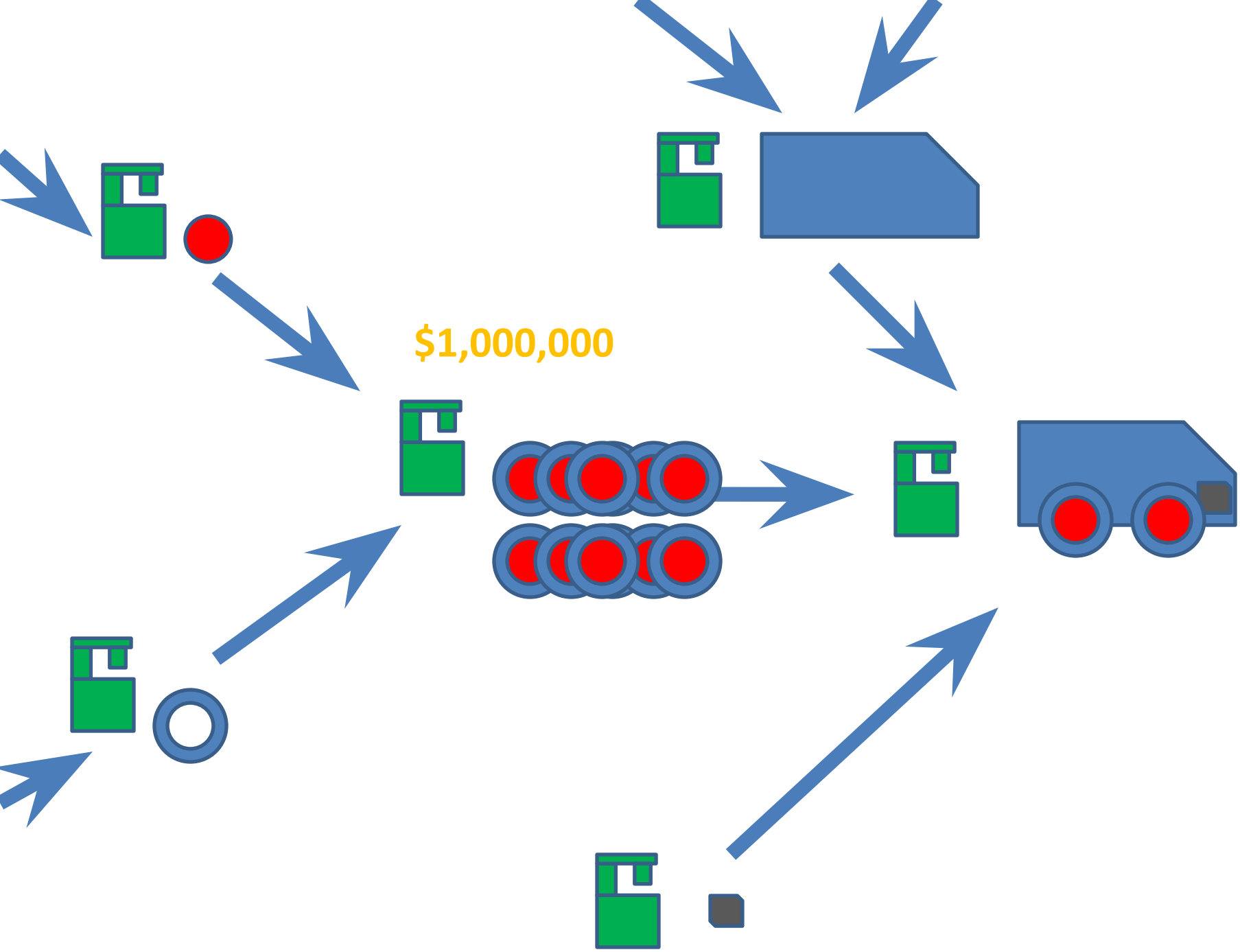
У сотрудника FIN заказ серверов – низкоприоритетная задача

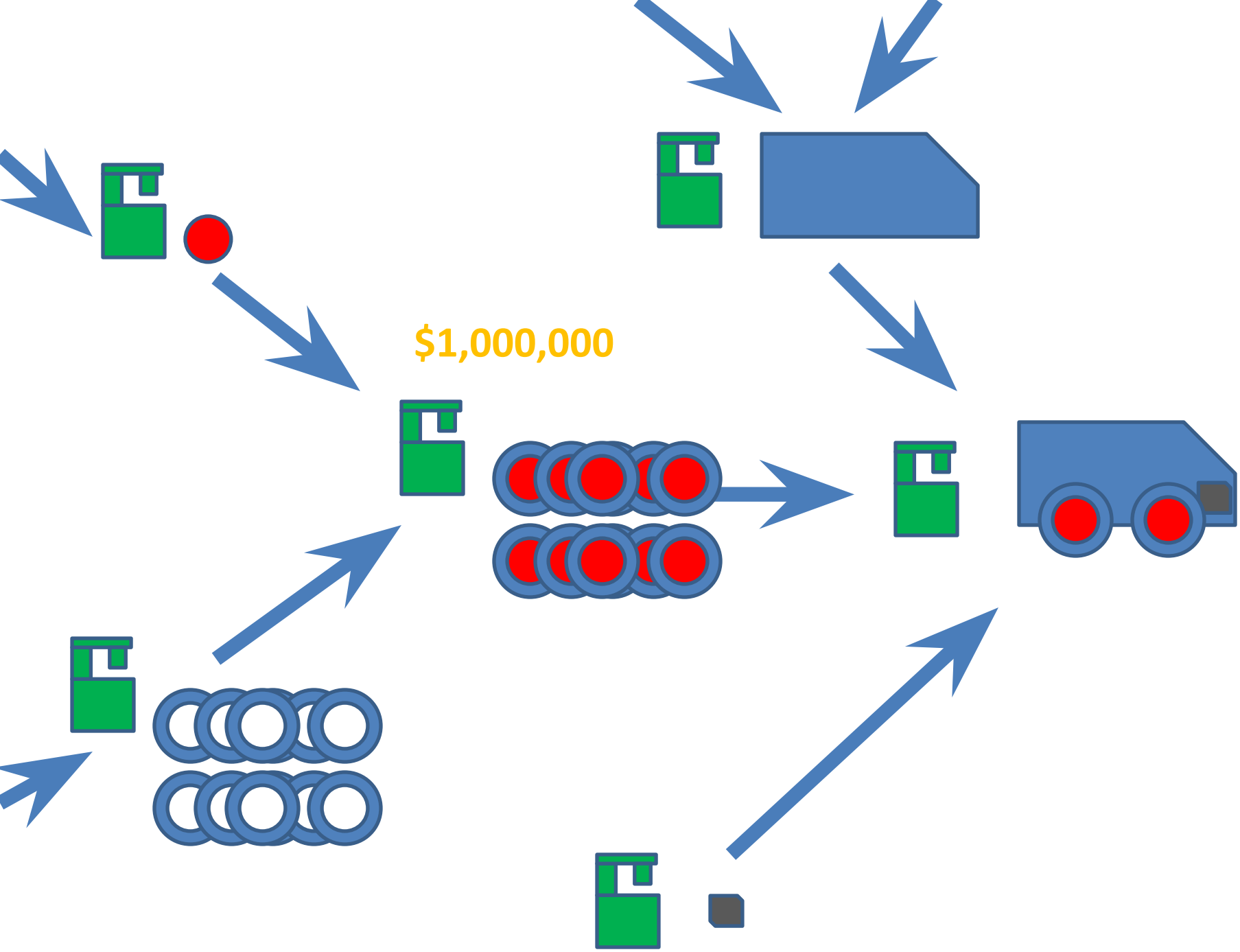
ОПЫТ ТОЙОТЫ

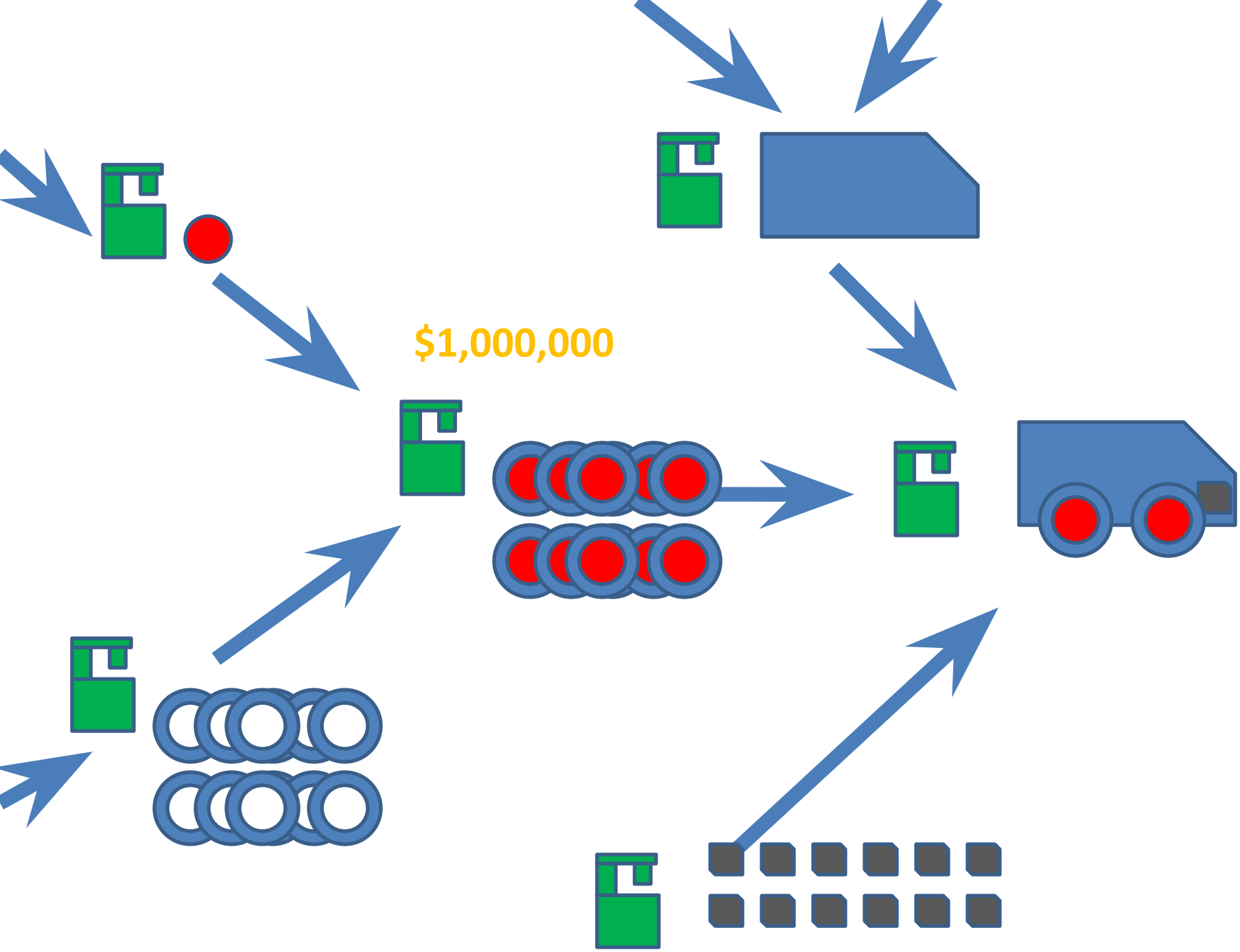
Производственная система Toyota

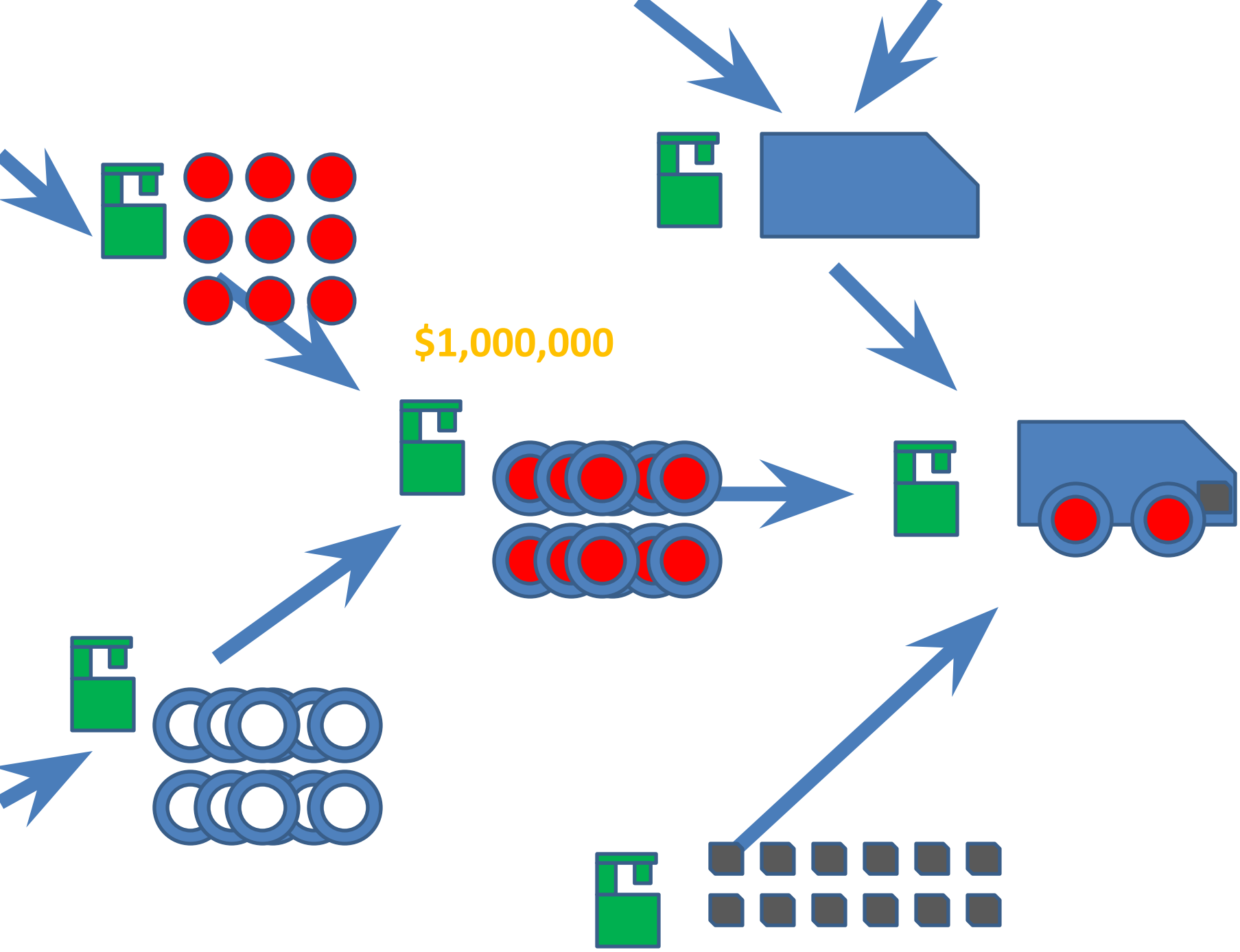
- Развивалась с 1948 по 1975 на заводах Тойота.
- С 2007 года Тойота – крупнейший производитель автомобилей в мире
- Адаптирована американскими учеными под названием *Lean Manufacturing (Бережливое производство)* (1988)
- Адаптирована к другим отраслям (медицина, логистика, почта, офис и так далее)
- Адаптирована к разработке ПО




$1,000,000


$1,000,000


$1,000,000

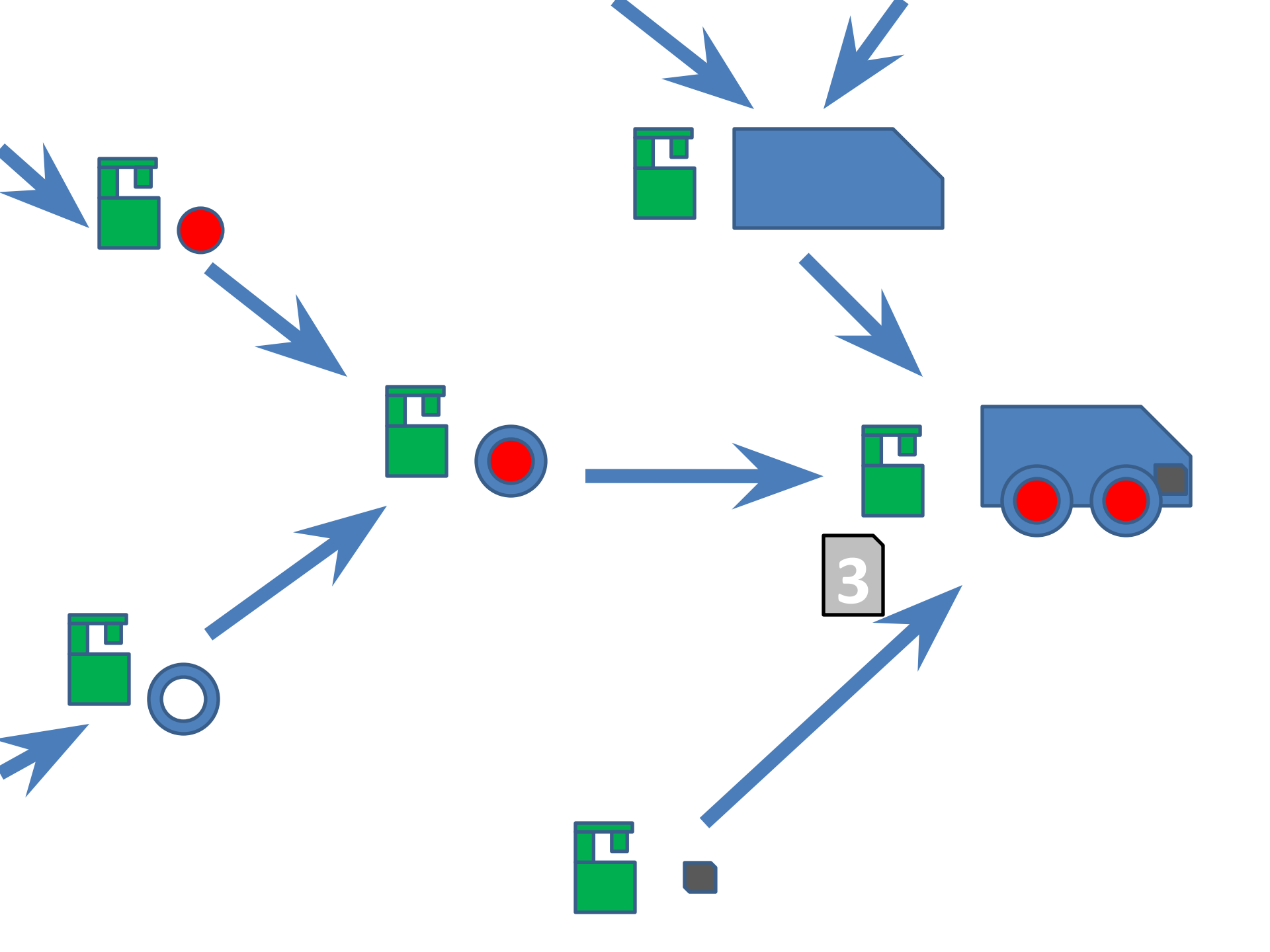

$1,000,000

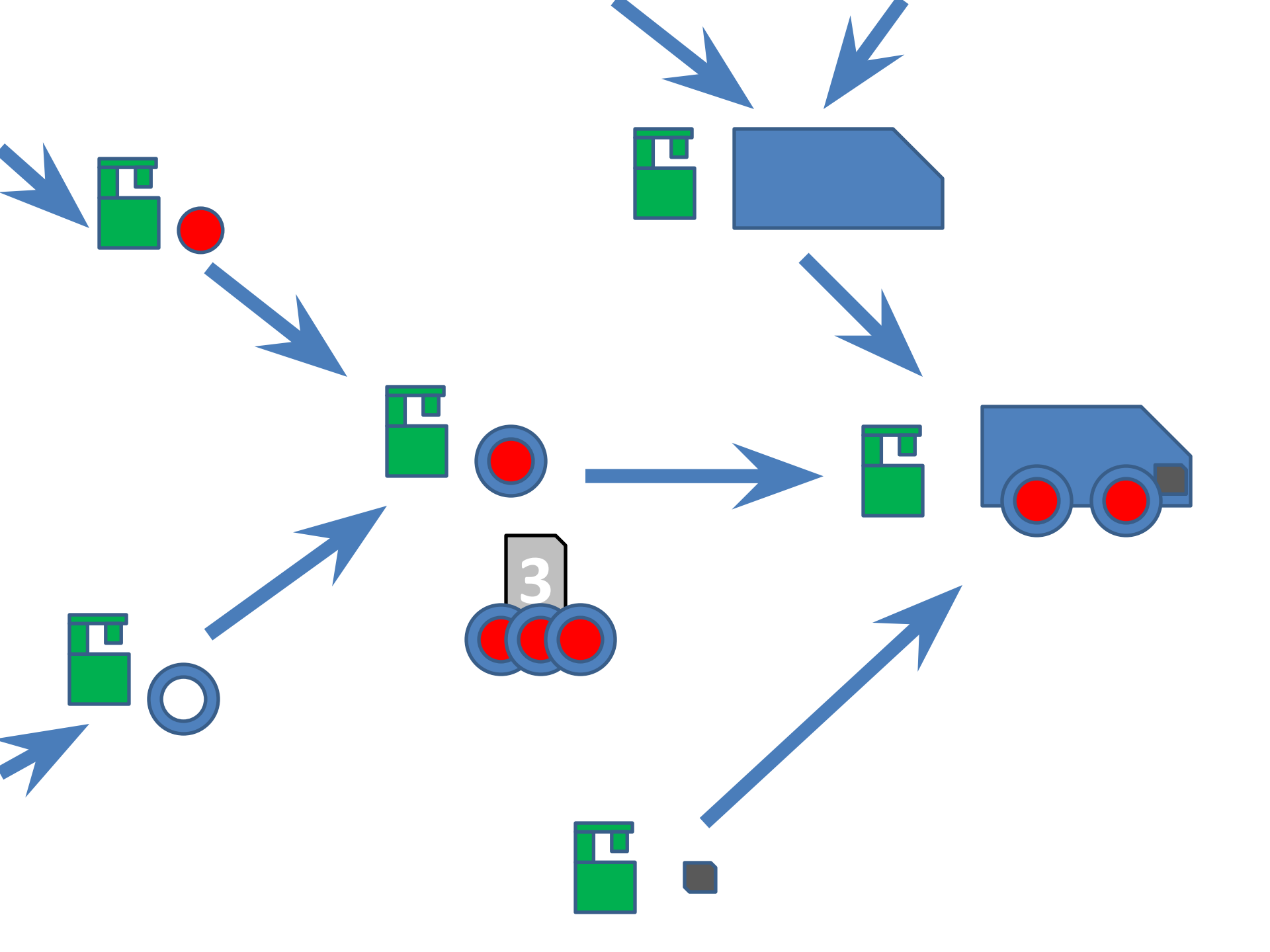
Характеристики массового производства

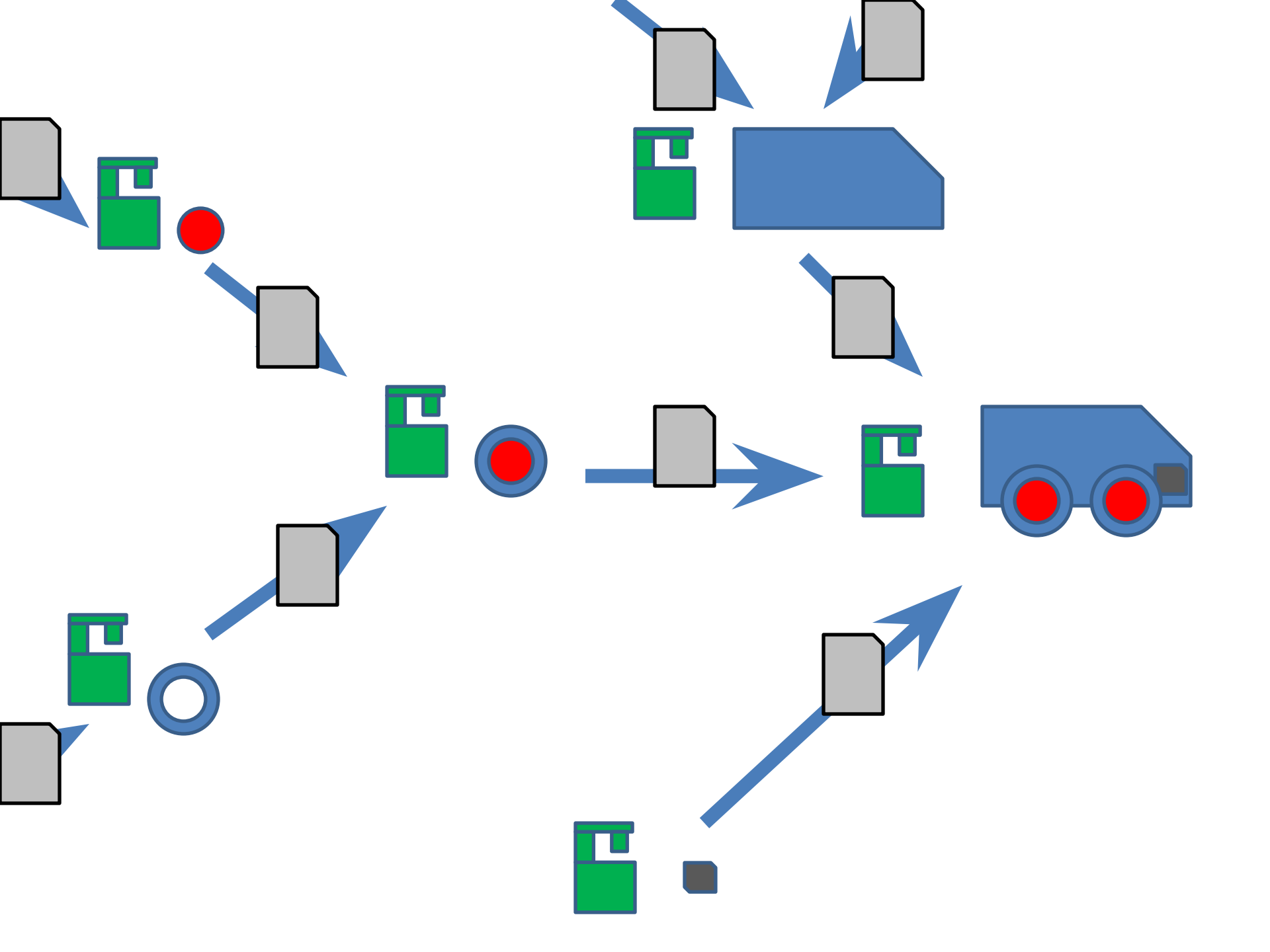
- Громоздкое и дорогое оборудование
- Склад готовых деталей
- Перемещение деталей
- Дефекты долго не обнаруживается

Taiichi Ohno

- Отец Toyota Production System


3


3



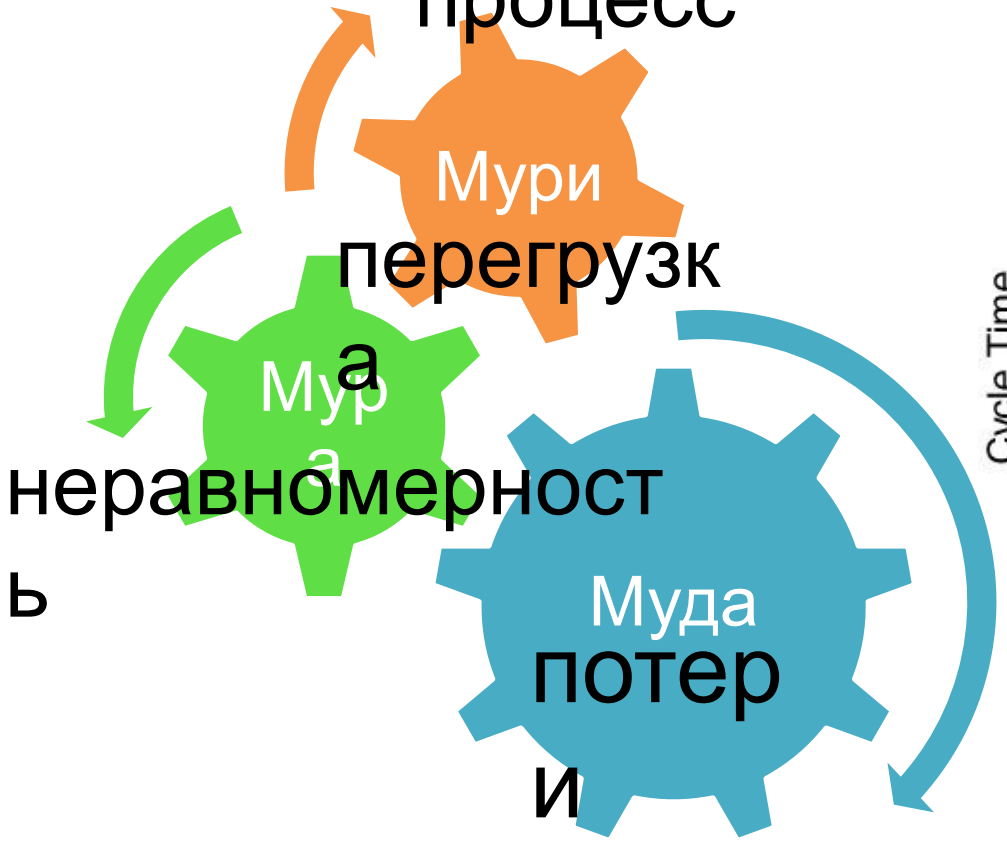
Основа TPS – «вытягивание»

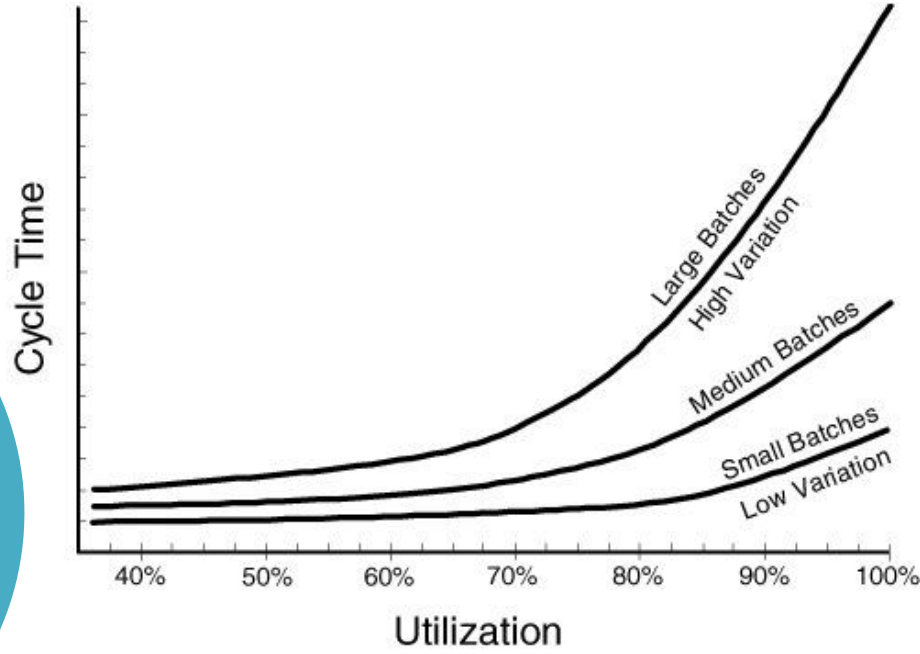
- Меньше времени от заказа до продажи
- Меньше запасов на складе

Канбан

Используй систему вытягивания, чтобы избежать перепроизводства

Ответственность за процесс





Выравнивай объем работ (хейдзунка)

Встроенное качество (дзидока)

АНДОН

Сделай остановку производства с целью решения проблем частью производственной культуры, если того требует качество


Комплектующие подаются к сборочной линии
Оператор проходит 25 футов, чтобы взять деталь
Оператор берет контейнер и вынимает детали
Оператор затягивает руку
Оператор берет и деталь
Оператор подбирает болты для кр детали
Оператор проходит 25 футов, возвращаясь к ш
Оператор устанавливает деталь на шасси
Оператор идет туда, где хранится инструмент
Оператор берет инструмент
Оператор возвращается и подносит инструмент к шасси
Оператор кладет инструмент
Оператор крепит поданную деталь болтами
Оператор затягивает болты с помощью инструмента
Оператор проходит 25 футов, чтобы взять следующую деталь
Все остальное представляет собой потери или не добавляет ценности
Добавляют ценность

Устранение потерь

1. Перепроизводство
2. Ожидание
3. Лишняя транспортировка
4. Излишняя обработка
5. Избыток запасов
6. Лишние движения
7. Дефекты

8. Нереализованный творческий потенциал сотрудников.

Процесс в виде непрерывного потока способствует выявлению проблем

используй визуальный контроль, чтобы ни одна проблема не осталась незамеченной

Пять этапов Лин

Определение ценности для потребителя

Выстраивание последовательного потока создания этой ценности

Обеспечение непрерывности этого потока

Обеспечение «вытягивания» от заказчика

Стремление к совершенству

Пять этапов Лин

Определение ценности для потребителя

Выстраивание последовательного потока создания этой ценности

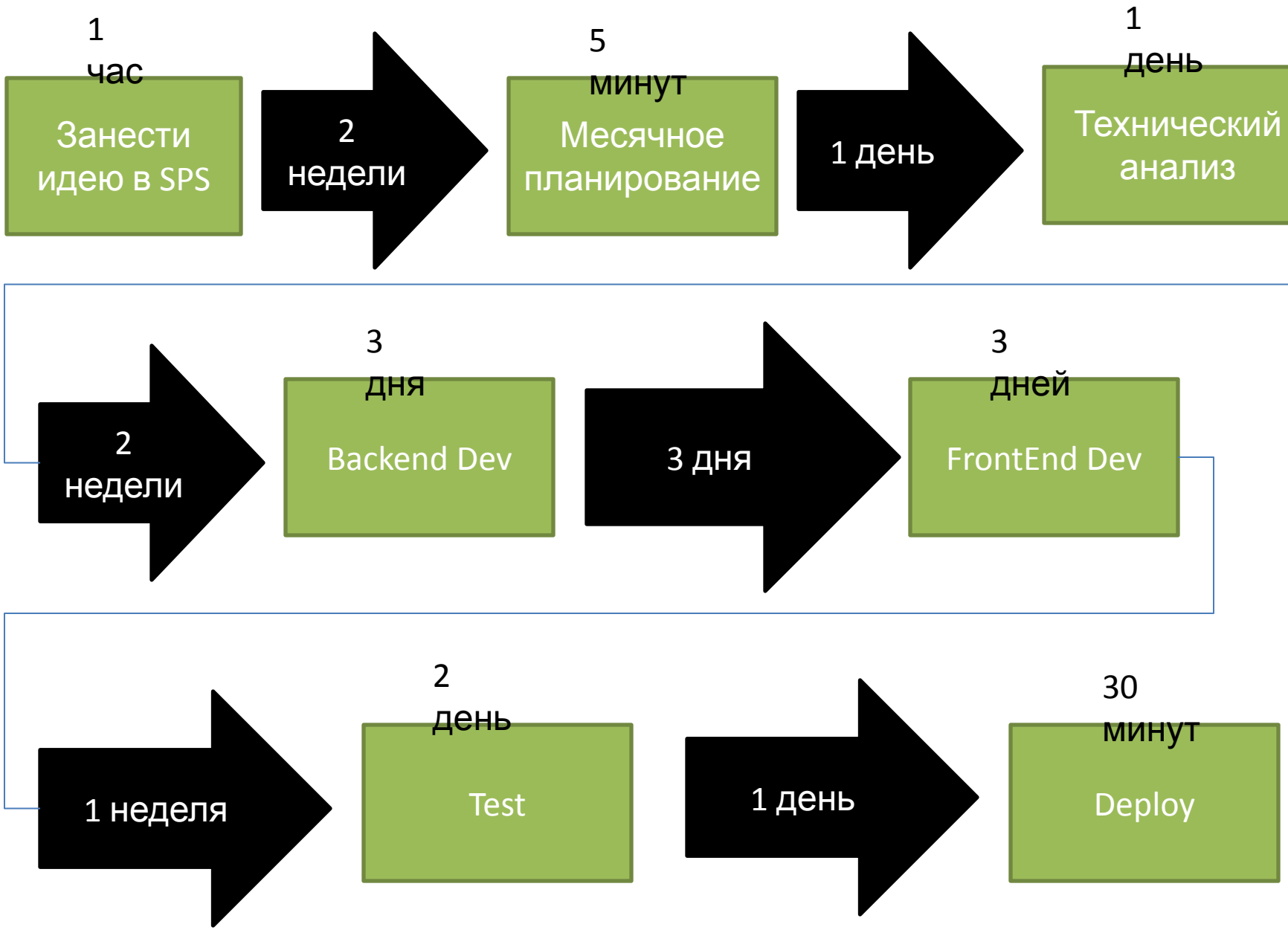
Обеспечение непрерывности этого потока

Обеспечение «вытягивания» от заказчика

Стремление к совершенству

Выстраивание последовательного потока создания ценности

- От грустного заказчика до веселого заказчика
- Эффективность цикла = полезная работа / полное время
- Создаем совместно со ВСЕМИ заинтересованными лицами



8 дней / 38 дней = 21%

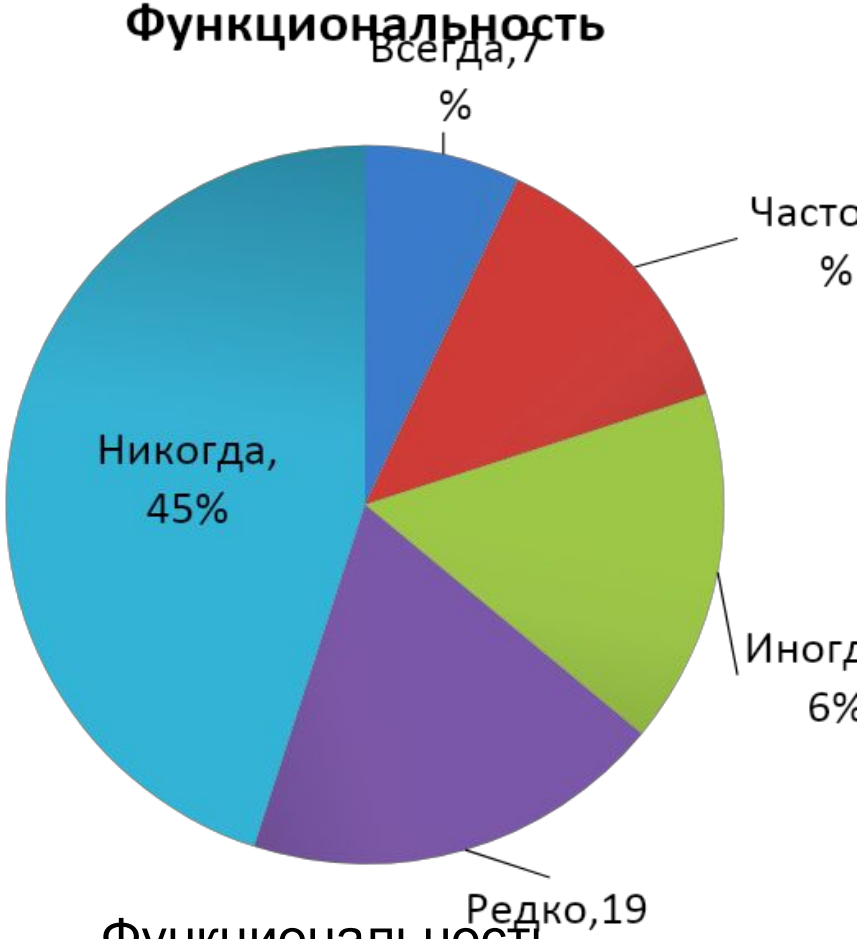
7 потерь

Производственная Система Toyota	Бережливая разработка ПО
Запасы	Недоделанная работа
Перепроизводство	Ненужная функциональность
Излишняя обработка	Повторное изучение (relearning)
Перевозка	Передача (handoff)
Движения	Переключение между задачами
Ожидание	Ожидание
Дефекты	Дефекты

Недоделанная работа

- Незапрограммированные требования
- Неинтегрированный код
- Нетестированный код
- Недокументированный код
- Незадеплоеный код

Ненужная функциональность



Функциональность, используемая в типичной системе

Standish Group Study Report

Повторное изучение

- Получение новой информации о продукте, коде, заказчике несет ценность для заказчика
- Повторное изучение – потери

Передача

- Разделение
  - Ответственности
  - Знаний
  - Действий
  - Обратной связи
- Самая распространенная проблема – разделение принятие решений и ответственности

Переключение между задачами

Ожидание

- Ожидание согласования с заказчиком
- Внутренние согласования
- Бесполезные митинги

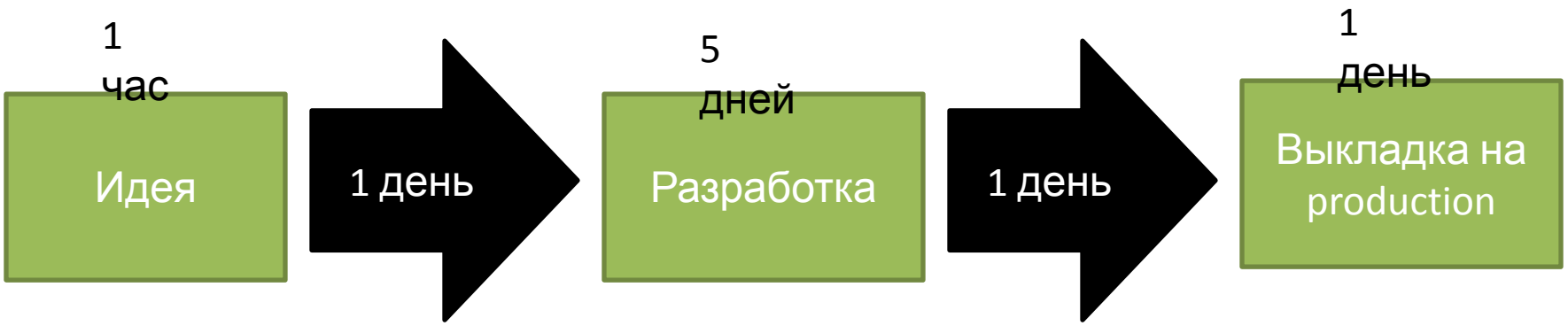
Дефекты

- ПОТЕРИ = ВЛИЯНИЕ ДЕФЕКТА * ВРЕМЯ ПОКА ДЕФЕКТ НЕ ОБНАРУЖЕН
- Чем позже дефект найден, тем он дороже

5 копеек про поток

- Очень полезен!
- Иногда выглядит тривиально
- Выводы иногда понятны и без всякого построения потока
  - (если вы только не закоренелый “вотерфольщик”)

Code &Fix



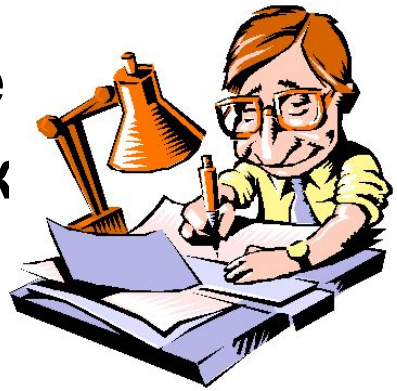
5 дней / 7 дней = 71%

1. Идея
2. ???
3. PROFIT!

Переключение контекста, лишняя работа, повторное изучение, дефекты

Реальный пример

- **Начальник отдела документирования**: «Я раньше техписом был. Теперь сделали меня начальником отдела документирования. Я должен заставлять разработчиков писать техническую документацию к продукту. Иначе потом будет очень много проблем у команды поддержки. Проблемы? Разработчики не хотят писать документацию! Еще мне трудно сформулировать требования к схеме развертывания, я в этом далеко не спе
- **В чем причина проблем и как их мож исправить?**



Кого на этом потоке НЕТ?

- Это важнее измерения эффективности потока
- Проверьте маркетологов
- HR-ов
- Архитекторов
- Сервисные команды/компонентные команды
- И просто Важные Принимающие Решения Шишки

- Чем они все занимаются?

В каких участники отношениях?

- Цепочка должна быть непрерывной
- Отношения внутри потока: заказчик-исполнитель

Кто у кого заказывает работу?

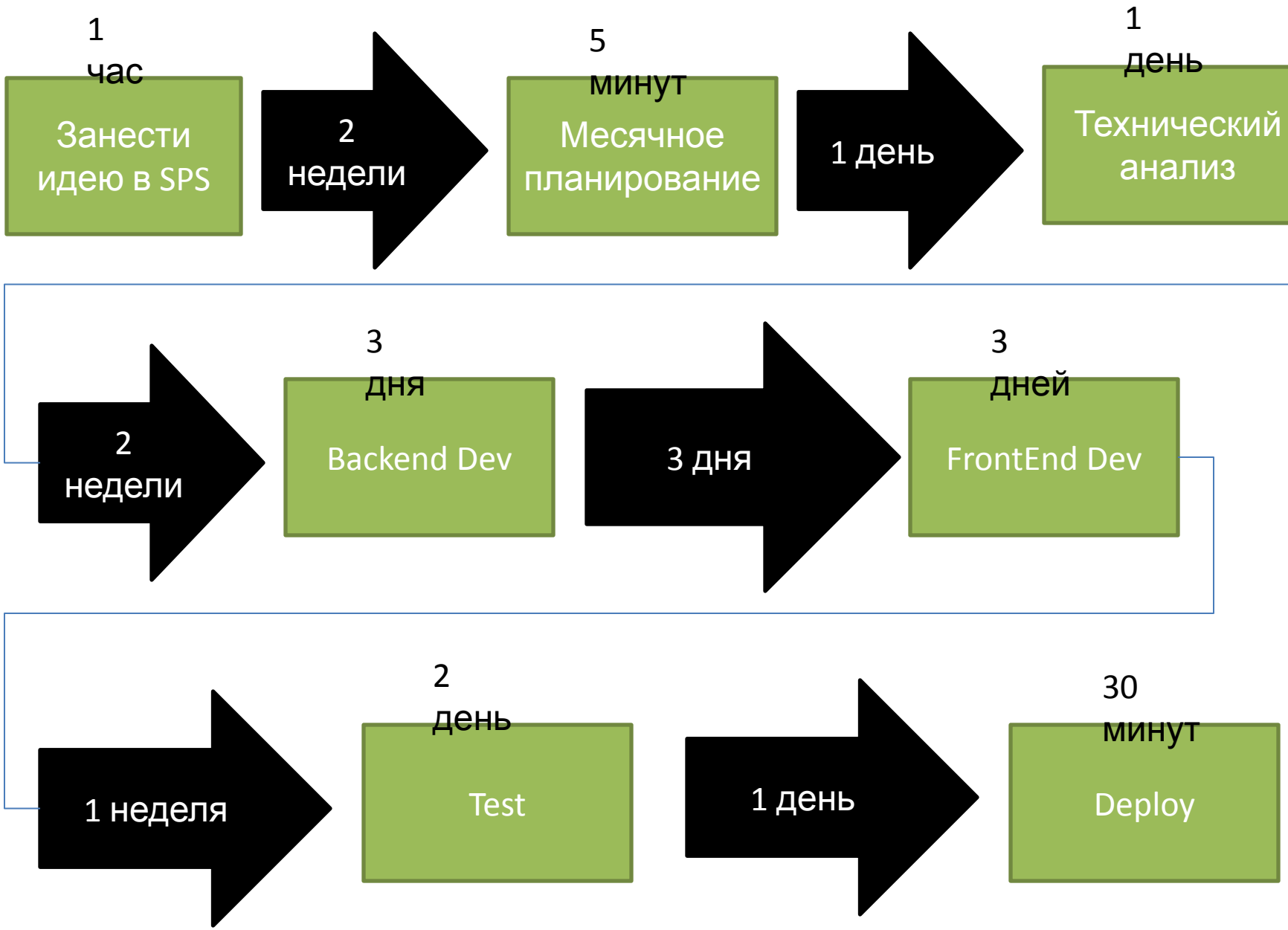
- Конечный пользователь
- Менеджер продукта
- Маркетинг
- Разработчики
- Архитектор
- Поддержка (support)
- HR-менеджер

Что тут делает ТАКАЯ ПРОРВА ЛЮДЕЙ?

- Слишком длинная цепочка – потенциальный источник потерь

Передача

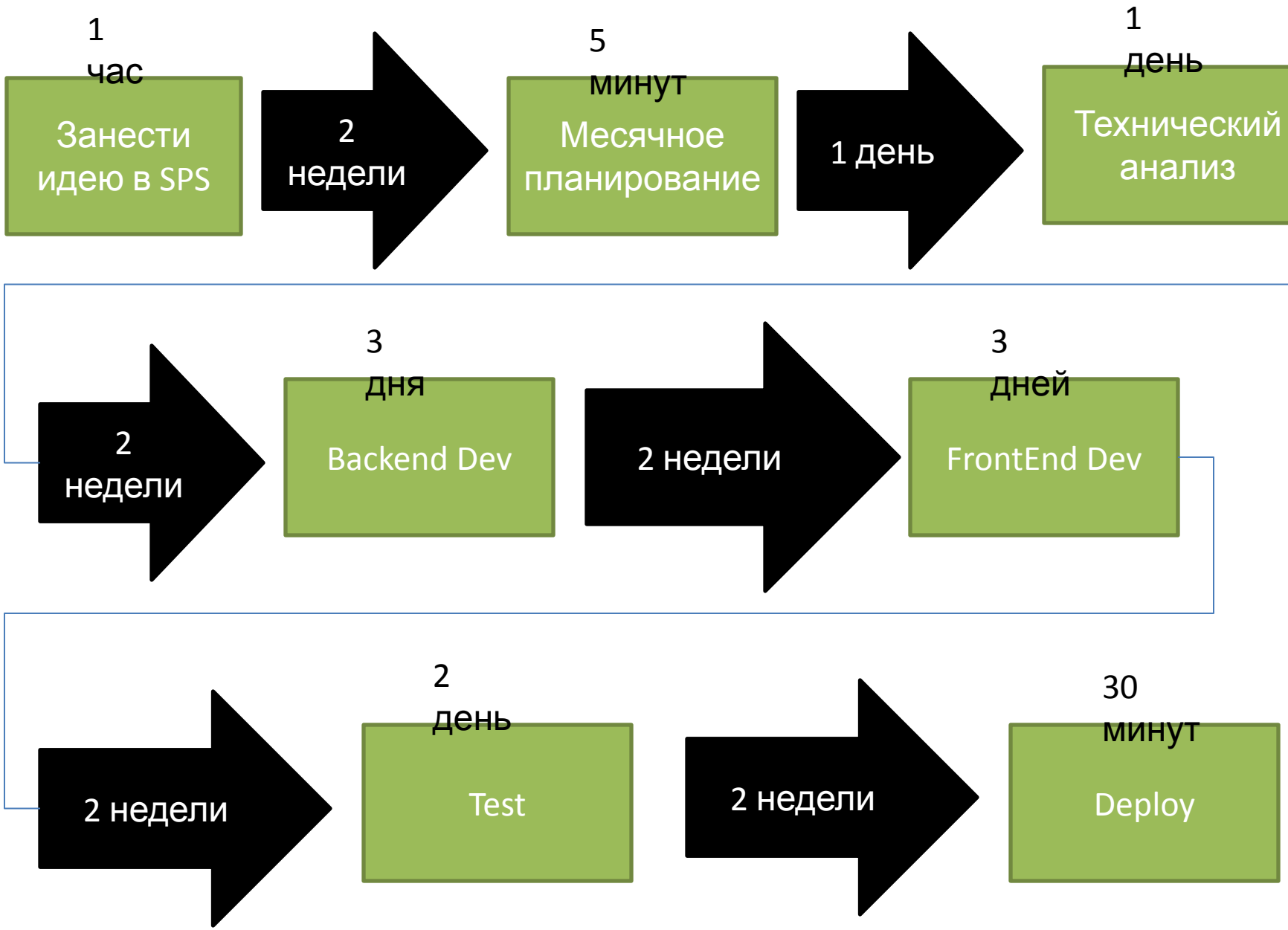
- Разделение
  - Ответственности
  - Знаний
  - Действий
  - Обратной связи
- Самая распространенная проблема – разделение принятие решений и ответственности



8 дней / 38 дней = 21%

Внедряем Agile

- Внедрение «снизу» в большой компании
- Итеративность, самоорганизация, ретроспективы, технические практики и т.д.



8 дней / 70 дней = 11%

Feature Team

- Команда, включающая всех специалистов для решения проблемы заказчи

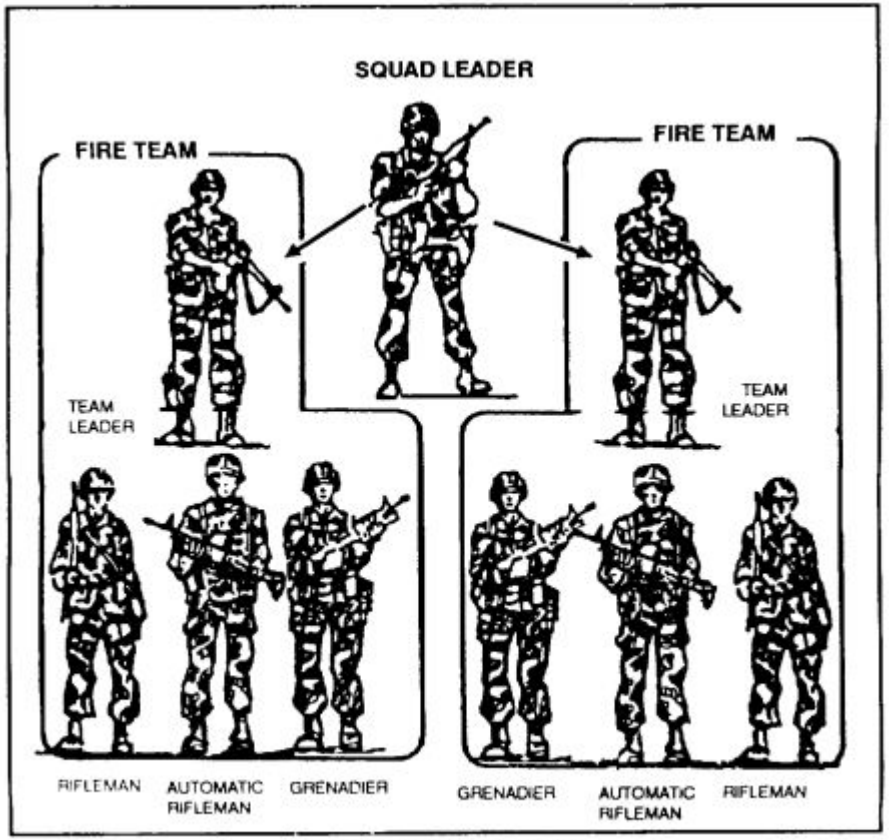
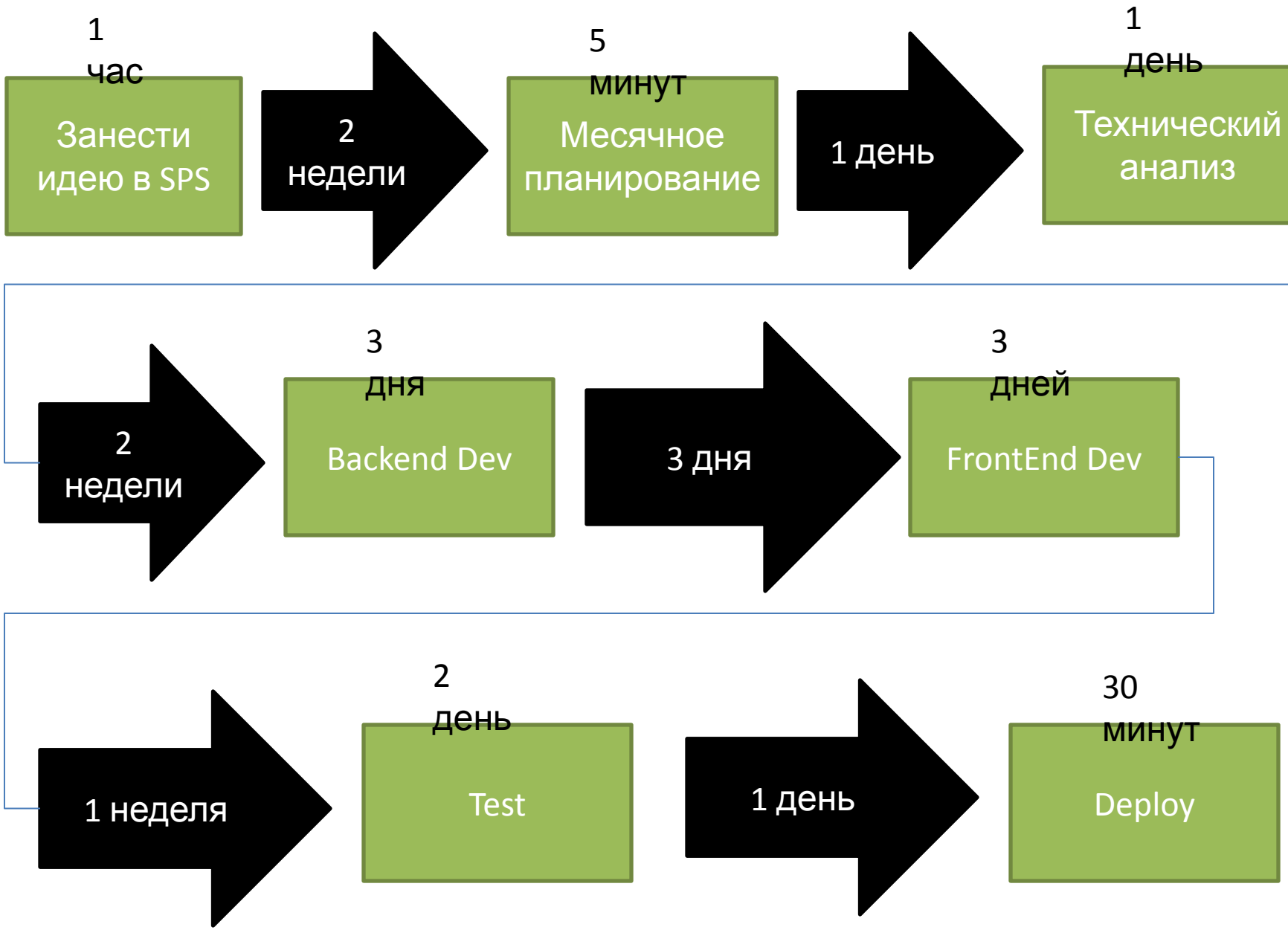
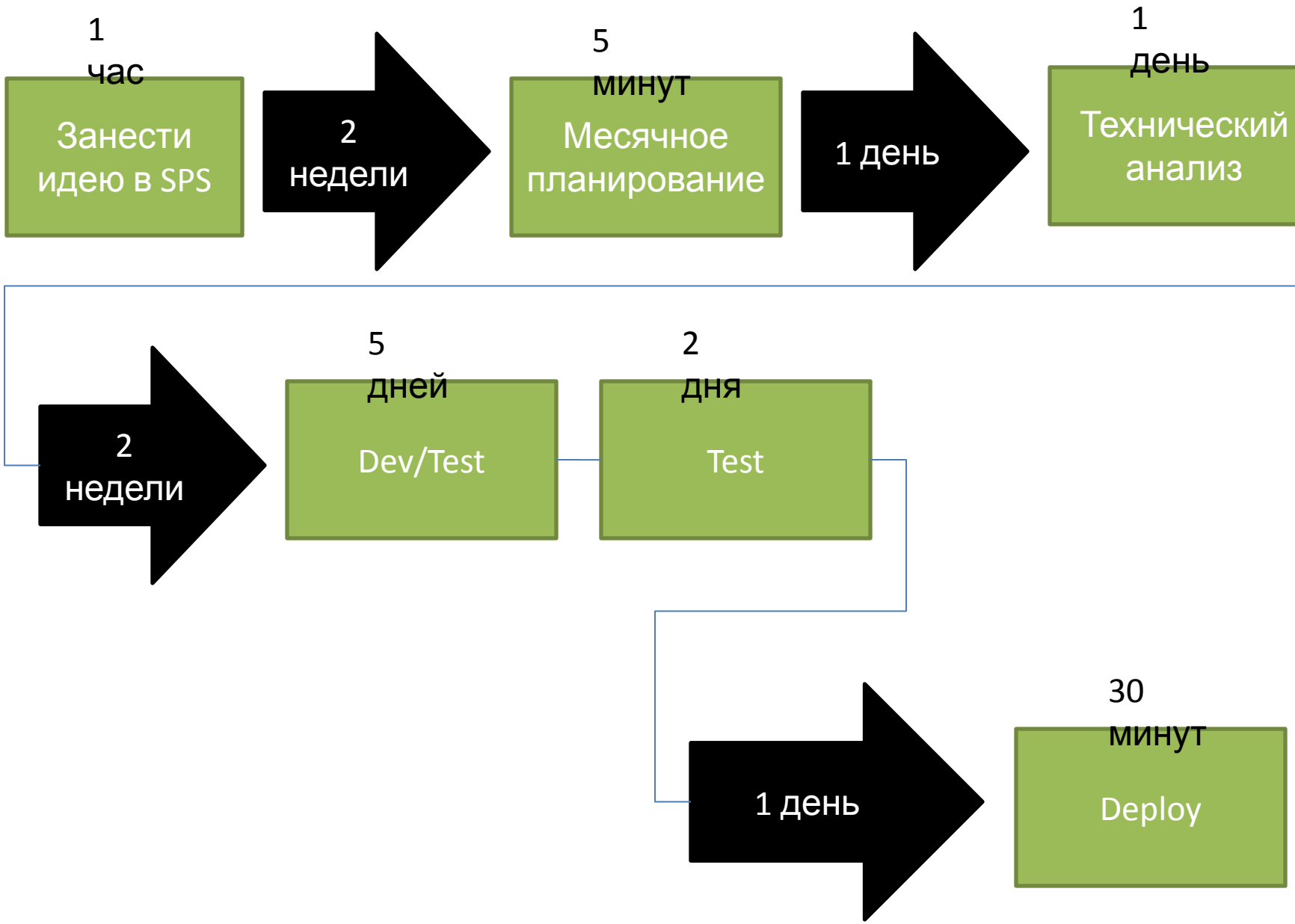


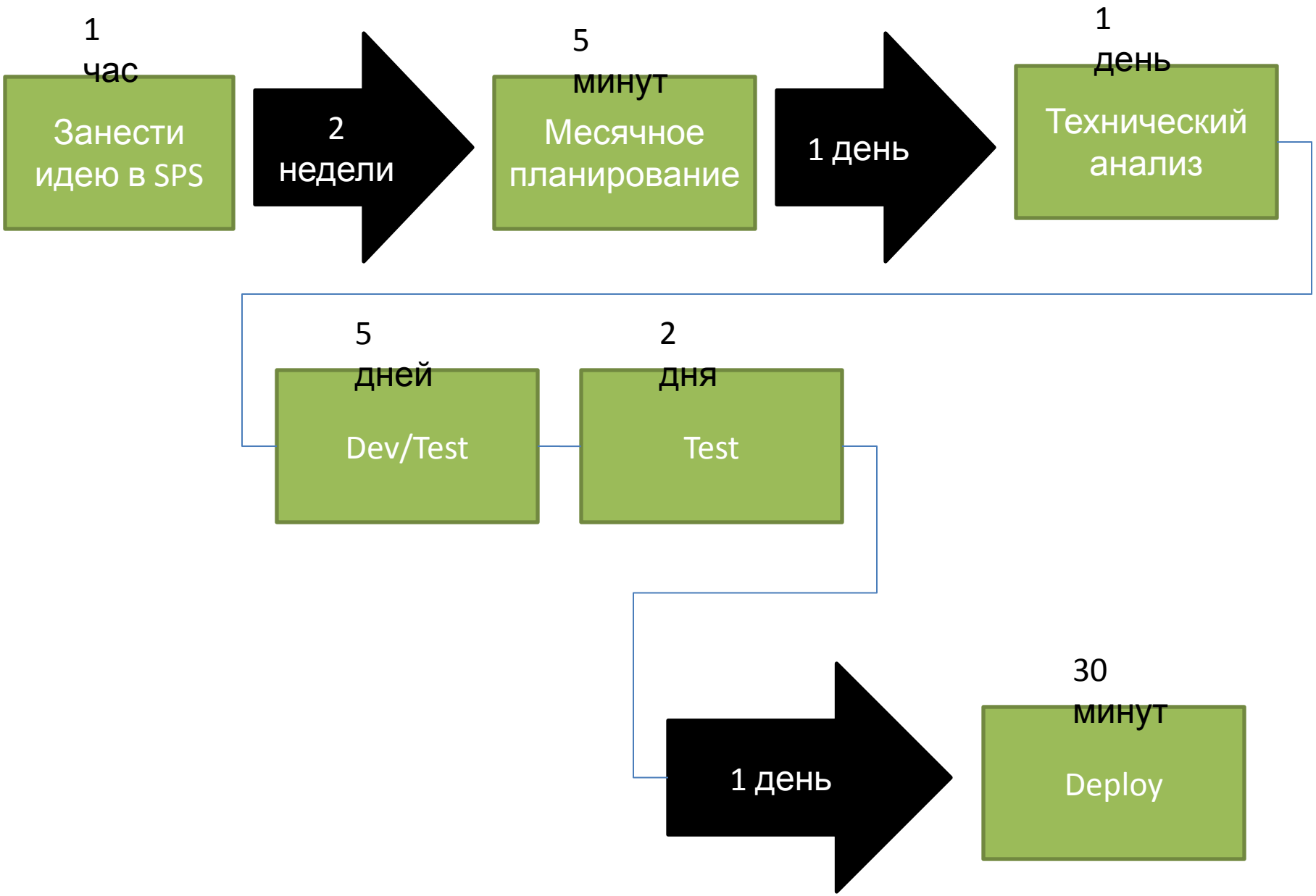
Figure A-6. Rifle squad.



8 дней / 38 дней = 21%



8 дней / 30 дней = 26%



8 дней / 20 дней = 40%

Пять этапов Лин

Определение ценности для потребителя

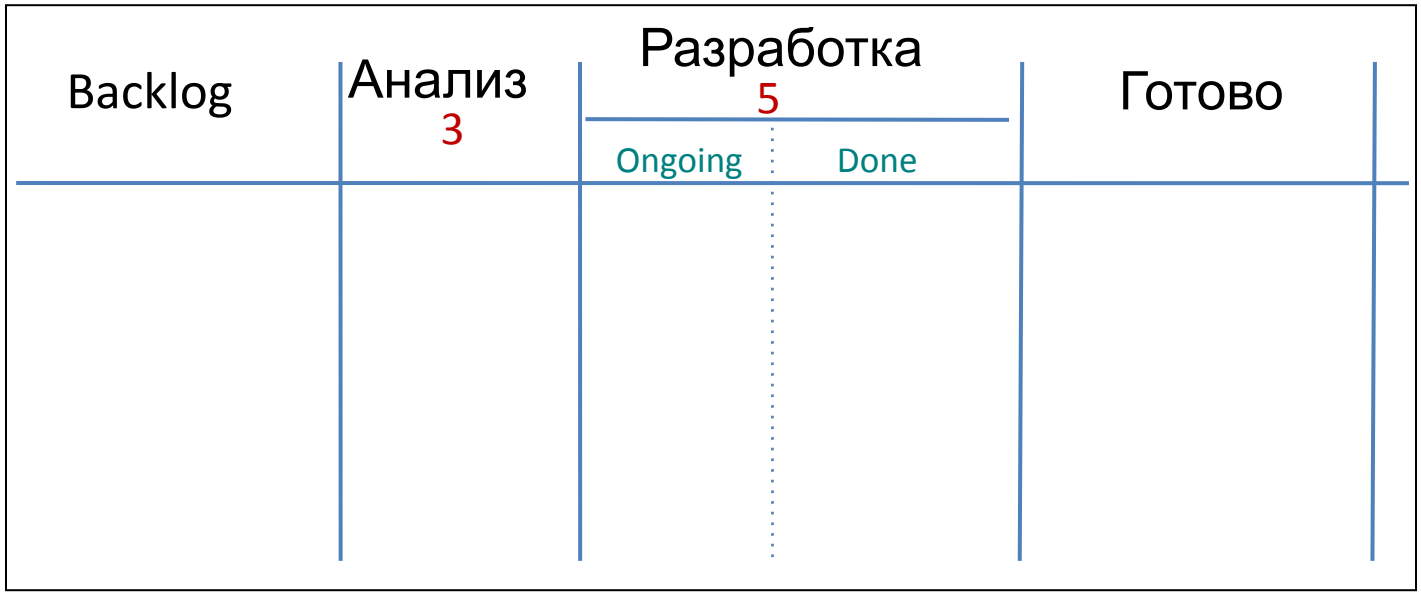
Выстраивание последовательного потока создания этой ценности

Обеспечение непрерывности этого потока

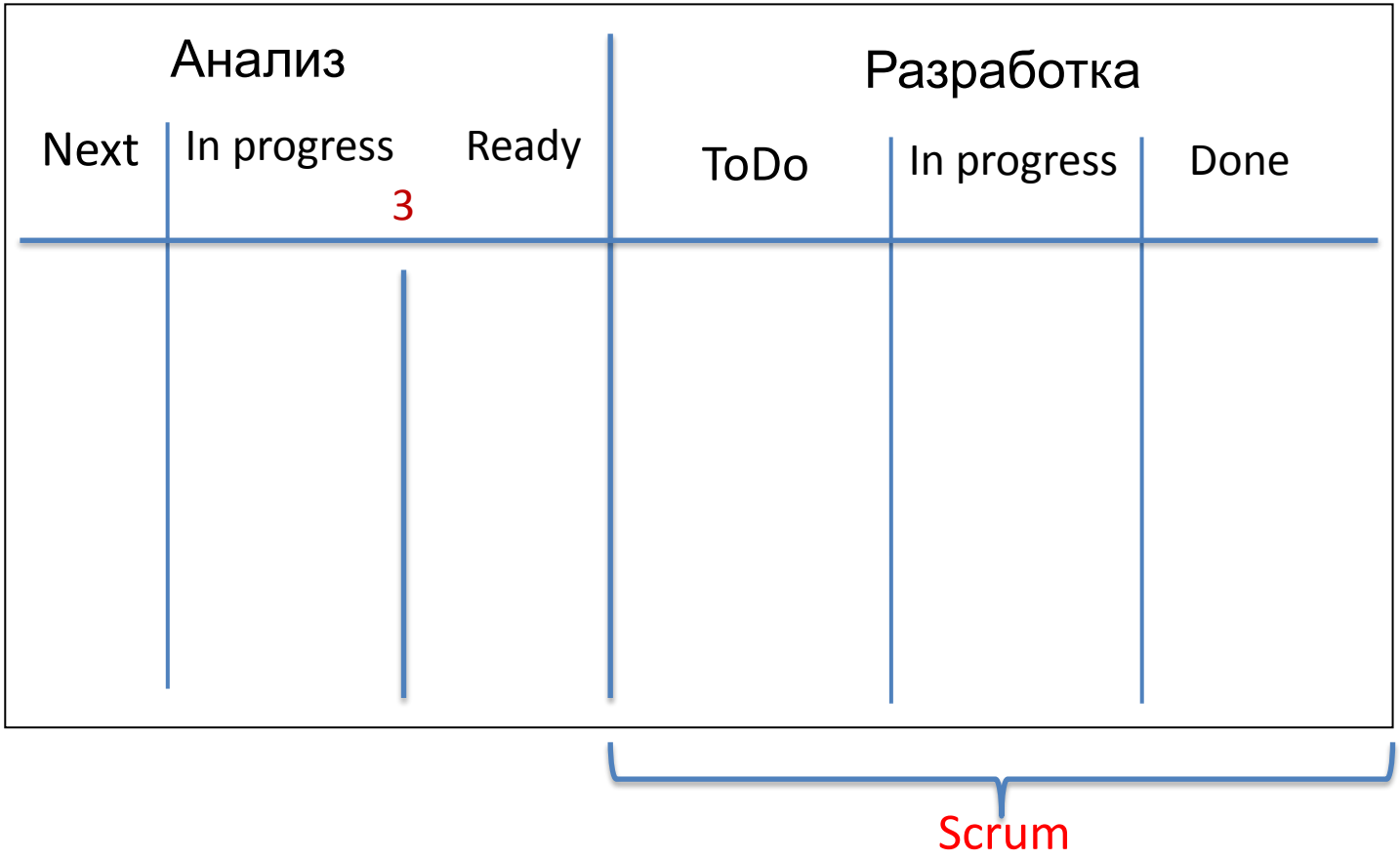
Обеспечение «вытягивания» от заказчика

Стремление к совершенству

Канбан



Канбан + Скрам

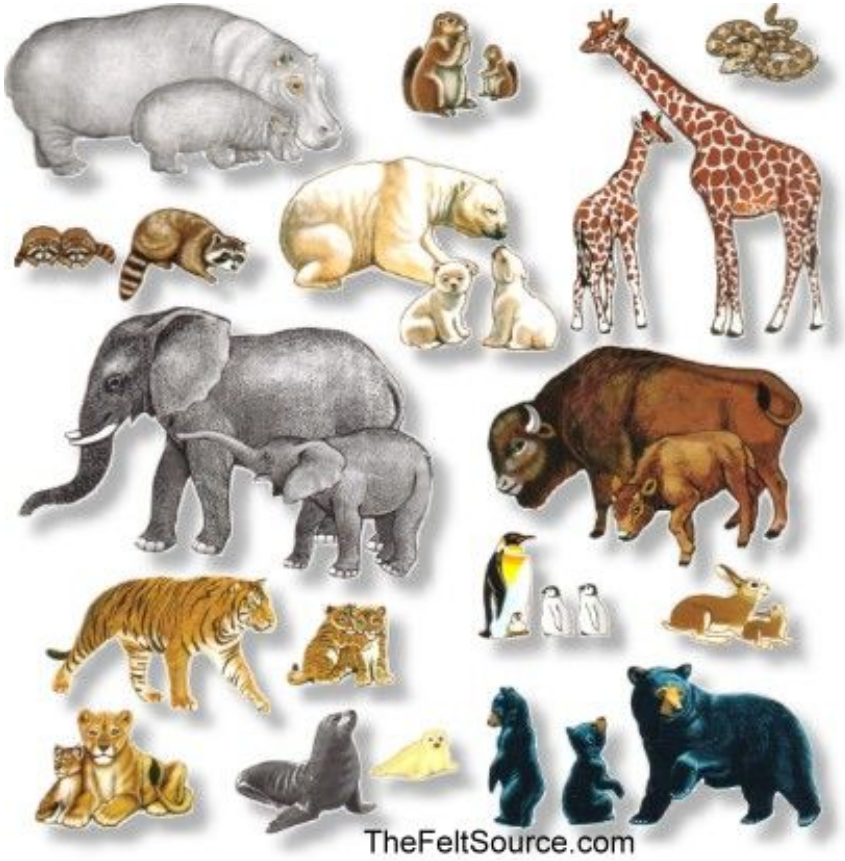


Верстка и бэкенд

- Выделение верстки и бекенда в последовательные стадии замедляет разработку

«Сворачиваем» цепочку где можем!

- Берем в команду
  - Аналитика
  - Разработчиков
  - Тестировщиков
  - Сисадминов



Принцип ЛИН – минимизация Cycle Time

- Минимизация времени цикла **фичи** ускоряет проект по закону Литтла
- Увеличивает накладные расходы
- Зачем **ЕЩЕ** нужно минимизировать время цикла?

Низкий Cycle Time

- Снижаем Cycle Time
- Меньше Work In Progress
- Малейшая проблема – застреваем
- Меняется отношение к проблемам

Dancing Elephants

Presentation

The Dancing Agile Elephant: IBM Software Group's Transition to Agile and Lean Development

Recorded at: QCon

Presented by Sue McKinney on Aug 12, 2009 Length 00:52:33

Community Agile Topics Agile in the Enterprise , Leadership Tags QCon San Francisco 2008 , Lean , QCon , IBM

NOTICE The next QCon is in London March 7-11, Join us!

Share

Select your view: **vertical** | horizontal

http://www.infoq.com/presentations/dan

Summary

This session explores the approach and challenges to transforming multi-thousand person division to adopt new approaches to developing software. Questions about how to inspire and motivate change, identifying the change agents, the tooling to enable the masses will be discussed.

Bio

Currently responsible for development transformational activities with IBM's software development group, her major emphasis is adoption of Agile and Lean principles into the mainstream. In addition to driving transformational activities within IBM, Sue works with large clients to share IBM's experience and help them scope opportunities for their own transformational activities.

Способ выйти из зоны комфорта

- Высокая прозрачность
- Любая неоптимальность – все начнет буксовать
- Сотрудники исправят процесс

- Pain = no motivation?

Pain = no motivation?

- Так работать МЕНЕЕ комфортно и БОЛЕЕ интересно
- Pain = motivation!

Пример - баг в production

- Баг в production
- Работа останавливается
- Пока баг не исправлен продолжать работу в итерации нельзя

Самая спорная потеря

Согласование требований это потеря

Работа по несогласованным требованиям это потеря

Достигай консенсуса

Принимай решение не торопясь, на основе консенсуса, взвесив возможные варианты, внедряя его – не медли (немаваси)

- Работает в области технических решений
  - В области избытка информации
- В бизнесе работает НЕ ВСЕГДА

Потери?

- Backlog это потери
- Оценка это потери

Пять этапов Лин

Определение ценности для потребителя

Выстраивание последовательного потока создания этой ценности

Обеспечение непрерывности этого потока

Обеспечение «вытягивания» от заказчика

Стремление к совершенству

Пример

- Вы не знаете, где расположить блок с рекламой

- Что делать?

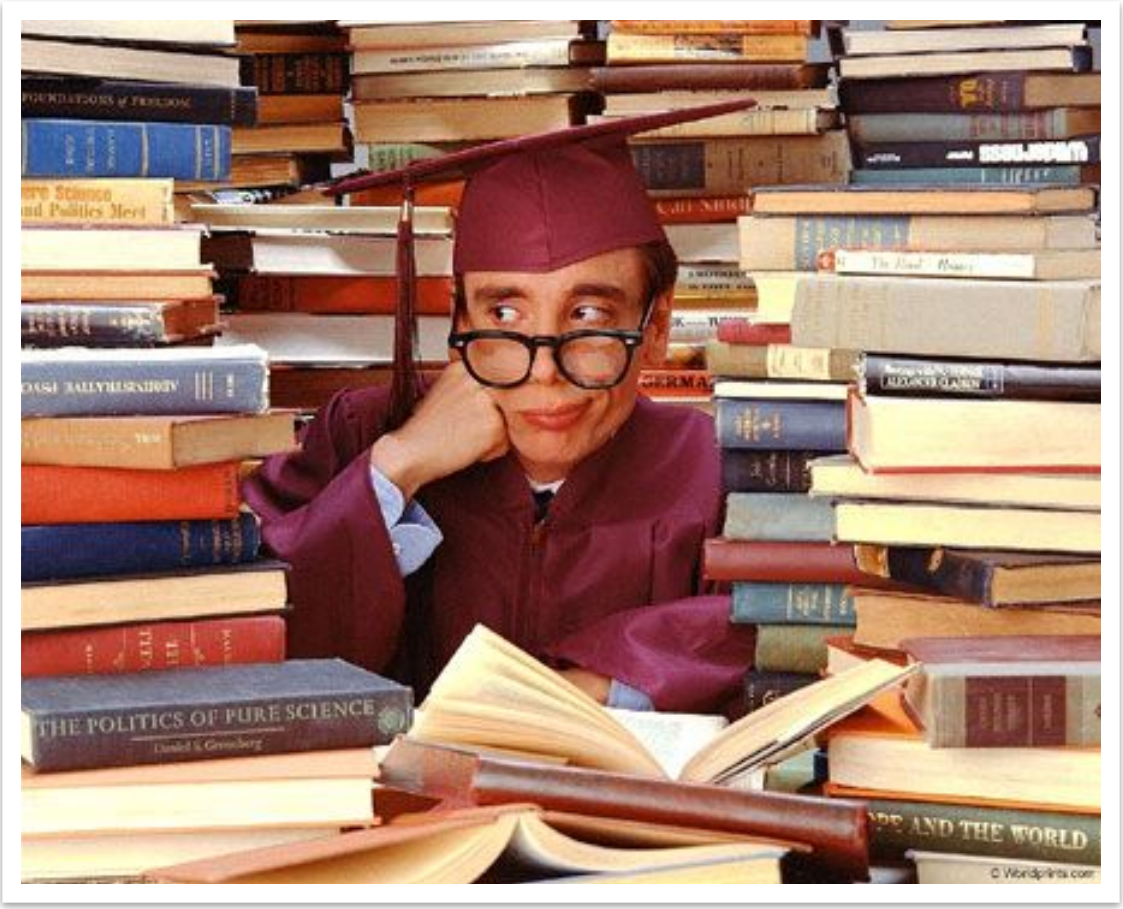
Пример

- Положить куда-нибудь
  - Есть проблемы и поважнее
- Эксперимент
  - Выложить в разные места и посмотреть, что будет
- Анализ
  - Изучать пользователя и рынок
  - учиться проектированию интерфейсов

От чего зависит ответ?

- Имеете ли вы информацию для принятия решений?
  - Нет. Контролируемый эксперимент для получения знаний
  - Да. Анализ, расчет и валидация результатов

Постоянный поиск новых знаний



Команда обретает самосознание

- Демо
- Планирование

Что говорит команда

Не Agile

Дайте нам четкое ТЗ!

Agile

Куда мы движемся?

Зачем мы делаем эту фичу?

Как можно улучшить фичу?

Принятие решений командой

- Сдвигать уровень принятия решений как можно ниже

Потери

- Переделывать Wording
- Переделывать функциональность
- Исправлять проблемы с Usability
- Исправлять внешний дизайн

…

Делать сразу правильно

- Делать сразу правильно там, где информация доступна или ее можно получить
- Разрабатывать пробный вариант там, где информации недостаточно или она в принципе недоступна
- Везде, где можно, добывать новую информацию

Уничтожение потерь

- Это возможно, если
  - Научиться разбираться в бизнес-домене
  - Научиться разбираться в смежных с разработкой областях (например, UX)
  - Учить заказчика взаимодействовать с командой
  - Свободно обмениваться информацией внутри команды и с заказчиком

К чему это приводит с практической точки зрения

- Внятный Vision до начала разработки
- Ready/ready – требования готовы к началу итерации
- Вовлечение команды в подготовку требований

Определение ценности – важнейший элемент Лин

- Постоянно продолжающийся и неустанный процесс
- Создание баклога, приоритезация и т.д. – часть процесса понимания ценности заказчика

Этапы развития организации

Delivery, прозрачность, предсказуемость

Ценность, бизнес, управление продуктом

Постоянное совершенствование

Пять этапов Лин

Определение ценности для потребителя

Выстраивание последовательного потока создания этой ценности

Обеспечение непрерывности этого потока

Обеспечение «вытягивания» от заказчика

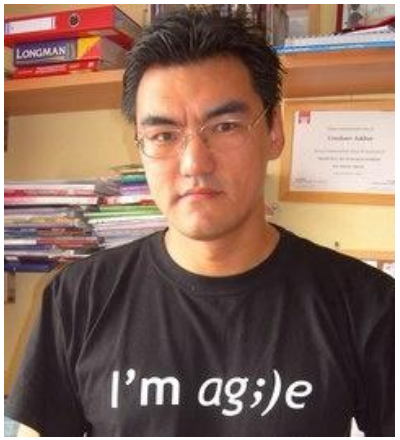
Стремление к совершенству

Принципы лидерства

1. Ничто не заменит непосредственного наблюдения.
2. Изменения вводятся в режиме эксперимента.
3. Как можно больше экспериментов.
4. Менеджер не решает проблем, а учит этому других.

Воспитывай лидеров, которые досконально знают свое дело, исповедуют философию компании и могут научить этому других

Асхат Уразбаев



- askhat@scrumtrek.ru
- Twitter: zibsun
- Skype: askhatu
- ЖЖ: zibsun.livejournal.com

ScrumTrek
culture of process

Agile Russia

ВОПРОСЫ?