

# Проектирование Распределенных Систем

**Первый семинар: Введение в курс**

Знакомство с основными проблемами распределенных  
систем и способами их решения

Алексей Шкель



# About Me

## Разработчик в ТБанке в отделе Лояльности

- Мы занимаемся:
  - Персонализацией предложений в сервисах Города
  - Запусками акций для раздачи кэшбэка в Мобильном Банке
- Разрабатывал:
  - Платформу персонализации для сервисов Города
  - Сервисы таргетирования, аналитики и документооборота в Выгоде
    - Плоды нашей работы можно увидеть в МБ в «Кэшбэк и Бонусы»

## Почему я здесь что-то рассказываю?

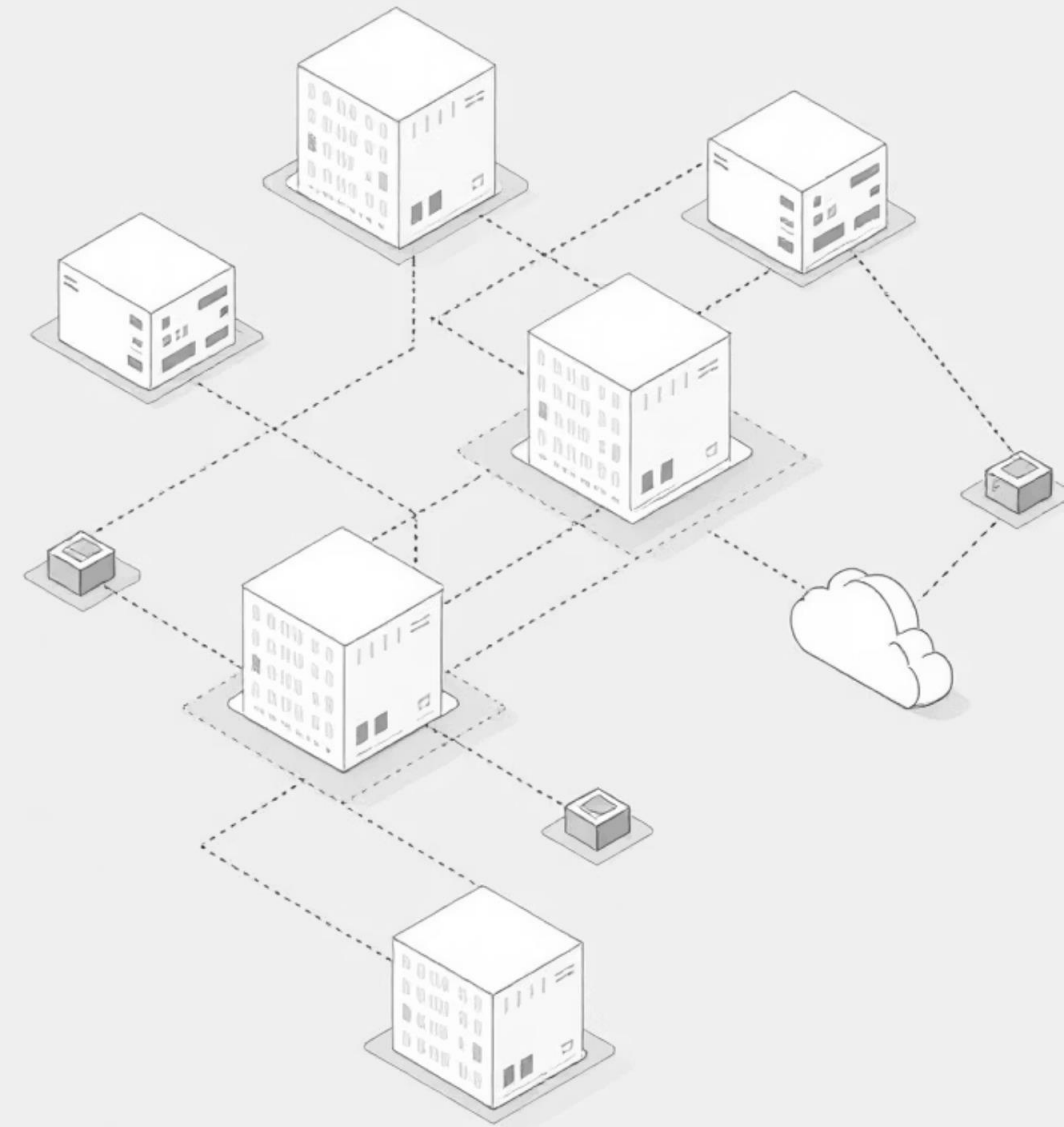
- Работаю с распределенными системами каждый день
- Имею опыт в несколько лет работы в сложных проектах
- «Поварился» в разных бизнесовых доменах и спроектировал ни один десяток взаимодействия систем, которые сейчас крутятся в production



Алексей Шкель  
tg: @alekksseeii

# План на семинар

1. Организационные вопросы
  - Формат проведения курса
  - Система оценивания
  - Коммуникации
2. Цели семинара
3. Введение основных понятий
4. Атрибуты качества системы
5. Ограничения централизованных систем
6. Что такое Распределенная Система?
7. Проблемы распределенных систем



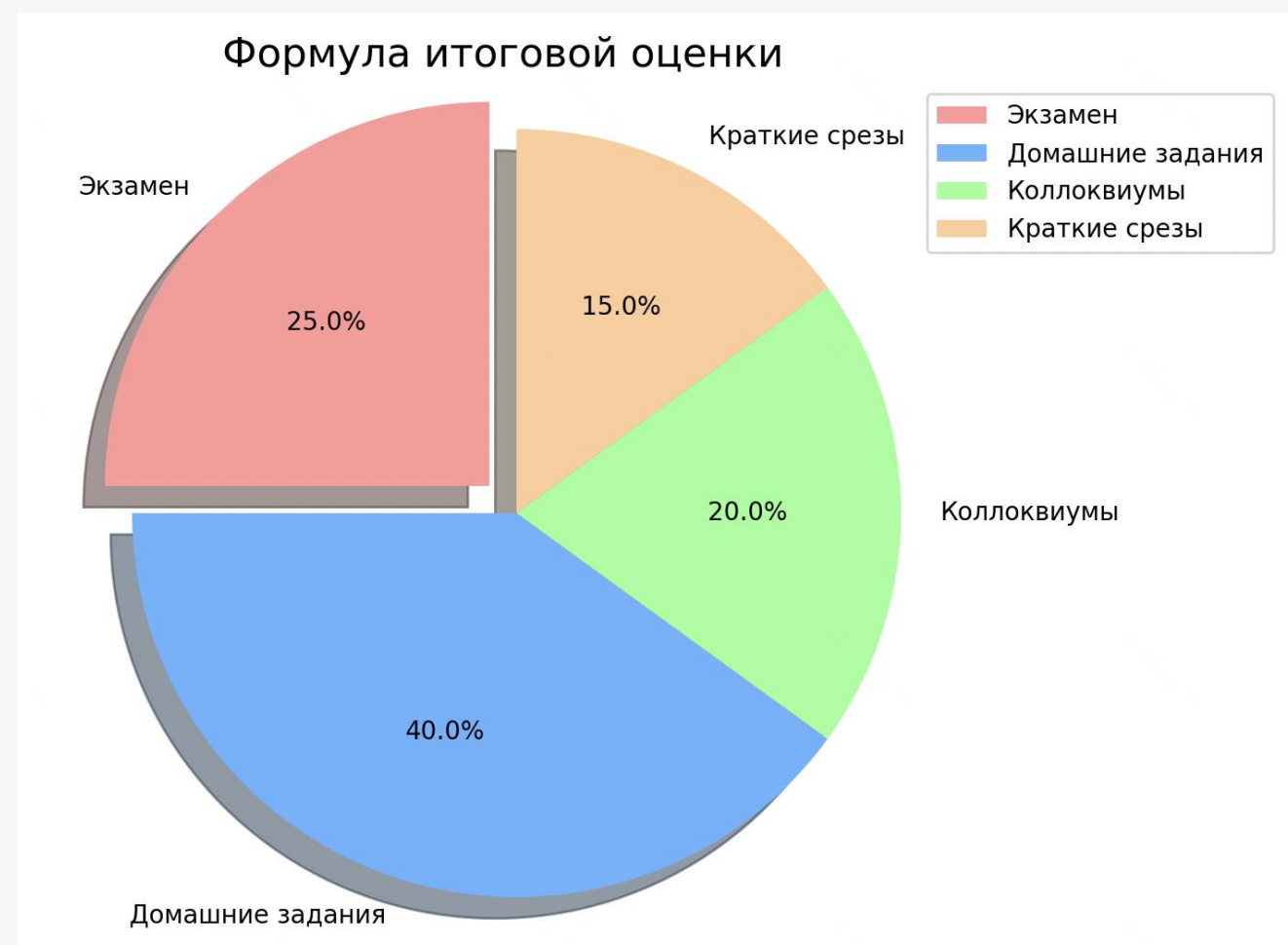
# Формат проведения курса и система оценивания

## Формат курса

**Проектные семинары** — пишем код каждый раз, применяем теорию на практике

## Формула итоговой оценки

- 25% — Экзамен
- 40% — Домашние задания (среднее)
- 20% — Коллоквиумы (среднее)
- 15% — Краткие срезы (среднее)

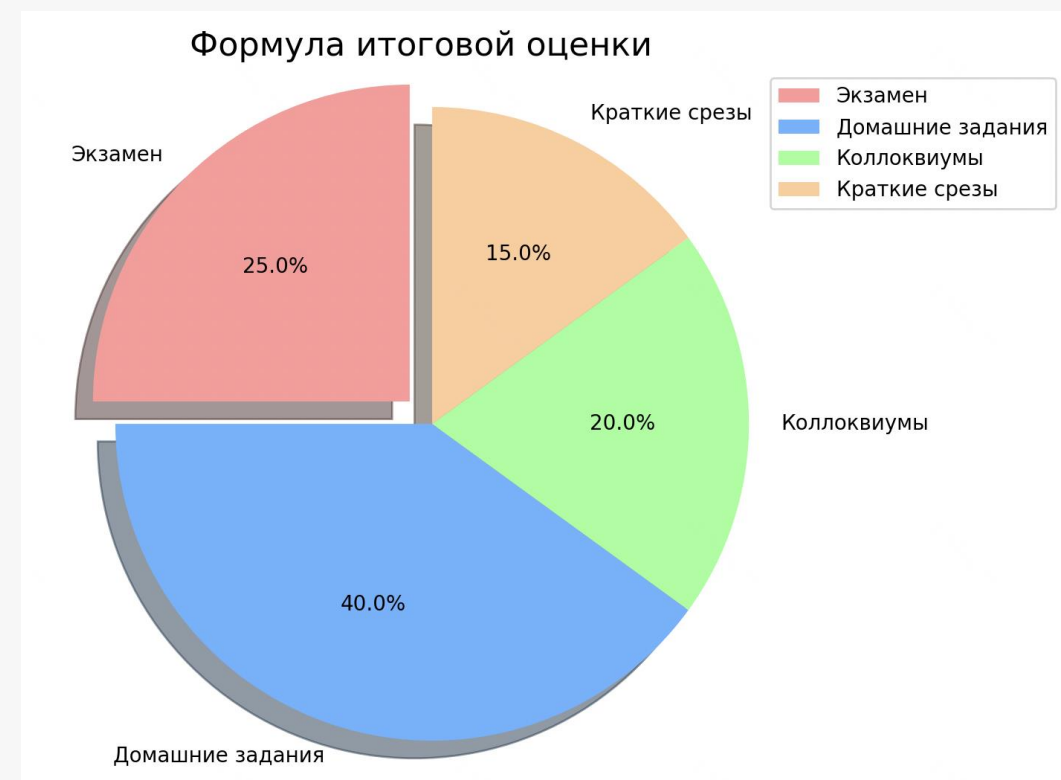




# Формат проведения курса и система оценивания

## Форматы контроля

- Экзамен — проводится по билетам один раз в конце года
- Коллоквиумы — проводятся по билетам в течение семестра
- Домашние задания — выдаются в течение семестра
  - Время выполнения: 2 недели
  - Формат проверки: устная защита перед преподавателем
- Краткие срезы — проводятся очно в начале некоторых семинаров
  - Описание краткое письменное тестирование на 5-10 минут
  - Проверяется письменно преподавателем до следующего семинара
  - Не пересдается



# Литература для подготовки



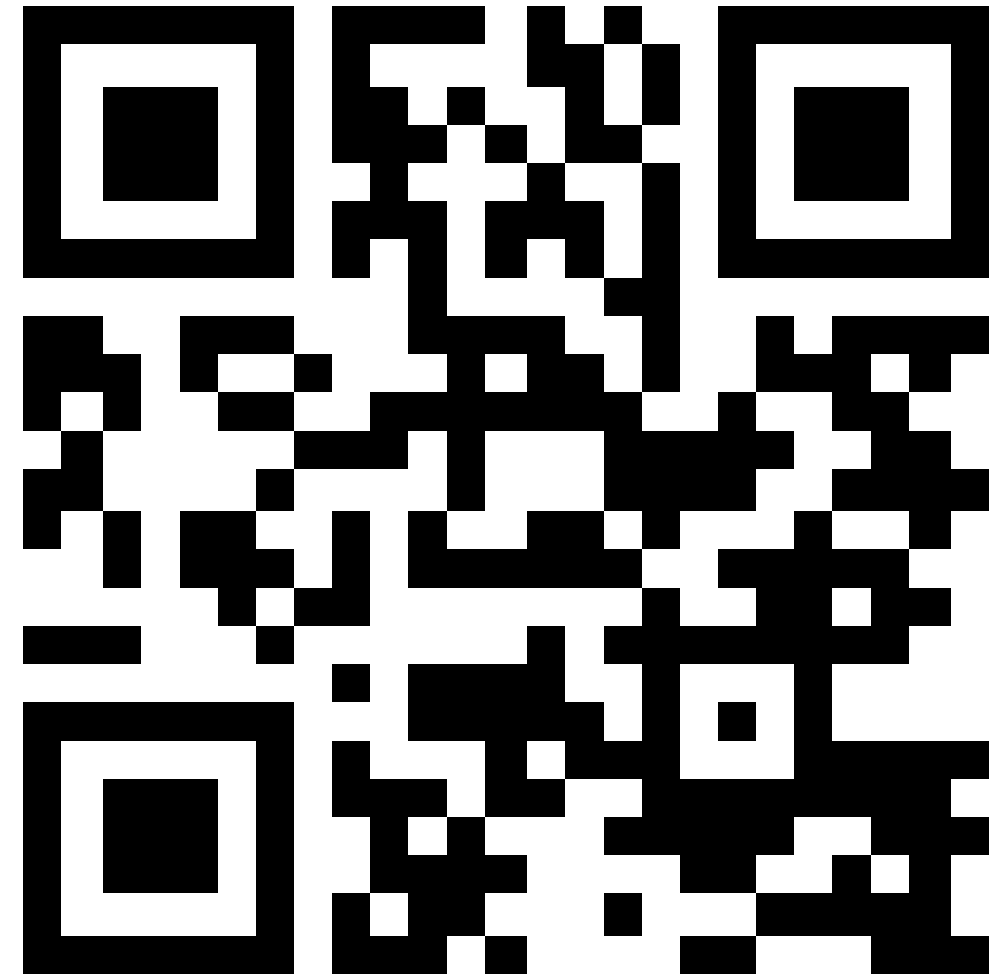
# Коммуникация

## Способы коммуникации на паре

- Вопросы по теме курса задаем сразу по пути прохождения материала
- Вопросы не по теме курса задаем после семинара
- Не общаемся во время:
  - Кратких срезов
  - Коллоквиумов
  - Экзаменов
- Во все остальное время — свободно общаемся, задаем вопросы по теме курса, спрашиваем, если что-то не понятно и т. д.

## Способы коммуникации вне пар

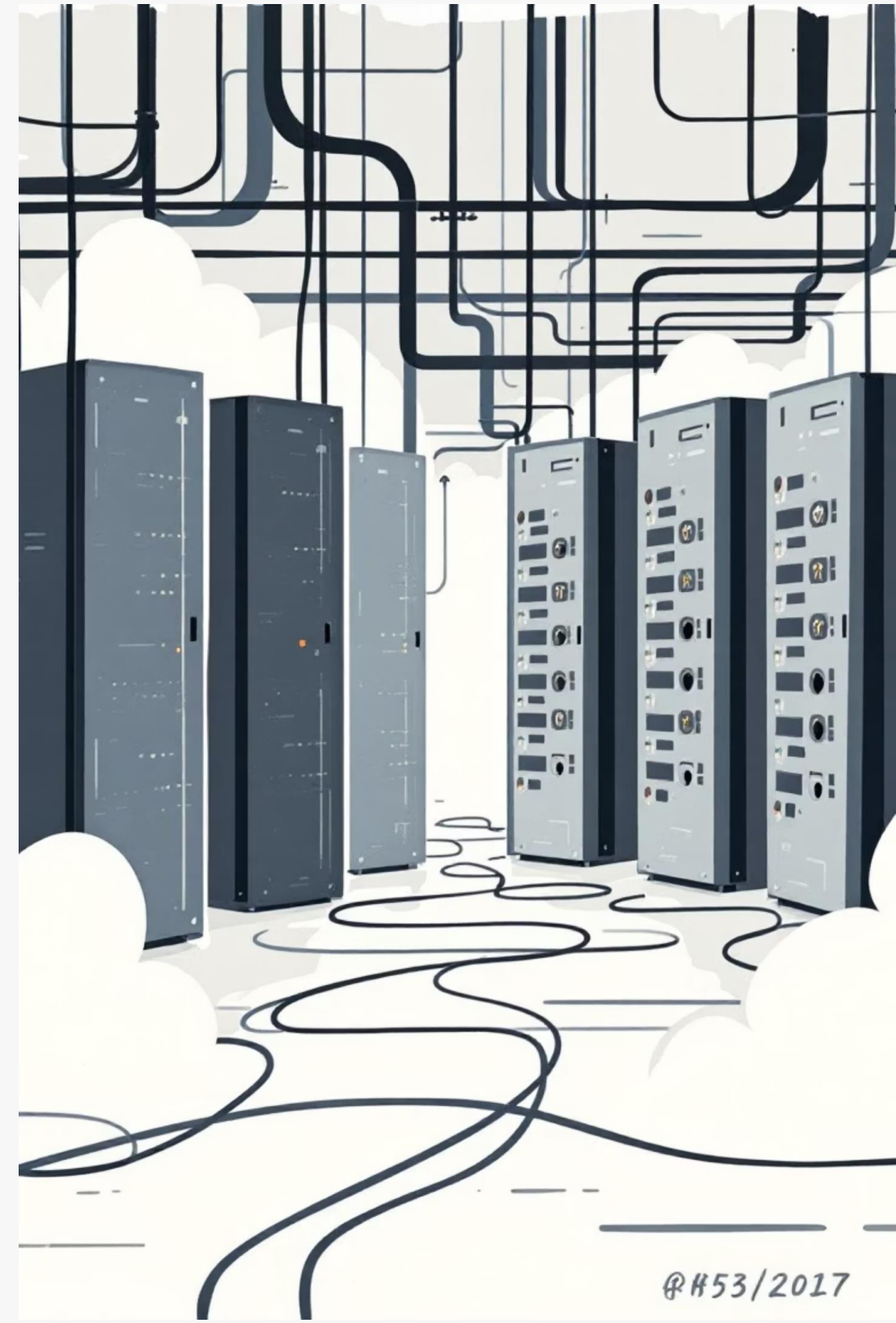
- Telegram группа по QR или по ссылке: <https://t.me/+YQS1PFfWXag3YjYy>
- Личка tg (при срочных и важных вопросах): @alekksseeii



# Цели семинара

## Чего мы сегодня достигнем?

1. Понимание сути распределенных систем
  - Сможете объяснить, что такое РС и почему они неотъемлемая часть жизни современного разработчика
2. Знание ключевых проблем
  - Сформулируем основные проблемы распределенных систем и способы их решения
3. Архитектурное мышление
  - Обсудим компромиссы, которые приходится делать при проектировании сложных систем







Что такое система?



## Что такое система?

Система – это набор взаимосвязанных компонентов, работающих вместе для достижения общей цели.



# Что такое система?

Система – это набор взаимосвязанных компонентов, работающих вместе для достижения общей цели.

Примеры систем вокруг нас





# Что такое система?

Система – это набор взаимосвязанных компонентов, работающих вместе для достижения общей цели.

## Примеры систем вокруг нас



Метрополитен

Поезда, станции, диспетчерские службы работают согласованно для перевозки пассажиров





# Что такое система?

Система – это набор взаимосвязанных компонентов, работающих вместе для достижения общей цели.

## Примеры систем вокруг нас



Метрополитен

Поезда, станции, диспетчерские службы работают согласованно для перевозки пассажиров



Энергосеть

Электростанции, трансформаторы, линии передач обеспечивают электричеством города



# Что такое система?

Система – это набор взаимосвязанных компонентов, работающих вместе для достижения общей цели.

## Примеры систем вокруг нас



### Метрополитен

Поезда, станции, диспетчерские службы работают согласованно для перевозки пассажиров



### Энергосеть

Электростанции, трансформаторы, линии передач обеспечивают электричеством города



### Автомобиль

Двигатель, трансмиссия, электроника взаимодействуют для обеспечения движения

А теперь давайте сузим фокус до **программных** систем и их особенностей.

# Программные (Software) системы

# Программные (Software) системы

Программная система – это совокупность компьютерных программ, данных, документации и конфигурационных файлов, работающих вместе для выполнения определенных задач.



# Программные (Software) системы

Программная система — это совокупность компьютерных программ, данных, документации и конфигурационных файлов, работающих вместе для выполнения определенных задач.

Программная система — это система, цели которой достигаются выполнением программного кода.

# Программные (Software) системы

Программная система — это совокупность компьютерных программ, данных, документации и конфигурационных файлов, работающих вместе для выполнения определенных задач.

Программная система — это система, цели которой достигаются выполнением программного кода.

Примеры программных систем

# Программные (Software) системы



Программная система — это совокупность компьютерных программ, данных, документации и конфигурационных файлов, работающих вместе для выполнения определенных задач.

Программная система — это система, цели которой достигаются выполнением программного кода.

## Примеры программных систем

- Мобильные приложения
- Веб-сайты и сервисы
- Компьютерные игры
- Операционные системы
- Базы данных

Какую систему можно назвать «хорошей»?



# Атрибуты качества системы



## Производительность и Масштабируемость (Performance and Scalability)

Система должна быстро обрабатывать  
запросы и масштабироваться при  
увеличении нагрузки

Производительность — это количественная мера того, насколько эффективно система выполняет свои задачи за единицу времени, учитывая скорость и использование ресурсов

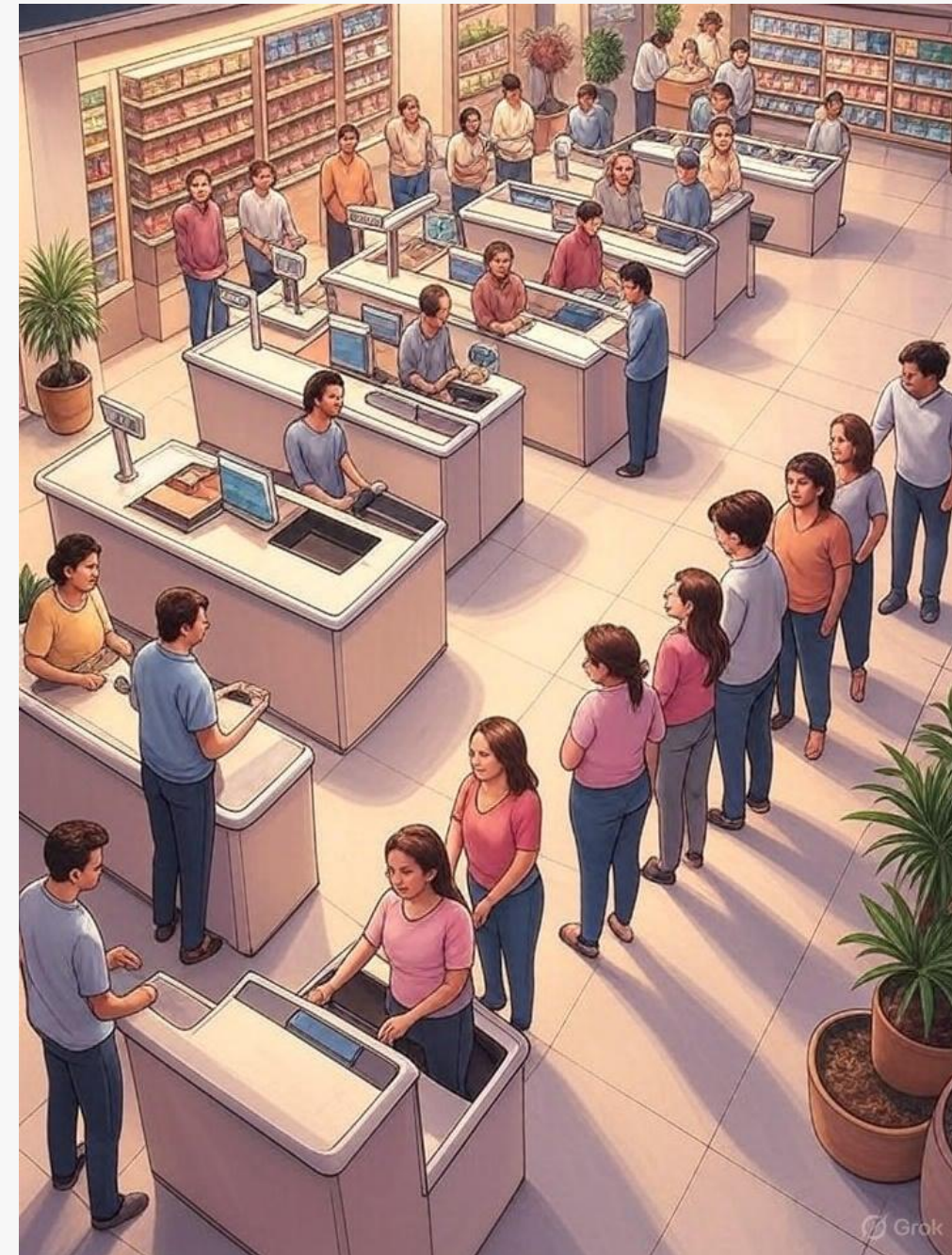
Масштабируемость - это способность системы справляться с увеличением рабочей нагрузки или объема решаемых задач путем добавления ресурсов, без потери производительности или функциональности

# Атрибуты качества системы

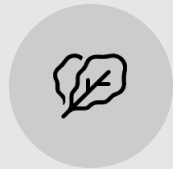


## Производительность и Масштабируемость (Performance and Scalability)

Система должна быстро обрабатывать  
запросы и масштабироваться при  
увеличении нагрузки



# Атрибуты качества системы



## Доступность (Availability)

Система должна быть готова обработать  
запрос, когда это нужно пользователю

Доступность системы — это способность системы выполнять свои функции по запросу в определенный период времени.

Измеряется в % времени, в которое система может обрабатывать клиентские запросы.

# Атрибуты качества системы

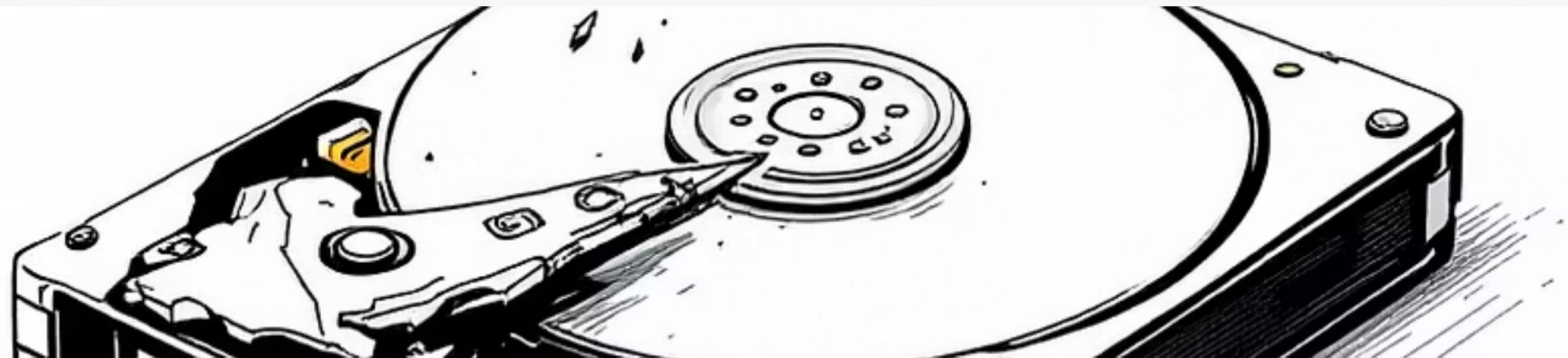


## Устойчивость (Durability)

Система не должна терять данные  
при любых обстоятельствах

Устойчивость - это способность системы не должна терять данные даже во время сбоев.

Если данные подтверждены – они сохранены навсегда, даже если сервер сгорел.



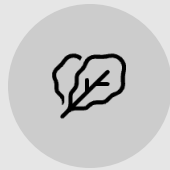


# Атрибуты качества системы



## Производительность и Масштабируемость (Performance and Scalability)

Система должна быстро обрабатывать запросы и масштабироваться при увеличении нагрузки



## Доступность (Availability)

Система должна быть готова обработать запрос, когда это нужно пользователю



## Устойчивость (Durability)

Система не должна терять данные при любых обстоятельствах



Не бывает идеальной системы на все случаи жизни, но есть ключевые атрибуты качества, которые определяют успех любого продукта.

# Попробуем создать свою систему

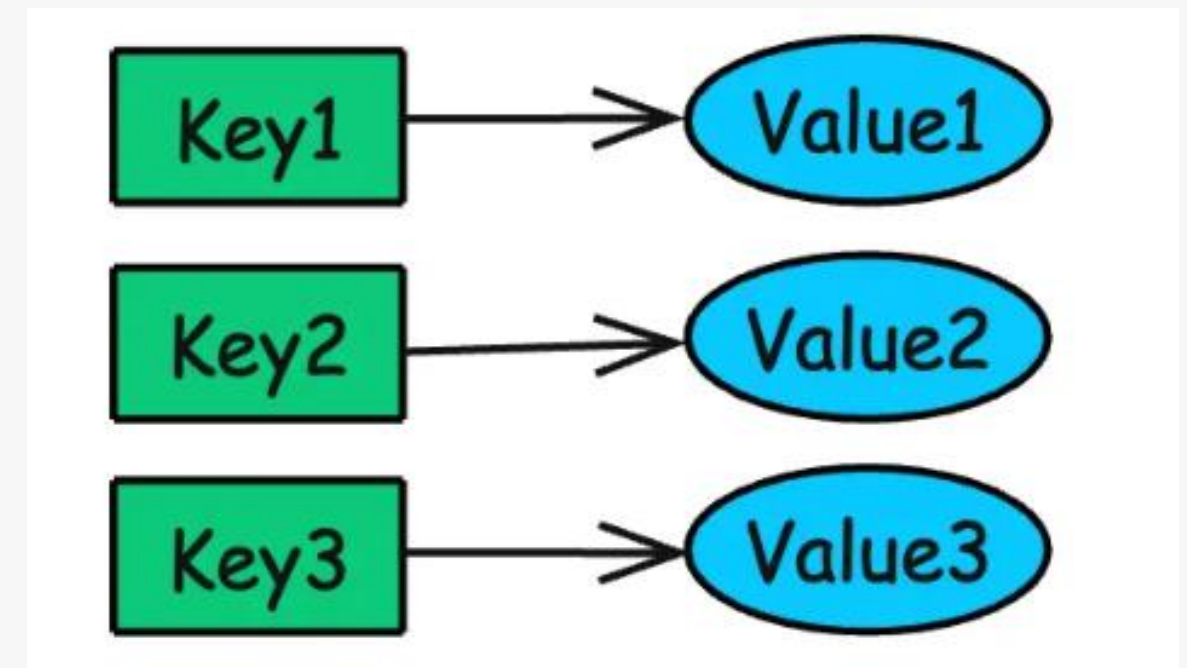
Упражнение: Проектируем систему

Представим, что мы создаем систему для хранения данных вида «ключ-значение»

У нас есть один большой, мощный сервер.

Давайте проверим, сможет ли он обеспечить наши атрибуты качества.

**Звучит просто, не так ли?**





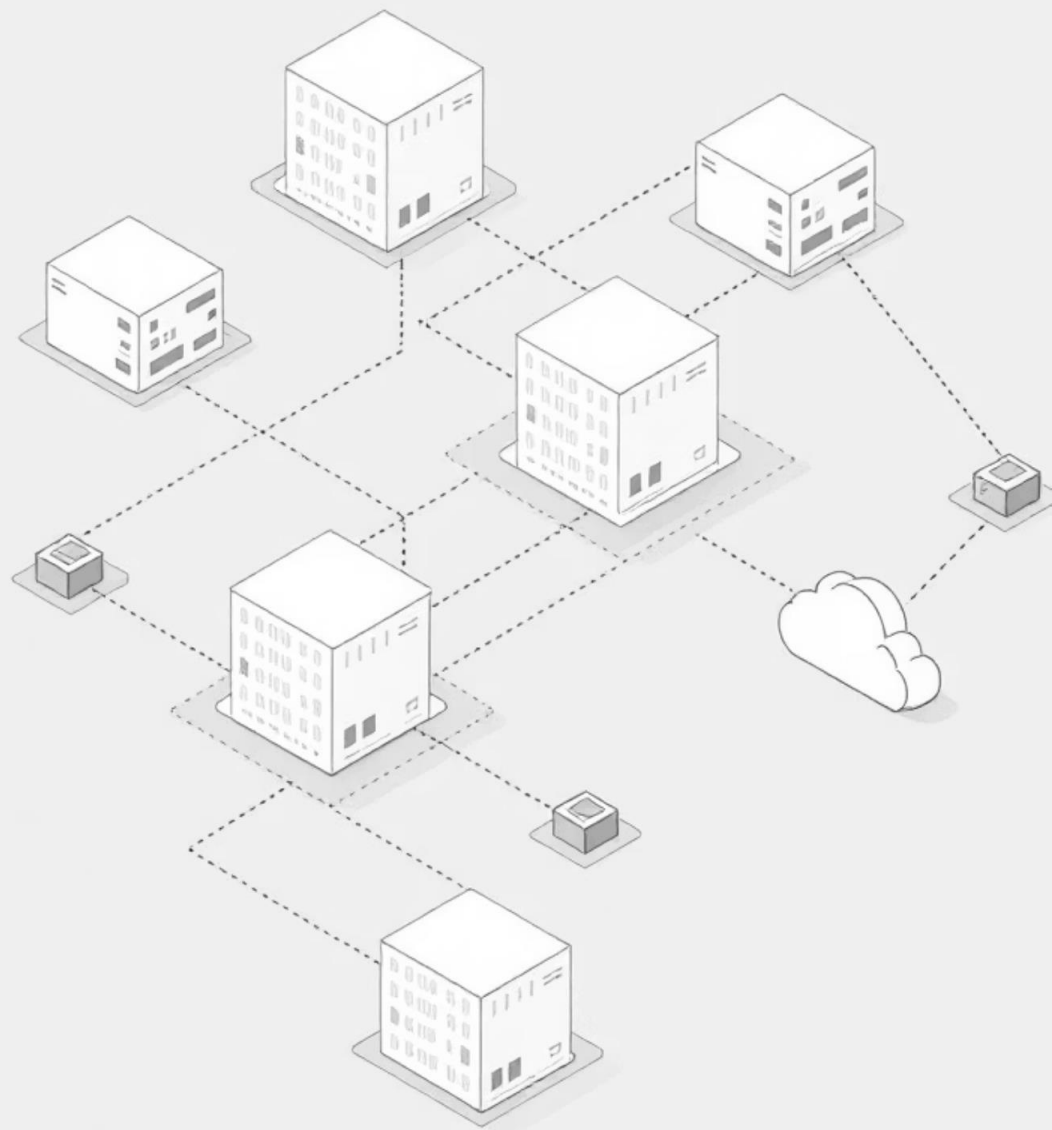
# Одна машина

## единая точка отказа

Централизованная архитектура фундаментально не может быть надежной, доступной и масштабируемой одновременно.

Можно купить сервер мощнее, но это дорого, есть физические ограничения, и это лишь откладывает проблему.

Нам нужен другой подход.



# Спасительный путь

## Что такое Распределенная Система?

Распределенная система — это система, компоненты которой расположены на различных сетевых компьютерах, которые координируют свои действия только путем обмена сообщениями и для пользователя выглядят как единое целое.

Звучит как решение всех проблем? Не совсем... У распределенных систем есть свои уникальные вызовы.

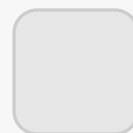
# The Network is Reliable... NOT

## Проблема №1: Ненадежная сеть

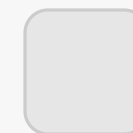
Сервер А отправил сообщение серверу Б. Мы не знаем, что случилось:



Сообщение затерялось в  
сети



Сообщение задержалось на  
неопределенное время



Сервер Б вообще упал и не  
отвечает



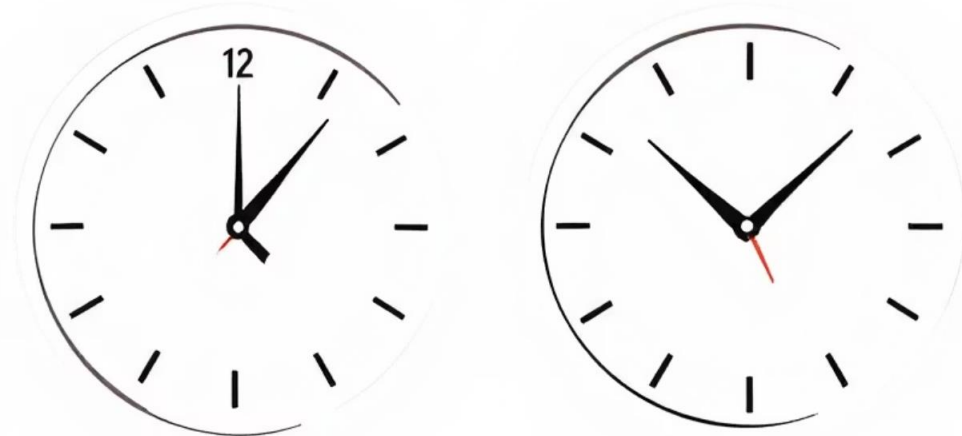
**Реальный пример:** Пользователь нажал "оплатить". Деньги списались, но ответ от сервера не дошел. Повторять ли запрос?

Эта неопределенность — корень всех зол в распределенных системах.



# Проблема №2: What time is it?

## Часы и порядок событий

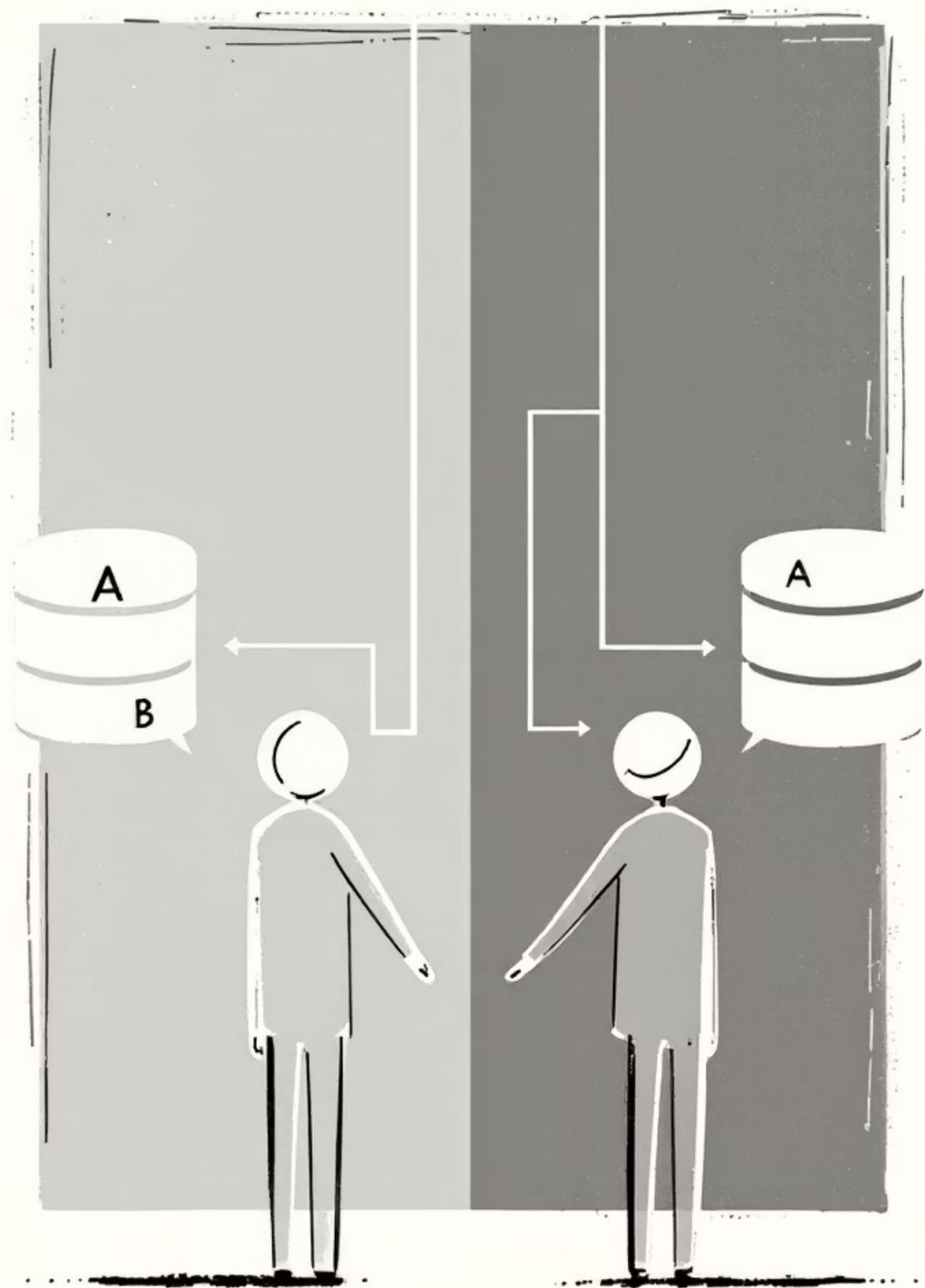


Часы на разных серверах идут с разной скоростью и могут расходиться.

**Критический вопрос:** Если два пользователя одновременно изменяют документ, как понять, чье изменение было позже?

⚠ **Реальный сценарий:** Лайк из Казани пришел с меткой времени 12:00:01, а лайк из Москвы – 12:00:02. Но часы в Москве могли отставать на 2 секунды. Чей лайк был первым?

Мы не можем полагаться на физическое время в распределенных системах.



# Проблема №3: Консистентность данных

## Согласованность данных

Мы сделали реплики данных в разных дата-центрах для доступности. Но возникает дилемма:

- 1 Ждем обновления везде  
Консистентность ✓ Производительность ✗
- 2 Отвечаем старыми данными  
Производительность ✓ Консистентность ✗
- 3 Не отвечаем при сбое  
Консистентность ✓ Доступность ✗

При сетевом сбое приходится жертвовать одним из атрибутов качества.  
(CAP-теорема)