

ARCHDAYS|

2022



Как подготовиться и пройти System Design Interview



Alexander
Polomodov

Tinkoff



Привет!

Меня зовут Александр

- X СТО мобильного банка, онлайн-привлечения, ...
- X Курирую архитектурные вопросы в компании
- X Курирую наши System Design Interview и Troubleshooting Interview
- X Провожу интервью по Менеджменту техническим руководителям



План нашей беседы

X Как выглядит процесс найма в общем



План нашей беседы

- X Как выглядит процесс найма в общем
- X Как выглядит system design interview



План нашей беседы

- X Как выглядит процесс найма в общем
- X Как выглядит *system design interview*
- X Как подготовится к интервью



План нашей беседы

- X Как выглядит процесс найма в общем
- X Как выглядит *system design interview*
- X Как подготовится к интервью
- X Как проявить себя на нем с лучшей стороны



План нашей беседы

- X Как выглядит процесс найма в общем
- X Как выглядит *system design interview*
- X Как подготовится к интервью
- X Как проявить себя на нем с лучшей стороны
- X Материалы для дальнейшего изучения



Как выглядит процесс найма

Предполагаемый уровень кандидата

ARCHDAYS|

2022



Senior+



Senior



Middle



Junior

Стадии найма

Предполагаемый уровень кандидата



Senior+



Senior



Middle



Junior

Предварительное
интервью

Технические
интервью

Интервью
с командой

ARCHDAYS|
2022

Оффер

Стадии найма

Предполагаемый уровень кандидата



Senior+



Senior



Middle



Junior

Предварительное
интервью

Интервью с
рекрутером

Технические
интервью

1. Алгоритмы

2. Языковая
секция

3. System
design

Интервью
с командой

ARCHDAYS|
2022

Оффер

Стадии найма

Предполагаемый уровень кандидата

ARCHDAYS|

2022

Оффер



Senior+



Senior



Middle



Junior

Предварительное
интервью

Интервью с
рекрутером



Технические
интервью

1. Алгоритмы

2. Языковая
секция

3. System
design



Секция по
проектированию систем
не требуется

Интервью
с командой



Стадии найма

Предполагаемый уровень кандидата

ARCHDAYS|

2022

Оффер

Предварительное
интервью

Технические
интервью

Интервью
с командой

Оффер



Senior+



Senior



Middle



Junior

Интервью с
рекрутером



1. Алгоритмы

2. Языковая
секция

3. System
design



Секция по
проектированию систем
в самый раз

Стадии найма



Как выглядят system design interview

1

Название

Контекст

Требования:

- функциональные
- нефункциональные



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



2

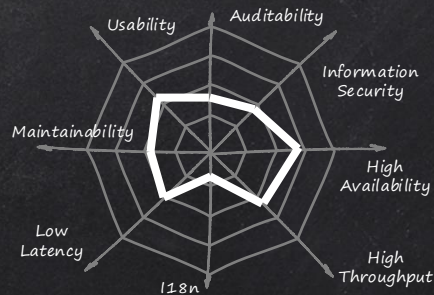
Формализация задачи:

- функциональность
- главные арх.

характеристики

Вопросы про сайзинг:

- users, requests, ...



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



2

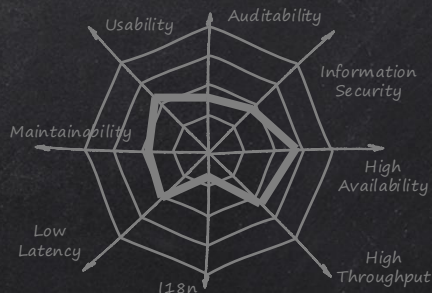
Формализация задачи:

- функциональность
- главные арх.

характеристики

Вопросы про сайзинг:

- users, requests, ...



ARCHDAYS|

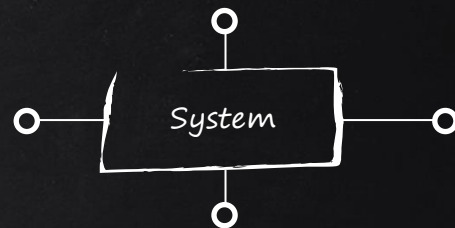
2022

3

Границы системы

Обрабатываемые сценарии

Публичный API



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



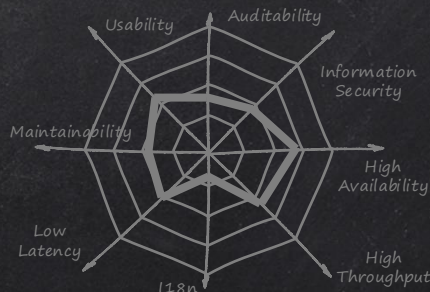
2

Формализация задачи:

- функциональность
- главные арх. характеристики

Вопросы про сайзинг:

- users, requests, ...



ARCHDAYS|

2022

3

Границы системы

Обрабатываемые сценарии

Публичный API



4

Потоки данных и компоненты системы

Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



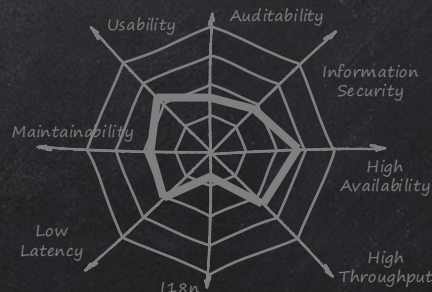
2

Формализация задачи:

- функциональность
- главные арх. характеристики

Вопросы про сайзинг:

- users, requests, ...



ARCHDAYS

2022

3

Границы системы

Обрабатываемые сценарии

Публичный API



4

Потоки данных и компоненты системы

Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются

5

Концептуальная схема готова:

- прописаны публичные API
- компоненты системы определены до классов систем (пример: k/v, rdbms, column family, etc)
- расписаны модели данных для обмена
- ясно что in-memory, а что persistent



1

Название
Контекст

Требования:

- функциональные
- нефункциональные



2

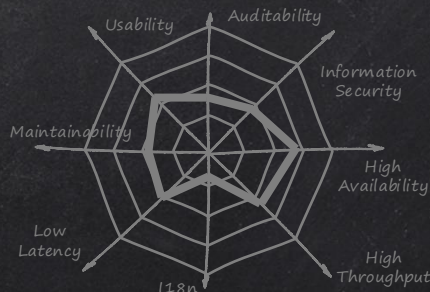
Формализация задачи:

- функциональность
- главные арх.

характеристики

Вопросы про сайзинг:

- users, requests, ...



ARCHDAYS

2022

3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются

6

Выбор конкретных технологий + расчет мощностей под нагрузку

5

Концептуальная схема готова:

- прописаны публичные API
- компоненты системы определены до классов систем (пример: k/v, rdbms, column family, etc)
- расписаны модели данных для обмена
- ясно что in-memory, а что persistent

Key-Value



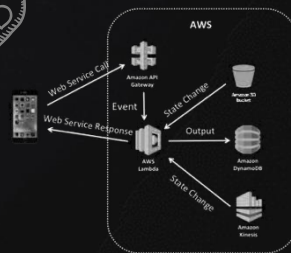
Column-Family



Graph



Document



1

Название
Контекст

Требования:

- функциональные
- нефункциональные



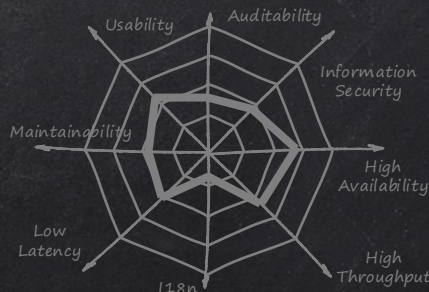
2

Формализация задачи:

- функциональность
- главные арх. характеристики

Вопросы про сайзинг:

- users, requests, ...



3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются

7

Дополнительные вопросы

5

Концептуальная схема готова:

- прописаны публичные API
- компоненты системы определены до классов систем (пример: k/v, rdbms, column family, etc)
- расписаны модели данных для обмена
- ясно что in-memory, а что persistent

6

Выбор конкретных технологий + расчет мощностей под нагрузку

Key-Value



Column-Family



Graph



Document





Как подготовиться к интервью

Формализация задачи

ARCHDAYS|
2022

Что стоит уметь



Формализация задачи

ARCHDAYS|
2022



Что стоит уметь

- X Задавать правильные вопросы и уточнять что входит в scope задачи, а что в него не входит

Формализация задачи

ARCHDAYS|
2022



Что стоит уметь

- X Задавать правильные вопросы и уточнять что входит в scope задачи, а что в него не входит
- X Собирать функциональные требования в виде желаемых сценариев системы

Формализация задачи

ARCHDAYS|
2022



Что стоит уметь

- Х Задавать правильные вопросы и уточнять что входит в scope задачи, а что в него не входит
- Х Собирать функциональные требования в виде желаемых сценариев системы
- Х Уточнять нефункциональные требования в виде самых важных архитектурных характеристик

Формализация задачи

ARCHDAYS|
2022



Что стоит уметь

- Х Задавать правильные вопросы и уточнять что входит в scope задачи, а что в него не входит
- Х Собирать функциональные требования в виде желаемых сценариев системы
- Х Уточнять нефункциональные требования в виде самых важных архитектурных характеристик
- Х Определять сравнительную важность этих требований между собой по важности

Формализация задачи

ARCHDAYS|

2022

Что стоит изучить



Формализация задачи

ARCHDAYS|

2022



Что стоит изучить

X Для функциональных требований

Формализация задачи



Что стоит изучить

- X Для функциональных требований
 - Use Cases из UML

Формализация задачи



Что стоит изучить

- X Для функциональных требований
 - Use Cases из UML
 - User story

Формализация задачи

ARCHDAYS|

2022



Что стоит изучить

- X Для функциональных требований
 - Use Cases из UML
 - User story
 - Jobs to be Done

Формализация задачи



Что стоит изучить

- Х Для функциональных требований
 - Use Cases из UML
 - User story
 - Jobs to be Done
- Х Для нефункциональных требований

Формализация задачи



Что стоит изучить

- Х Для функциональных требований
 - Use Cases из UML
 - User story
 - Jobs to be Done
- Х Для нефункциональных требований
 - Architecture Tradeoff Analysis Method

Формализация задачи



Что стоит изучить

- X Для функциональных требований
 - Use Cases из UML
 - User story
 - Jobs to be Done
- X Для нефункциональных требований
 - Architecture Tradeoff Analysis Method
- X Книгу Бугерса "Software Requirements. Third Edition" - amzn.to/3DfJea5

Формализация задачи

ARCHDAYS|
2022



Что стоит изучить

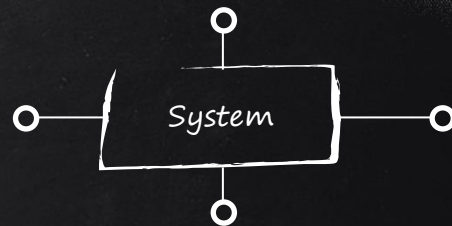
- X Для функциональных требований
 - Use Cases из UML
 - User story
 - Jobs to be Done
- X Для нефункциональных требований
 - Architecture Tradeoff Analysis Method
- X Книгу Бугерса "Software Requirements. Third Edition" - amzn.to/3DfJea5
- X Книгу "Software Architecture for Busy Developers" - bit.ly/busyArch

Границы системы

ARCHDAYS|

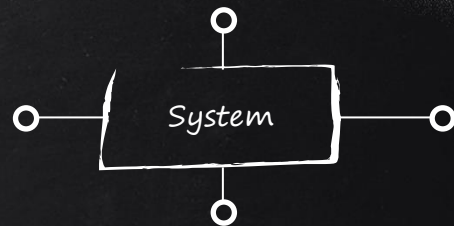
2022

Что стоит уметь



Границы системы

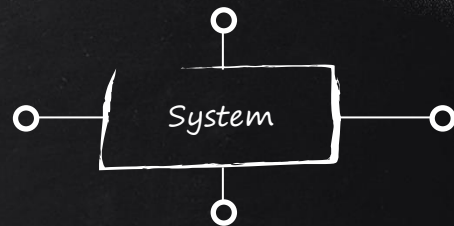
Что стоит уметь



- X Выбирать правильный способ интеграции нашей системы с другими системами – есть 4 вида интеграций: Files, DBs, API, Messaging

Границы системы

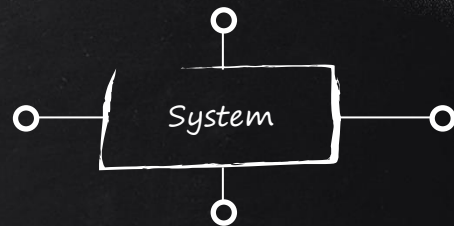
Что стоит уметь



- X Выбирать правильный способ интеграции нашей системы с другими системами – есть 4 вида интеграций: Files, DBs, API, Messaging
- X Проектировать точки интеграции системы под нужные сценарии

Границы системы

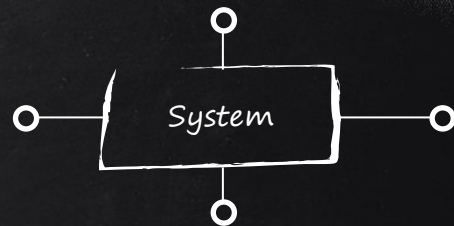
Что стоит уметь



- X Выбирать правильный способ интеграции нашей системы с другими системами – есть 4 вида интеграций: Files, DBs, API, Messaging
- X Проектировать точки интеграции системы под нужные сценарии
- X Если интеграция через API, то выбрать правильно подход, например, REST, RPC, GraphQL, AsyncAPI ...

Границы системы

Что стоит уметь



- X Выбирать правильный способ интеграции нашей системы с другими системами – есть 4 вида интеграций: Files, DBs, API, Messaging
- X Проектировать точки интеграции системы под нужные сценарии
- X Если интеграция через API, то выбрать правильно подход, например, REST, RPC, GraphQL, AsyncAPI ...
- X Описывать контракты в выбранном подходе

Границы системы

ARCHDAYS|

2022



Что стоит изучить

Границы системы

ARCHDAYS|
2022



Что стоит изучить

X System Context Diagram из C4 model - c4model.com

Границы системы

ARCHDAYS|
2022



Что стоит изучить

- X System Context Diagram из C4 model – c4model.com
- X Вспомнить сети и протоколы – OSI, UDP, TCP/IP, DNS, HTTPs 1/2/3, Websockets, ...

Границы системы

ARCHDAYS|
2022



Что стоит изучить

- X System Context Diagram из C4 model – c4model.com
- X Вспомнить сети и протоколы – OSI, UDP, TCP/IP, DNS, HTTPs 1/2/3, Websockets, ...
- X Балансировщики нагрузки – Nginx, HaProxy, ...

Границы системы

ARCHDAYS|

2022



Что стоит изучить

- X System Context Diagram из C4 model – c4model.com
- X Вспомнить сети и протоколы – OSI, UDP, TCP/IP, DNS, HTTPs 1/2/3, Websockets, ...
- X Балансировщики нагрузки – Nginx, HaProxy, ...
- X Подходы к интеграции систем между собой – files, DBs, API, Messaging, ...

Границы системы

ARCHDAYS|
2022



Что стоит изучить

- X System Context Diagram из C4 model – c4model.com
- X Вспомнить сети и протоколы – OSI, UDP, TCP/IP, DNS, HTTPs 1/2/3, Websockets, ...
- X Балансировщики нагрузки – Nginx, HaProxy, ...
- X Подходы к интеграции систем между собой – files, DBs, API, Messaging, ...
- X Книгу Tannenbaum "Computer Networks" – amzn.to/3sj51XS

Границы системы

ARCHDAYS|
2022



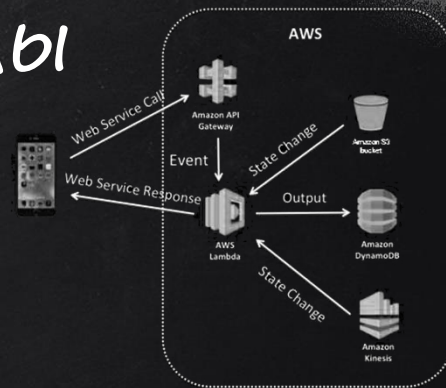
Что стоит изучить

- X System Context Diagram из C4 model - c4model.com
- X Вспомнить сети и протоколы – OSI, UDP, TCP/IP, DNS, HTTPs 1/2/3, Websockets, ...
- X Балансировщики нагрузки – Nginx, HaProxy, ...
- X Подходы к интеграции систем между собой – files, DBs, API, Messaging, ...
- X Книгу Tannenbaum "Computer Networks" - amzn.to/3sj51XS
- X Книгу Hohpe "Enterprise Integration Patterns" - amzn.to/3TvnTia

Основные потоки и компоненты системы

Что стоит уметь

ARCHDAYS|
2022

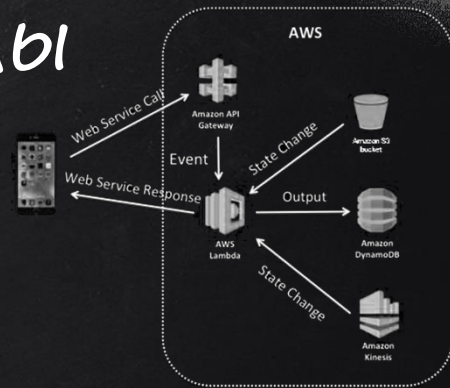


Основные потоки и компоненты системы

ARCHDAYS|
2022

Что стоит уметь

- X Описать основной поток работы (happy path), проходящий через компоненты системы

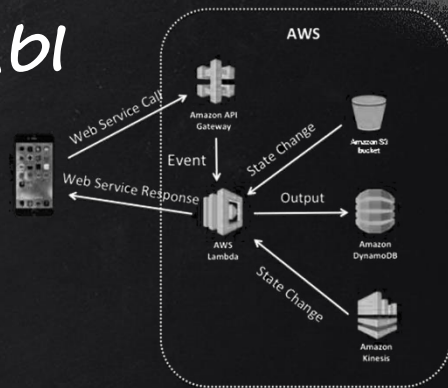


Основные потоки и компоненты системы

ARCHDAYS|
2022

Что стоит уметь

- Х Описать основной поток работы (happy path), проходящий через компоненты системы
- Х Очертить exceptional flows и как наша система их обрабатывает

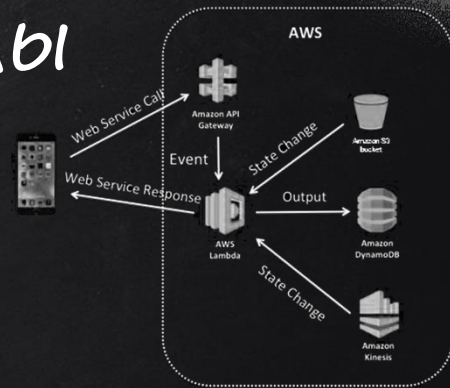


Основные потоки и компоненты системы

ARCHDAYS|
2022

Что стоит уметь

- X Описать основной поток работы (happy path), проходящий через компоненты системы
- X Очертить exceptional flows и как наша система их обрабатывает
- X Научиться определять read/write path для системы с учетом архитектурных характеристик



Основные потоки и компоненты системы

ARCHDAYS|

2022



Что стоит изучить

Основные потоки и компоненты системы

ARCHDAYS|

2022



Что стоит изучить

X Container Diagram из C4 Model - c4model.com

Основные потоки и компоненты системы

ARCHDAYS|

2022



Что стоит изучить

- X Container Diagram из C4 Model - c4model.com
- X Sequence и Activity Diagrams из UML

Основные потоки и компоненты системы

ARCHDAYS|

2022



Что стоит изучить

- X Container Diagram из C4 Model - c4model.com
- X Sequence и Activity Diagrams из UML
- X IDEFO и DFD диаграммы

Основные потоки и компоненты системы

ARCHDAYS|

2022



Что стоит изучить

- X Container Diagram из C4 Model - c4model.com
- X Sequence и Activity Diagrams из UML
- X IDEFO и DFD диаграммы
- X BPMN диаграммы

Основные потоки и компоненты системы



Что стоит изучить

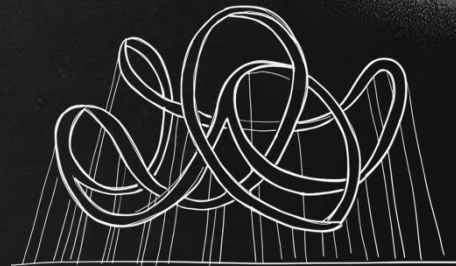
- X Container Diagram из C4 Model - c4model.com
- X Sequence и Activity Diagrams из UML
- X IDEFO и DFD диаграммы
- X BPMN диаграммы
- X Как балансировать read/write path для системы
Это хорошо расписано в книге Kleppmann "Designing Data-Intensive Applications" - amzn.to/3zO6vdo

Концептуальная схема

ARCHDAYS|

2022

Что стоит уметь

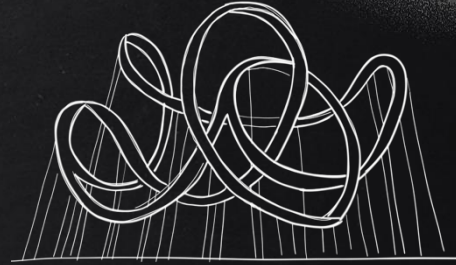


Концептуальная схема

ARCHDAYS|
2022

Что стоит уметь

✗ Разделять рабочие нагрузки на *stateful/stateless*

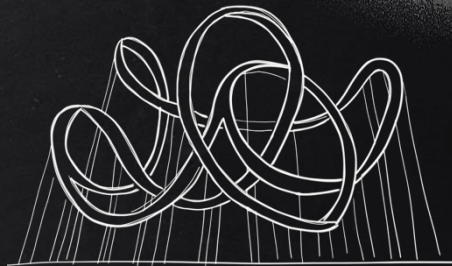


Концептуальная схема

ARCHDAYS|
2022

Что стоит уметь

- X Разделять рабочие нагрузки на *stateful/stateless*
- X Связывать между собой компоненты

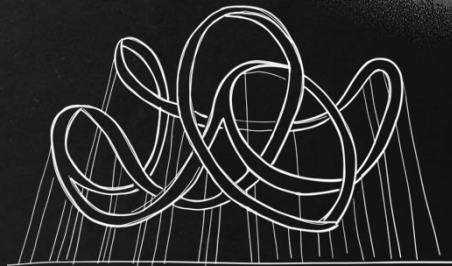


Концептуальная схема

ARCHDAYS|
2022

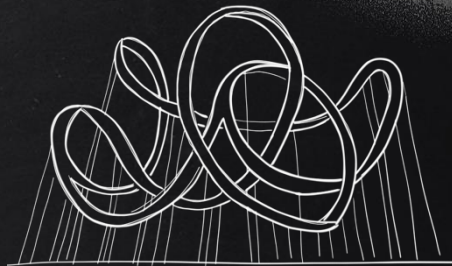
Что стоит уметь

- X Разделять рабочие нагрузки на *stateful/stateless*
- X Связывать между собой компоненты
- X Проектировать модели данных



Концептуальная схема

ARCHDAYS|
2022

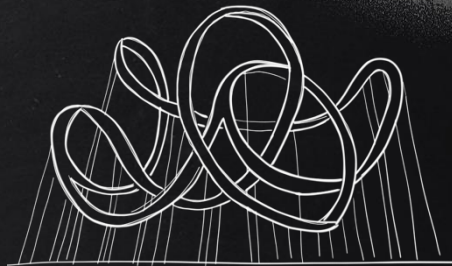


Что стоит уметь

- X Разделять рабочие нагрузки на *stateful/stateless*
- X Связывать между собой компоненты
- X Проектировать модели данных
- X Выбирать классы для *stateful* компонентов

Концептуальная схема

ARCHDAYS|
2022

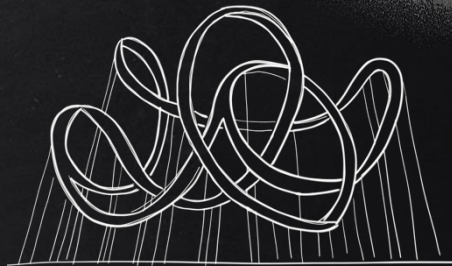


Что стоит уметь

- X Разделять рабочие нагрузки на *stateful/stateless*
- X Связывать между собой компоненты
- X Проектировать модели данных
- X Выбирать классы для *stateful* компонентов
 - RDBMS

Концептуальная схема

ARCHDAYS|
2022



Что стоит уметь

- X Разделять рабочие нагрузки на *stateful/stateless*
- X Связывать между собой компоненты
- X Проектировать модели данных
- X Выбирать классы для *stateful* компонентов
 - RDBMS
 - NoSQL по основным типам: *K/V, Document-oriented, Column-oriented, Graph*

Концептуальная схема

ARCHDAYS|

2022



Что стоит изучить

Концептуальная схема

ARCHDAYS|

2022



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*

Концептуальная схема



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*
- X Как моделировать данные *ER Diagrams*, *Class Diagrams*

Концептуальная схема

ARCHDAYS|
2022



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*
- X Как моделировать данные *ER Diagrams*, *Class Diagrams*
- X Принципы для *stateless* приложений из *The Twelve-Factor App*

Концептуальная схема

ARCHDAYS|
2022



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*
- X Как моделировать данные *ER Diagrams*, *Class Diagrams*
- X Принципы для *stateless* приложений из *The Twelve-Factor App*
- X Какие базы данных бывают и их границы применимости

Концептуальная схема

ARCHDAYS|
2022



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*
- X Как моделировать данные *ER Diagrams*, *Class Diagrams*
- X Принципы для *stateless* приложений из *The Twelve-Factor App*
- X Какие базы данных бывают и их границы применимости
- X Как работают оркестраторы рабочих нагрузок, навроде *K8s*

Концептуальная схема

ARCHDAYS|
2022



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*
- X Как моделировать данные *ER Diagrams*, *Class Diagrams*
- X Принципы для *stateless* приложений из *The Twelve-Factor App*
- X Какие базы данных бывают и их границы применимости
- X Как работают оркестраторы рабочих нагрузок, навроде *K8s*
- X Книгу “*Software Architecture: The Hard parts*” – bit.ly/softArchHardParts

Концептуальная схема

ARCHDAYS|
2022



Что стоит изучить

- X Как делить систему на части – Domain Driven Design и конкретно виды *subdomains* и *Bounded Context*
- X Как моделировать данные *ER Diagrams*, *Class Diagrams*
- X Принципы для *stateless* приложений из *The Twelve-Factor App*
- X Какие базы данных бывают и их границы применимости
- X Как работают оркестраторы рабочих нагрузок, навроде *K8s*
- X Книгу “*Software Architecture: The Hard parts*” – bit.ly/softArchHardParts
- X Книгу “*K8s Patterns*” – bit.ly/k8sPatterns

Реальная схема и желаемые архитектурные характеристики

Что стоит уметь



Реальная схема и желаемые архитектурные характеристики

Что стоит уметь

- X Выбирать конкретных представителей систем из перечисленных в концептуальной схеме



Реальная схема и желаемые архитектурные характеристики

Что стоит уметь

- X *Выбирать конкретных представителей систем из перечисленных в концептуальной схеме*
- X *Думать в концепции failure domains*



Реальная схема и желаемые архитектурные характеристики

Что стоит уметь

- Х Выбирать конкретных представителей систем из перечисленных в концептуальной схеме
- Х Думать в концепции *failure domains*
- Х Проектировать систему с учетом *day-2 operations*: эксплуатация и развитие (*logs, monitoring, migrations, ...*)



Реальная схема и желаемые архитектурные характеристики

Что стоит изучить



Реальная схема и желаемые архитектурные характеристики

Что стоит изучить

- X Потрогать руками основных представителей систем из перечисленных в концептуальной схеме



Реальная схема и желаемые архитектурные характеристики

Что стоит изучить



- Х Потрогать руками основных представителей систем из перечисленных в концептуальной схеме
- Х Понять гарантии, которые дадут эти решения и возможный диапазон рабочих нагрузок, которые они потянут

Реальная схема и желаемые архитектурные характеристики

Что стоит изучить



- X Потрогать руками основных представителей систем из перечисленных в концептуальной схеме
- X Понять гарантии, которые дают эти решения и возможный диапазон рабочих нагрузок, которые они потянут
- X Изучить SRE подходы и потрогать руками инструменты для logs, monitoring, migrations, ...

Реальная схема и желаемые архитектурные характеристики

Что стоит изучить

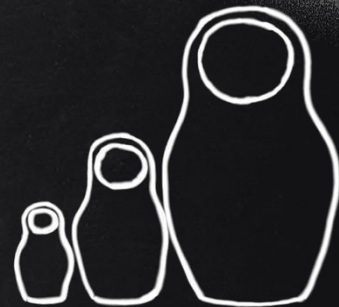


- X Потрогать руками основных представителей систем из перечисленных в концептуальной схеме
- X Понять гарантии, которые дают эти решения и возможный диапазон рабочих нагрузок, которые они потянут
- X Изучить SRE подходы и потрогать руками инструменты для logs, monitoring, migrations, ...
- X SRE книги от Google и обязательно "Building Secure and Reliable Systems" - bit.ly/ReliableSystems

Масштабирование под нагрузку

ARCHDAYS|
2022

Что стоит уметь

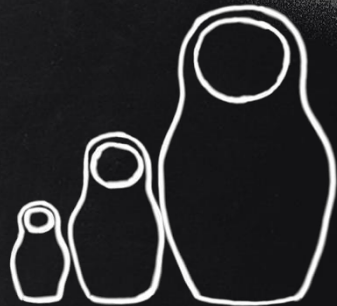


Масштабирование под нагрузку

ARCHDAYS|
2022

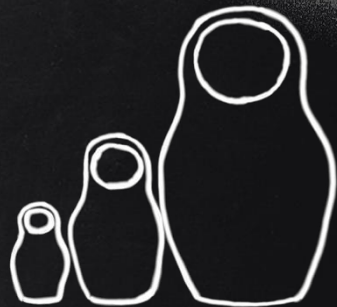
Что стоит уметь

X Масштабировать *stateless* компоненты



Масштабирование под нагрузку

ARCHDAYS|
2022

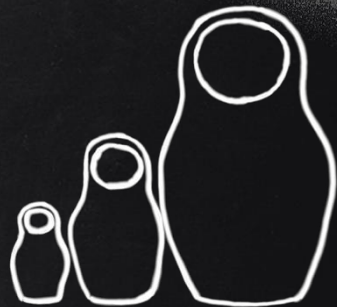


Что стоит уметь

- X Масштабировать *stateless* компоненты
 - С использованием оркестраторов и HPA, VPA

Масштабирование под нагрузку

ARCHDAYS|
2022

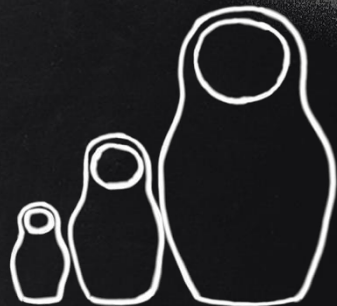


Что стоит уметь

- X Масштабировать *stateless* компоненты
 - С использованием оркестраторов и HPA, VPA
- X Масштабировать *stateful* компоненты

Масштабирование под нагрузку

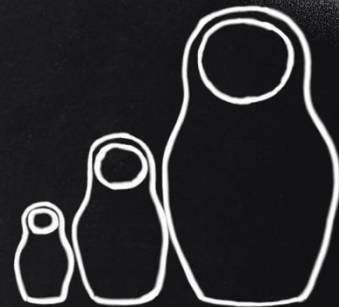
ARCHDAYS|
2022



Что стоит уметь

- X Масштабировать *stateless* компоненты
 - С использованием оркестраторов и HPA, VPA
 - X Масштабировать *stateful* компоненты
 - Масштабировать по чтению – Replication
- Здесь могут быть ослаблены требования по *consistency*

Масштабирование под нагрузку



Что стоит уметь

- X Масштабировать *stateless* компоненты
 - С использованием оркестраторов и HPA, VPA
- X Масштабировать *stateful* компоненты
 - Масштабировать по чтению – *Replication*
Здесь могут быть ослаблены требования по *consistency*
 - Масштабировать по записи – *Partitioning* и *Sharding*
Здесь нужен определенный паттерн доступа к данным, чтобы схема работала

Масштабирование под нагрузку

ARCHDAYS|
2022

Что стоит изучить



Масштабирование под нагрузку

ARCHDAYS|
2022

Что стоит изучить

X Книгу Kleppmann "Designing Data-Intensive Applications" –
amzn.to/3zO6vdo



Масштабирование под нагрузку

Что стоит изучить

- X Книгу Kleppmann "Designing Data-Intensive Applications" – amzn.to/3z06vdo
- X Книгу "K8s Patterns" – bit.ly/k8sPatterns



Масштабирование под нагрузку

ARCHDAYS|
2022

Что стоит изучить



- X Книгу Kleppmann "Designing Data-Intensive Applications" – amzn.to/3z06vdo
- X Книгу "K8s Patterns" – bit.ly/k8sPatterns
- X Книгу "Database Internals" – bit.ly/storageEng и bit.ly/distrsystems

Масштабирование под нагрузку

ARCHDAYS|
2022

Что стоит изучить



- X Книгу Kleppmann "Designing Data-Intensive Applications" – amzn.to/3z06vdo
- X Книгу "K8s Patterns" – bit.ly/k8sPatterns
- X Книгу "Database Internals" – bit.ly/storageEng и bit.ly/distrsystems
- X Книгу "Distributed Systems" – bit.ly/distrSystems

Масштабирование под нагрузку

Что стоит изучить



- X Книгу Kleppmann "Designing Data-Intensive Applications" – amzn.to/3zO6vdo
- X Книгу "K8s Patterns" – bit.ly/k8sPatterns
- X Книгу "Database Internals" – bit.ly/storageEng и bit.ly/distrsystems
- X Книгу "Distributed Systems" – bit.ly/distrSystems
- X Посчитать и погонять нагрузочные тесты с любимыми вашими инструментами



Советы для успешного прохождения интервью

Список советов

ARCHDAYS|

2022

X Помнить о времени



Список советов

ARCHDAYS|

2022



- X Помнить о времени
- X Не начинать проектировать сразу

Список советов

ARCHDAYS|
2022

- X Помнить о времени
- X Не начинать проектировать сразу
- X Делиться своими размышлениями при проектировании с интервьюером



Список советов

ARCHDAYS|

2022



- X Помнить о времени
- X Не начинать проектировать сразу
- X Делиться своими размышлениями при проектировании с интервьюером
- X Проявлять самостоятельность в проектировании – вести интервью

Список советов

ARCHDAYS|
2022



- X Помнить о времени
- X Не начинать проектировать сразу
- X Делиться своими размышлениями при проектировании с интервьюером
- X Проявлять самостоятельность в проектировании – вести интервью
- X Реагировать на наводящие вопросы и уточнения

Список советов

ARCHDAYS|

2022



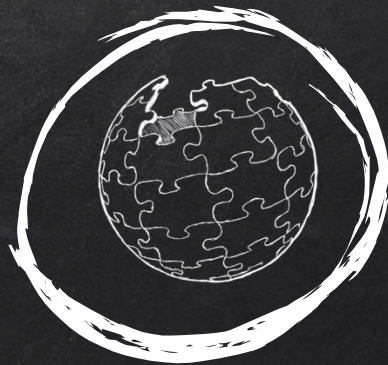
- Х Помнить о времени
- Х Не начинать проектировать сразу
- Х Делиться своими размышлениями при проектировании с интервьюером
- Х Проявлять самостоятельность в проектировании – вести интервью
- Х Реагировать на наводящие вопросы и уточнения
- Х Говорить уверенно о том, что знаешь, а не то

Список советов

ARCHDAYS|
2022



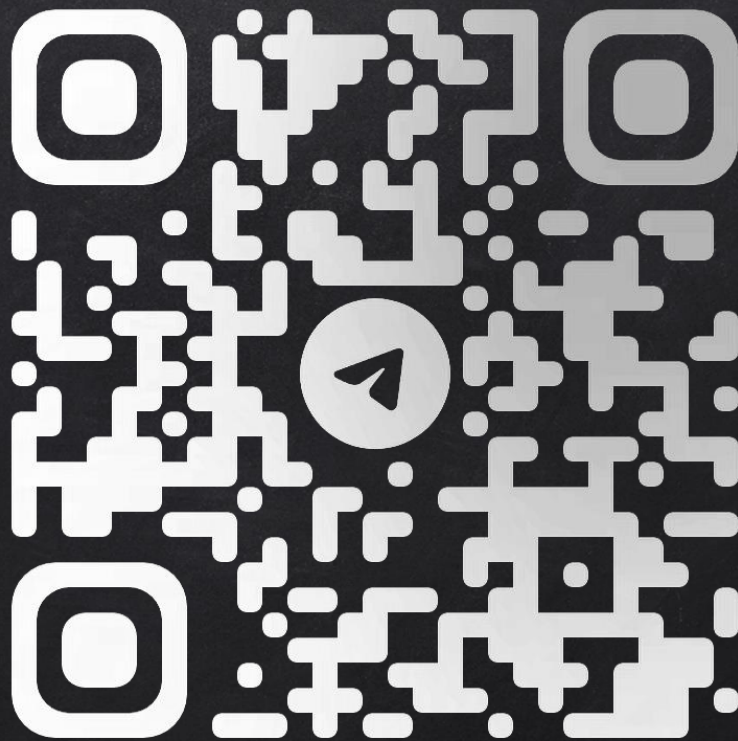
- Х Помнить о времени
- Х Не начинать проектировать сразу
- Х Делиться своими размышлениями при проектировании с интервьюером
- Х Проявлять самостоятельность в проектировании – вести интервью
- Х Реагировать на наводящие вопросы и уточнения
- Х Говорить уверенно о том, что знаешь, а не то
- Х Подготовить вопросы к интервьюеру на случай, если для них останется время



Материалы для дальнейшего изучения

ARCHDAYS|

2022



@BOOK_CUBE

t.me/book_cube/629