Skillbox

Атрибуты качества

Андрей Гордиенков
Solution Architect
ABAX

Skillbox

В прошлом уроке

- Сценарии атрибутов качества и их типы
- Как прорабатывать и описывать?

В этом уроке

- Атрибуты качества и их таксономия
- Тактики реализации

Skillbox

Атрибуты качества

accessibility
accountability
accuracy
adaptability
administrability
affordability
agility
auditability
autonomy
availability
compatibility
composability
configurability
correctness
credibility
customizability
debuggability
degradability
determinability
demonstrability
dependability
deployability
discoverability
distributability
durability
effectiveness
efficiency
evolvability
extensibility
failure transparency
fault-tolerance
fidelity
flexibility
inspectability
installability
integrity
interchangeability
interoperability
learnability
localizability
maintainability
manageability
mobility
modifiability
modularity
observability
operability
orthogonality
portability
precision
predictability
process capabilities
producibility
provability
recoverability
relevance
reliability
repeatability
reproducibility
resilience
responsiveness
reusability
robustness
safety
scalability
seamlessness
self-sustainability
serviceability
curability
simplicity
stability
standards compliance
survivability
sustainability
tailorability
testability
timeliness
traceability
transparency
ubiquity
understandability
upgradability
usability
vulnerability

Модели качества

- **ISO 20010**
- **ГОСТ 34**
- **МакКолл** (McCall’s Quality Model)
- **Боэм** (Boehm’s Quality Model)
- **Дроми** (Dromey’s Quality Model)

Группы атрибутов качества

Runtime

Атрибуты, относящиеся ко времени работы ПО.

- Доступность
- Надёжность
- Масштабируемость
- Время хранения данных
- Удобство использования
- Безопасность
- Конфигурируемость
- Производительность

Design

Атрибуты, определяющие аспекты проектирования ПО.

- Повторное использование
- Расширяемость
- Переносимость
- Взаимодействие с другими системами
- Поддерживаемость
- Модульность
- Тестируемость
- Локализация

- **Доступность** (Availability)
- **Изменяемость** (Modifiability)
- **Скорость** (Performance)
- **Безопасность** (Security)
- **Тестируемость** (Testability)
- **Удобство использования** (Usability)

Skillbox

Доступность

Доступность — это период времени, в течение которого система функционирует нормально и без сбоев.

$$\alpha = \frac{MTBF}{MTBF + MTTR}$$

MTBF — средняя наработка на отказ. Mean time between failures.

MTTF — средняя наработка до отказа. Mean time to failure.

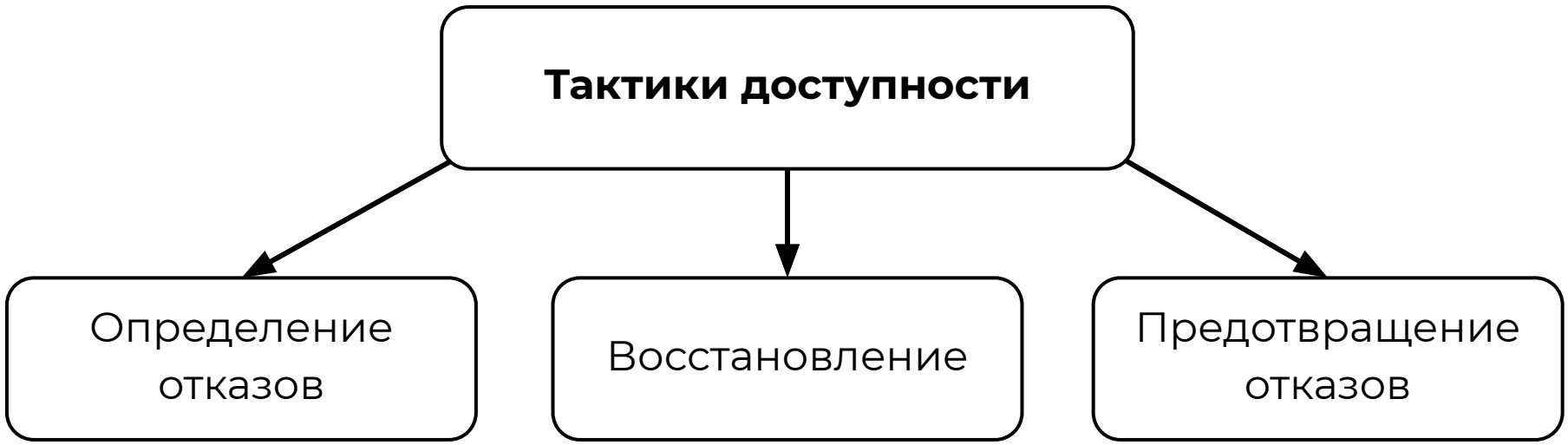
MTTR — среднее время восстановления. Mean time to Recovery.

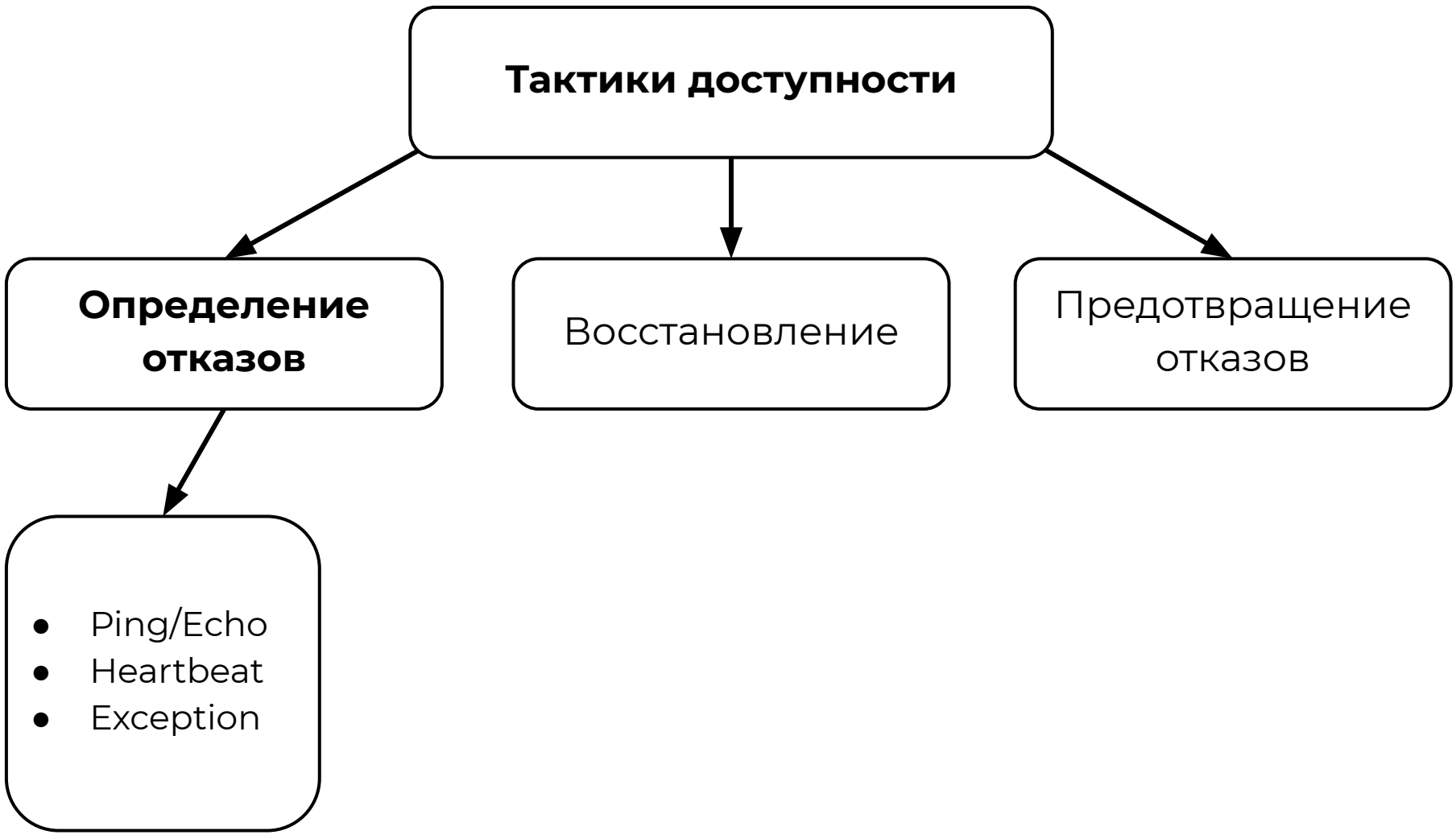
Skillbox

Доступность

Нормальное значение для большинства бизнес-приложений — 95–99 %.

Доступность	Время простоя
99 %	3 дня 15,6 часа
99,9 %	8 часов 6 сек.
99,99 %	52 минуты 34 сек.
99,999 %	5 минут 15 сек.
99,9999 %	32 сек.


Тактики доступности
Определение отказов
Восстановление
Предотвращение отказов


Тактики доступности
Определение отказов
Восстановление
Предотвращение отказов
Ping/Echo
Heartbeat
Exception


Тактики доступности
Определение отказов
Восстановление
Предотвращение отказов
Ping/Echo
Heartbeat
Exception
Подготовка и починка
Повторный запуск
Voting
Active redundancy
Passive redundancy
Spare
Voting
Active redundancy
Passive redundancy
Spare


Тактики доступности
Определение отказов
Восстановление
Предотвращение отказов
Ping/Echo
Heartbeat
Exception
Подготовка и починка
Voting
Active redundancy
Passive redundancy
Spare
Повторный запуск
Voting
Active redundancy
Passive redundancy
Spare
Removal from service
Transactions
Process monitor

Доступность

Уточняющие характеристики:

- локализация операций
- требования работы без доступности сети
- восстанавливаемость
- устойчивость

Вопросы для самопроверки

- Принято ли во внимание восстановление данных из неполных или повреждённых резервных копий?
- Как хорошо система будет реагировать на неисправности, в том числе регистрирует ли их и сообщает о них должным образом администраторам и пользователям?
- Определено ли время, необходимое для восстановления после сбоя?
- Обеспечивает ли технология резервного копирования транзакционную целостность восстановленных данных?
- Соответствует ли целевая архитектура требованиям доступности?

Вопросы для самопроверки

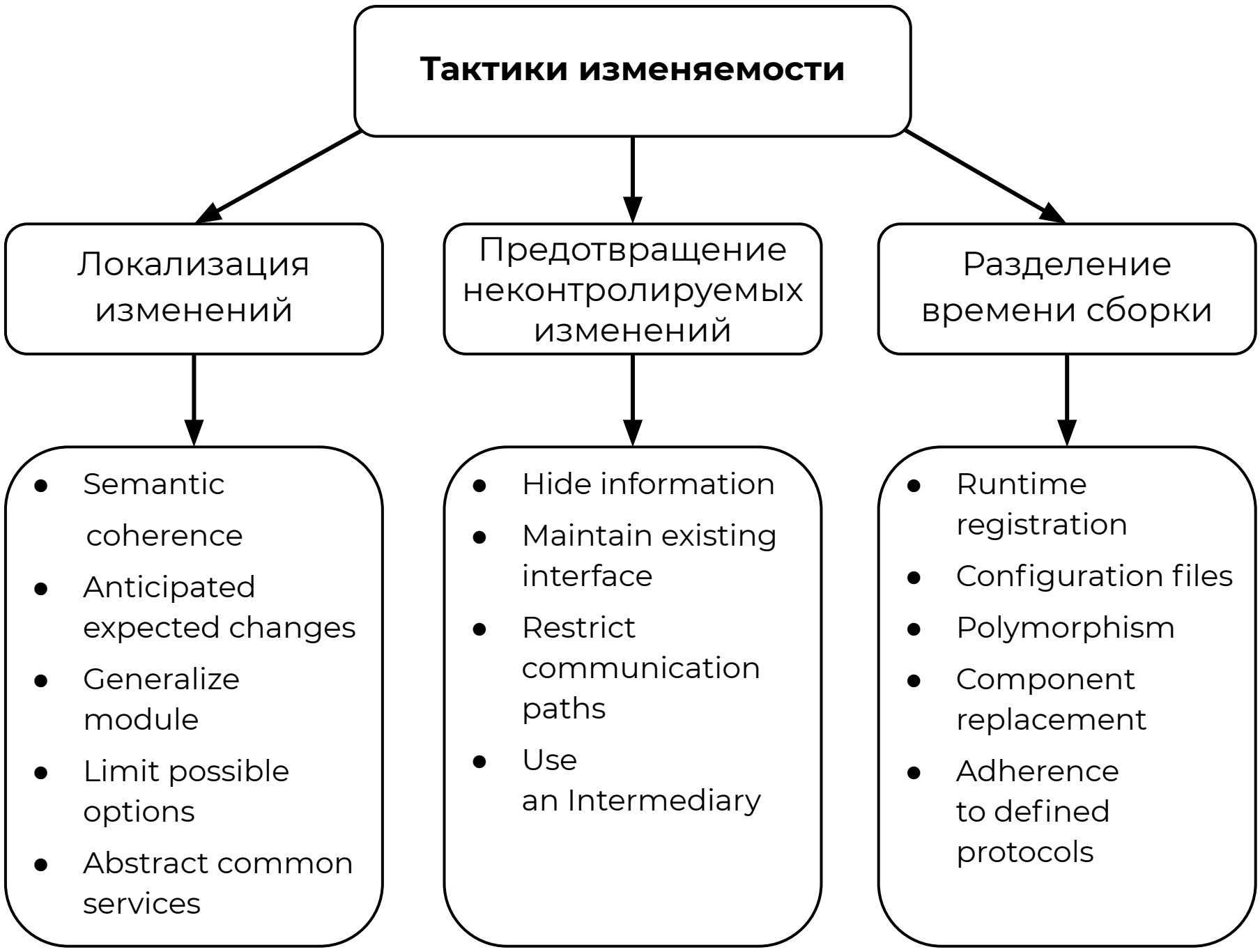
- Поддерживает ли резервное копирование в режиме онлайн с приемлемым снижением производительности?
- Установлены ли механизмы перехода от основного к резервному сайту?
- Оценивается ли архитектура на предмет наличия узких мест, отдельных точек отказа и других недостатков?
- Распространяется ли модель отказоустойчивости на все уязвимые объекты в ландшафте системы?
- Определена ли процедура резервного копирования?

Изменяемость

Изменяемость — это стоимость изменений и лёгкость, с которой проектируемая система может аккумулировать изменения.

Изменяемость

- Средняя стоимость изменения внутри одной области ответственности
- Связанность компонентов — coupling
- Сплочённость компонентов — cohesion
- Время внесения изменений в жизни приложения


Тактики изменяемости
Локализация изменений
Предотвращение неконтролируемых изменений
Разделение времени сборки
Semantic coherence
Anticipated expected changes
Generalize module
Limit possible options
Abstract common services
Hide information
Maintain existing interface
Restrict communication paths
Use an Intermediary
Runtime registration
Configuration files
Polymorphism
Component replacement
Adherence to defined protocols

Изменяемость

	Modifiability									
	Increase Cohesion		Reduce coupling					Defer Binding Time		
Pattern	**Maintain Semantic Coherence**	**Abstract Common Services**	**Use Encapsulation**	**Use a Wrapper**	**Restrict Comm. Paths**	**Use an Intermediary**	**Raise the Abstraction Level**	**Use Runtime Registration**	**Use Start-Up Time Binding**	**Use Runtime Binding**
Layers	X	X	X		X	X	X			
Pipe-and-Filter	X		X		X	X			X	
Blackboard	X	X			X	X	X	X		X
Broker	X	X	X		X	X	X	X		
Model-View-Controller	X		X			X				X
Presentation-Abstraction-Control	X		X			X	X			
Microkernel	X	X	X		X	X				
Reflection	X		X							

Изменяемость

- Насколько легко будет ввести новые типы данных?
- Как будут вводиться или потребляться новые интеграции?
- Отслеживаемость операций в распределённом окружении

Вопросы для самопроверки

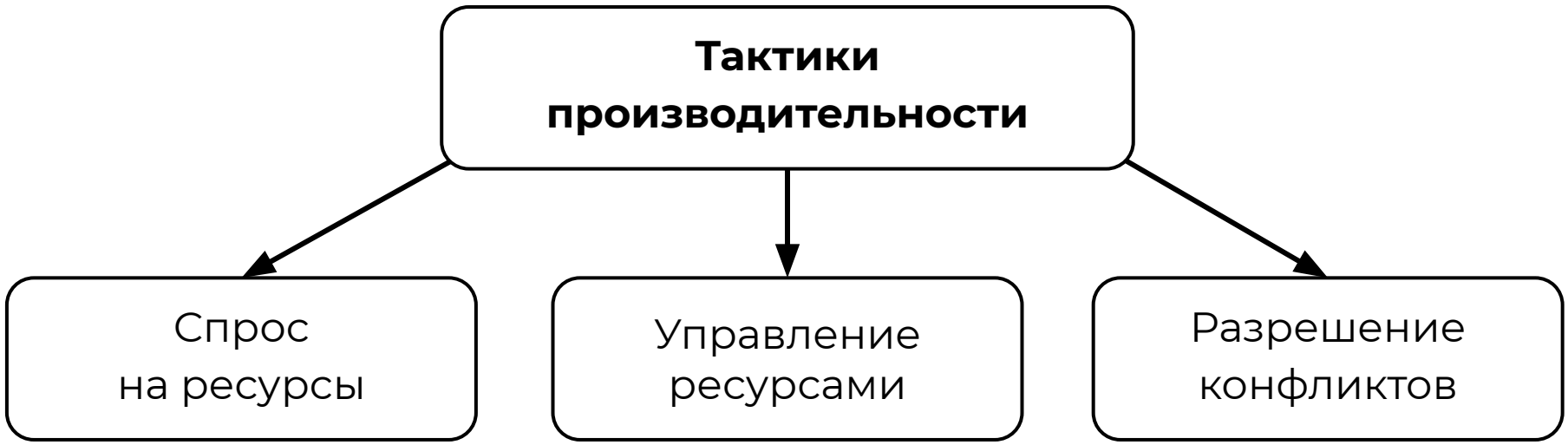
- Сможет ли операционная команда контролировать ландшафт приложений в среде исполнения?
- Определены ли в архитектуре точки расширения?
- Отделены ли операции ввода/вывода данных от обработки данных?
- Разделены ли процессы обработки данных на подпроцессы с точками сохранения?
- Предоставляют ли операции обработки данных информацию о прогрессе?

Skillbox

Производительность

Производительность определяется как отзывчивость приложения на выполнение конкретных задач в заданный промежуток времени.

- **Пропускная способность** — это количество событий в заданном промежутке времени
- **Задержка** — это время, необходимое для реакции на событие

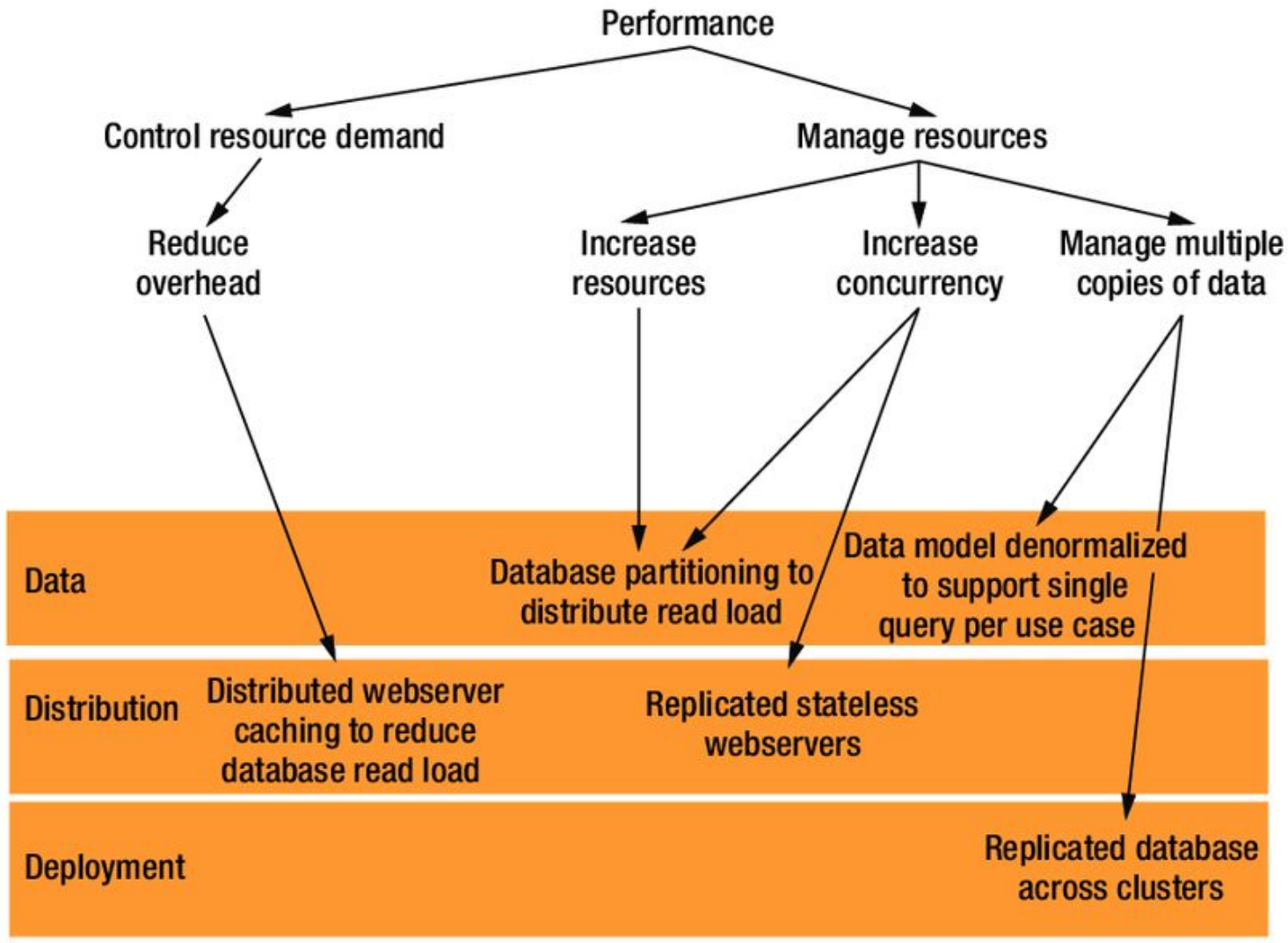

Тактики производительности
Спрос на ресурсы
Управление ресурсами
Разрешение конфликтов


Тактики производительности
Спрос на ресурсы
Управление ресурсами
Разрешение конфликтов
Increase computation efficiency
Reduce computation efficiency
Manage event rate
Control frequency of sampling

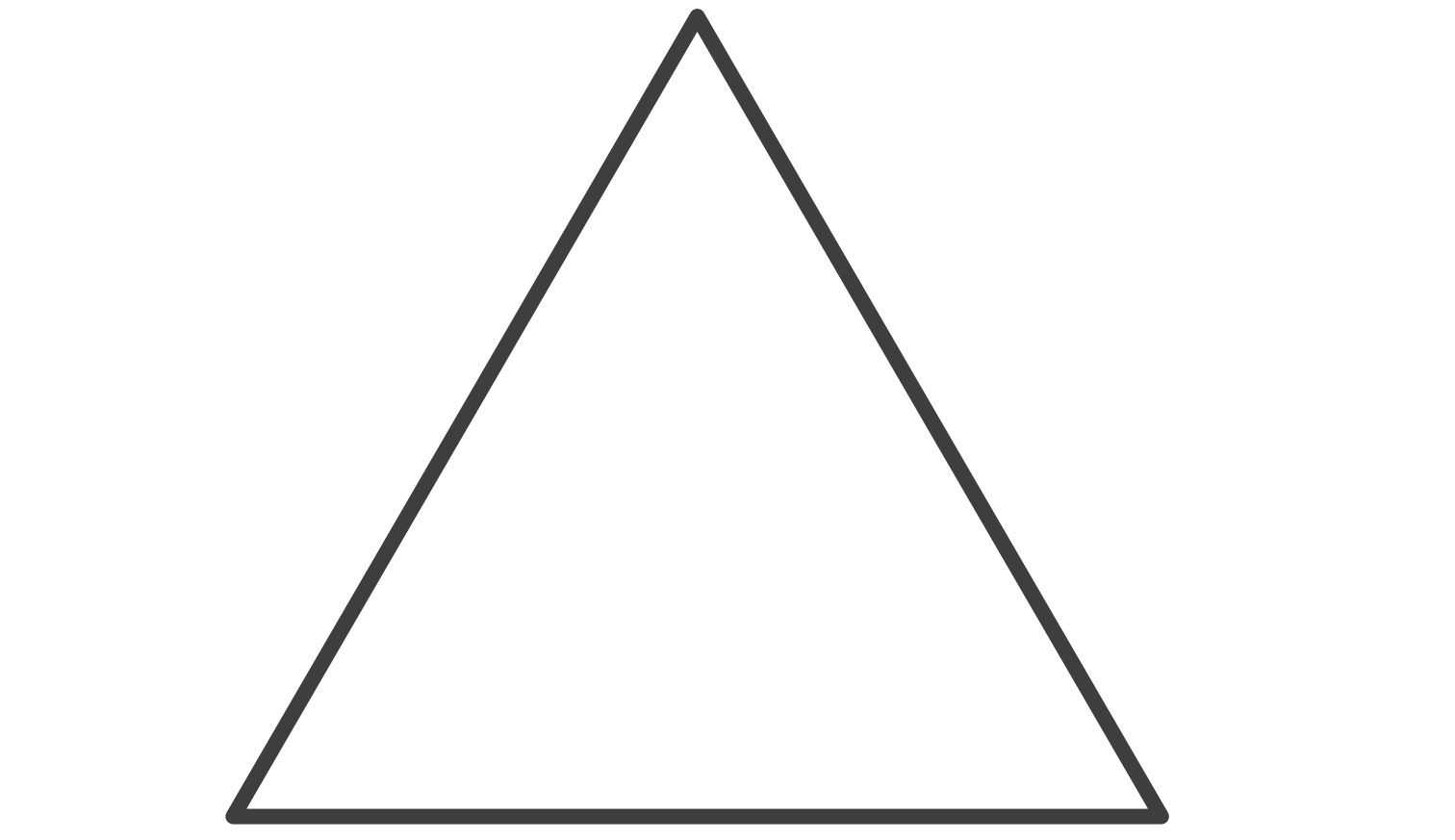

Тактики производительности
Спрос на ресурсы
Управление ресурсами
Разрешение конфликтов
- Increase computation efficiency
- Reduce computation efficiency
- Manage event rate
- Control frequency of sampling
- Introduce concurrency
- Maintain multiple copies
- Increase available resources


Тактики производительности
Спрос на ресурсы
Управление ресурсами
Разрешение конфликтов
Increase computation efficiency
Reduce computation efficiency
Manage event rate
Control frequency of sampling
Introduce concurrency
Maintain multiple copies
Increase available resources
Scheduling policy
Synchronization policy

Производительность



Компромиссы



Вопросы для самопроверки

- Определены и подтверждены ли допущения, связанные с производительностью?
- Как ошибки и отказы влияют на производительность?
- Насколько детально было произведено тестирование и анализ сторонних компонентов, чтобы понять вероятные последствия для производительности системы?
- Обозначены ли потенциальные проблемы с производительностью в архитектуре?
- Знаете ли вы, в какой степени предлагаемая архитектура может масштабироваться без значительных изменений?
- Какие рабочие нагрузки приоритетны?

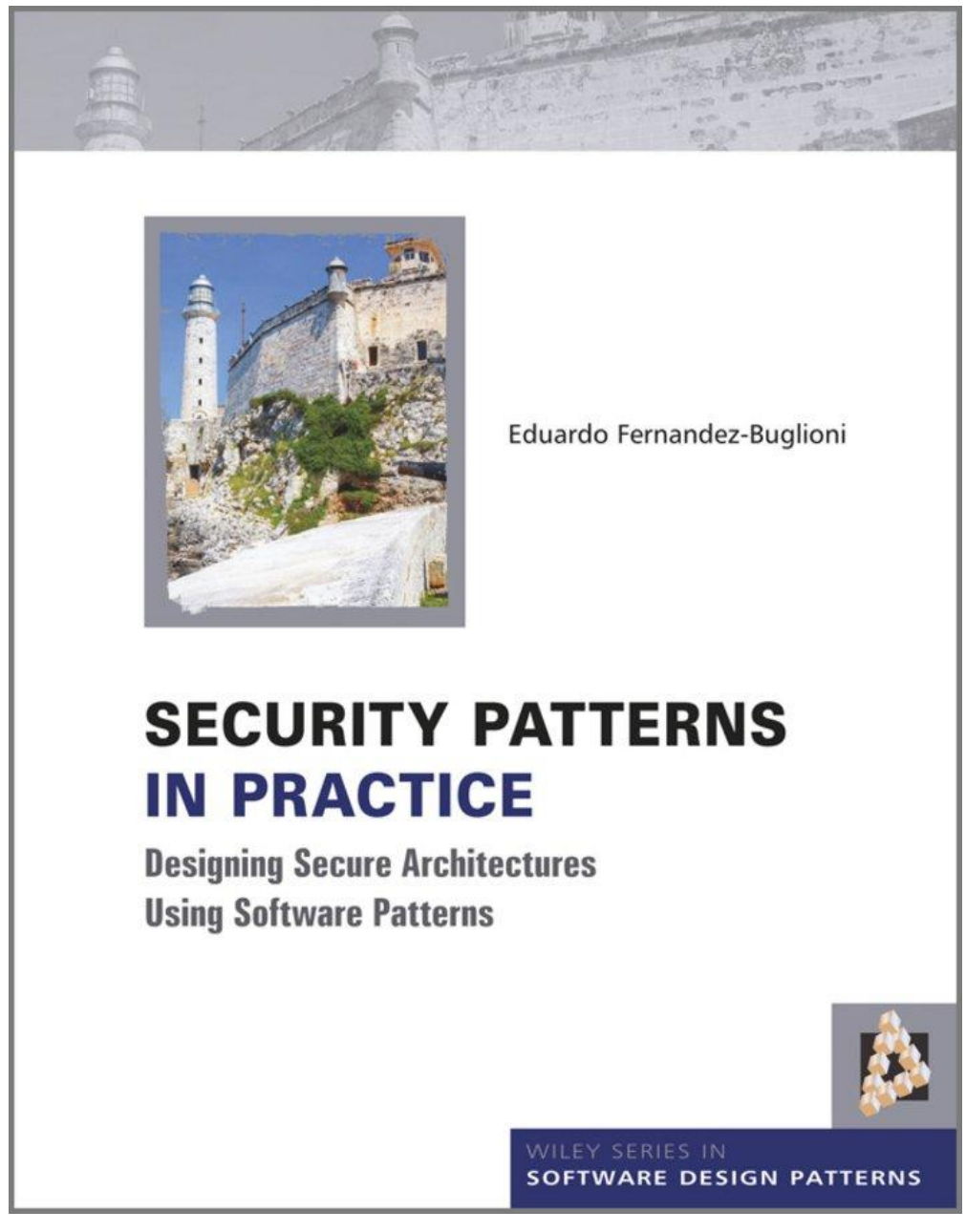
Skillbox

Безопасность

Безопасность — это возможность избежать вредоносных событий и инцидентов при проектировании использования системы, а также предотвратить потерю информации.

Skillbox

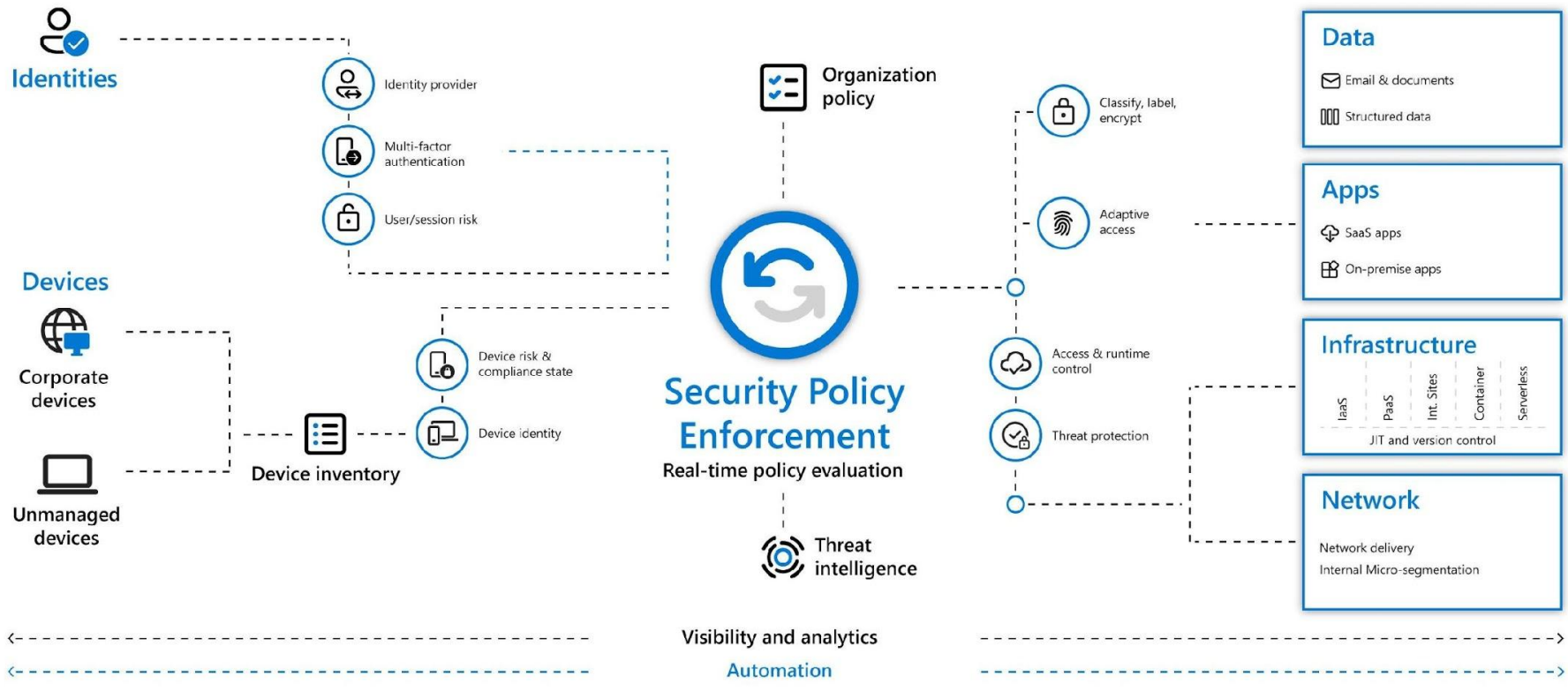
Безопасность




Тактики безопасности
Определение атак
Verify message integrity
Verify storage integrity
Maintain audit trail
Identify intrusion
Остановка и смягчение атак
Authenticate subjects
Authorize subjects
Detect origin of messages
Establish security channel
Manage security data
Hide data
Реакция на атаки
Alert subjects
Apply institutions policies
Восстановление после атак
Audit actions
Apply institutions policies

Zero tolerance

Модель «Никому не доверяй» основана на предположении, что сеть в пределах корпоративного брандмауэра небезопасна, а все запросы рассматриваются так, как если бы они поступали из открытой сети.



Компромиссы

Skillbox

OWASP-10

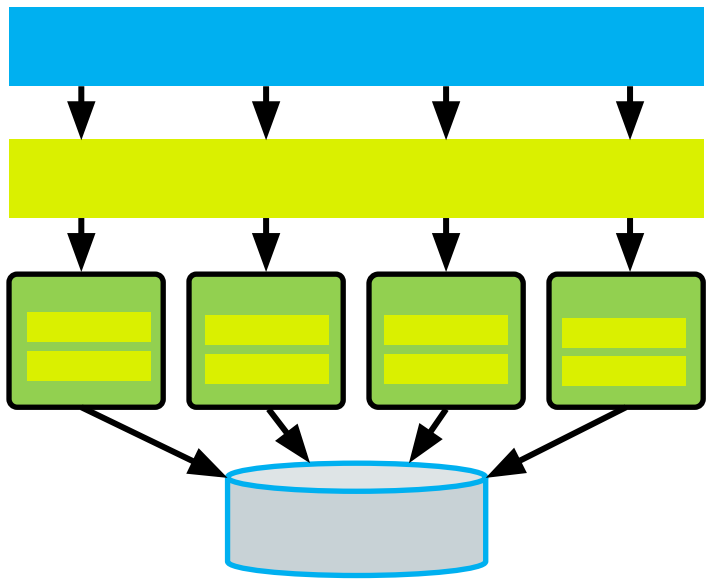
1. Инъекции кода
2. Ошибки настройки авторизации
3. Раскрытие чувствительных данных
4. XML External Entities
5. Ошибки настройки прав доступа
6. Общие ошибки настройки безопасности
7. Cross-Site Scripting
8. Небезопасная десериализация данных
9. Использование компонентов с известными уязвимостями
10. Недостаточное логирование и мониторинг

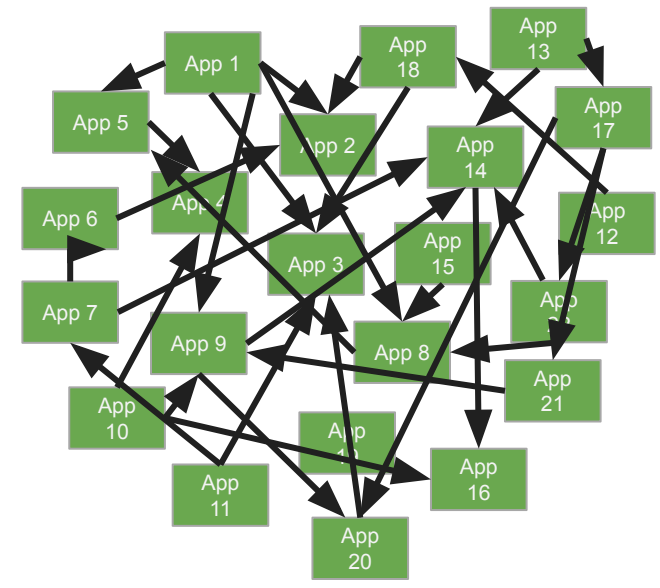
Безопасность

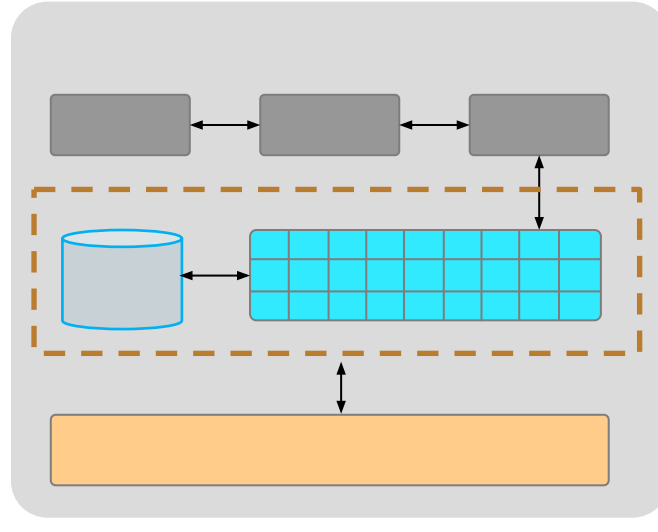
- Аутентификация: правильная идентификация сторон, пытающихся получить доступ к системам, и защита систем от несанкционированного доступа
- Авторизация: механизм, необходимый для предоставления пользователям полномочий на выполнение различных функций в рамках систем
- Шифрование (данных в хранилище и при передаче): все внешние связи между сервером данных и клиентами должны быть зашифрованы
- Конфиденциальность данных: все данные должны иметь категорию конфиденциальности, и соответствующие средства защиты должны быть применены
- Соответствие стандартам: процесс подтверждения соответствия систем стандартам и политикам безопасности организации

Skillbox

Тестируемость



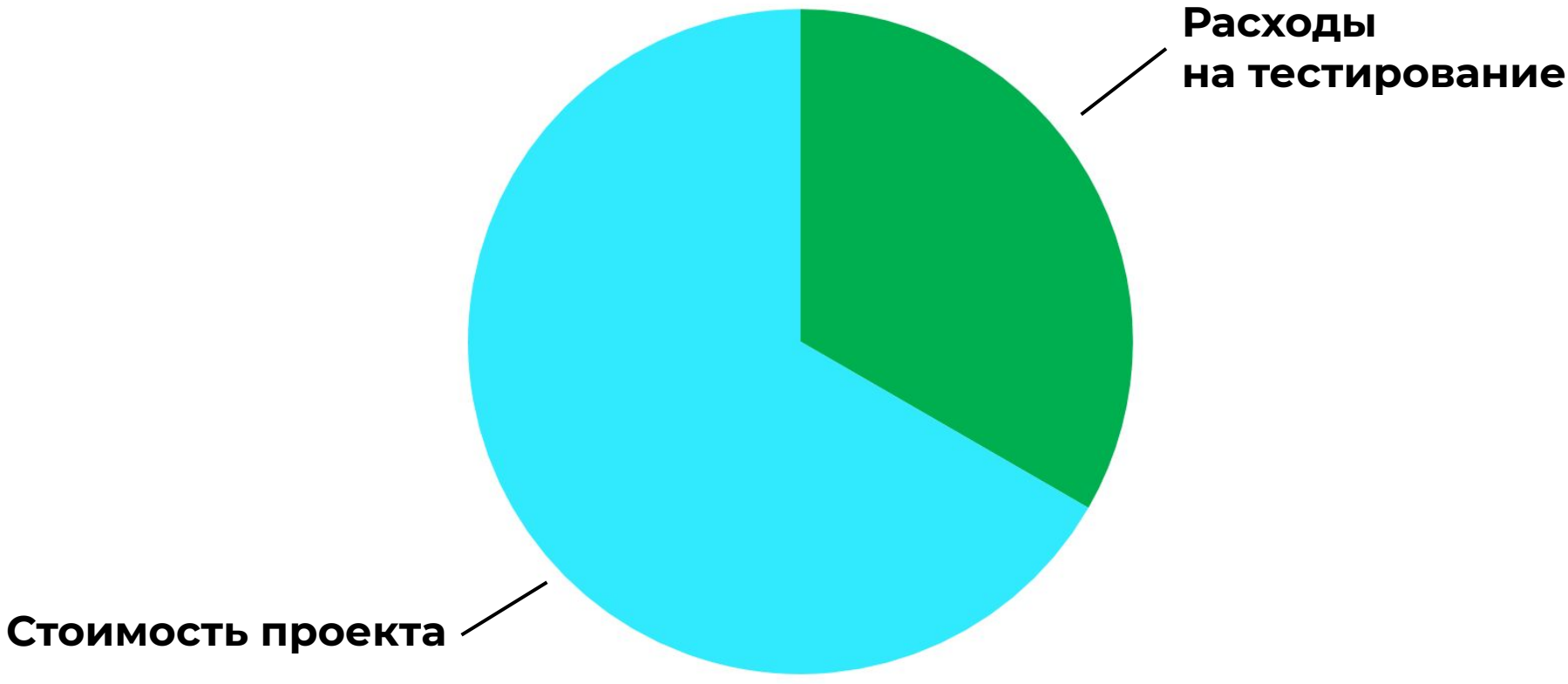




Skillbox

Тестируемость

Система тестируема, если она легко «отдаёт» свои ошибки.



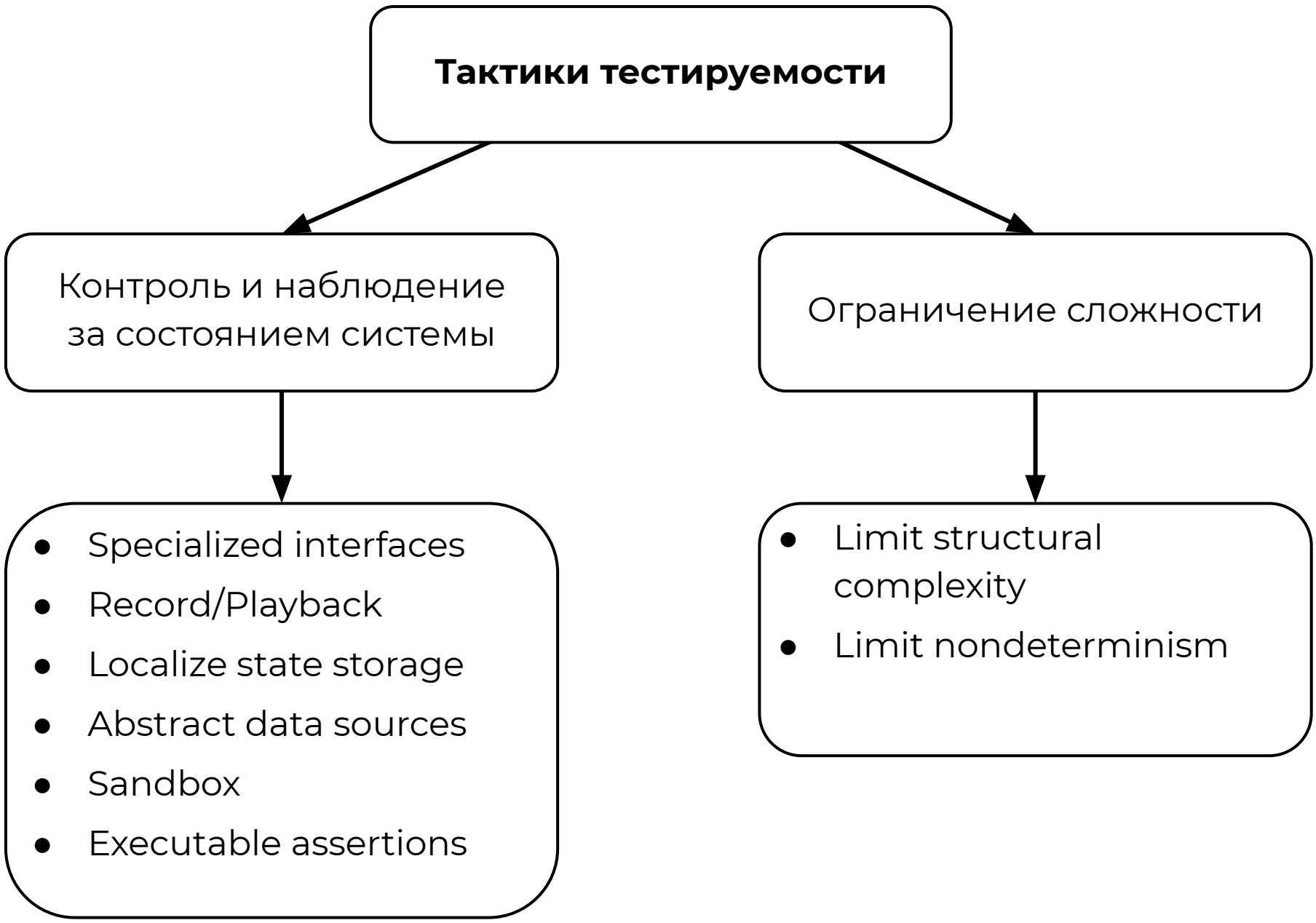
Тестируемость

Облегчают тестирование следующие аспекты:

- контроль над вводом данных
- низкая связанность
- низкая согласованность данных (eventual consistency)

Skillbox

Тестируемость


Тактики тестируемости
Контроль и наблюдение за состоянием системы
Specialized interfaces
Record/Playback
Localize state storage
Abstract data sources
Sandbox
Executable assertions
Ограничение сложности
Limit structural complexity
Limit nondeterminism

Безопасность

- Существуют ли операции, пересекающие границы доменов, и как определить успешность их завершения?
- Насколько легко запустить каждый отдельный модуль в изоляции для тестирования?
- Как сервисы сообщают о своей неработоспособности?
- Есть ли возможность определения нарушения согласованности данных в рамках одного хранилища?
- Позволяет ли API/интерфейс написание автоматизированных тестов и обращение к отдельным компонентам независимо и однозначно?

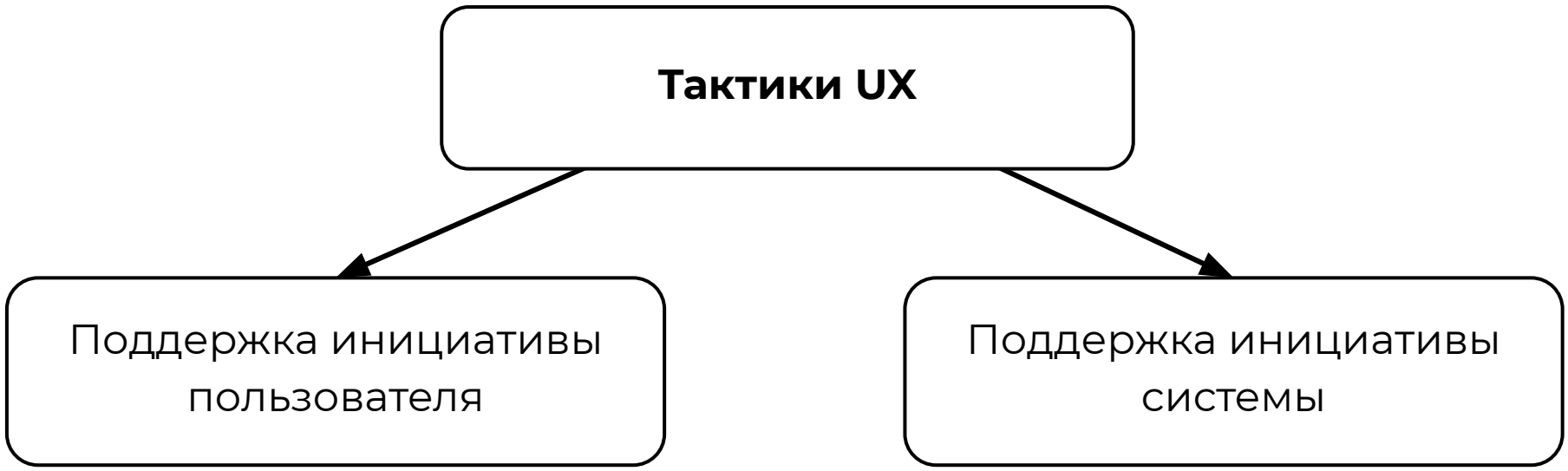
Удобство использования

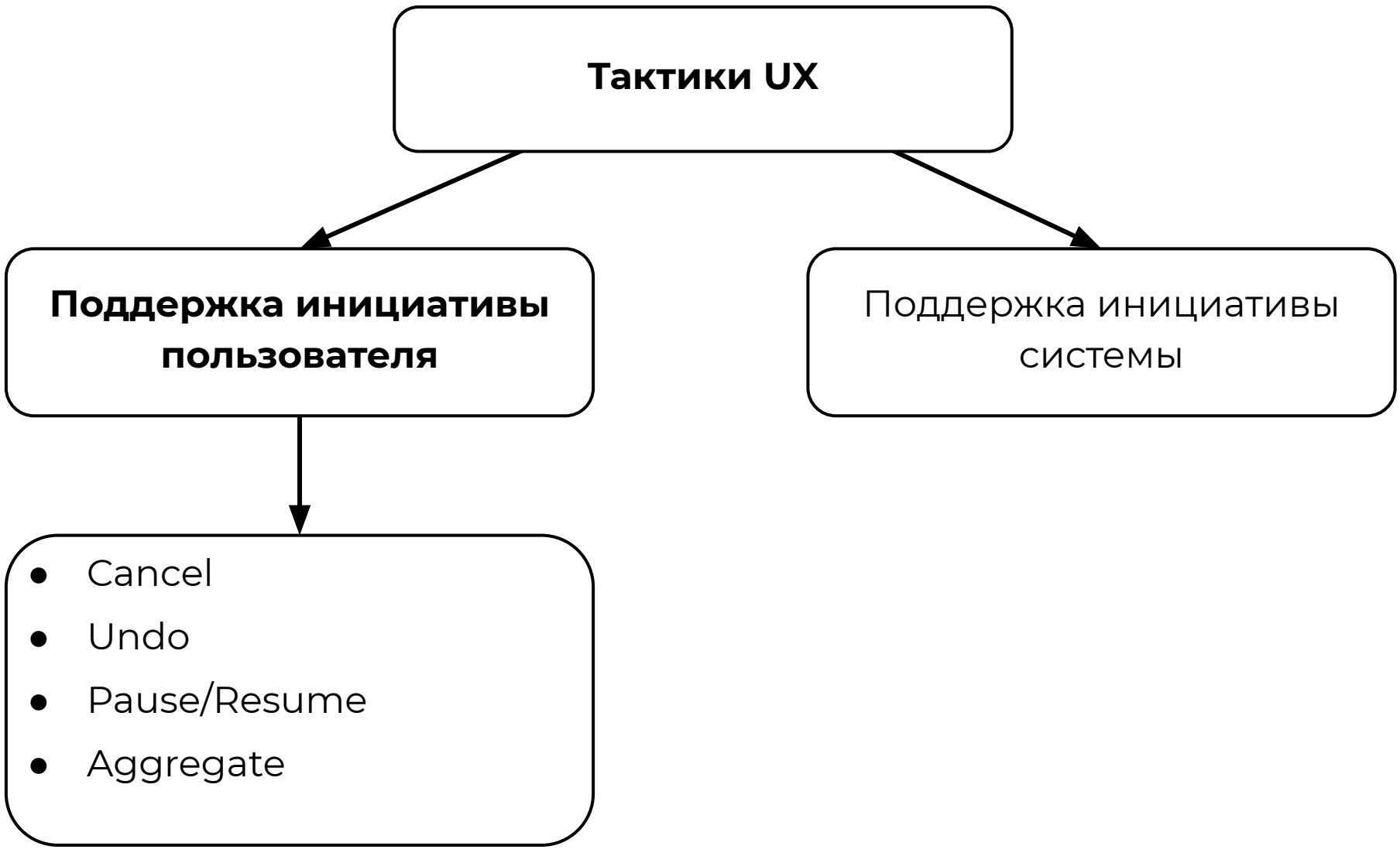
Юзабилити заботится о том, насколько легко пользователю выполнить желаемую задачу, а также о том, какую пользовательскую поддержку обеспечивает система.

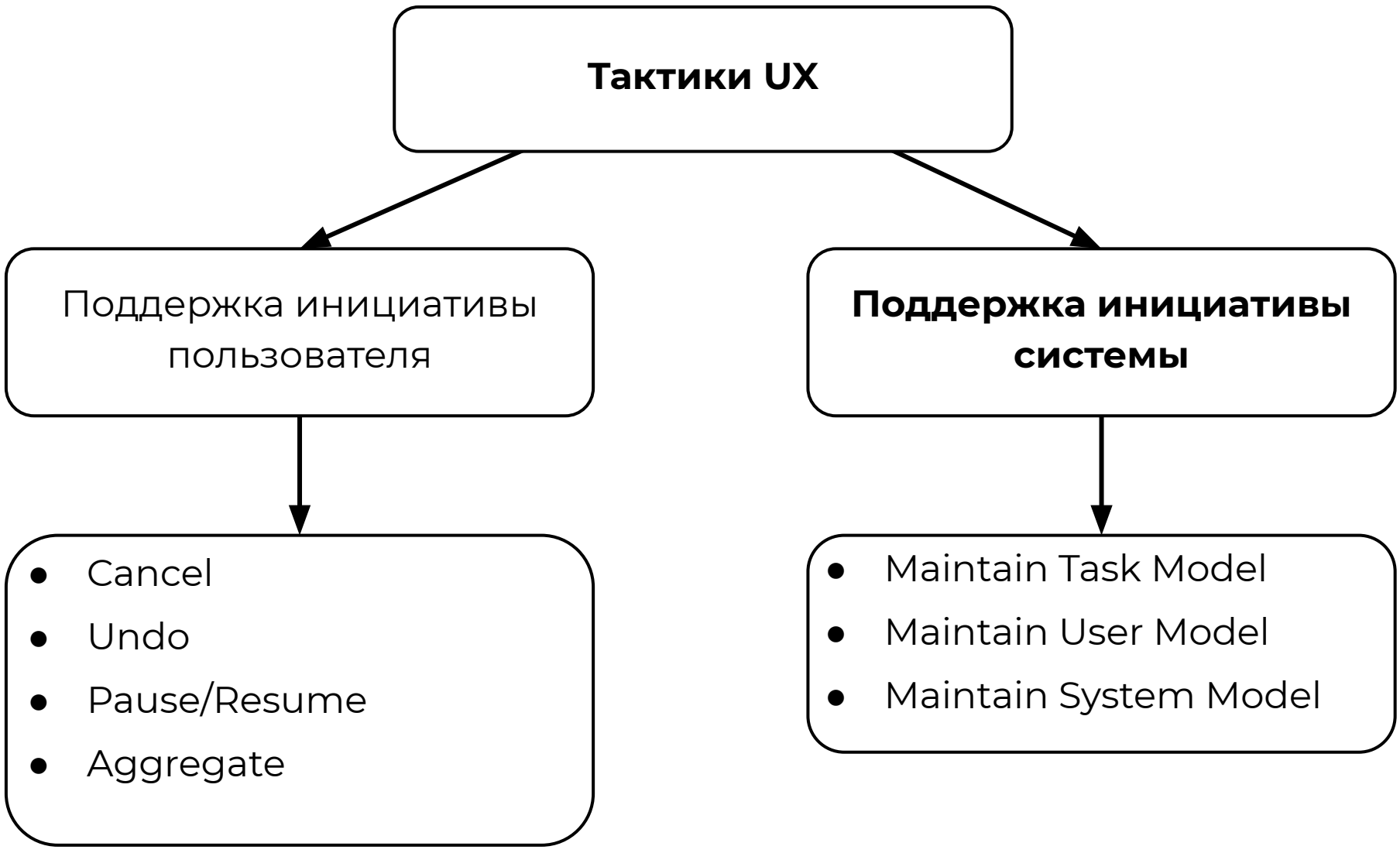
Skillbox

Удобство использования

- Изучение возможностей ПО
- Эффективное использование системы
- Минимизация влияния ошибок
- Адаптация системы под нужды пользователя
- Повышение уверенности и удовлетворённости


Тактики UX
Поддержка инициативы пользователя
Поддержка инициативы системы


Тактики UX
Поддержка инициативы пользователя
Поддержка инициативы системы
Cancel
Undo
Pause/Resume
Aggregate


Тактики UX
Поддержка инициативы пользователя
Поддержка инициативы системы
- Cancel
- Undo
- Pause/Resume
- Aggregate
- Maintain Task Model
- Maintain User Model
- Maintain System Model

Удобство использования

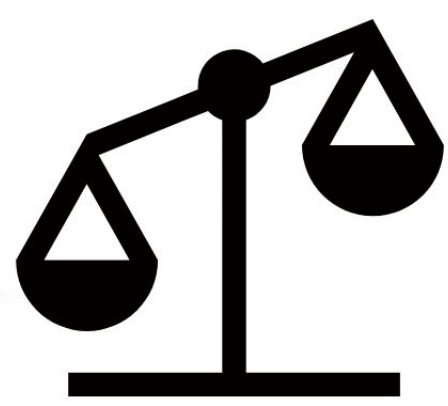
- Стандарты восприятия и отображения: расположение компонентов, плотность элементов на экране, присутствие привычных горячих клавиш, метафоры пользовательского интерфейса, характерные для системы, цвета, размер шрифтов
- Локализация: поддержка различных алфавитов, переводы, единицы измерения и так далее

Вопросы для самопроверки

- Учтена ли способность интерфейса минимизировать необходимость в ручной манипуляции. Например, информация уже содержится в системных таблицах, что приводит к появлению соответствующих выпадающих списков вместо того, чтобы вводить её?
- Есть ли возможность включить использование локальных настроек компьютера (например, локальные, региональные и настройки разрешения) в браузере?
- Одинакова ли поддержка нескольких браузеров: Edge, Chrome, Firefox, Safari?
- Доступна ли поддержка различных разрешений экрана для ПК: HDready, HD, 4K, Ultra Wide Screen, низкие разрешения экрана?
- Есть ли адаптивное представление для: ПК, мобильные телефоны, планшеты?

Skillbox

Компромиссы



Компромиссы

★ x 20

Доступность	★★★★★
Изменяемость	★★★★★
Скорость	★★★★★
Безопасность	★★☆☆☆
Тестируемость	★☆☆☆☆
Удобство использования	★★☆☆☆

Компромиссы

★ x 20

Доступность	★★★★★
Изменяемость	★★☆☆☆
Скорость	★★★☆☆
Безопасность	★★★☆☆
Тестируемость	★★★★★
Удобство использования	★★☆☆☆

Самое важное

Конфигурация атрибутов качества уникальна для каждой системы.

Выводы

- НФТ направлены на удовлетворение одного или нескольких атрибутов качества
- Атрибуты качества должны описываться с измерениями
- Выделенные атрибуты качества подсказывают, какие шаблоны и подходы использовать
- Усиление одного атрибута ведёт к ослаблению других

Skillbox

Что дальше?

Как использовать атрибуты качества для построения концептуальной архитектуры?

Skillbox

Спасибо за внимание!