

РАЗПРЕДЕЛЕНИ ПРИЛОЖЕНИЯ

Павел Кюркчиев
Ас. към ПУ „Паисий Хилендарски“
@rkyurkchiev

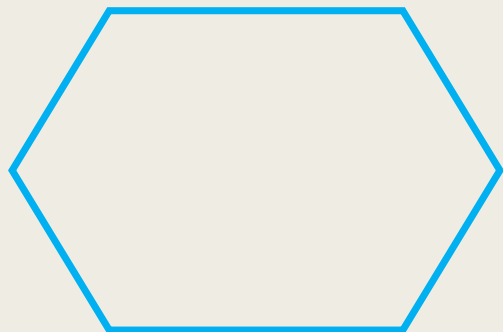
АРХИТЕКТУРА „БЕЗ СЪРВЪР“

Традиционна архитектура

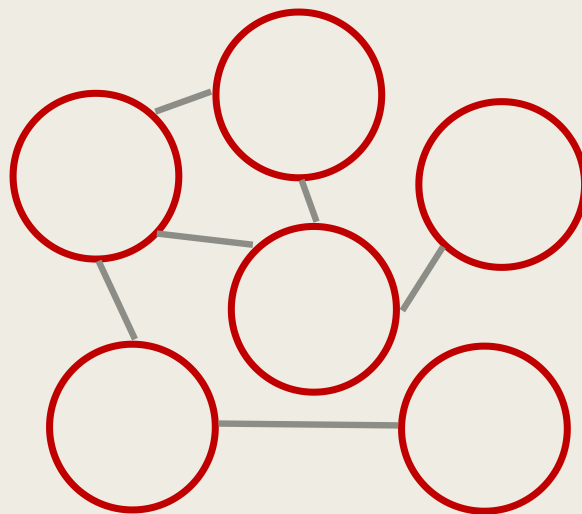
Традиционна архитектура 3 tier client-oriented



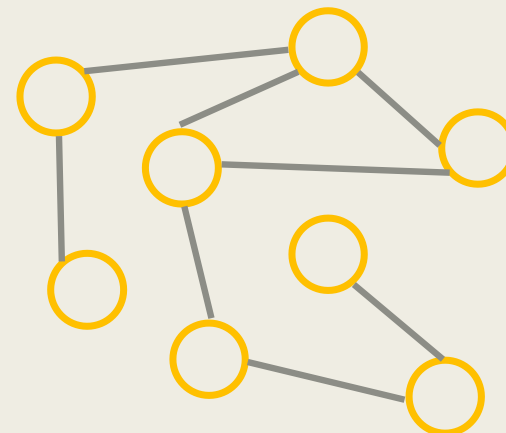
Еволюцията на бизнес логиката



Monolith



Microservices



Functions

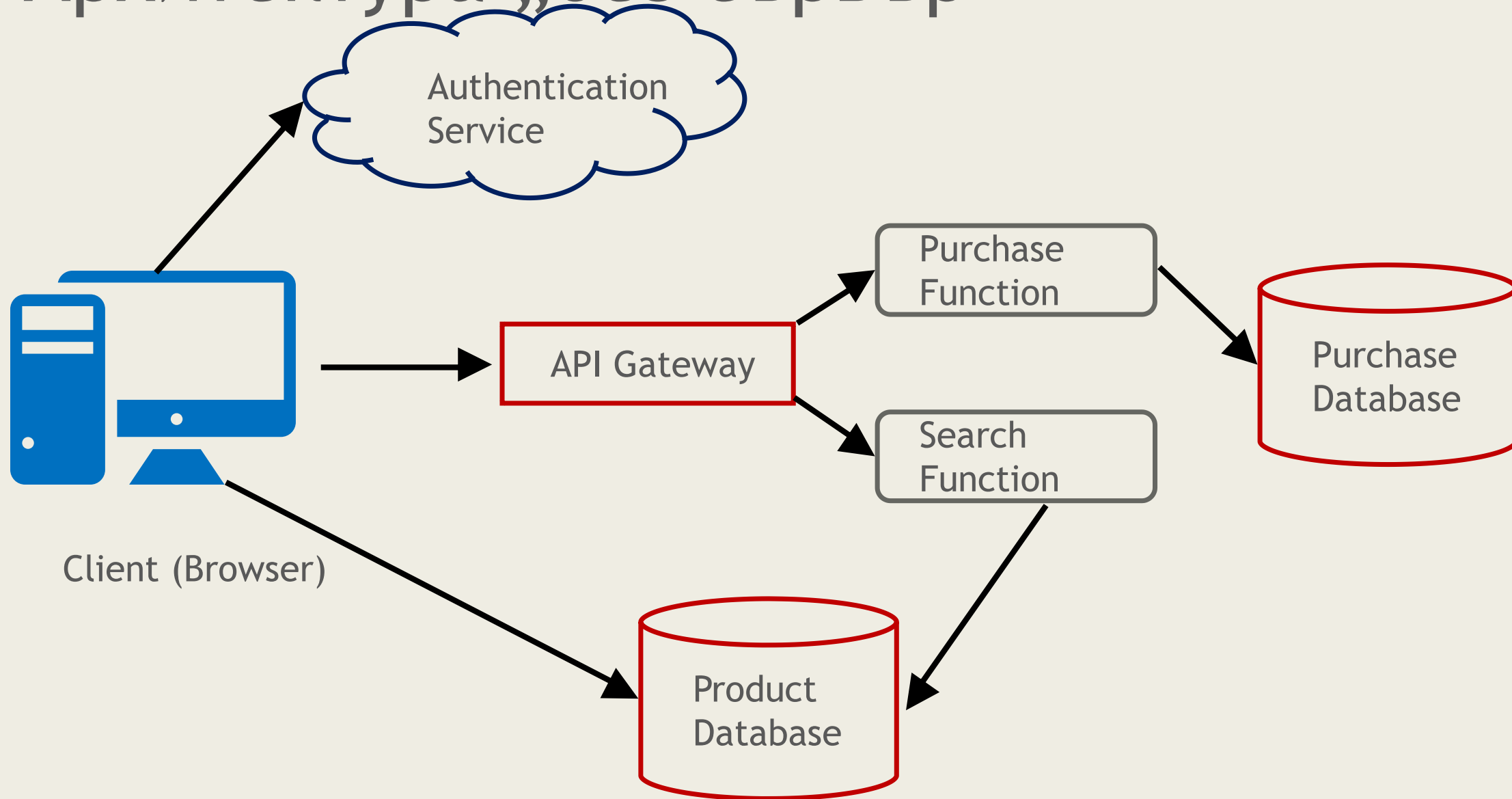
Какво е архитектура „без сървър“?

- „Без сървър“ архитектура наричаме приложенията, които в голямата си част или изцяло зависят от трета страна за обработката и обслужването на техните нужни. Този тип услуги е описван като BaaS ((Mobile) Backend as a Service).



- „Без сървър“ архитектура може да означава и приложения, на които голяма част от логиката е написана от програмисти, но е качена на отдалечени контейнери, които се извикват и изпълняват при определени събития. Тези контейнери са напълно поддържана от трета страна. За тях можем да мислим като за (Functions as a service / FaaS).

Архитектура „без сървър“



Примери за Functions

- Azure
 - *Azure Functions - C#, F#, Node.js, Java, or PHP*
- Amazon AWS
 - *Lambda Functions - Node.js, Java, C#, Go and Python*
- Google Firebase
 - *Firebase Functions - Node.js*

„Без сървър“ означава

- Без нужда от собствен сървър или от управлението му
- Плащаме само това, което използваме
- Разширение само според нашите потребности
- Достъпност и толерантност към средата

Регионална услуга



Анатомия на Serverless Functions

- Handler() function
 - *Функцията да бъде извикана при възникването на някакъв обект*
- Event object
 - *Изпращаната информация по време на извикване на функцията*
- Context object
 - *Достъпен метод отговарящ при изпълнението на функцията*

```
exports.myHandler = function(event, context, callback) {  
    console.log("value1 = " + event.key1);  
    console.log("value2 = " + event.key2);  
    callback(null, "some success message");  
}
```

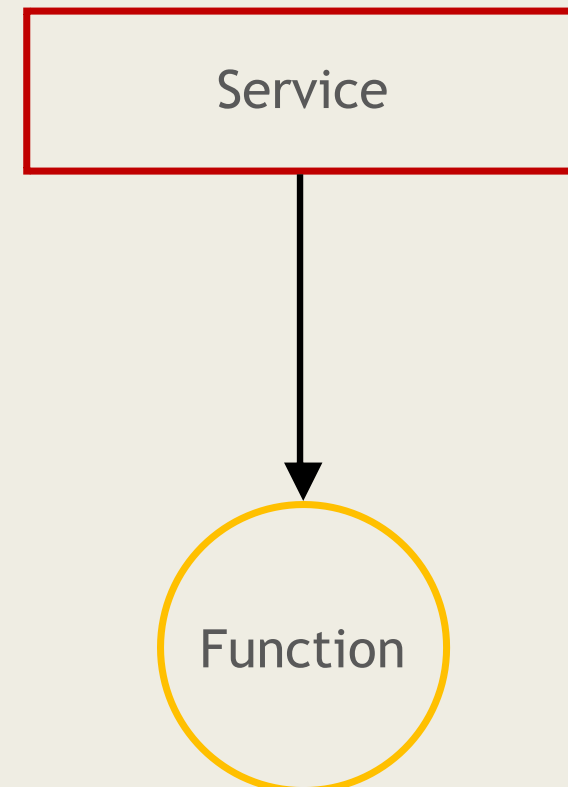
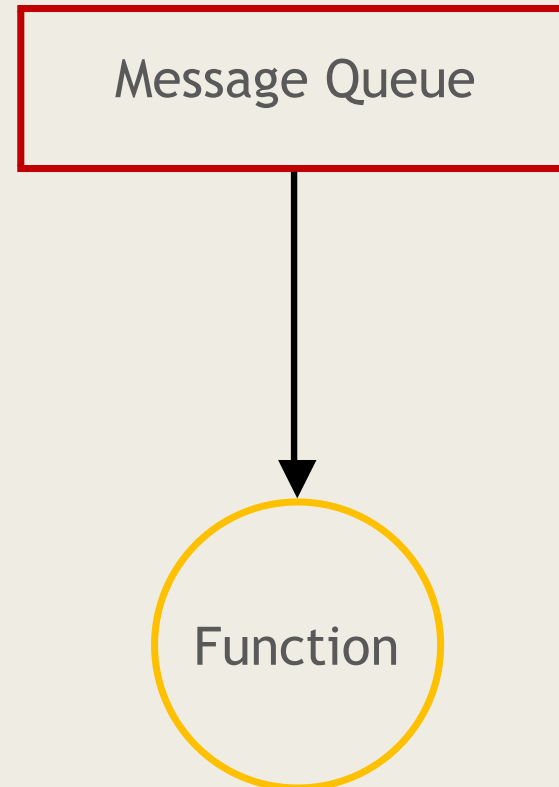
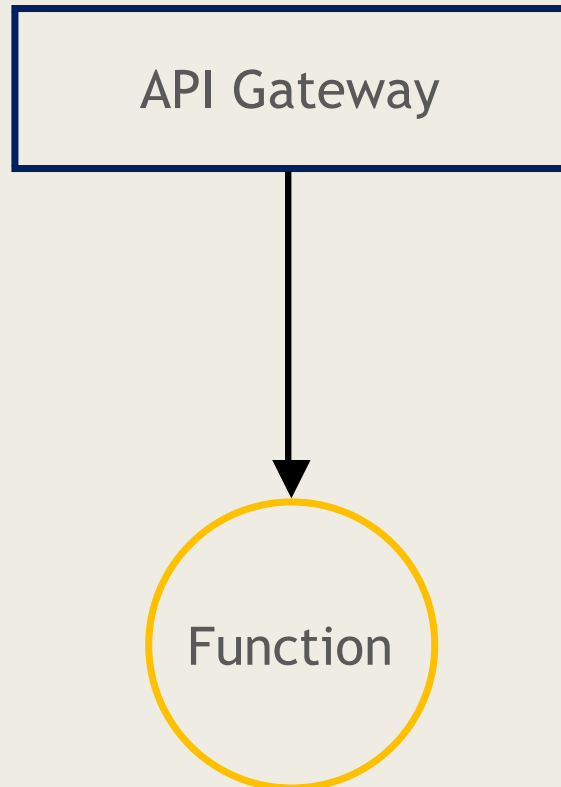
AWS Lambda function

Functions модел на изпълнение

Synchronous (push)

Asynchronous (event)

Stream-based



Добри практики Functions

- Да се минимизира големината на пакетите
- Да се разделя handler от основната логика на функцията
- Да се използват Environment Variables за модифициране поведението
- Да се възползваме от “Max Memory Used” за да определим правилно големината на function
- Да се премахнат големите неизползвани функции

DEMO FIREBASE FUNCTIONS

