

Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря Сікорського”  
Кафедра автоматизації проектування енергетичних процесів і систем

---

**Застосування  
мікросервісної  
архітектури для  
побудови відмовостійкої  
хмарної системи**

Виконав:  
Студент 6 курсу, ТВ-71мп  
Соломкін Максим

Дипломний керівник:  
Доц. каф. АПЕПС, к.т.н.  
Смаковський Д.С.

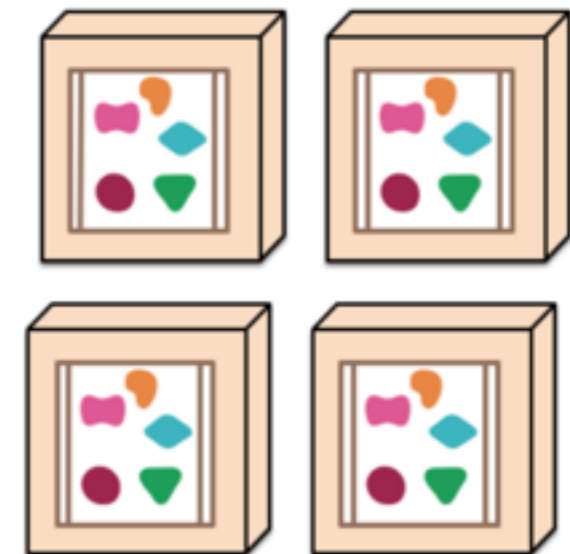
- ▶ Хмарна інфраструктура дуже популярна і широко використовується багатьма компаніями
- ▶ Хмарна інфраструктура дозволяє економити як на обслуговуванні і персоналі, так і на інфраструктурі
- ▶ Хмарна інфраструктура не надійна
- ▶ Системи повинні бути спроектовані для роботи у хмарному середовищі

- ▶ **Мета:**  
Підвищити надійність та стійкість розподілених хмарних систем.
- ▶ **Задачі:**  
Створити систему для вивчення нових слів, що спроектована для роботи у хмарній інфраструктурі та використовує переваги і мінімізує недоліки хмарного середовища.

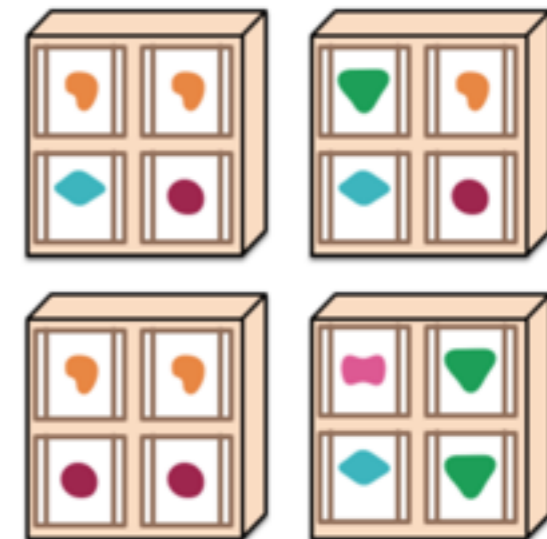
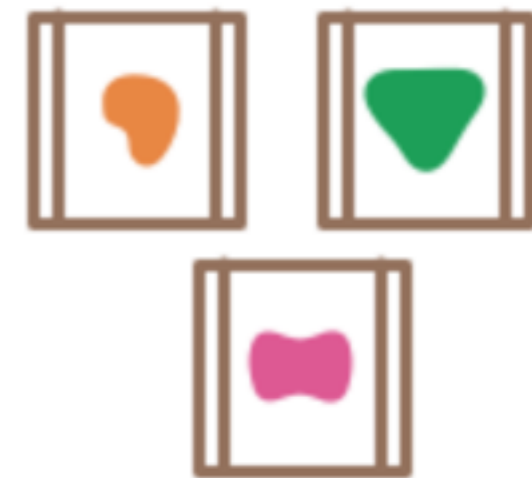
- ▶ єдиний шлюз для виконання всіх запитів
- ▶ реєстрація екземплярів сервісів у загальному реєстрі
- ▶ аналіз розподілених запитів та часу їх виконання
- ▶ автоматичне масштабування сервісів за рівнем використання CPU або оперативної пам'яті
- ▶ автоматичне масштабування сервісів за рівнем використання CPU або оперативної пам'яті або залежно від метрик користувача
- ▶ балансування навантаження
- ▶ режими для додавання та вивчення слів

- ▶ Розподілена за своєю природою
- ▶ Дозволяє масштабувати незалежні компоненти
- ▶ Сервіси швидко запускаються та виконують роботу

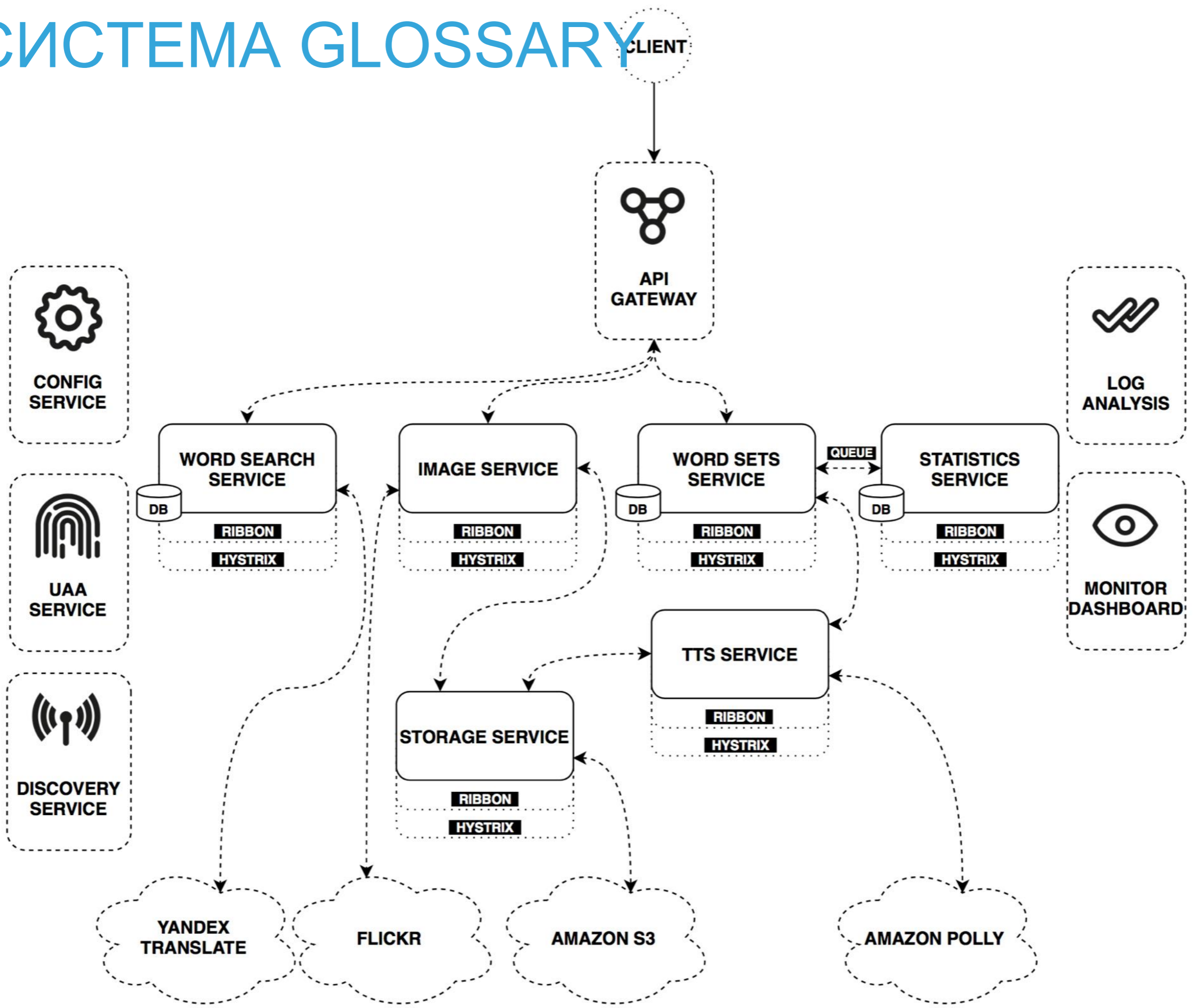
- ▶ Монолітна програма має всю свою функціональність у єдиному процесі
- ▶ Та масштабується через багатократне підняття всього застосунку на декількох серверах



- ▶ Мікросервісна система виділяє функціональні одиниці у окремі сервіси
- ▶ Та масштабується через багатократне підняття необхідних функцій на декількох серверах



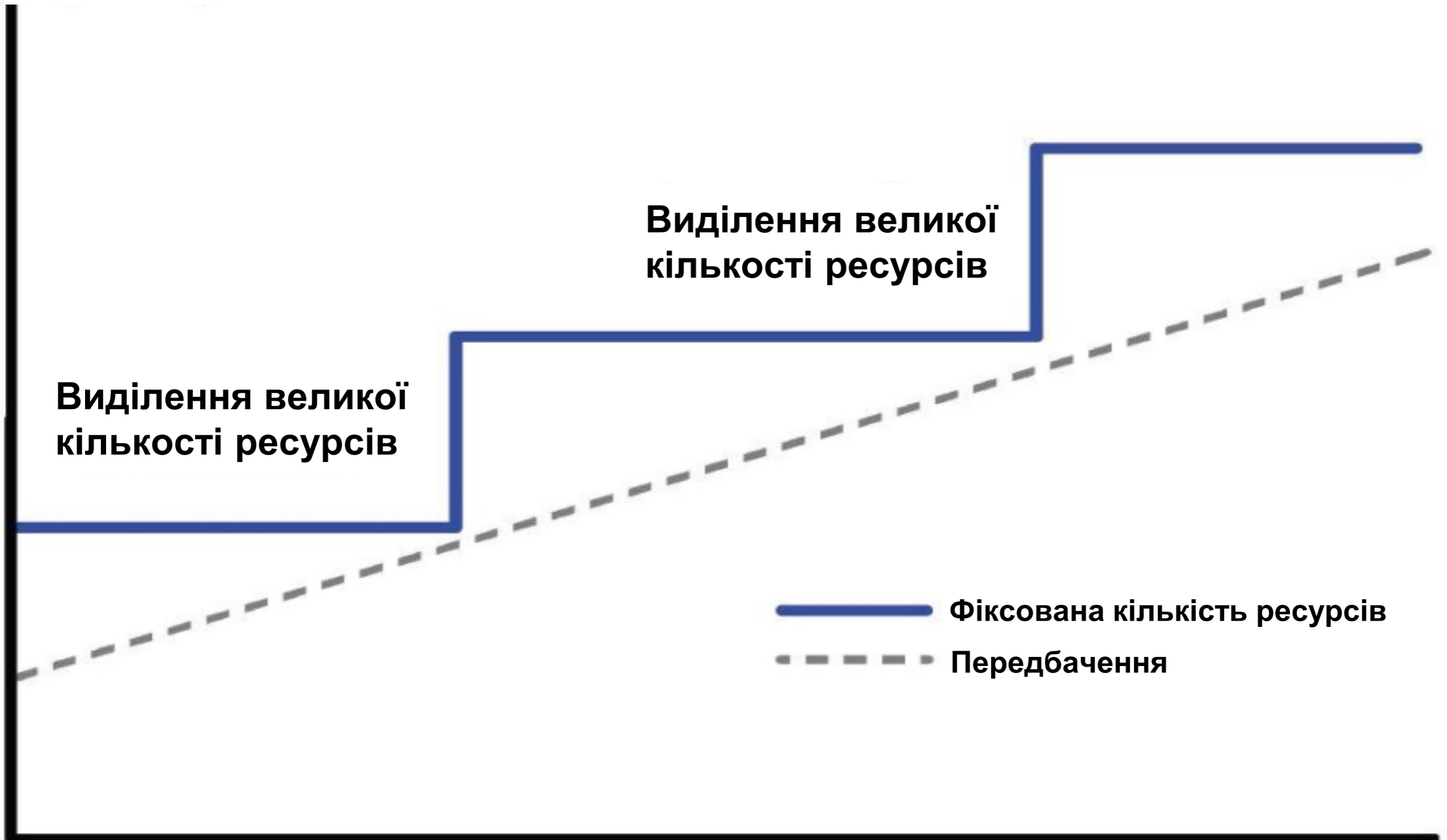
# СИСТЕМА GLOSSARY





# ВИДІЛЕННЯ РЕСУРСІВ ЗАВЧАСНО

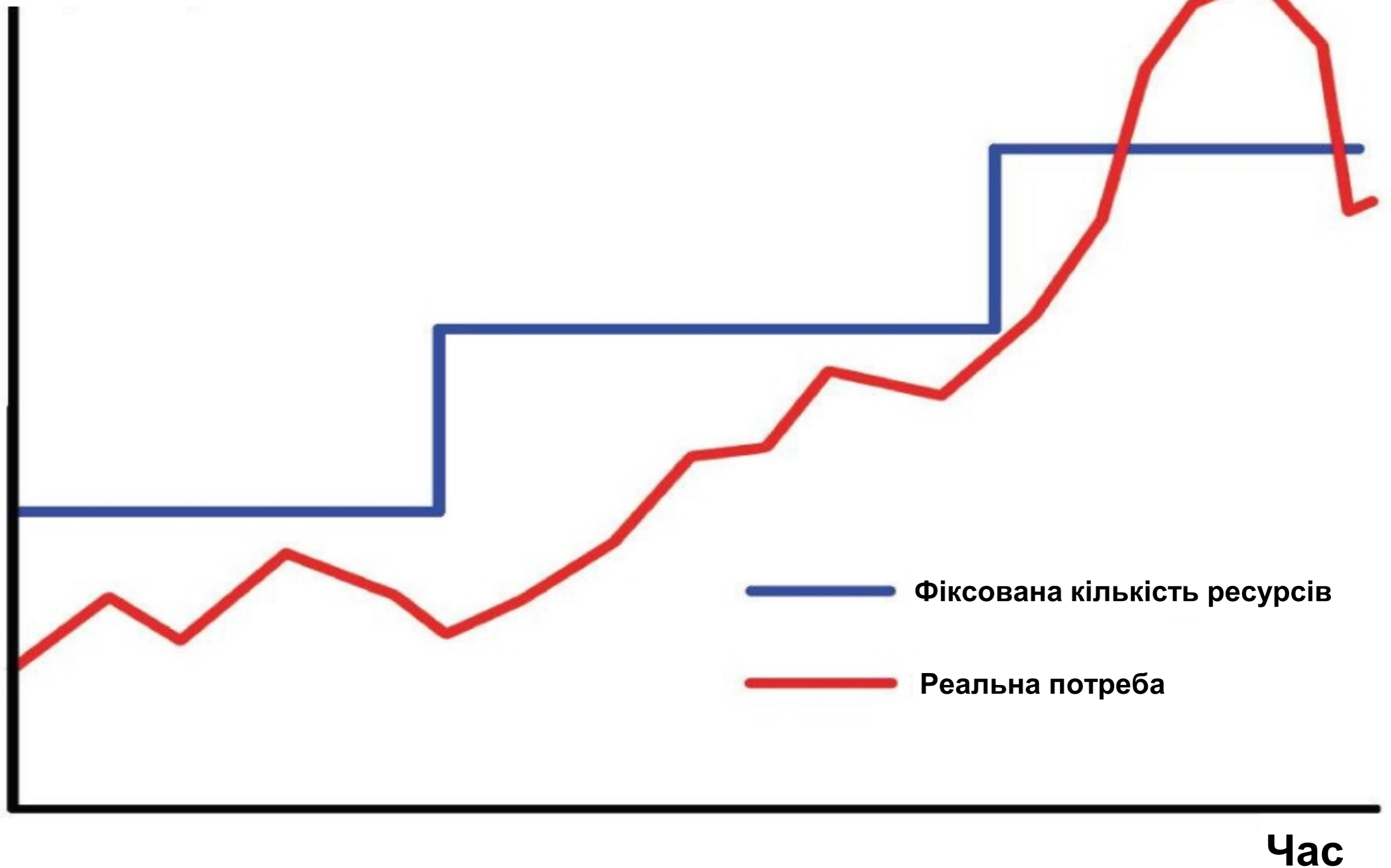
Ресурси



Час

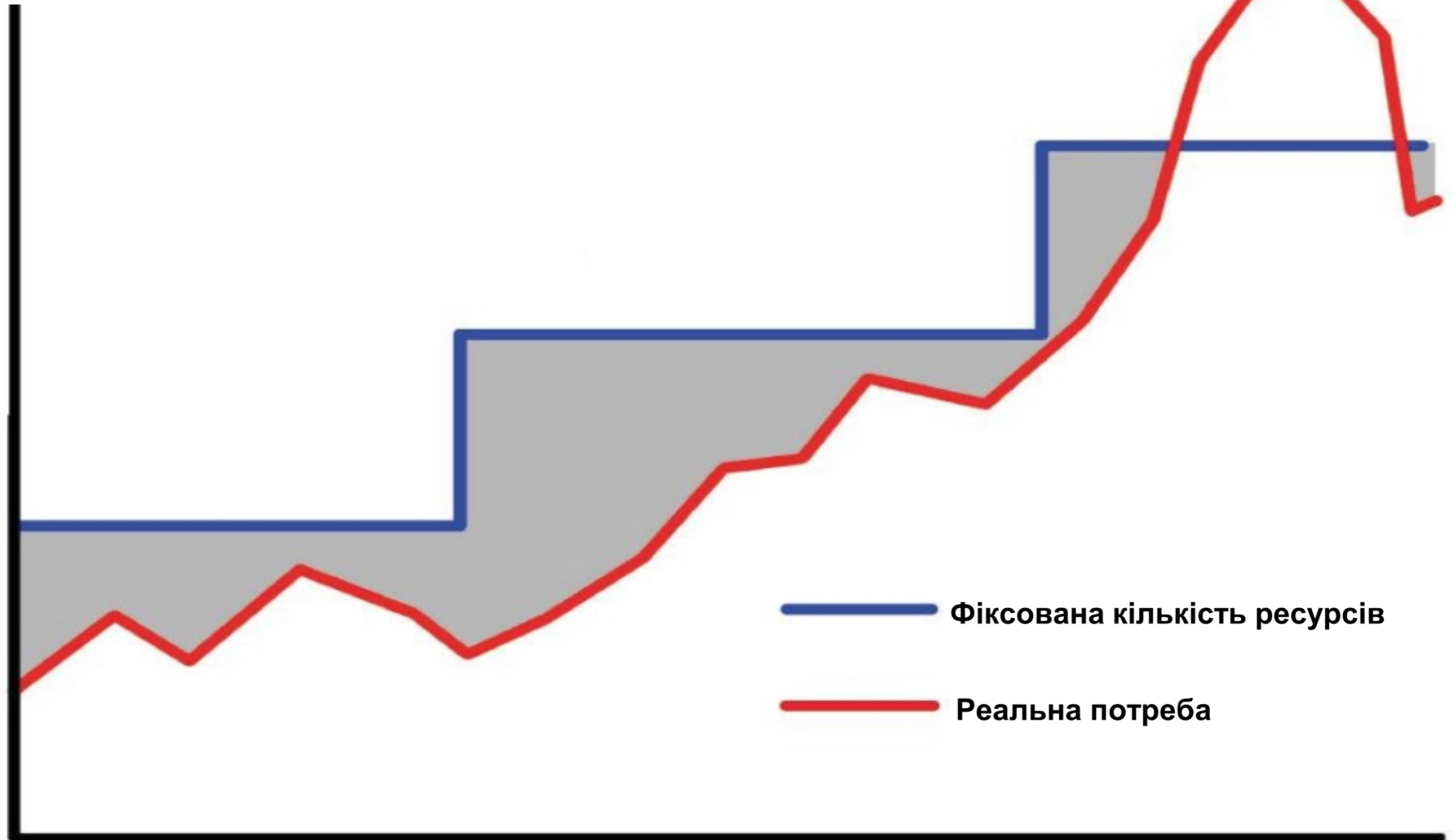
# НЕОБХІДНІ ТА ДОСТУПНІ РЕСУРСИ

Ресурси



# ПРОБЛЕМА 1: ДАРМА ВИТРАЧЕНІ РЕСУРСИ

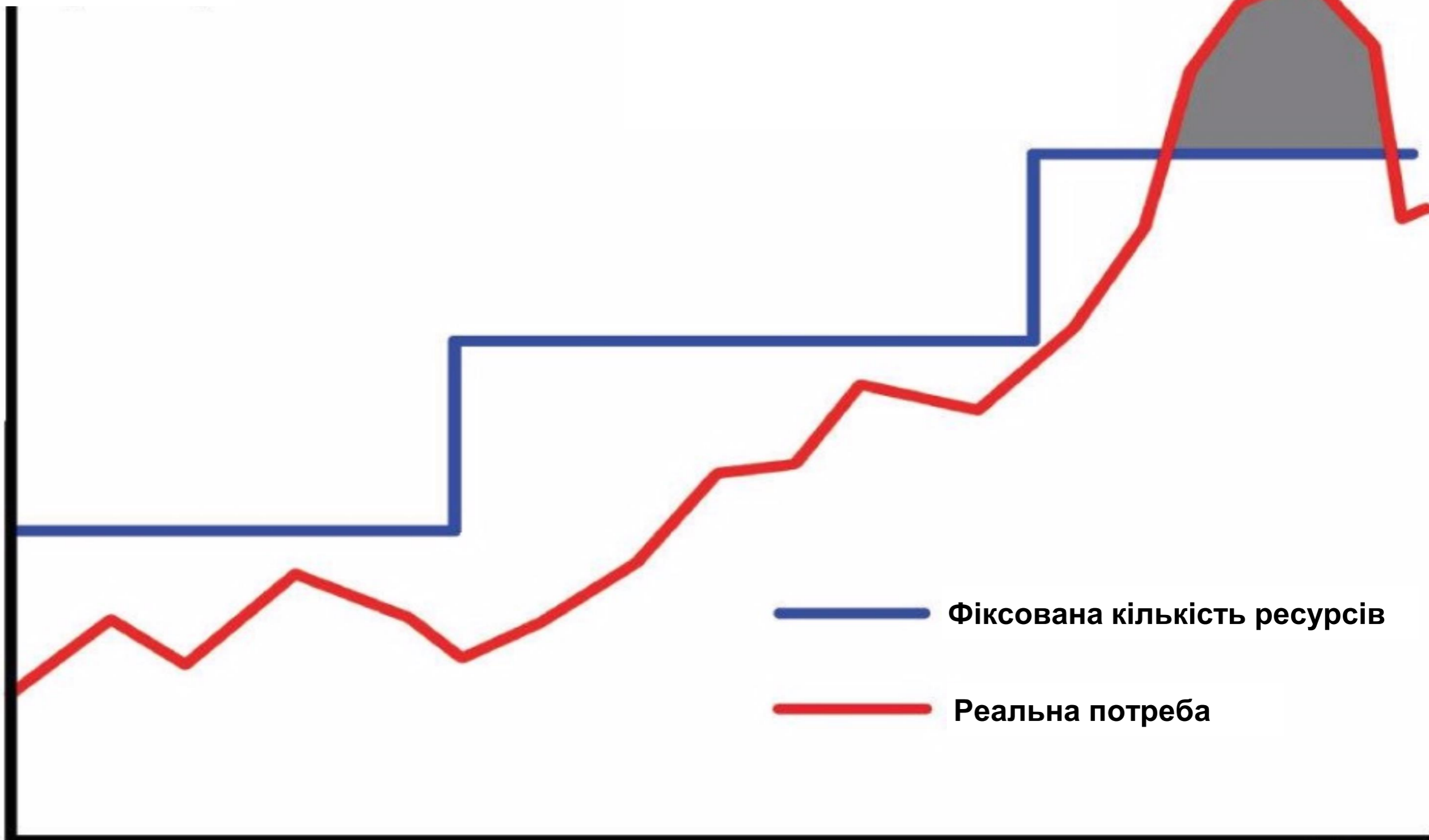
Ресурси



Час

# ПРОБЛЕМА 2: НЕСПОДІВАНІ ПІКИ НАВАНТАЖЕННЯ

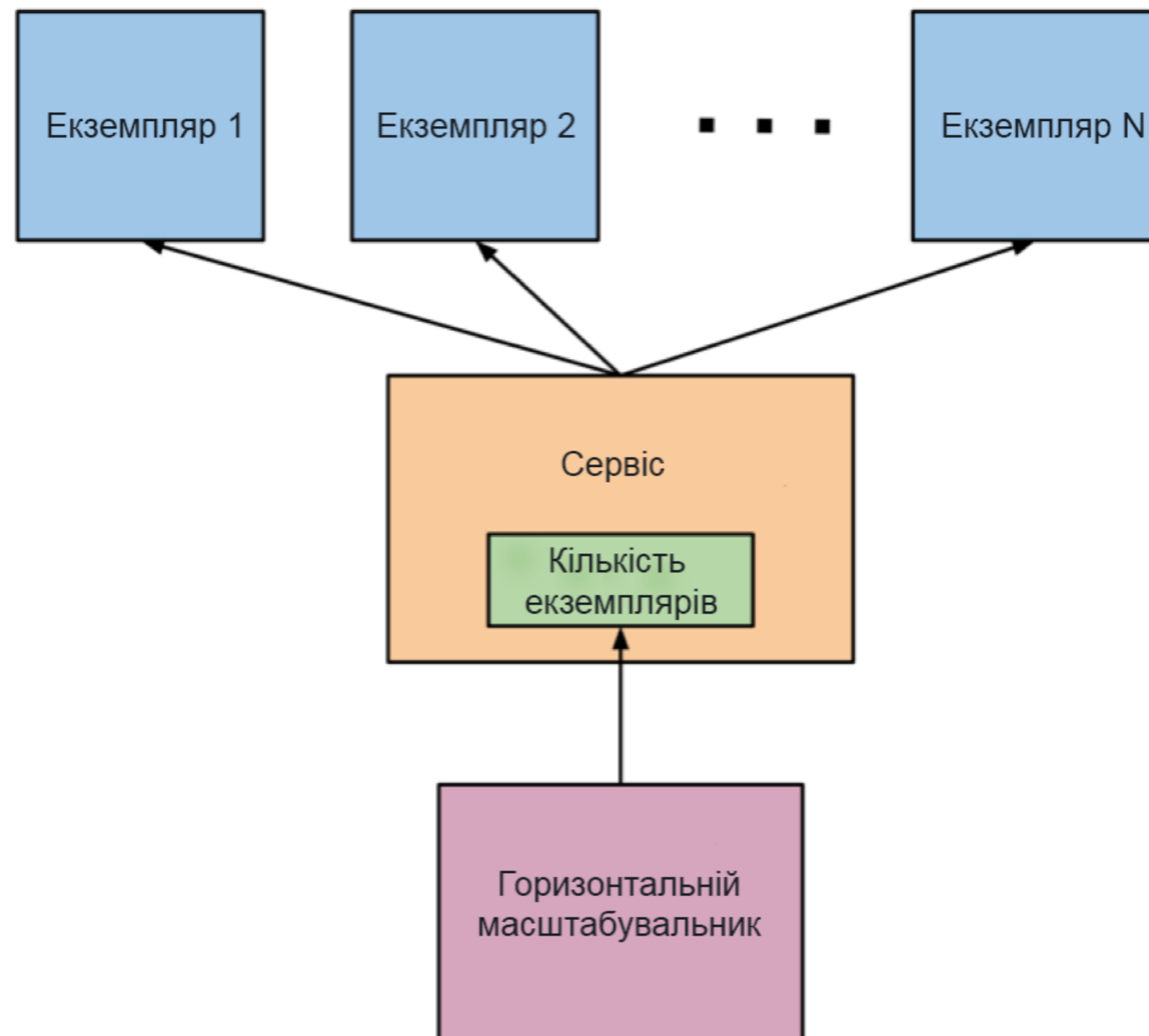
Ресурси



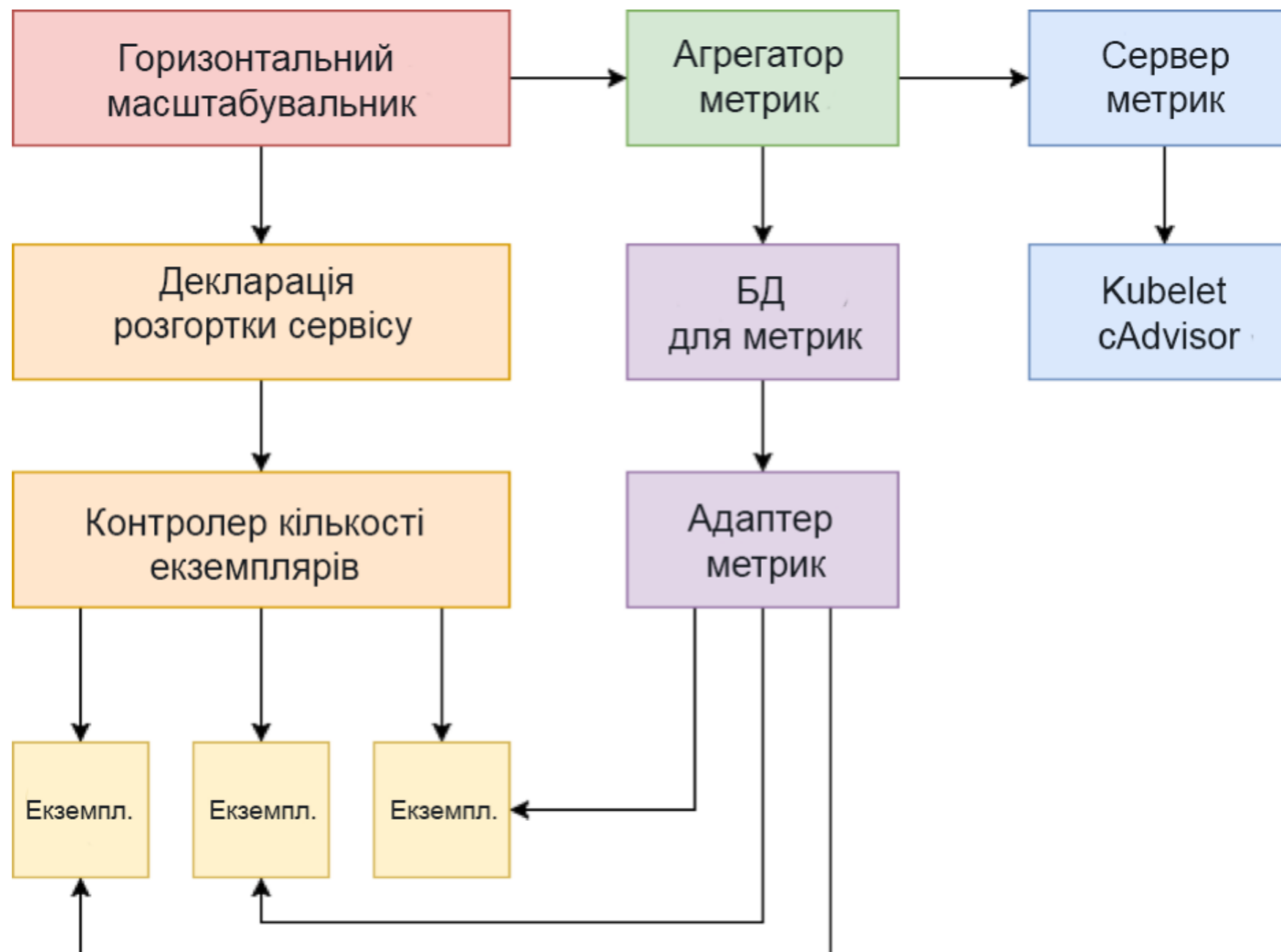
Час

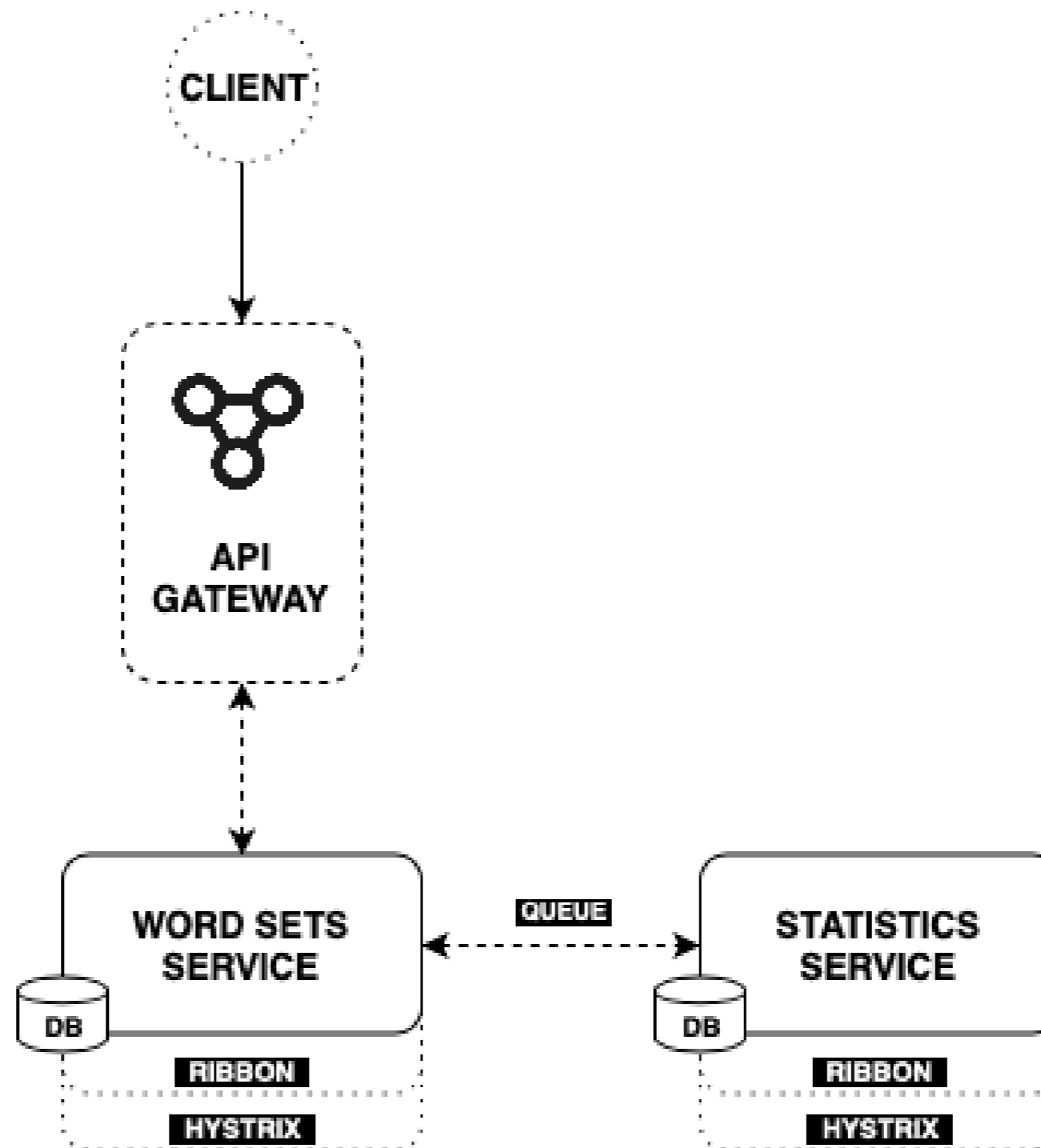
- ▶ Для автоматизації масштабування був використаний оркестратор Kubernetes та його можливості по автоматизації горизонтального масштабування
- ▶ Автоматичне масштабування ресурсів у хмарному провайдері та масштабування в Kubernetes дозволяє досягти динамічної зміни обчислювальної потужності

- ▶ Функція горизонтального автоматичного масштабування відстежує кількість екземплярів сервісів за різними метриками



- ▶ Метрики ресурсів збираються з kubelet агентів
- ▶ Метрики сервісів збираються до бази даних Prometheus та передаються до масштабувальника

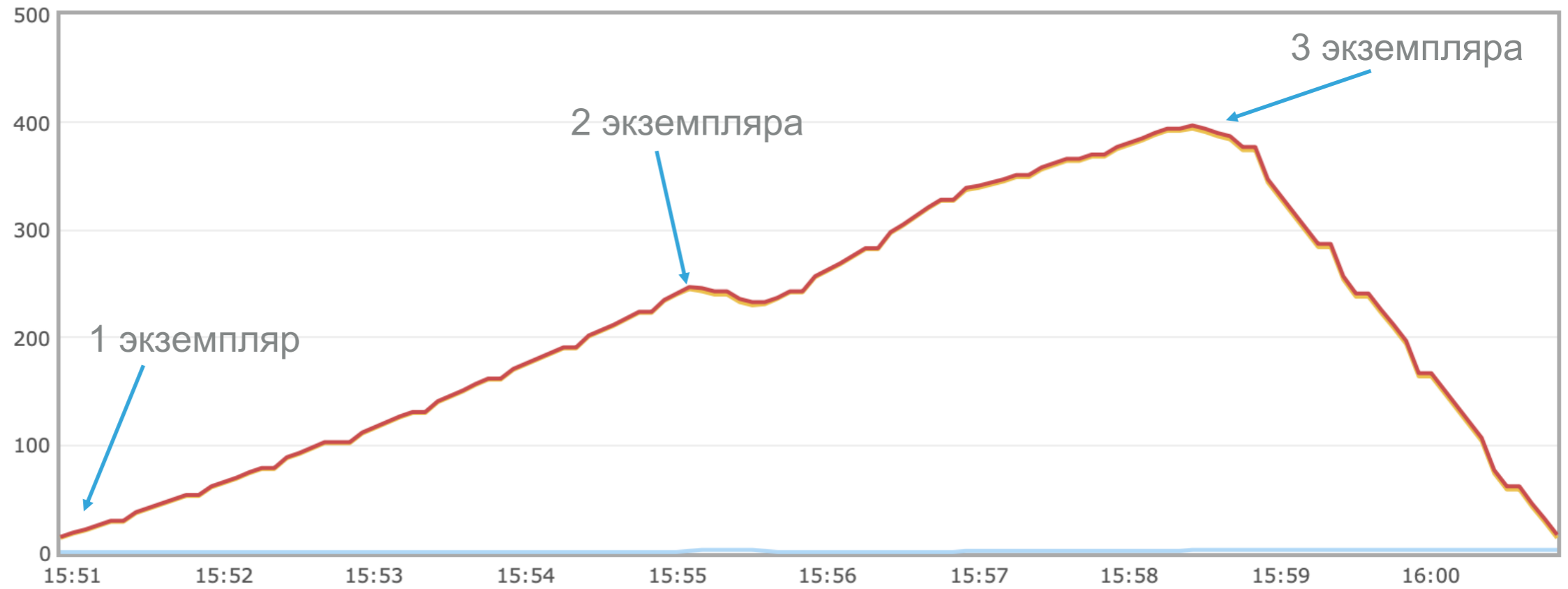






Queued messages last ten minutes ?

Довжина  
черги, шт.



Час, хвилини



Search

+ CREATE |

## Overview

### Cluster

- Namespaces
- Nodes
- Persistent Volumes
- Roles
- Storage Classes

Namespace

default

### Overview

### Workloads

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets

### Discovery and Load Balancing

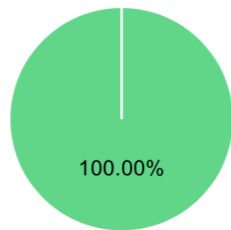
- Ingresses
- Services

### Config and Storage

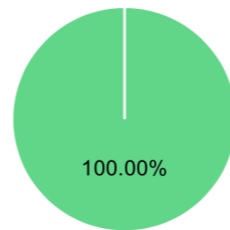
- Config Maps
- Persistent Volume Claims

## Workloads

### Workloads Statuses



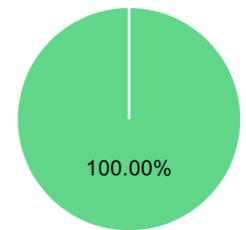
Deployments



Pods



Replica Sets



Stateful Sets

### Deployments

Name	Labels	Pods	Age	Images
✓ <a href="#">statistics-service</a>	app: glossary	1 / 1	2 months	solomkinmv/statistics-service
✓ <a href="#">dashboard</a>	app: glossary	1 / 1	2 months	solomkinmv/dashboard
✓ <a href="#">zipkin</a>	app: glossary	1 / 1	2 months	openzipkin/zipkin
✓ <a href="#">tts-service</a>	app: glossary	1 / 1	2 months	solomkinmv/tts-service
✓ <a href="#">translate-service</a>	app: glossary	1 / 1	2 months	solomkinmv/translate-service
✓ <a href="#">storage-service</a>	app: glossary	1 / 1	2 months	solomkinmv/storage-service
✓ <a href="#">rabbitmq</a>	app: glossary	1 / 1	2 months	rabbitmq:management
✓ <a href="#">mysql-words</a>	app: glossary	0 / 0	2 months	mariadb:latest
✓ <a href="#">registry</a>	app: glossary	1 / 1	2 months	solomkinmv/discovery-server
✓ <a href="#">mysql-keycloak</a>	app: glossary	0 / 0	2 months	mysql:5.7.22

- ▶ Мікросервісна архітектура дозволяє точно налаштовувати продуктивність системи
- ▶ Застосування автоматичного масштабування дозволяє будувати системи що підлаштовуються під поточне навантаження. Цей підхід не тільки підвищує стабільність систем, але також дозволяє зберігати ресурси на серверах, що дозволяє клієнтам мінімізувати обсяг необхідних ресурсів, коли рівень навантаження мінімальний. Використання користувацьких показників покращує точність автоматичного масштабування та надає широкий спектр можливостей налаштування розподіленої системи.
- ▶ За результатами інтеграції та ручних тестів підтверджено коректність роботи системи.