



ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

Corso di Laurea in Ingegneria Informatica



Performance di Architetture EventMesh a supporto di Microservizi

Anno accademico 2022/2023
Sessione di laurea I - 20/07/2023

Tesi di:
Leonardo Dominici

Relatore:

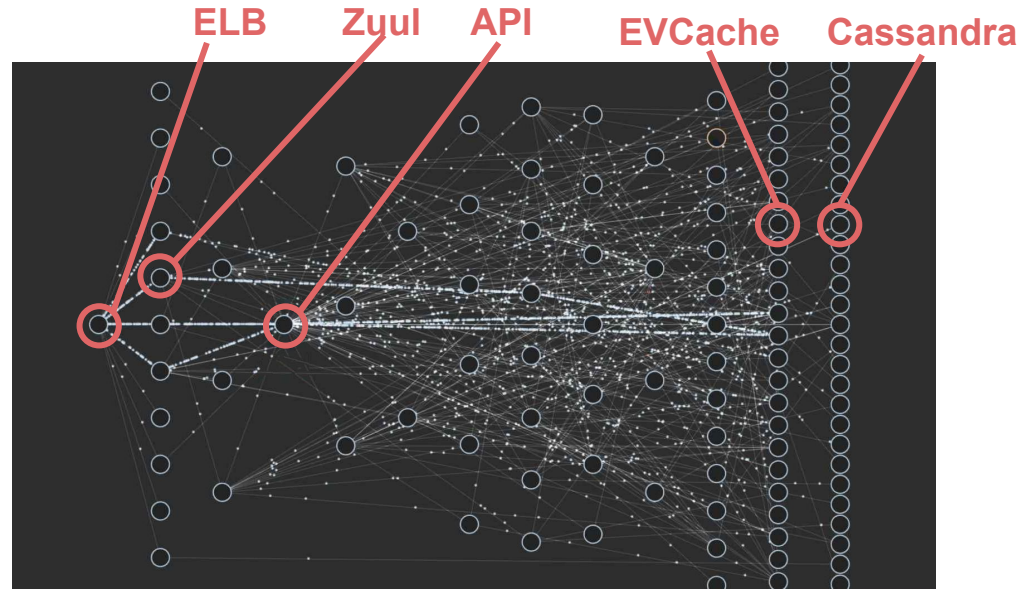
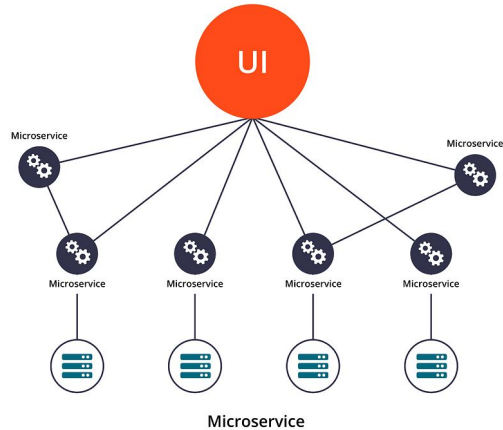
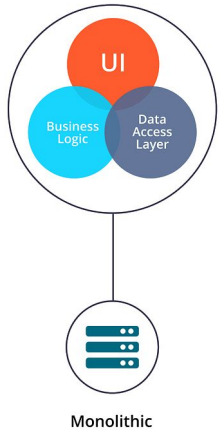


Correlatori:



Microservizi

Le architetture a microservizi
frazionano in piccoli servizi
indipendenti le funzionalità logiche



Numero di microservizi impiegati:

Netflix: 700

Uber: 1300

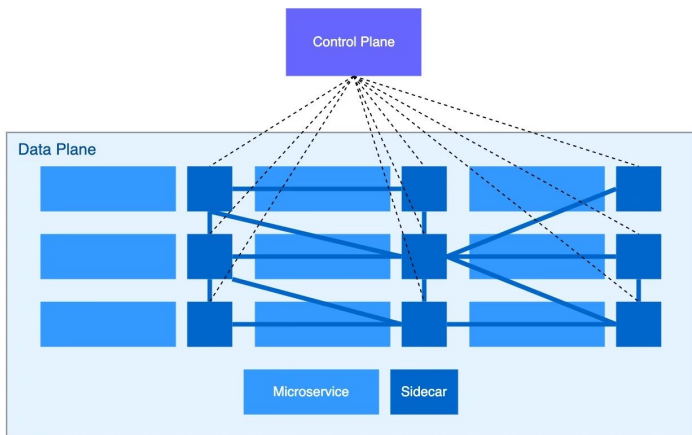
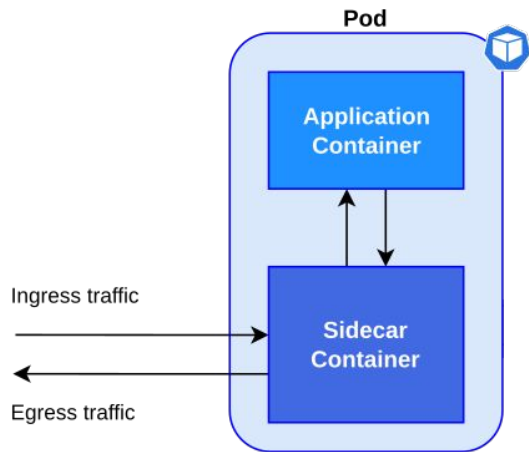
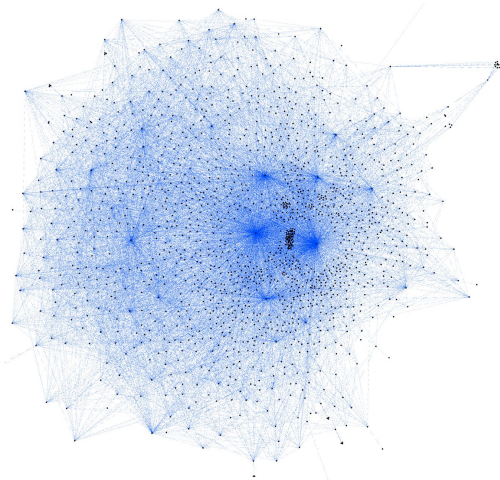
Spotify: 1200

Monzo: 1500

ServiceMesh

Coordina e realizza:

- Routing fra microservizi
- Sicurezza delle connessioni
- Politiche di load balancing, ritrasmissione e circuit breaking
- Monitoraggio
- Collegamenti extracluster



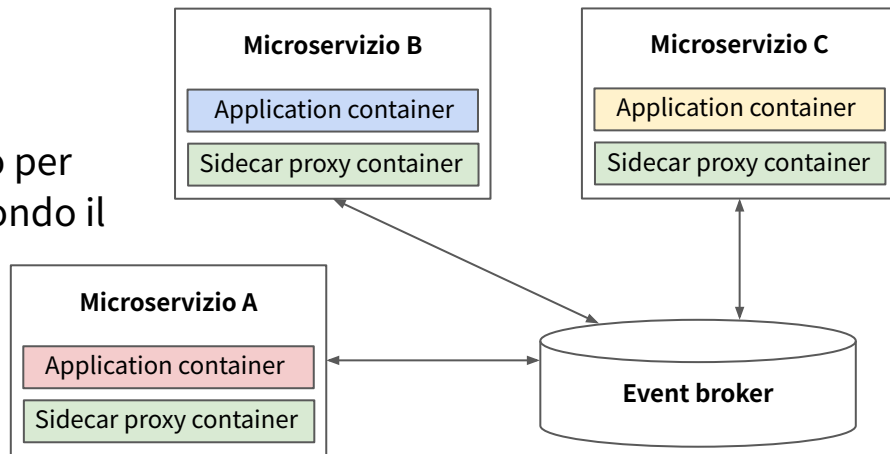
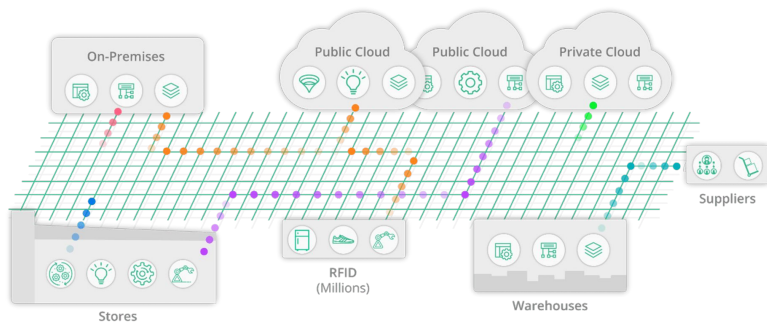
Il proxy garantisce modularità, flessibilità tecnologica e garantisce che il principio di singola responsabilità sia rispettato. Spesso l'implementazione impiegata è quella del sidecar proxy.

EventMesh

Consiste in un livello infrastrutturale dinamico per realizzare comunicazione fra microservizi secondo il modello ad eventi.

Garantisce:

- Basse latenze
- Alto throughput
- Scalabilità
- Disaccoppiamento
- Osservabilità



A differenza di EventMesh il tramite delle comunicazioni è l'event broker.

I proxy funzionano in modo analogo a ServiceMesh ma rendono comunicano con l'event broker attraverso gli eventi, indipendentemente dal protocollo usato con i microservizi.



Implementazione

Proxy:

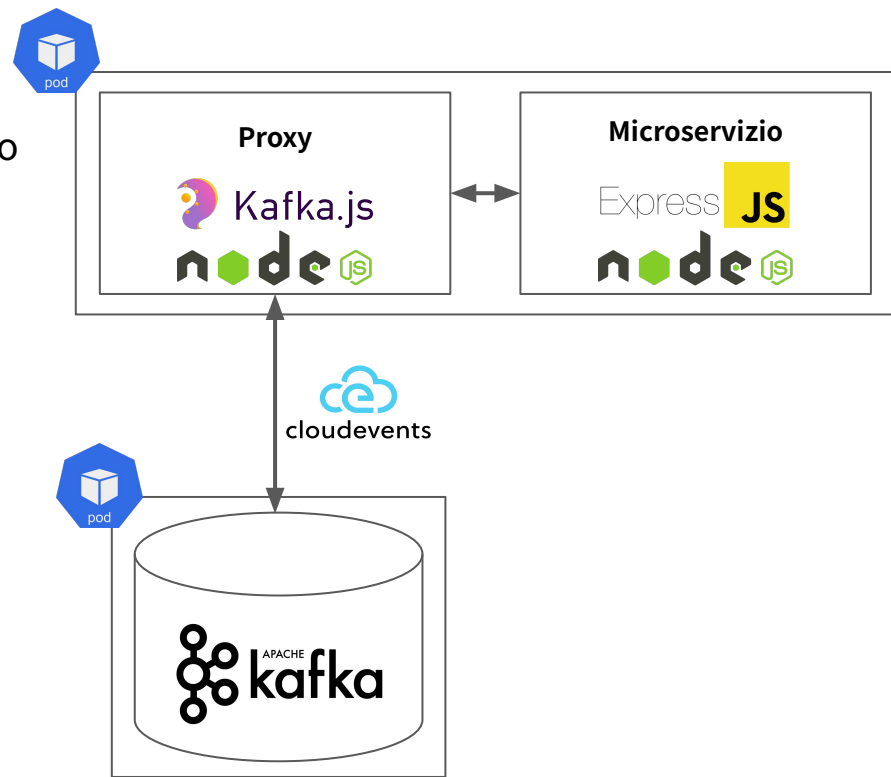
- Sidecar proxy nello stesso pod del microservizio
- Server NodeJS
- KafkaJS per connessione con Kafka
- Utilizzano CloudEvents (specifica CNCF)

Microservizi:

- Consumer e producer generici implementati
- Server NodeJS
- API esposte con Express.js
- Comunicazione con proxy tramite WebSocket

Event broker:

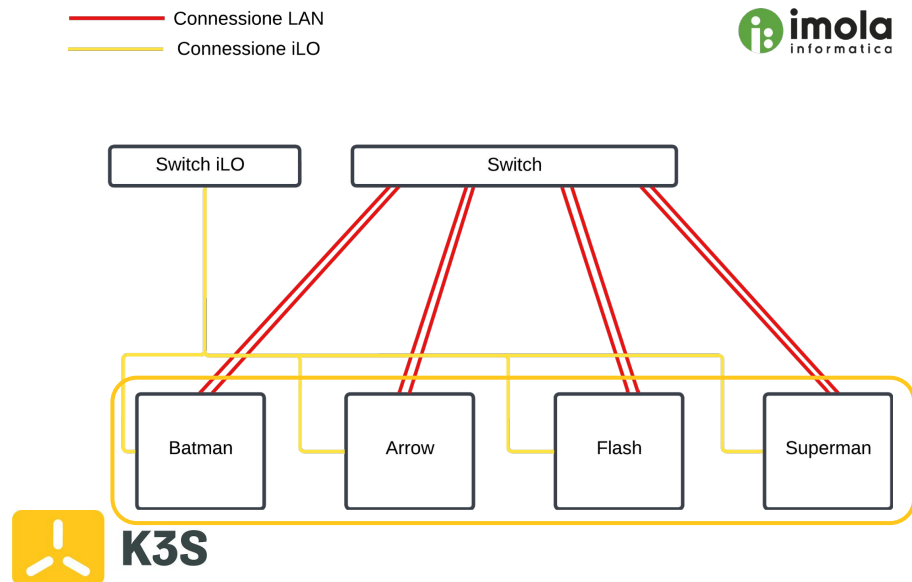
- Apache Kafka
- Deployment su un pod Kubernetes
- API esposte con Express.js
- Comunicazione con proxy tramite WebSocket



Deployment

Il deployment è stato realizzato all'interno di un laboratorio R&D su un cluster K3S (Kubernetes) composto da 4 macchine di cui una master.

OS	RAM	CPU	Ethernet
Ubuntu 22.04 LTS	11917MiB DDR4	Intel Celeron G1610T@2.3Ghz	2x1Gib/s + 100Mib/s iLO



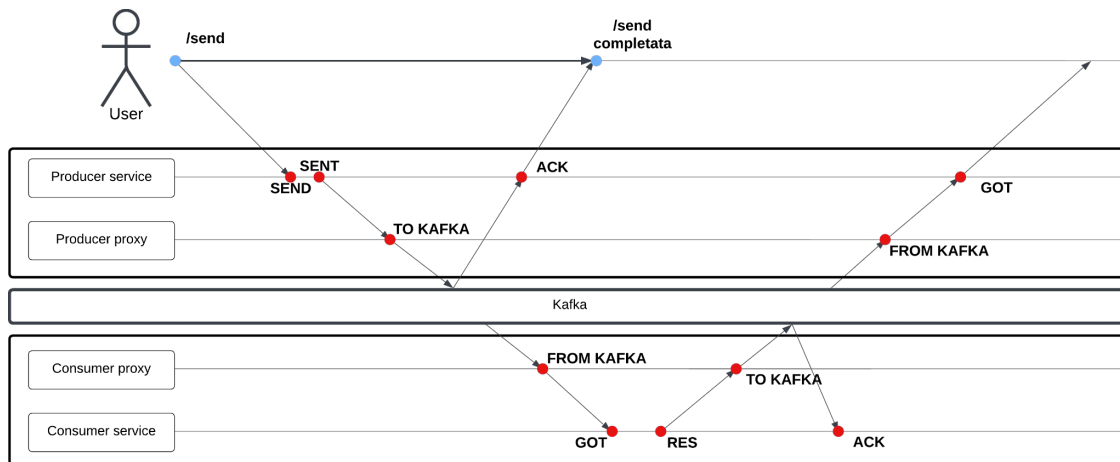
Per ragioni legate alla sincronizzazione degli orologi, durante i test tutti i componenti di EventMesh sono stati eseguiti sullo stesso nodo.

Log e metriche

Ogni attore riporta entry in un file di log composte da:

<azione> <UUID> <timestamp>

Notare come dopo la risposta dell'API /send la comunicazione continui



Le metriche che più ci interessano sono gli intervalli:

- SENT-ACK: tempo per elaborazione del proxy e conferma ricezione di Kafka
- TO KAFKA-FROM KAFKA: tempo instradamento attraverso Kafka
- SEND-GOT: tempo totale producer-consumer

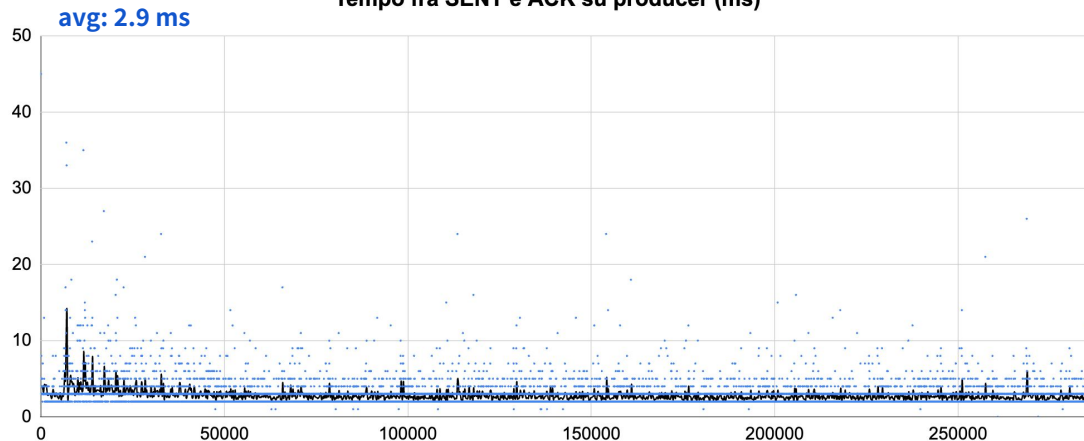
Single Send

Invio di richieste in maniera sincrona bloccante al fine di verificare la latenza minima dal sistema.

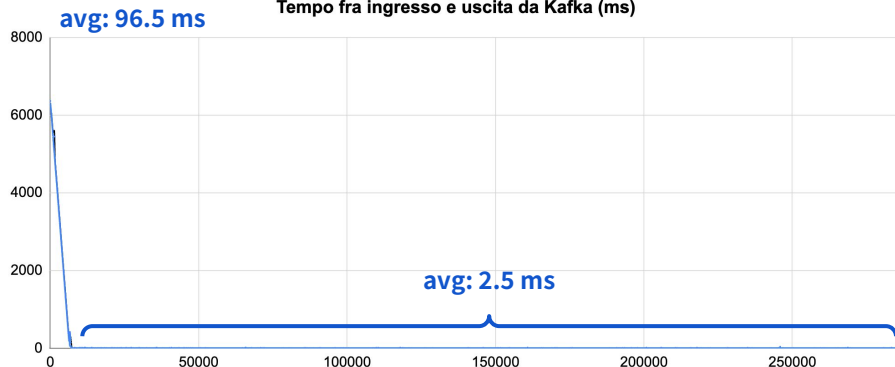
Chiamate realizzare tramite curl da una macchina Ubuntu all'interno del cluster.

13.5K richieste in 5 minuti: **45 req/sec**

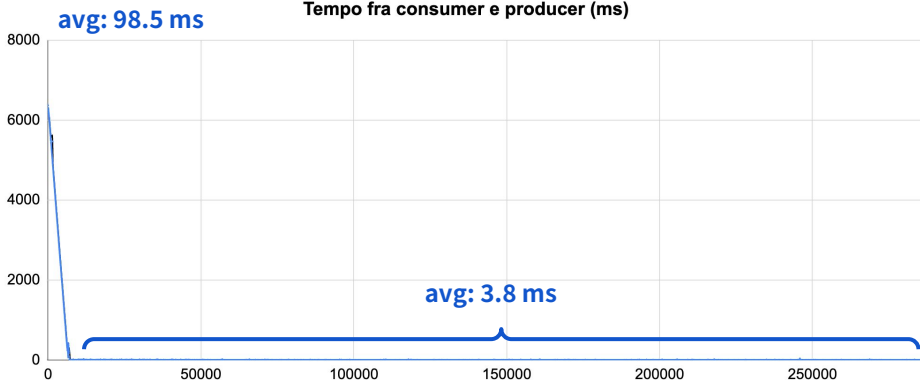
Tempo fra SENT e ACK su producer (ms)



Tempo fra ingresso e uscita da Kafka (ms)



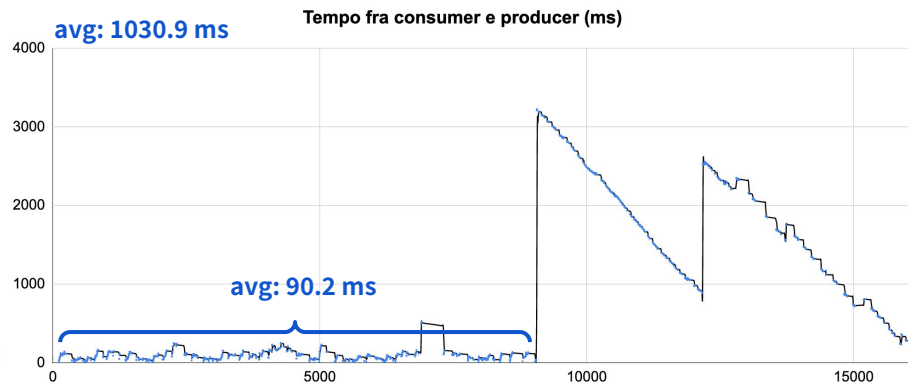
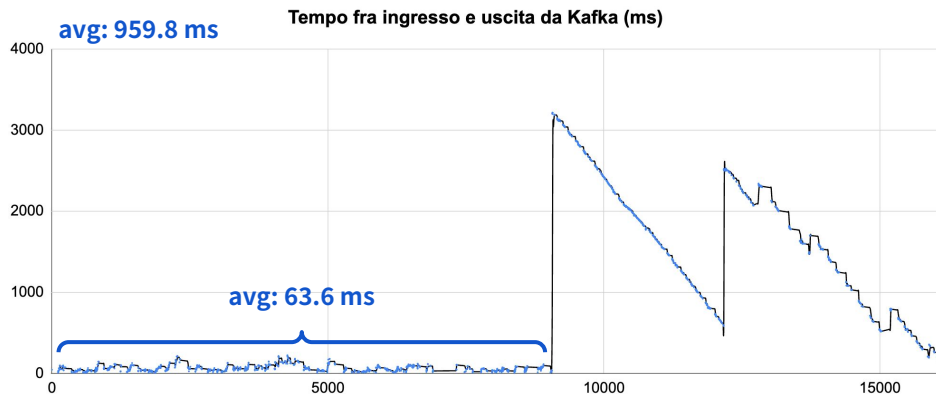
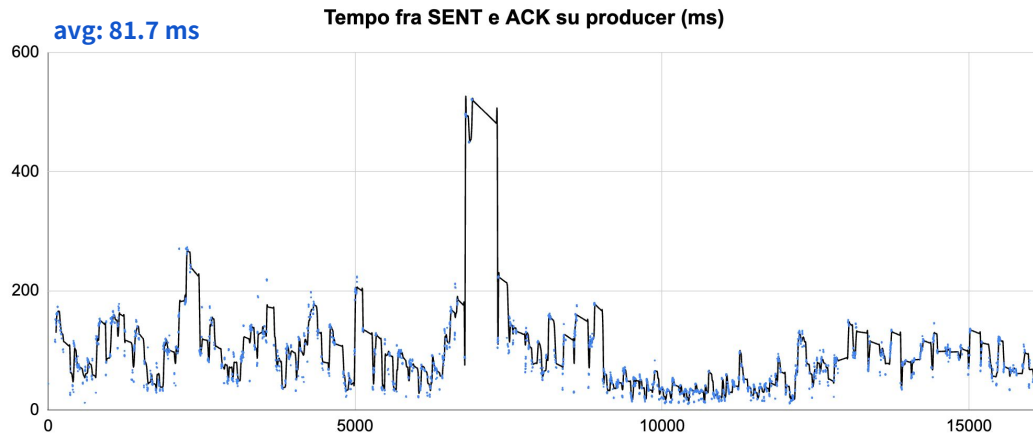
Tempo fra consumer e producer (ms)



Parallel Send

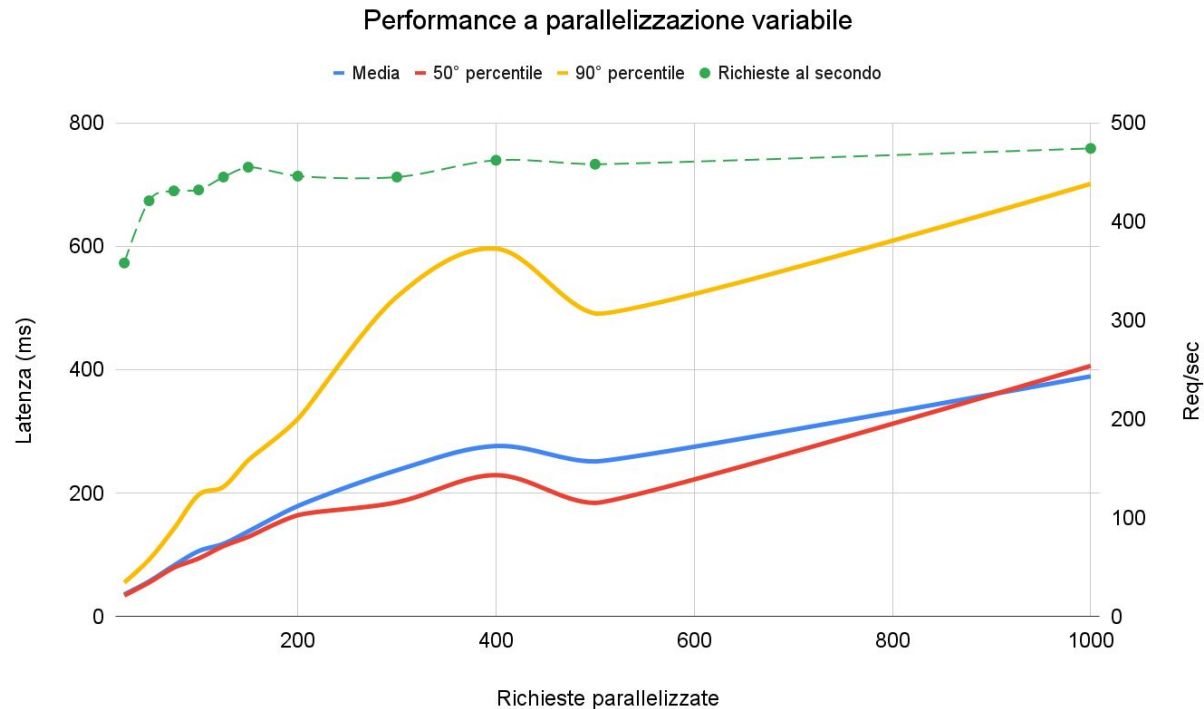
Invio di richieste in parallelo tramite Apache bench per verificare le latenze in casi di carichi paralleli.
Sempre 50 chiamate all'API attive in ogni momento.

5K richieste in 16 secondi: **312 req/sec**



Resistenza al carico

Invio di richieste in parallelo tramite Apache bench a parallelizzazione variabile. In ogni test sono state inviate 5K richieste indipendentemente dalla parallelizzazione.



Conclusioni

EventMesh e ServiceMesh sono tecnologie necessarie in architetture molto estese.

Kafka richiede configurazioni più personalizzati per migliorare le performance dell'implementazione.

Apache EventMesh è ancora acerba, non considerabile un prodotto.

Un refactoring dell'implementazione gioverebbe alla facilità di utilizzo.



ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA



Performance di Architetture EventMesh a supporto di Microservizi

Anno accademico 2022/2023
Sessione di laurea I - 20/07/2023

Leonardo Dominici
Matricola: s0000971128

Mail: leonardo.dominici2@studio.unibo.it
leonardodominici01@gmail.com

Implementazione EventMesh: <https://github.com/Leodom01/EventMeshThesis>