

---

## PROGETTO 2

# ANALISI REAL-TIME DI DIFETTI NELLA PRODUZIONE L-PBF CON APACHE FLINK

- Giuseppe Marseglia – 0350066
- Alessandro Finocchi – 0340543
- A.A. 2024/2025



# AGENDA

---



1 INTRODUZIONE

2 ARCHITETTURA

3 DATA INGESTION

4 DATA PROCESSING

5 RESULTATI

6 CONCLUSIONI

# INTRODUZIONE



- Sviluppare il deployment per una piattaforma di analisi dei difetti nella produzione l-pbf, una tecnologia di stampa 3D per realizzare parti in metallo a partire a un letto di polveri metalliche fuse.
- Analisi delle immagini OT<sup>1</sup> provenienti dal Dataset della conferenza ACM DEBS 2025 fornito tramite un server REST chiamato **challenger**
  - Ogni letto è suddiviso in strati denominati **layers**
  - Ogni strato è suddiviso in sezioni denominate **tiles**
- Un elemento dello stream del challenger ha il seguente formato  
(seq\_id, print\_id, tile\_id, layer, tiff)

## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

1. Istantanee termiche del letto di polvere il cui colore dei pixel indica la temperatura del punto

# INTRODUZIONE



- Realizzare 3 query in 4 step
  1. **Q1: Filtraggio** dei punti in cui la temperatura ha un valore anomalo
  2. **Windowing** per analizzare gli ultimi 3 layer insieme
  3. **Q2: Classificazione** degli outliers in funzione della dev. di temperatura <sup>1</sup>
  4. **Q3: Clustering** degli outliers della Q2
- Effettuare benchmarking delle query con diverse configurazioni
  - Livello di **replicazione**
  - Livello di **parallelismo**
  - **Implementazione** della query 2
- Misurare le performance delle query in termini di **throughput** e **latenza** variando unidimensionalmente le configurazioni

## AGENDA

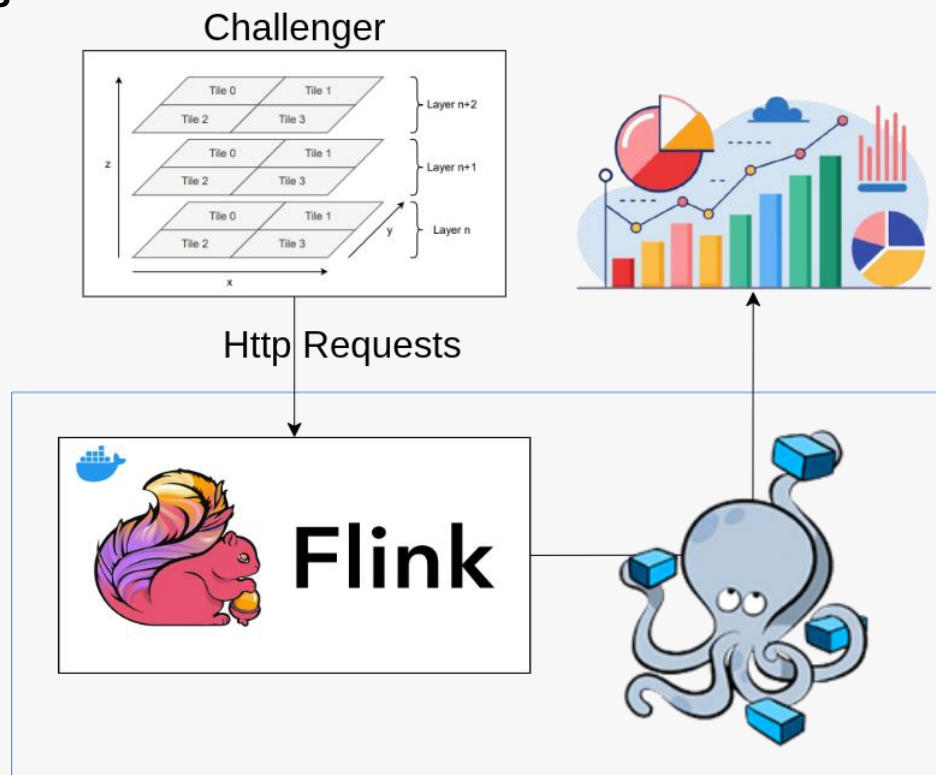
- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

1. Definita come la differenza assoluta tra la media delle temperature calcolate sui "vicini prossimi" e quella calcolata sui "vicini esterni"

# ARCHITETTURA



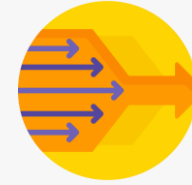
- Orchestrator: **Docker compose**
- Data ingestion e data processing: **Apache Flink**
- Data visualization: **JMP**



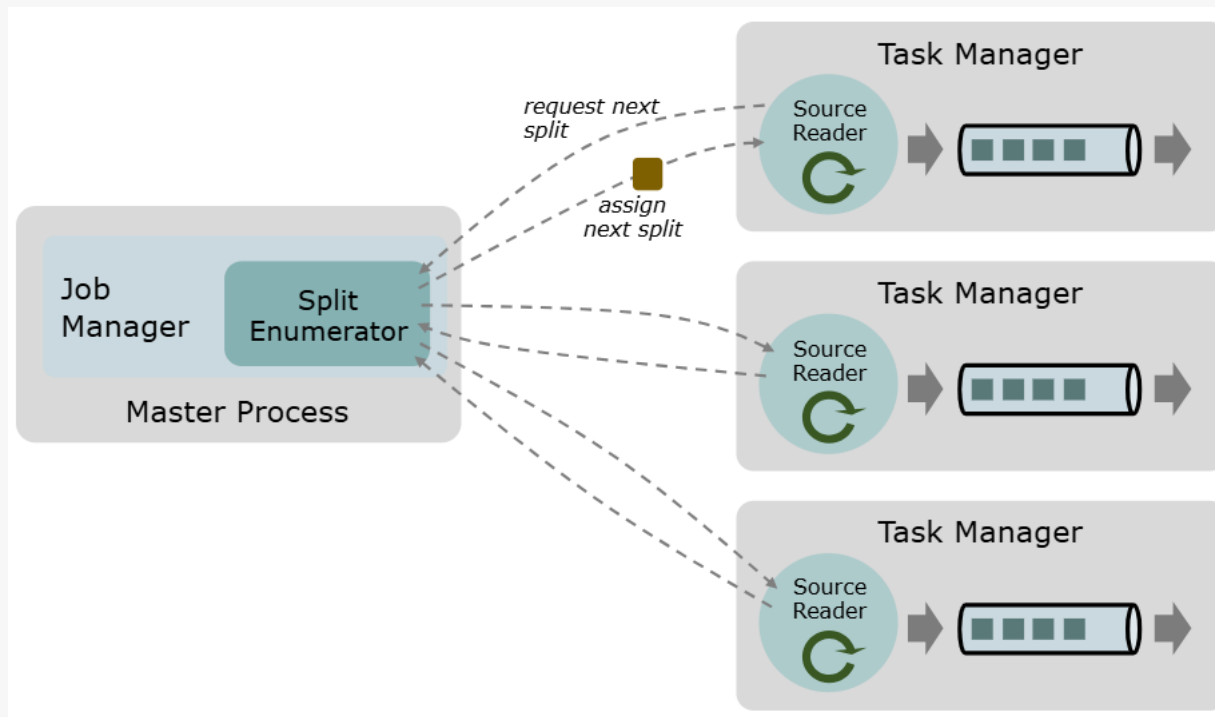
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA INGESTION



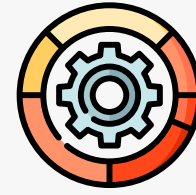
- Sono state implementate le richieste HTTP in Java attraverso le REST API per interfacciarsi con il Challenger
- L'API **Source** nativa di Flink svolge il ruolo di wrapper di queste API e permette ai dati di fluire nel sistema arrivando all'entry point del pre-processamento



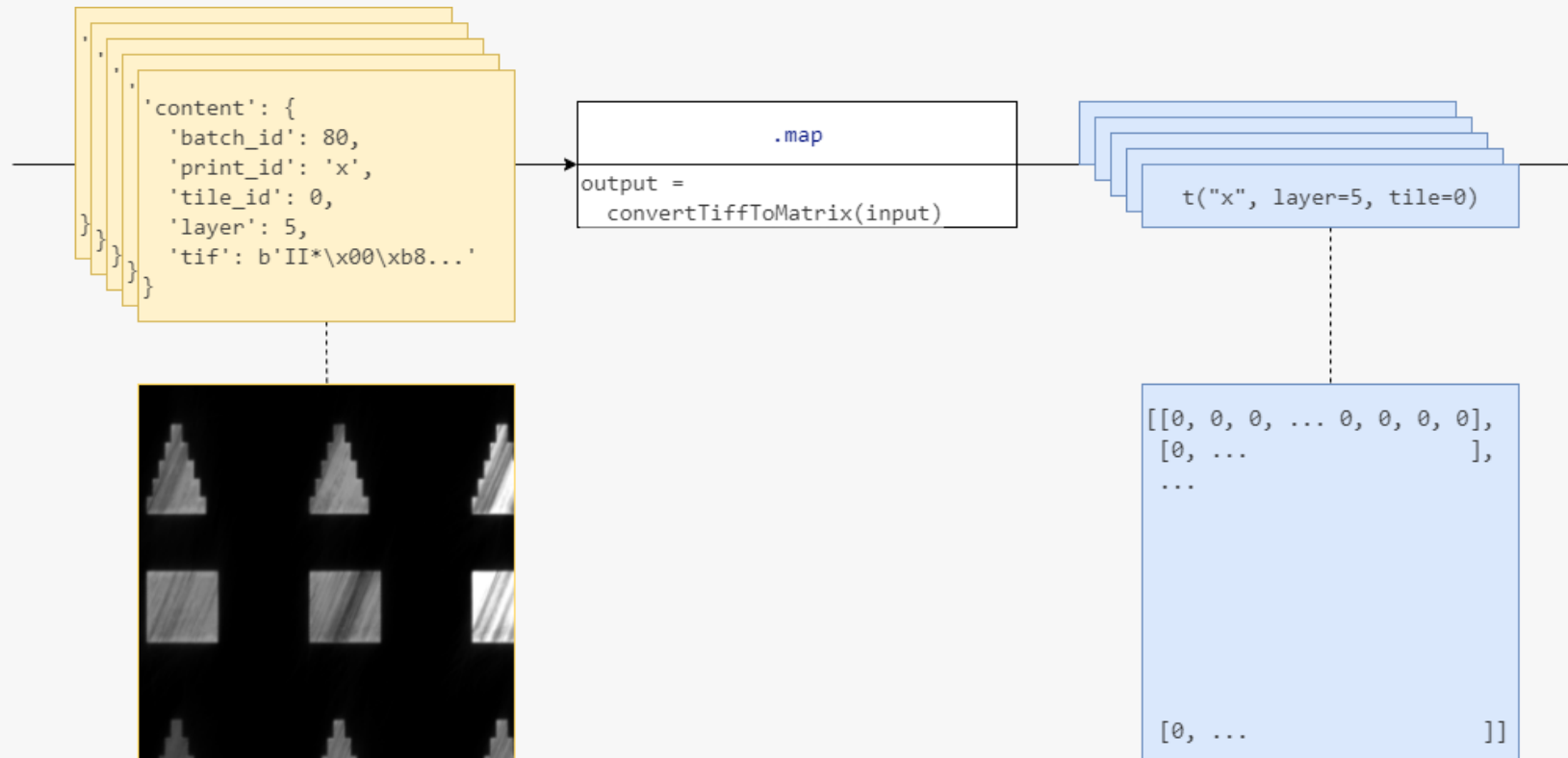
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA PRE-PROCESSING



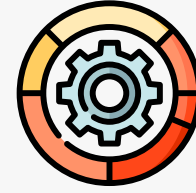
- Scopo: convertire i byte dell'immagine TIFF in una matrice di interi.



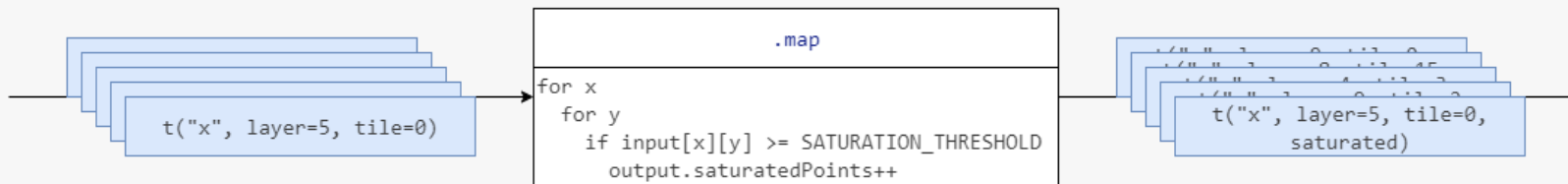
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA PROCESSING: QUERY 1



- Scopo: contare il numero di punti saturati.
  - Punti con un valore di temperatura maggiore di una soglia.

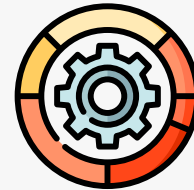


## AGENDA

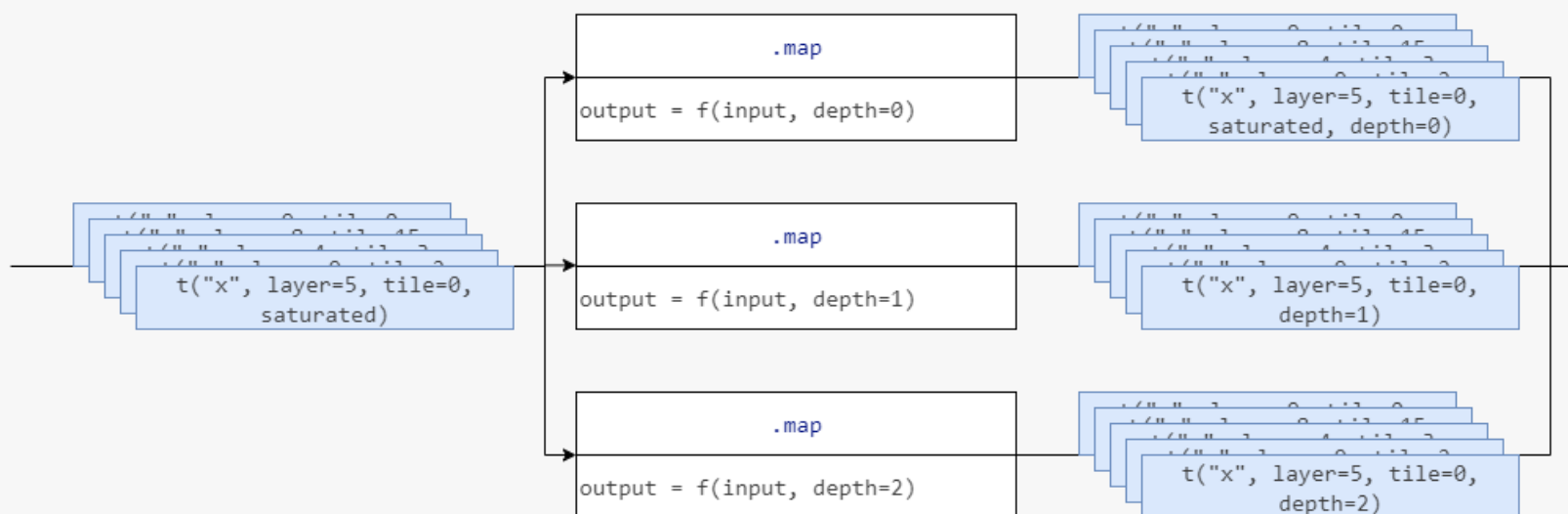
- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)



# DATA PROCESSING: QUERY 2



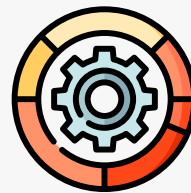
- Scopo: individuare gli outliers.
- 2 diverse implementazioni: naive e kernel.
- Fase 1: Preparare lo stream per la computazione parallela
  - Applicare prima convoluzione se implementazione con kernel



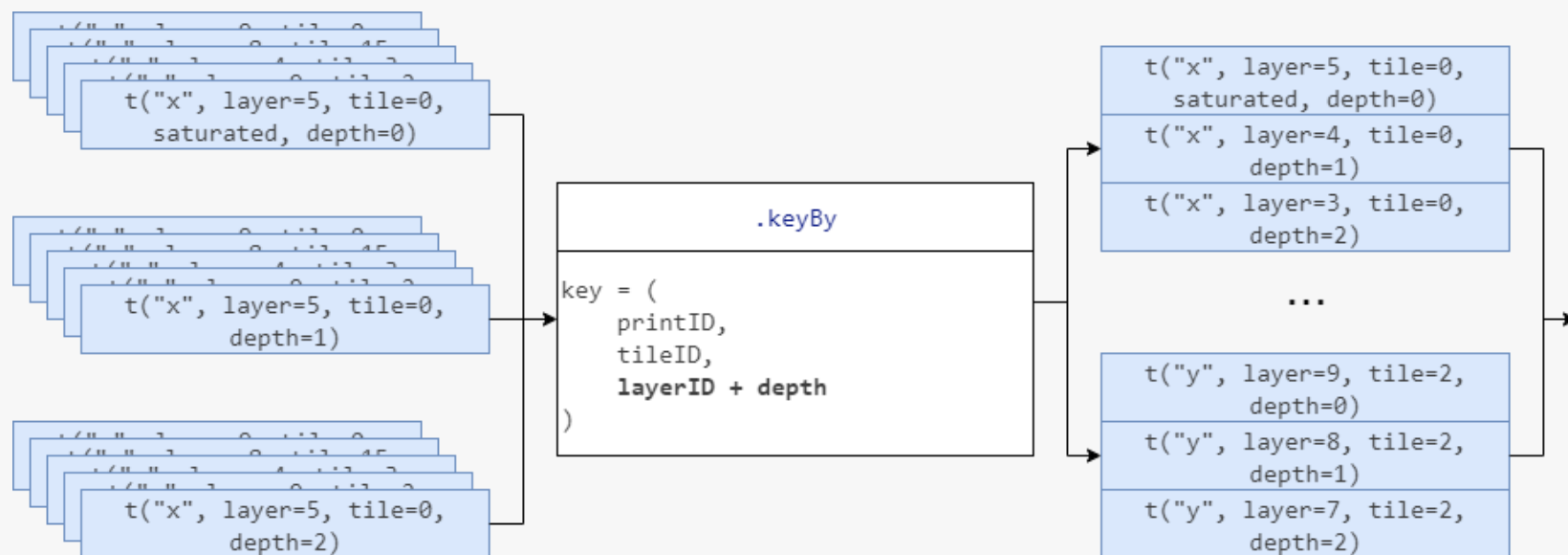
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA PROCESSING: QUERY 2



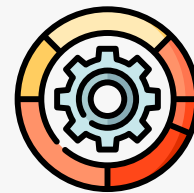
- Fase 2: windowing (KeyBy).
- Una chiave per ogni finestra.



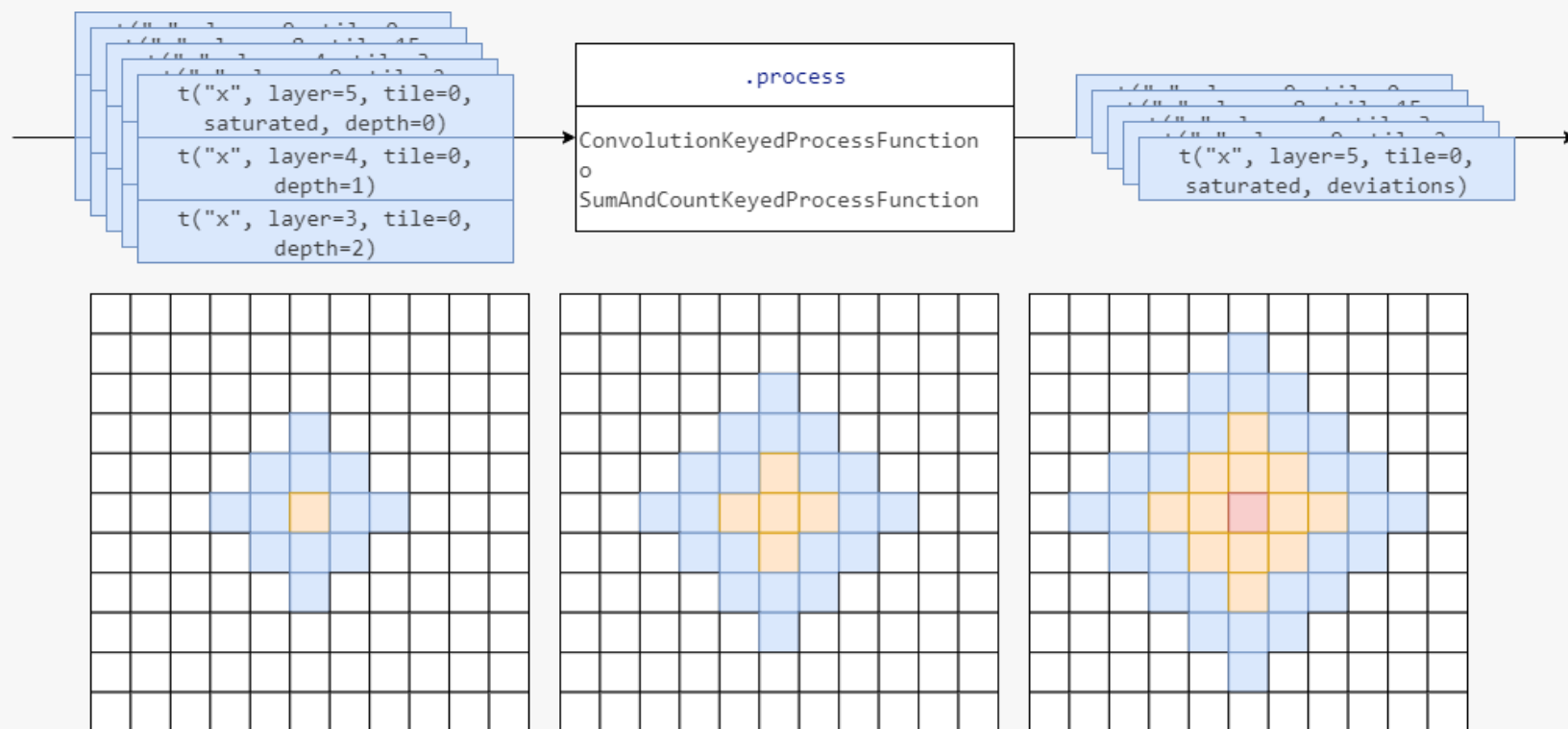
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA PROCESSING: QUERY 2



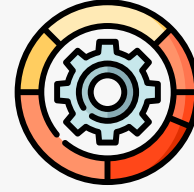
- Fase 3: windowing (KeyedProcessFunction) e calcolare il valore della deviazione di temperatura locale.
  - Approccio basato su convoluzione o approccio naive.



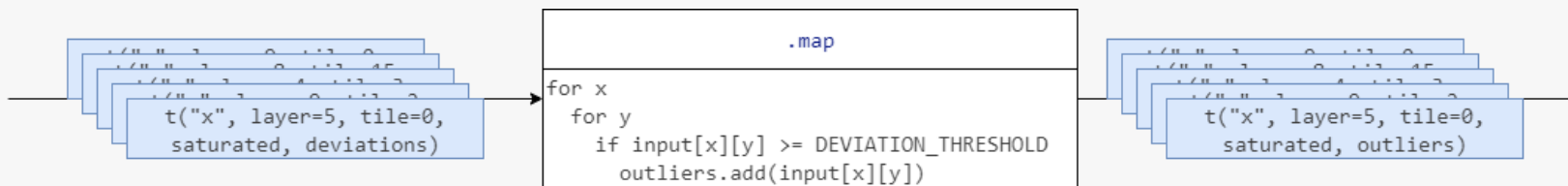
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA PROCESSING: QUERY 2



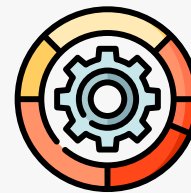
- Fase 4: individuare gli outliers.
  - Punti la cui deviazione di temperatura locale supera una soglia.



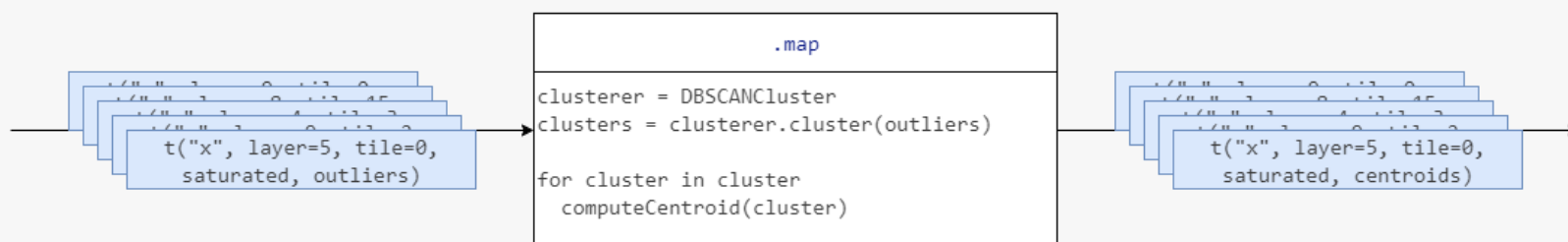
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# DATA PROCESSING: QUERY 3



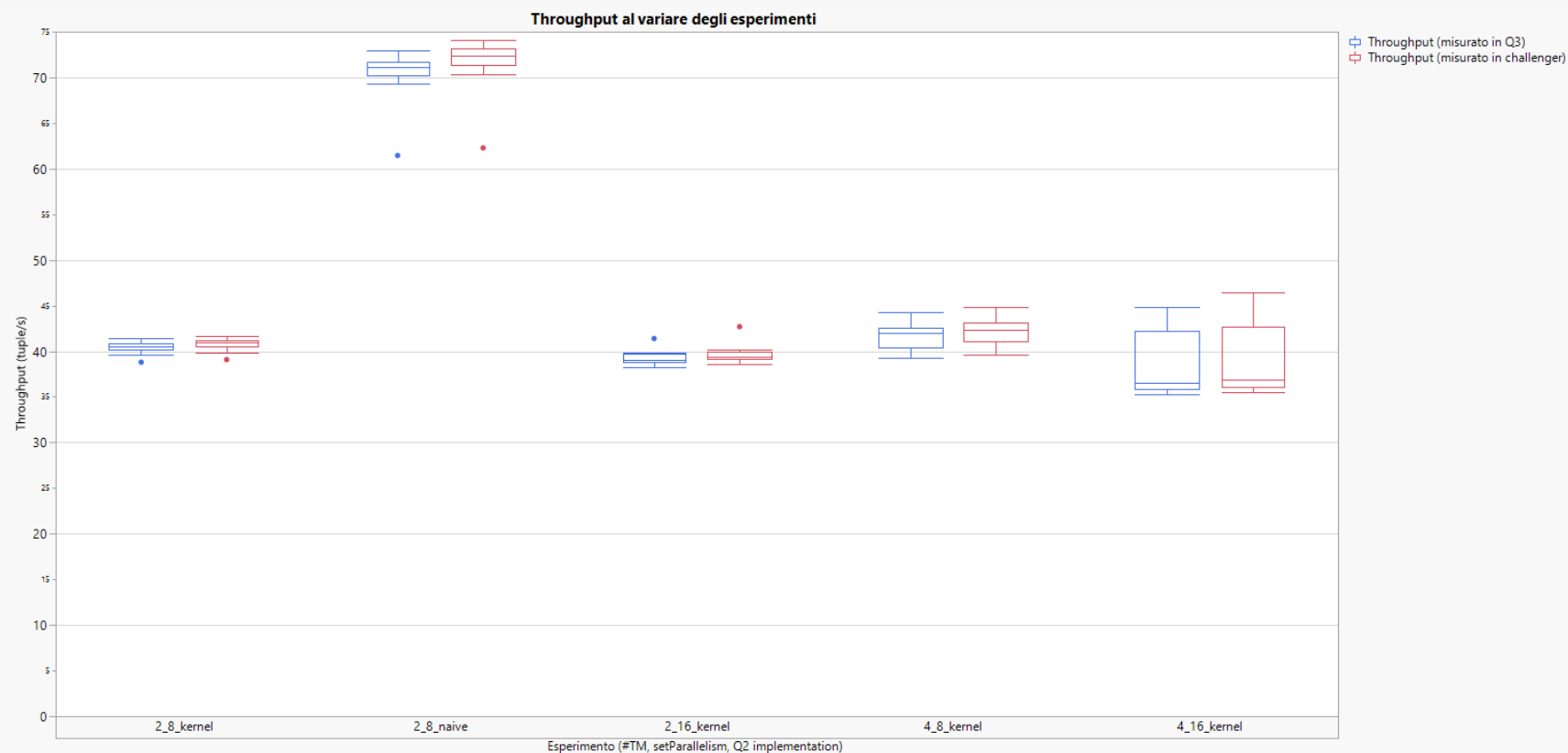
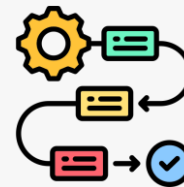
- Scopo: clusterizzare gli outliers tramite l'algoritmo DBSCAN.
  - DBSCANClusterer di Apache Commons Math.



## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

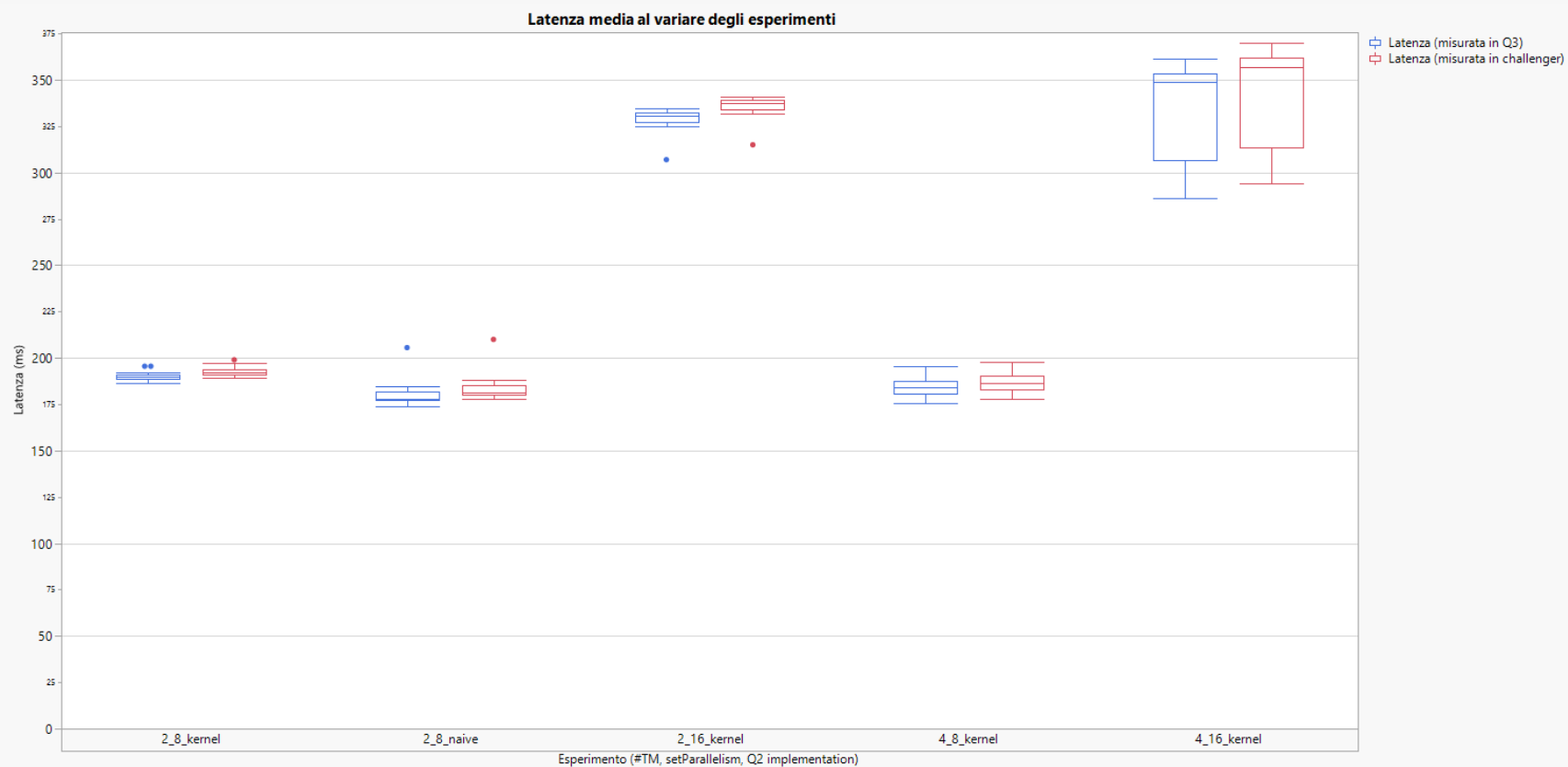
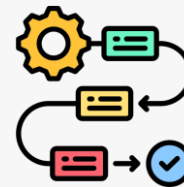
# RISULTATI



## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

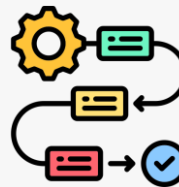
# RISULTATI



## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

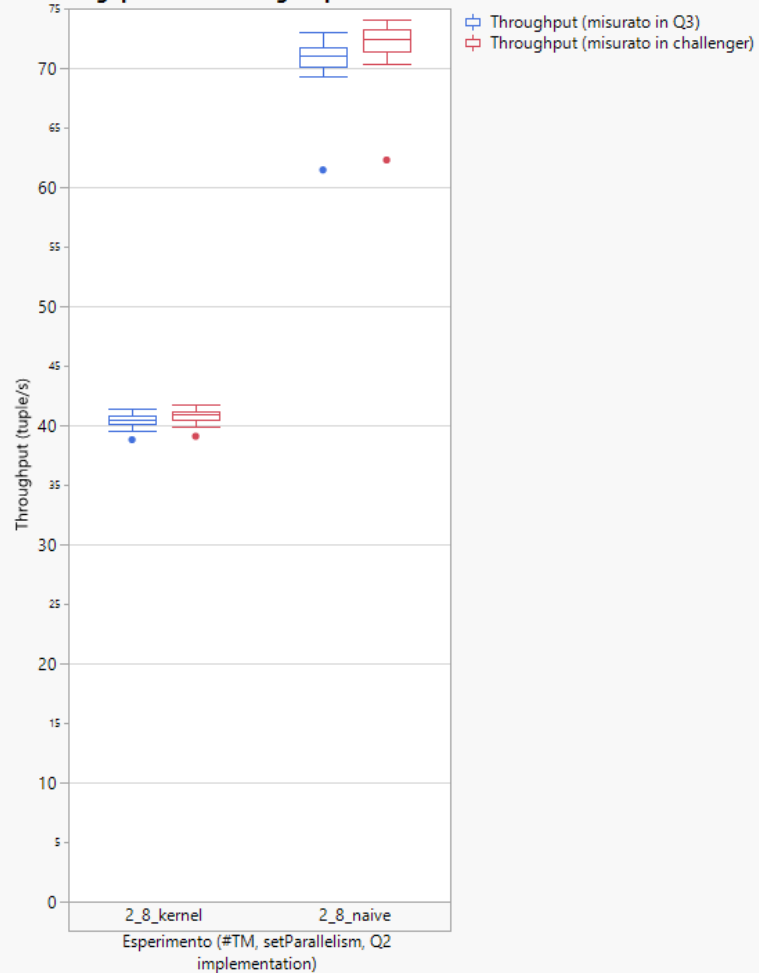
# RISULTATI



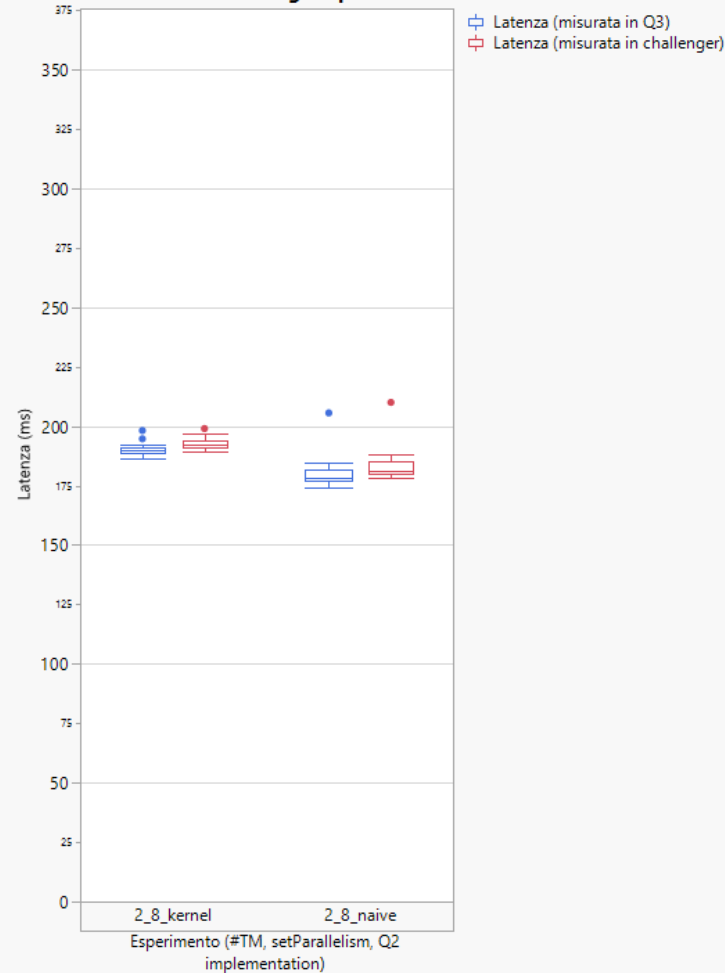
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

Throughput al variare degli esperimenti

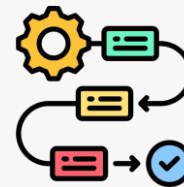


Latenza media al variare degli esperimenti

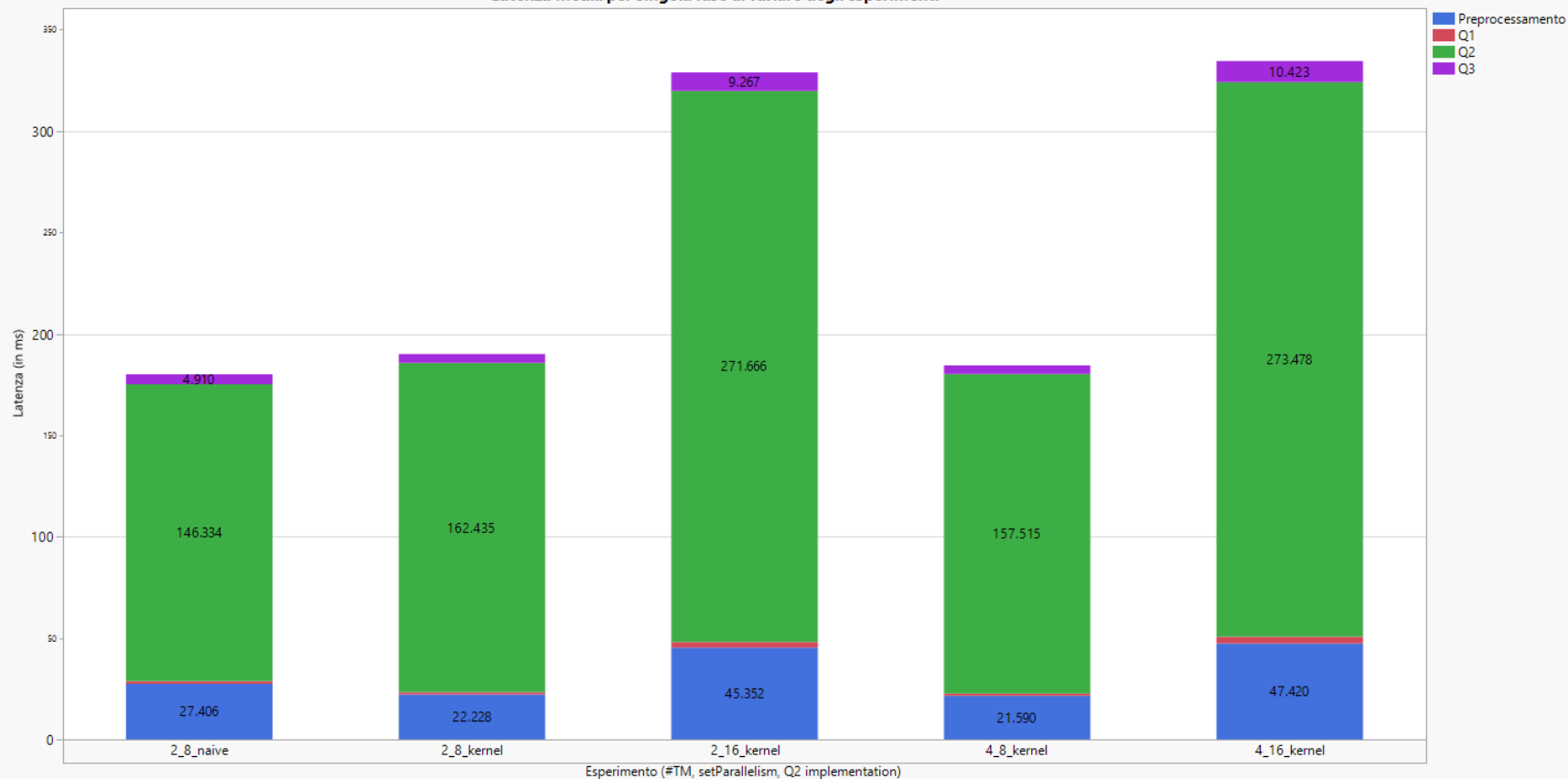




# RISULTATI



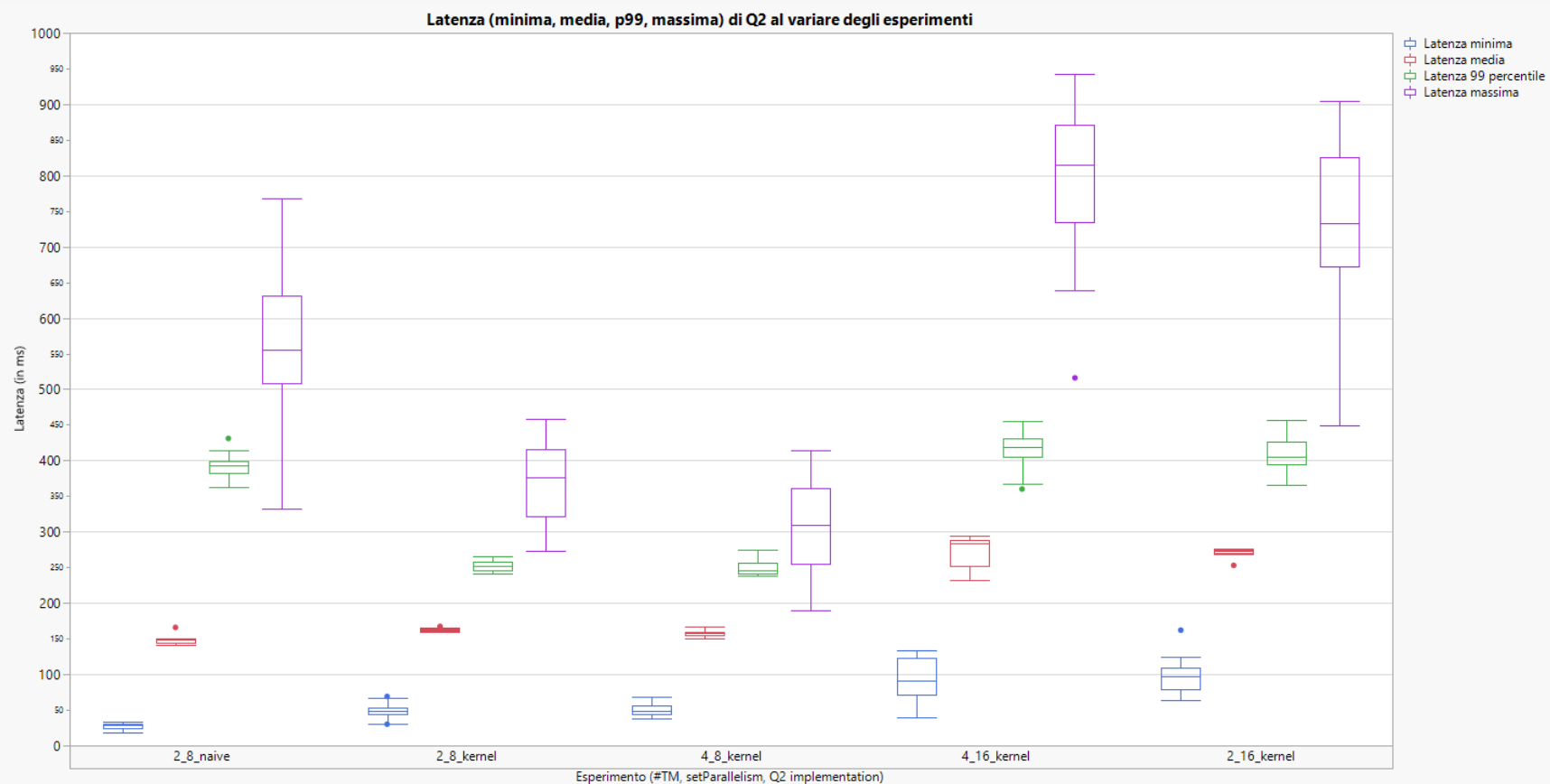
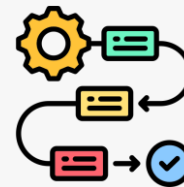
Latenza media per singola fase al variare degli esperimenti



## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

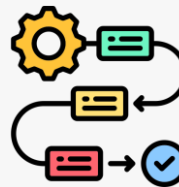
# RISULTATI



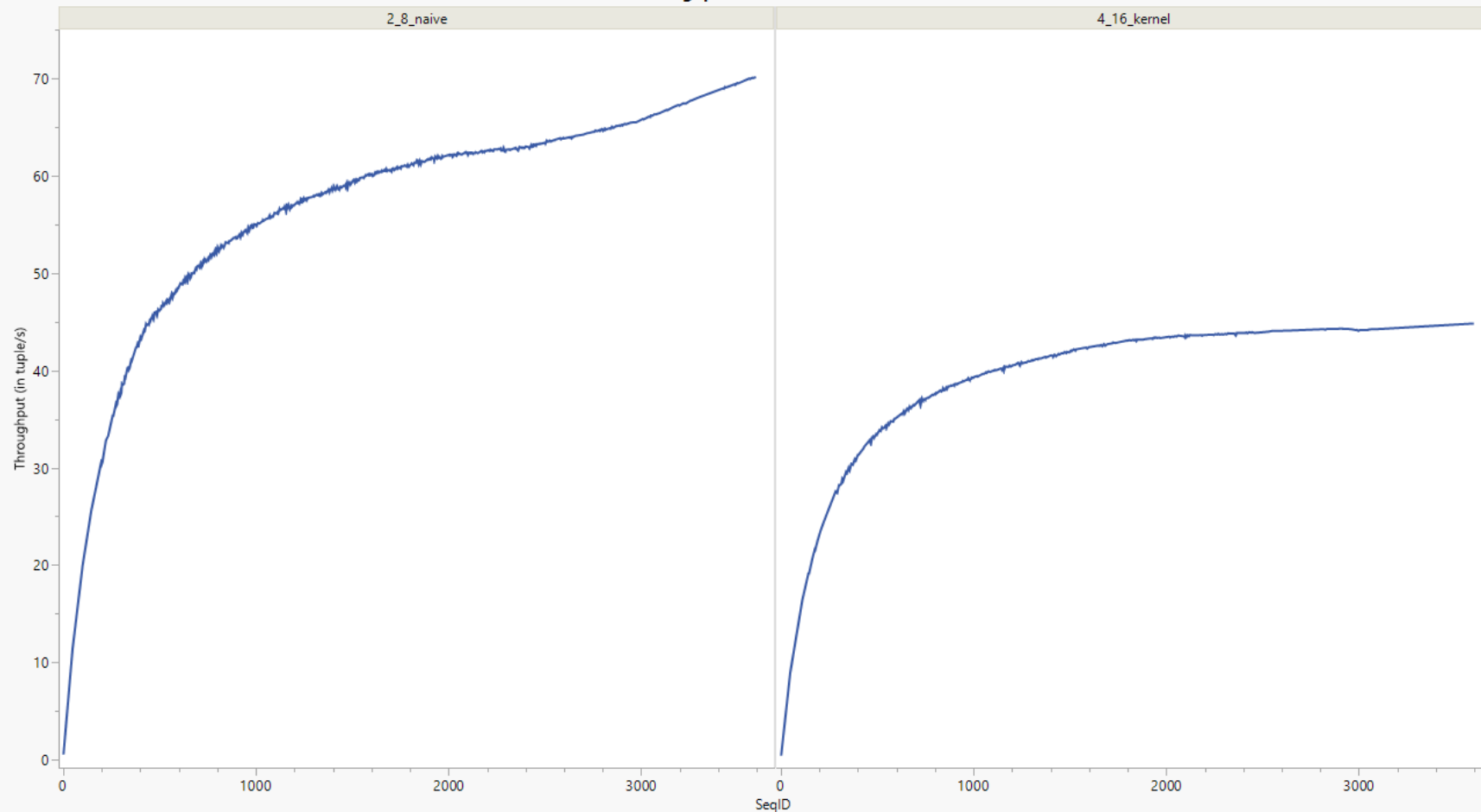
## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# RISULTATI



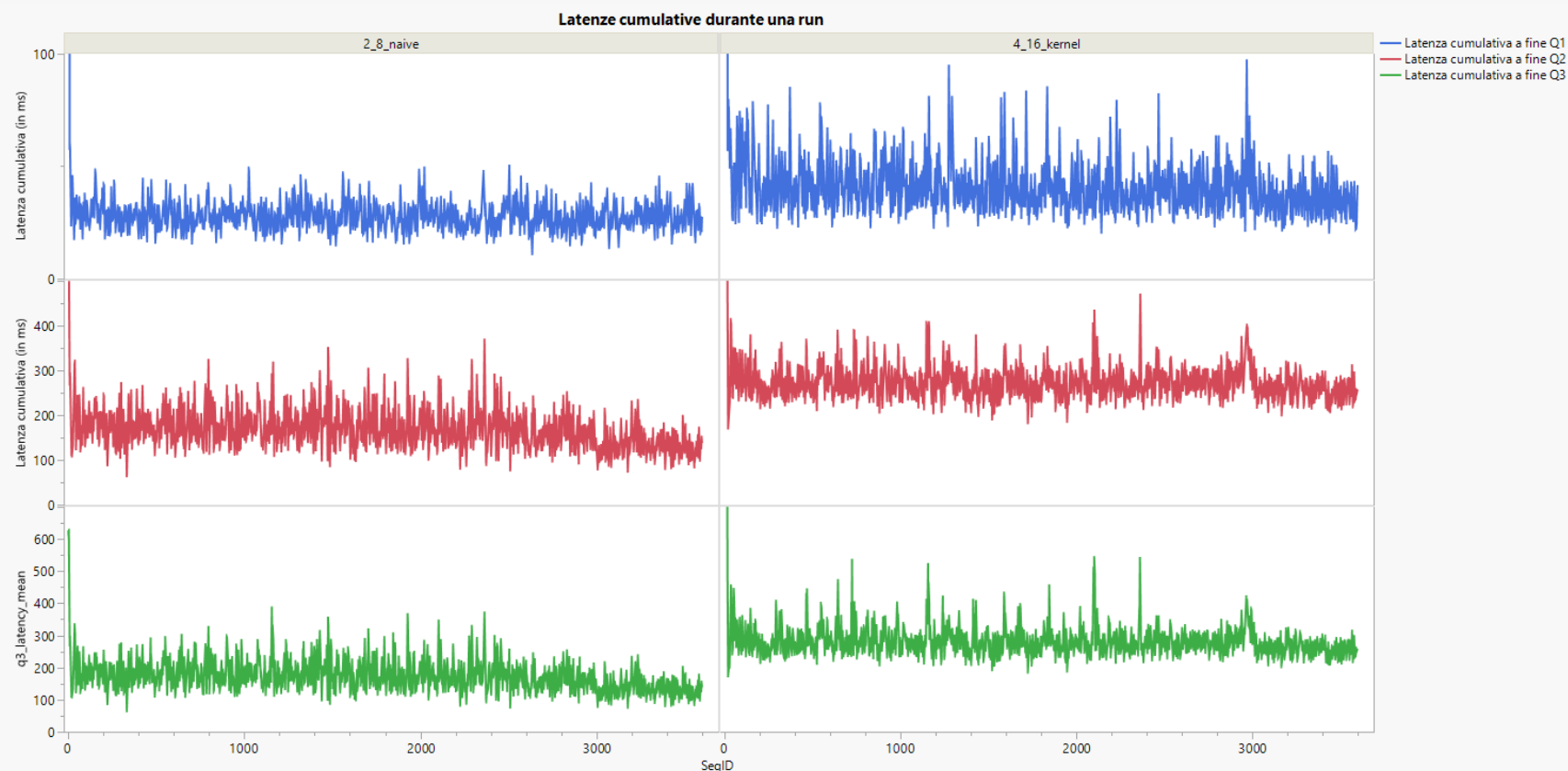
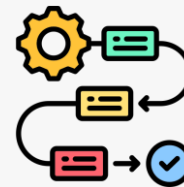
Throughput durante una run



## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# RISULTATI



## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# CONCLUSIONI



- Throughput massimo ottenuto con la configurazione 2\_8\_naive, la quale ottiene una latenza simile a quella di 2\_8\_naive
  - La mancanza di accelerazione hardware può portare l'approccio convolutivo a non sfruttare appieno le sue potenzialità
- Raddoppiare il parallelismo di 8 a 16 comporta raddoppiare anche la latenza, mentre variare il numero di taskmanager non sembra avere effetto sulle metriche
  - Probabilmente, l'insufficiente disponibilità di risorse fisiche ha portato la maggiore allocazione di risorse virtuali ad introdurre ulteriori ritardi di comunicazione che hanno controbilanciato i benefici del parallelismo
- La fase più computazionalmente onerosa è la Query 2
  - L'approccio naive presenta variabilità maggiore

## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# MINACCE ALLA VALIDITÀ



Riguardo alle performance delle query:

- ! **Validità esterna:** il grado con cui si può generalizzare la scoperta di uno studio ad altre situazioni, persone, contesti e misure
  - Avere a disposizione più risorse hardware potrebbe portare la Query 2 con l'approccio kernel ad essere preferibile
  
- ! **Validità conclusiva:** si riferisce alla correttezza delle conclusioni
  - Le metriche di latenza e throughput potrebbero non essere il focus in altri contesti
  - Le configurazioni scelte potrebbero non essere appropriate
  
- ! **Affidabilità:** si riferisce alla consistenza e ripetibilità delle misure
  - I risultati dipendono fortemente dalla macchina su cui sono calcolati
  - I risultati non sono riproducibili data l'imprevedibilità delle CPU

## AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)

# FINE



## GRAZIE PER L' ATTENZIONE!

Link al progetto:



[GitHub page](#)

### AGENDA

- [Introduzione](#)
- [Architettura](#)
- [Data ingestion](#)
- [Data processing](#)
- [Risultati](#)
- [Conclusioni](#)