



# Design e Sviluppo di una Soluzione per la Valutazione di Sistemi Distribuiti

Laure Triennale in Sicurezza dei Sistemi e delle Reti Informatiche

**Enea Manzi** (987326)

Relatore: Prof. Marco Anisetti

Correlatore: Dottor. Filippo Berto

22 Ottobre 2024

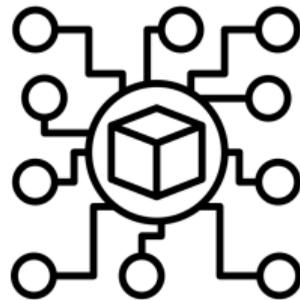


UNIVERSITÀ  
DEGLI STUDI  
DI MILANO



## Contesto

- Esigenze di **performance elevate**
- **Limiti dei sistemi centralizzati**
- Avvento del **cloud computing**:
  - Sistemi **decentralizzati**
  - Sistemi **distribuiti**
  - Architetture basate su **microservizi**





## Contesto

- Necessità di strumenti di **monitoraggio avanzato**
- Raccogliere **informazioni** dai **diversi sistemi**
- Necessità di avere **garanzie sui servizi**

**Obiettivo finale:** valutare il **comportamento complessivo** del sistema





## Gaps

- **G1 Generalizzabilità** Necessità di un sistema di valutazione **generico** che si **interfaccia** a vari strumenti di **monitoraggio**
- **G2 Valutazioni Performance** Focus ristretto su aspetti di **performance**, peccando di un'approccio **olistico e continuativo** che valuti il **comportamento** del sistema
- **G3 Elasticità Contratti** Framework attuali non permettono l'**elasticità** che cerchiamo





## Obbiettivi

- **O1:** Estensione dell'Assurance Engine per supportare il monitoraggio basato su **Elasticsearch**
- **O2:** Implementazione di **contratti** basati su **evidence** per eseguire **valutazioni** di PNF
- **O3:** Valutazione sperimentale delle **performance del sistema** in un cluster Kubernetes





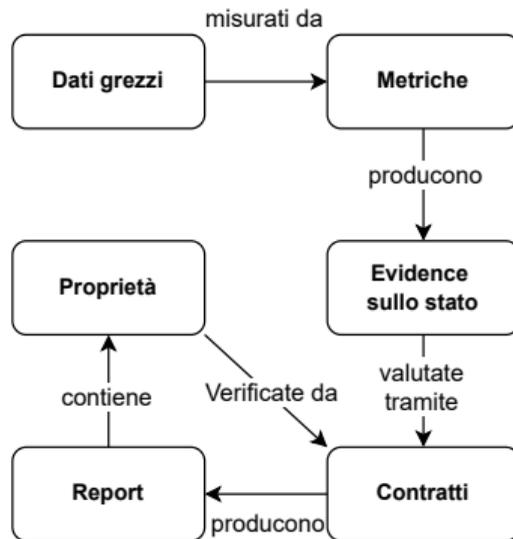
## Processo di Assurance e Proprietà

- Necessità di un nuovo **processo di assurance** per **valutare** sistemi distribuiti su **Proprietà Non Funzionali**
- **Verifiche di Assurance**: valutare se un sistema **soddisfa i contratti** definiti, implicando la **garanzia di proprietà** specifiche
  - Valutazione **continuativa**
  - Valutazione **su richiesta**
- **Proprietà**: definiscono delle caratteristiche **comportamentali** del sistema (*Scalabilità, Affidabilità, Confidenzialità, Integrità, Disponibilità, ...*)



# Metodologia di Assurance

- **Metriche:** misurano gli **stati** rilevanti del sistema
- **Evidence:** **dati** raccolti dalle metriche, utilizzati come **prove per i contratti**
- **Contratti:** **verificano** formalmente la validità di **PNF**
- **Report:** lista delle **proprietà valutate** con il **relativo esito** e le evidence a supporto





## Assurance Engine ed Elasticsearch

- L'Assurance Engine **implementa** il **processo di assurance** precedente **interagendo** con svariati **sistemi di monitoring** (Elasticsearch, Prometheus)
- **API REST** esposte tramite OpenAPI per interagire con l'engine
- Integrazione di **Elasticsearch** con l'Assurance Engine come **strumento di monitoring**
  - Efficienza nella **raccolta continua** di dati, nella **ricerca** e nell'**analisi**



## Necessità e Vantaggi del Framework Intermediario

- **Estende** le capacità di Elasticsearch, utilizzando il linguaggio di programmazione **Rust**
- Infrastruttura **request-reflector**
- Fornisce **scalabilità** e **replicabilità senza coordinazione**:
  - Sistema deployato come **una o più** istanze
  - Istanze **intercambiabili** e **stateless** non necessitano di coordinazione aggiuntiva



# Funzionamento dell'Assurance Engine

1. **Richiesta** POST all'endpoint dell'Assurance Engine
2. Valutazione di un contratto in **tre fasi**:
  - 2.1 **Measurement collection**: interrogazione dei **servizi di monitoring**
  - 2.2 **Contract evaluation**: valutazione dei **contratti**
  - 2.3 **Report composition**: restituzione del **risultato** della valutazione



## Valutazione sperimentale

- Benchmark **performance** esecuzione metriche e contratti
- Analisi in diverse circostanze
- Risultati:
  - **efficienza** nella **raccolta** e **valutazione** di evidenze tramite *metriche e contratti*
  - **latenza** di rete



## Conclusioni

- Metodologia innovativa di Assurance continuativa
- Estensione Assurance Engine integrandolo con Elasticsearch
- Implementazione di metriche e contratti formali
- Risultati sperimentali confermano l'efficienza del sistema

Fornisce complessivamente un sistema generico, flessibile e specializzabile per verifiche di assurance





## Lavori futuri

- **Integrazione** con ulteriori tecnologie di monitoraggio
- Sviluppo di nuovi **contratti** più complessi
- Applicazione del sistema ad **ambienti in produzione**





# Design e Sviluppo di una Soluzione per la Valutazione di Sistemi Distribuiti

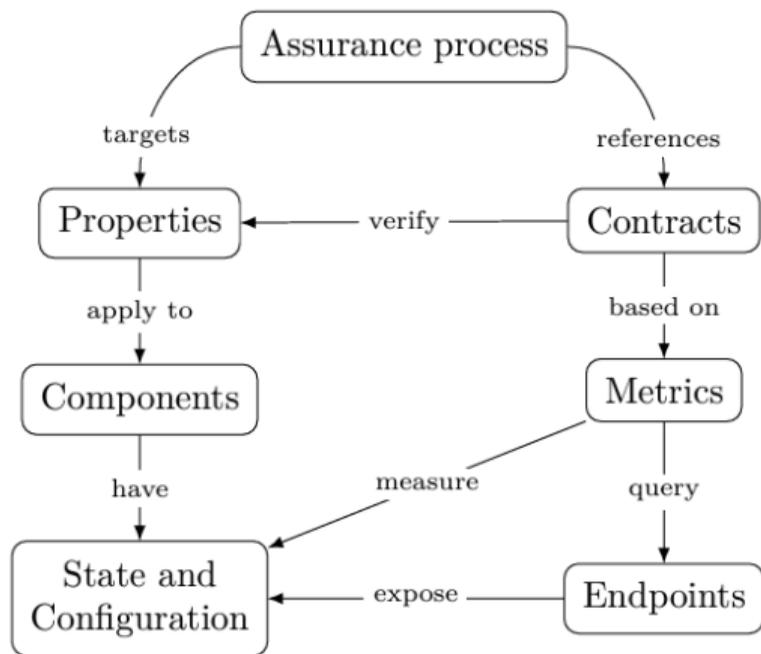
*Grazie per l'ascolto! Qualche domanda?*

**Enea Manzi: [enea.manzi@studenti.unimi.it](mailto:enea.manzi@studenti.unimi.it)**



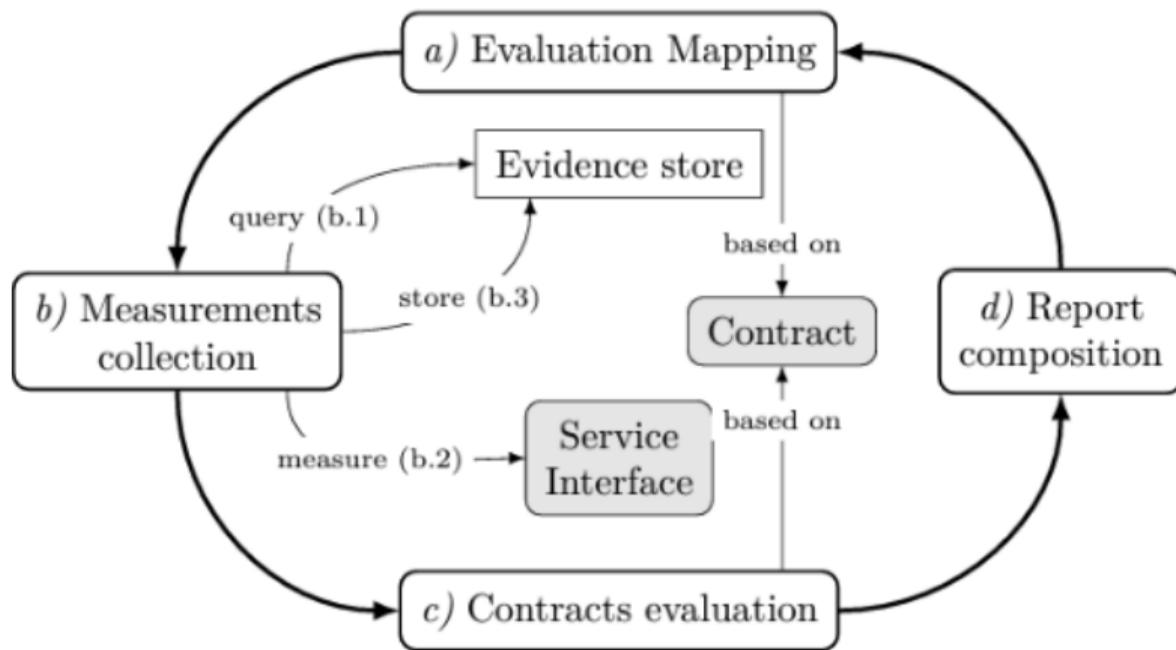


# Assurance methodology



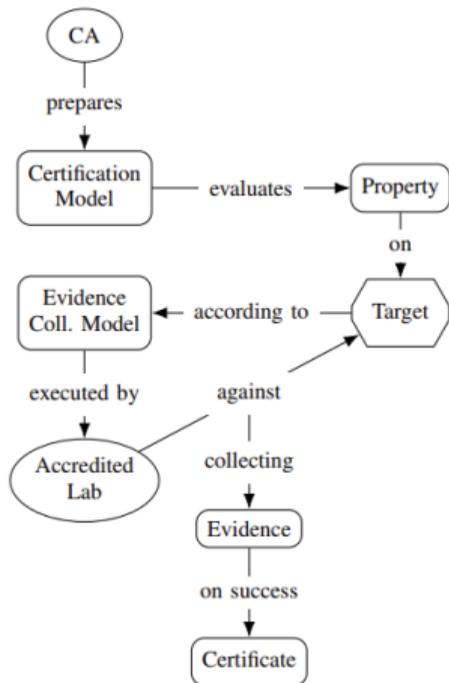


## Assurance process



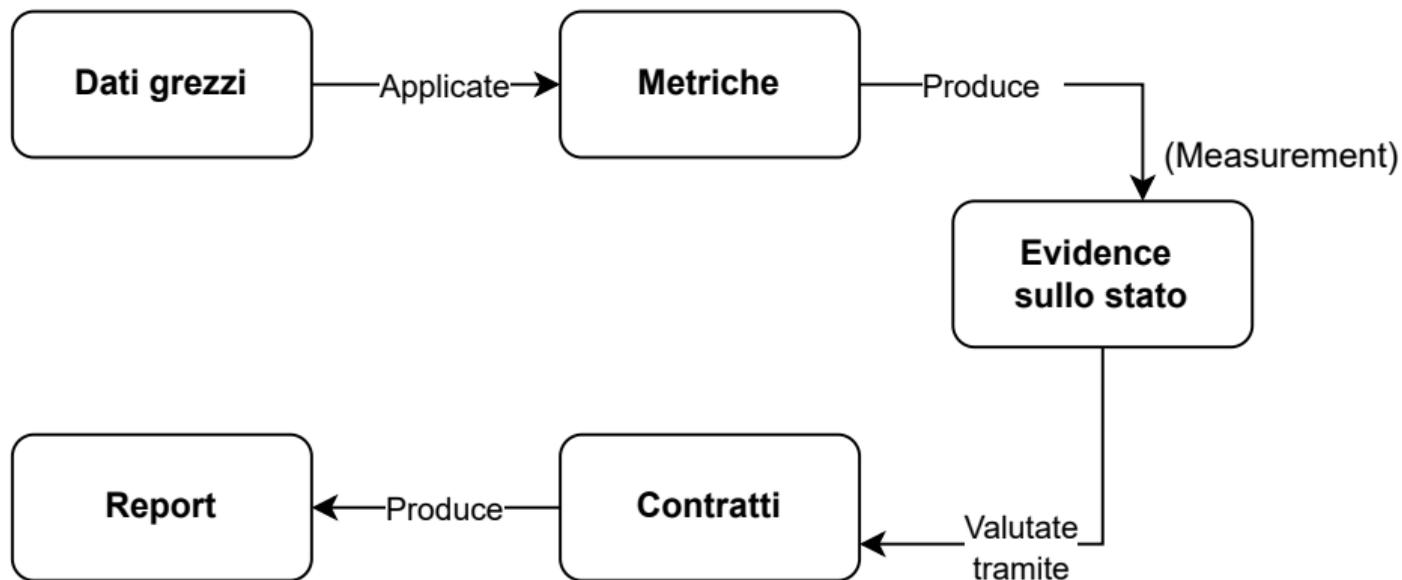


# Certification process



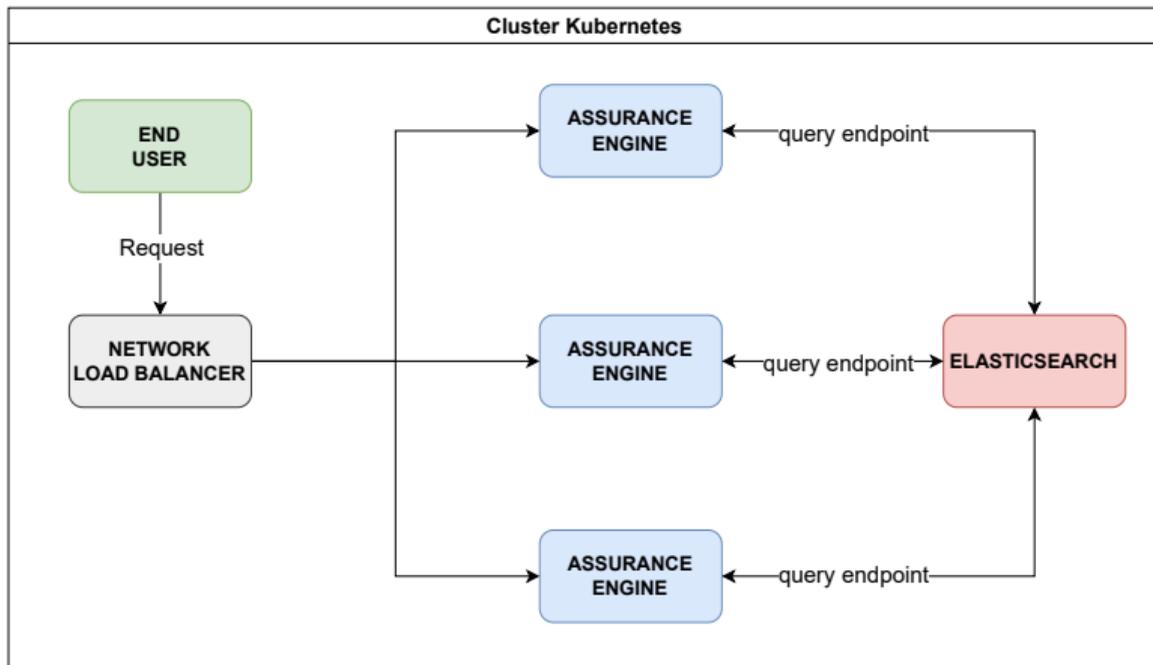


## Diagramma funzionamento



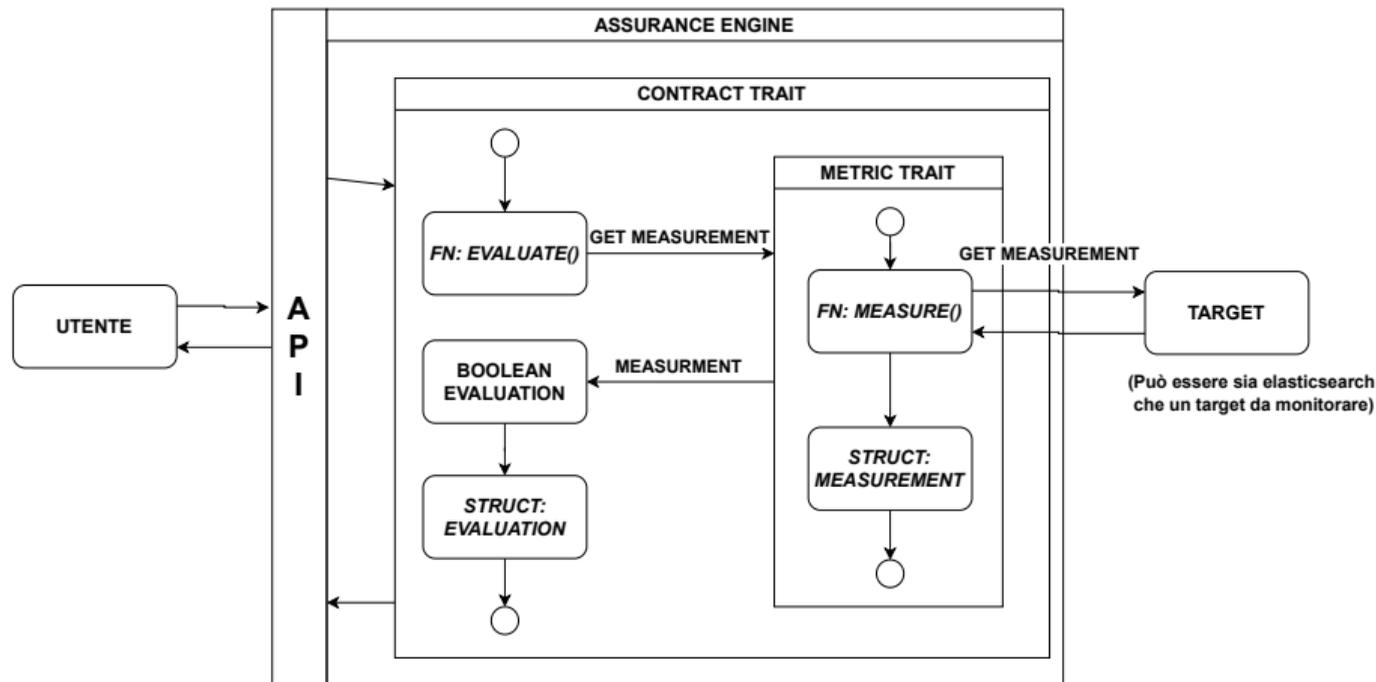


## Diagramma AE deployato su 3 nodi





## AE flusso di esecuzione





# Swagger richieste POST

**POST** /elastic\_basic\_query Elastic basic query measure ^

**Parameters** Try it out

No parameters

**Request body** required application/json; charset=utf-8 ▾

**Example Value** | Schema

```
{
  "query": "string",
  "index": [
    "string"
  ],
  "from": 0,
  "size": 0
}
```



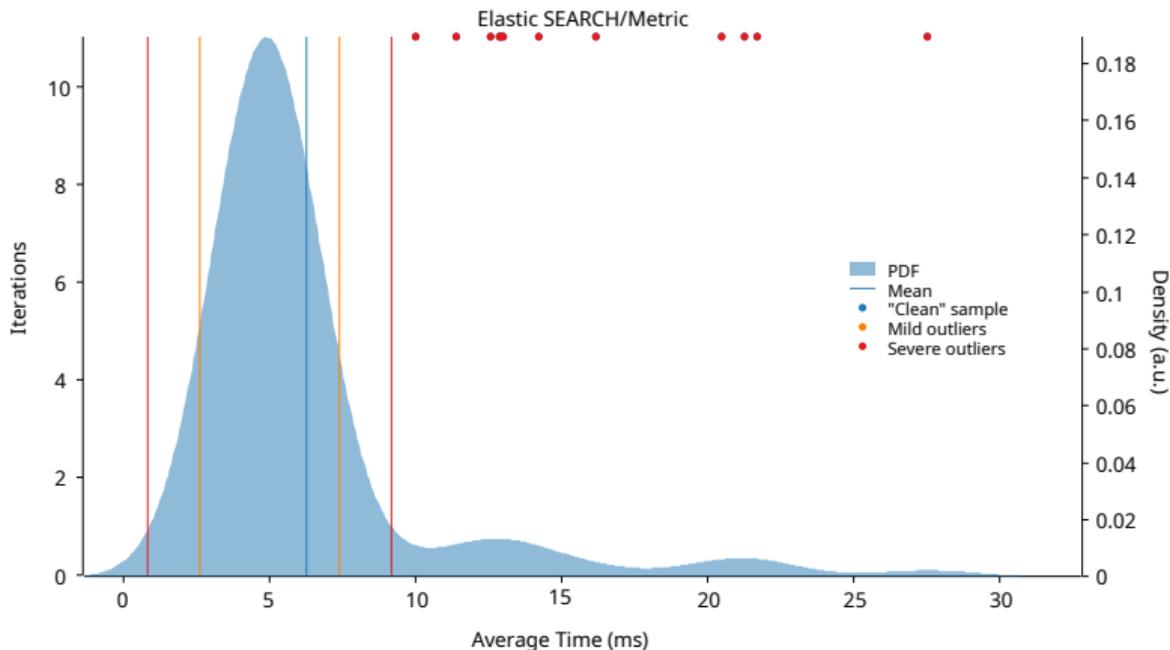
## Swagger schema JSON richieste POST

```
BasicSearchArgs {
  description: SEARCH BASIC generic types definition for args
  query*
  index*
  from
  size
}
```

```
BasicSearchOutput {
  description: SEARCH BASIC, PAGE, SQL generic types definition for output
  output*
}
```

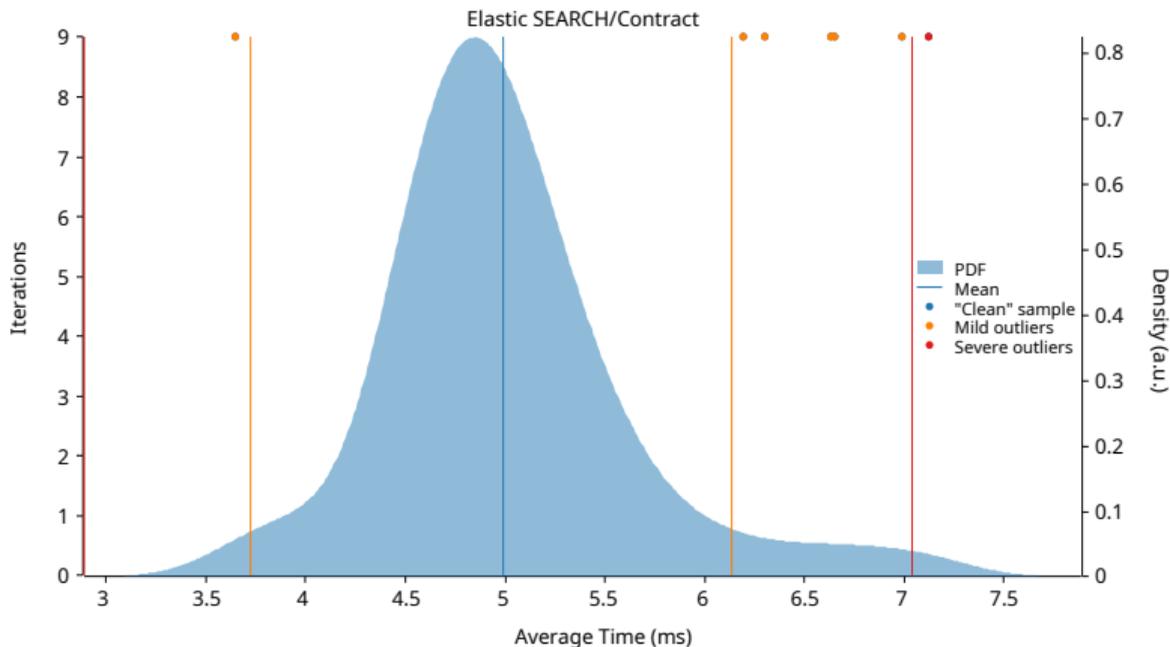


# Diagramma densità probabilistica richieste measure di tipo SEARCH



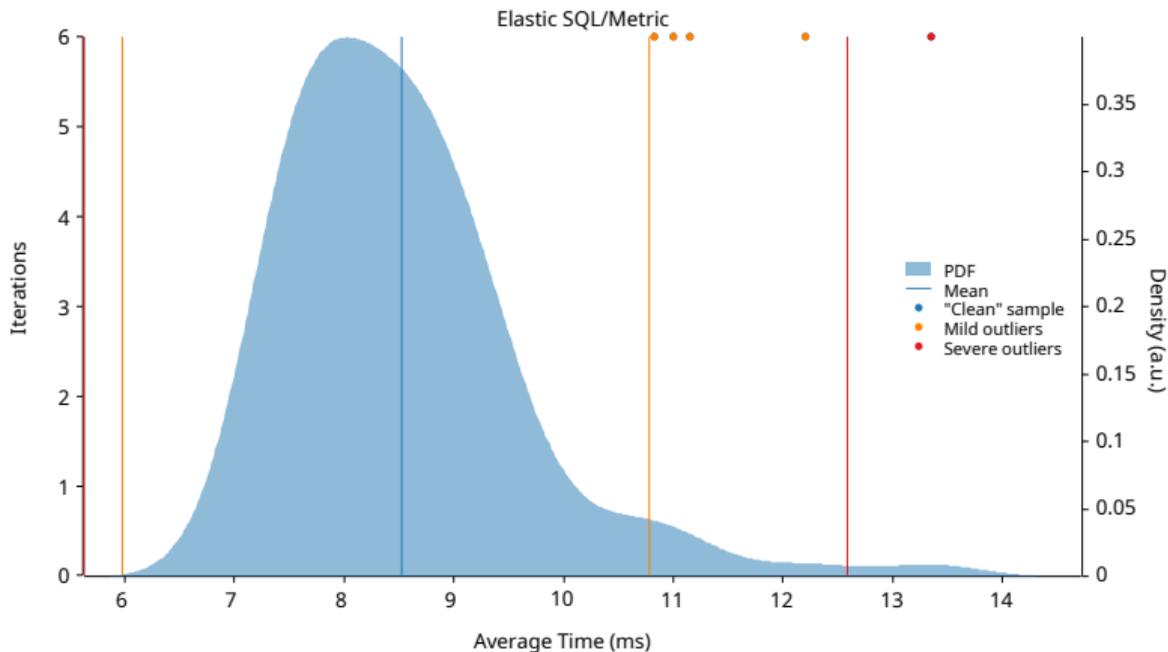


## Diagramma densità probabilistica richieste evaluate di tipo SEARCH





## Diagramma densità probabilistica richieste measure di tipo SQL





# Diagramma densità probabilistica richieste evaluate di tipo SQL

