

## **Sistema di monitoraggio dei flussi giudiziari\***

---

Sviluppato dall'Università degli Studi del Sannio nell'ambito del progetto:

MODELLI ORGANIZZATIVI E INNOVAZIONE DIGITALE: IL NUOVO UFFICIO  
PER IL PROCESSO PER L'EFFICIENZA DEL SISTEMA-GIUSTIZIA

\*Eugenio Marotti, Francesco C. Mazzitelli, Achille Melillo, Davide Paragona, Cristina Stelletti,  
Giancarlo Tretola, Eugenio Zimeo

*alpha release: 1.0.1*

## Sommario

<b>Indice delle figure.....</b>	<b>4</b>
<b>Indice delle tabelle.....</b>	<b>5</b>
<b>Introduzione.....</b>	<b>6</b>
Scopo del documento.....	6
<b>Descrizione generale.....</b>	<b>7</b>
Scopo del prodotto.....	7
Classi utente e caratteristiche.....	7
Ambiente operativo.....	7
Vincoli di progettazione e implementazione.....	8
Documentazione utente.....	8
Assunzioni e dipendenze.....	8
Dizionario.....	8
<b>Requisiti funzionali.....</b>	<b>9</b>
<b>Monitoraggio dei flussi giudiziari mediante cruscotti.....</b>	<b>9</b>
Utente: Giudice.....	9
Utente: Presidente di Sezione.....	11
Utente: Presidente del Tribunale.....	12
<b>Generazione di report.....</b>	<b>14</b>
<b>Previsione degli andamenti.....</b>	<b>17</b>
<b>Requisiti di interfaccia esterna.....</b>	<b>18</b>
<b>Requisiti non funzionali.....</b>	<b>19</b>
<b>Architettura software.....</b>	<b>20</b>
<b>Paradigma di riferimento: le architetture per Big Data.....</b>	<b>20</b>
Architettura del sistema HYPERION IUS.....	22
Componenti del sistema.....	23
Dispiegamento del Sistema.....	24
<b>Flusso dei dati e funzionalità.....</b>	<b>25</b>
Eventi nel sistema SICID.....	26
Stream Processing.....	28
Funzione di rischio.....	32
<b>Avvisi temporali e threshold.....</b>	<b>33</b>
<b>Avvisi di sfioramento e threshold.....</b>	<b>33</b>
<b>Modello previsionale del ritardo.....</b>	<b>33</b>
<b>Tecnologie utilizzate.....</b>	<b>34</b>
Debezium.....	34
Apache Kafka.....	35
Apache Flink.....	35
Opensearch.....	36
Opensearch Dashboard.....	36
<b>Integrazione del prototipo nei sistemi IT della giustizia.....</b>	<b>36</b>

<b>Manuale d'uso.....</b>	<b>38</b>
<b>Installazione.....</b>	<b>38</b>
Requisiti di sistema.....	38
Installazione del software.....	38
Configurazione del software.....	39
Aggiornamento del software.....	42
Disinstallazione del software.....	43
<b>Guida all'uso.....</b>	<b>43</b>
Accesso al software.....	43
Interfaccia utente.....	44
Funzionalità di base.....	44
Funzionalità avanzate.....	55
Disconnessione dal software.....	57
Interfaccia utente - Visualizzazione dei report.....	58
Disconnessione dal software - Visualizzazione dei report.....	61
<b>Appendice.....</b>	<b>62</b>
<b>Architetture Lambda.....</b>	<b>62</b>
<b>Architetture Kappa.....</b>	<b>63</b>
<b>Change Data Capture.....</b>	<b>64</b>
<b>Considerazioni sull'impatto della riforma Cartabia sul sistema di monitoraggio.....</b>	<b>65</b>
Aspetti della riforma.....	65
Impatti della riforma Cartabia sul monitoraggio.....	66
Adeguamenti per il monitoraggio dei processi aderenti alla riforma.....	67

## Indice delle figure

Figura 1 Componenti logici architettura per Big Data	19
Figura 2 Architettura di Hyperion IUS	20
Figura 3 Diagramma dei componenti	21
Figura 4 Diagramma di deploy	22
Figura 5 Flussi dati	23
Figura 6 Task dello stream processing	26
Figura 7 Modello dati dello stream di Fascicoli ed Eventi	27
Figura 8 Dato prodotto dal task di Join	28
Figura 9 Dato prodotto dal task Phase Duration Processing	28
Figura 10 Indirizzo di Hyperion IUS	40
Figura 11 Pagina di autenticazione	40
Figura 12 Pagina di selezione delle viste	41
Figura 13 Grafici per il cruscotto Tempestività di giudizio (giudice)	42
Figura 14 Grafici per il cruscotto Carico di giustizia (giudice)	43
Figura 15 Grafici per il cruscotto Tempestività di giudizio (presidente di sezione)	44
Figura 16 Grafici per il cruscotto Tempestività di giudizio (presidente di sezione)	45
Figura 17 Grafici per il cruscotto Carico di giustizia (presidente di sezione)	47
Figura 18 Grafici per il cruscotto Tempestività di giudizio (presidente tribunale)	49
Figura 19 Grafici per il cruscotto Carico di giustizia (presidente tribunale)	51
Figura 20 Esempio di informazioni avanzate	52
Figura 21 Esempio di informazioni avanzate	53
Figura 22 Accesso alla modalità full screen	53
Figura 23 Impostazione di un filtro	54
Figura 24 Risultato dell'applicazione del filtro	54
Figura 25 Esecuzione del log out	54
Figura 26 Esecuzione del log out	
Figura 27 Schermata principale	
Figura 28 Selezione di uno specifico range temporale	
Figura 29 Visualizzazione report disponibili	
Figura 30 Visualizzazione dettagli del report - Magistrato	
Figura 31 Visualizzazione dettagli del report - Presidente di Sezione	
Figura 32 Visualizzazione dettagli del report - Presidente di Tribunale	
Figura 33 Esecuzione del log out	55
Figura 34 Architettura lambda	56
Figura 35 Architettura kappa	57
Figura 36 CDC con tecnica basata sui log	59

## Indice delle tabelle

Tabella 1 Descrizione campi	12
Tabella 2 Esempio di report periodico del magistrato	12
Tabella 3 Descrizione campi	13
Tabella 4 Esempio di report periodico del Presidente di Sezione (Versione 1)	13
Tabella 5 Descrizione campi	14
Tabella 6 Esempio di report periodico del Presidente di Sezione (Versione 2)	14
Tabella 7 Descrizione campi	14
Tabella 8 Esempio di report periodico del Presidente di Tribunale	15

## Introduzione

### Scopo del documento

La presenza di un deficit di conoscenza, rappresentato dalla carenza di statistiche sul funzionamento degli uffici giudiziari, raccolte in modo continuo e capillare, costituisce un importante ostacolo ad una programmazione efficiente del servizio giudiziario soprattutto in relazione alla necessità di prendere decisioni sugli assetti organizzativi in presenza di cambiamenti come quelli indotti dall'introduzione degli Uffici per il Processo (UPP).

Occorre, quindi, far sì che le statistiche giudiziarie non restino più un oggetto evanescente e inaffidabile, ma diventino uno strumento di conoscenza completo della situazione organizzativa di ciascun ufficio e dell'attività espletata da ciascun magistrato.

Si è, quindi, ritenuto opportuno sviluppare un sistema di monitoraggio informatico dei flussi giudiziari, di seguito denominato **HYPERION IUS**, in modo da fornire un insieme di indicatori che possano rendere conto in tempo reale degli andamenti delle principali variabili che intervengono direttamente o indirettamente sull'andamento dei processi e più in generale sulle prestazioni del sistema giustizia nel suo complesso. Infatti, la complessità del sistema giudiziario, come di altre organizzazioni complesse, non consente di assumere come punto di riferimento un unico indicatore; diversi, infatti, sono i fattori che condizionano la gestione e la durata dei processi, sia dal punto di vista funzionale (es. attività dei giudici, degli avvocati e delle cancellerie), sia dal punto di vista strutturale (es. dotazione organica "reale" degli uffici).

Nel documento viene quindi presentato il sistema HYPERION IUS, partendo dai requisiti funzionali che hanno guidato lo sviluppo del software, per poi passare all'analisi dell'architettura software scelta per il sistema. Infine, viene fornita una guida all'utilizzo dello strumento.

## Descrizione generale

### Scopo del prodotto

Il sistema è stato concepito con l'obiettivo di agevolare l'accesso alle informazioni di natura statistica sui flussi giudiziari, fornendo agli utenti, in base alla loro tipologia (Giudici, Presidenti di Sezione e Presidenti di Tribunale), una serie di indicatori grafici raggruppati in cruscotti che consentono di monitorare in tempo reale l'andamento dei singoli fascicoli, di ottenere una serie di metriche aggregate e di confrontare tra loro i vari andamenti anche dal punto di vista storico.

La versione corrente del prototipo è pensata per i procedimenti civili ed in particolare per quelli che rientrano nel modello dei procedimenti ordinari a cognizione piena.

Il sistema offre, anche, la possibilità di ottenere report periodici sugli andamenti delle varie controversie e di essere tempestivamente informati qualora vi sia un'alta probabilità di ritardo nella trattazione dei fascicoli ai fini della loro definizione.

### Classi utente e caratteristiche

Il sistema prevede quattro tipologie di utenti:

- **Amministratore:** è l'utente che ha il compito di configurare i vari cruscotti e i vari indicatori messi a disposizione dal sistema;
- **Giudice:** utilizza il sistema per visualizzare l'andamento dei propri fascicoli e per confrontarli con gli andamenti aggregati della sua sezione e/o di fascicoli della stessa materia;
- **Presidente di Sezione:** utilizza il sistema per visualizzare l'andamento dei fascicoli in carico alla sua sezione di competenza e dei relativi indicatori;
- **Presidente del Tribunale:** utilizza il sistema per visualizzare l'andamento dei fascicoli dell'intero Tribunale e dei relativi indicatori.

### Ambiente operativo

HYPERION IUS è costituito da un insieme di microservizi, ognuno dispiegabile mediante un container. Ai fini della scalabilità, si raccomanda l'adozione di ambienti per l'autoscaling, quali Kubernetes.

## Vincoli di progettazione e implementazione

Il vincolo principale è rappresentato dalla dipendenza di HYPERION IUS dal sistema SICID, sviluppato da un'azienda esterna. Tale vincolo è stato considerato fin dalla progettazione del sistema, prevedendo una tecnica ed una tecnologia poco invasive per l'aggancio al sistema esterno in caso di dispiegamento in esercizio.

## Documentazione utente

Sarà fornita una "Guida Operativa" che accompagnerà l'utilizzo dello strumento, consentendo agli utenti di usare le funzionalità in modo efficace e proficuo.

## Assunzioni e dipendenze

- Ambiente per il dispiegamento delle unità logiche necessarie per l'esecuzione dei servizi del sistema e l'eventuale esigenza di scaling;
- **Accesso alla base di dati del sistema SICID** per il recupero dei dati mediante Change Data Capture.
- **Accesso al sistema di gestione degli utenti del SICID** per il recupero delle informazioni sugli utenti. Nel caso il SICID utilizzi LDAP o altro sistema di gestione degli utenti, l'accesso sarà diretto e realizzato mediante tale modulo integrato in Keycloak; nel caso, venga usato un approccio integrato, gli utenti del SICID saranno mappati sugli utenti di Hyperion IUS mediante i loro identificativi.

## Dizionario

**Early threshold:** il valore per il quale l'utente si trova in una zona ad alto rischio di ritardo.

**Early warning:** avverte l'utente dell'opportunità di migliorare le proprie performance per evitare possibili ritardi nella risoluzione della controversia in questione.

**Late threshold:** il valore oltre il quale si contribuisce a causare ritardo.

**Avvisi di sfornamento:** notificano il magistrato dello sfornamento dell'early threshold o della late threshold.

**Avvisi temporali:** notificano il magistrato del raggiungimento del 25% o del 50% della late threshold.

**Tasso di smaltimento:** il numero dei fascicoli definiti nel periodo diviso per il numero dei fascicoli sopravvenuti sommato al numero dei fascicoli pendenti all'inizio del periodo.

**Clearance rate:** è il rapporto, espresso in termini percentuali, tra il numero dei fascicoli definiti nel periodo e il numero dei fascicoli sopravvenuti.

**Disposition time:** è il rapporto, espresso in termini percentuali, tra il numero dei fascicoli pendenti alla fine del periodo e il numero dei fascicoli definiti.

## Requisiti funzionali

Saranno ora analizzati i requisiti funzionali del software. Come detto in precedenza, scopo del prodotto è quello di: a) consentire il monitoraggio in tempo reale dei flussi giudiziari mediante un'interfaccia user friendly, b) di fornire report periodici ai diversi profili giudiziari in relazione alla propria area di competenza e, infine, c) di fornire la possibilità di ottenere una previsione degli andamenti mediante l'analisi dei dati storici.

Di seguito sono elencate le caratteristiche principali, divise per le tre macroaree discusse.

## Monitoraggio dei flussi giudiziari mediante cruscotti

Verranno ora presentate le funzionalità messe a disposizione da Hyperion IUS per facilitare il monitoraggio in tempo reale dei fascicoli. Come detto in precedenza, il monitoraggio avviene mettendo a disposizione degli utenti una modalità di accesso e lettura delle informazioni intuitiva, organizzata in **cruscotti** ossia un tipo di interfaccia utente grafica che fornisce viste a colpo d'occhio di una serie di indicatori di prestazione. Ogni cruscotto, infatti, contiene una serie di "widget", ognuno dedicato ad un particolare indicatore. I cruscotti sono divisi per tipologia di utente (Giudice, Presidente di Sezione e Presidente del Tribunale) e contengono informazioni personalizzate per ciascuno di essi.

Di seguito sono discusse nel dettaglio le funzionalità implementate in ciascun cruscotto.

### Utente: Giudice

#### Cruscotto: Tempestività del Giudizio

1. **Visualizzazione della media mobile della durata delle fasi dei fascicoli del giudice:**
  - a. Il giudice deve poter visualizzare, in maniera grafica, la durata media delle fasi dei propri fascicoli.
2. **Confronto della media mobile della durata delle fasi dei fascicoli del giudice con quella della sezione:**
  - a. Il giudice deve poter confrontare, in maniera grafica, la durata media delle fasi dei propri fascicoli con la durata media delle fasi dei fascicoli in carico alla sua sezione di appartenenza.

3. **Confronto della media mobile della durata delle fasi dei fascicoli del giudice con quella del tribunale:**
  - a. Il giudice deve poter confrontare, in maniera grafica, la durata media delle fasi dei propri fascicoli con la durata media delle fasi dei fascicoli in carico al tribunale di appartenenza.
4. **Visualizzazione della mediana della durata delle fasi dei fascicoli del giudice:**
  - a. Il giudice deve poter visualizzare, in forma grafica, la mediana della durata delle fasi dei propri fascicoli.
5. **Confronto tra la mediana della durata delle fasi dei fascicoli del giudice con quella del tribunale di appartenenza:**
  - a. Il giudice deve poter confrontare, in forma grafica, la mediana della durata delle fasi dei propri fascicoli con la mediana della durata delle fasi dei fascicoli del tribunale di appartenenza.
6. **Visualizzazione della media mobile della durata delle fasi dei fascicoli suddivisi per materia:**
  - a. Il giudice deve poter visualizzare, in forma grafica, la media mobile della durata delle fasi dei propri fascicoli raggruppati per materia.
7. **Confronto tra la media mobile della durata delle fasi dei fascicoli per materia con quella della sezione di appartenenza:**
  - a. Il giudice deve poter confrontare, in forma grafica, la media mobile della durata delle fasi dei propri fascicoli raggruppati per materia con la media mobile della durata delle fasi dei fascicoli della sua sezione, raggruppati per materia.
8. **Visualizzazione della media mobile della durata dello stato di quiescenza dei fascicoli del giudice**
  - a. Il giudice deve poter visualizzare, in maniera grafica, la durata media dello stato di quiescenza dei propri fascicoli.
9. **Confronto della media mobile della durata dello stato di quiescenza dei fascicoli del giudice con quella della sezione:**
  - a. Il giudice deve poter confrontare, in maniera grafica, la durata media dello stato di quiescenza dei propri fascicoli con la durata media dello stato di quiescenza dei fascicoli in carico alla sua sezione di appartenenza.

#### **Cruscotto: Carico di Giustizia**

1. **Visualizzazione del Clearance Rate per i fascicoli del giudice:**
  - a. Il giudice deve poter visualizzare, mediante grafici, il clearance rate calcolato prendendo in esame i suoi fascicoli.
2. **Visualizzazione del Disposition Time per i fascicoli del giudice:**
  - a. Il giudice deve poter visualizzare, mediante grafici, il disposition time calcolato prendendo in esame i suoi fascicoli.
3. **Visualizzazione del tasso di smaltimento:**
  - a. Il giudice deve poter visualizzare, mediante grafici, il tasso di smaltimento dei fascicoli da lui presi in carico.

## Utente: Presidente di Sezione

### Cruscotto: Tempestività del Giudizio

1. **Visualizzazione della distribuzione dei fascicoli per sezione all'interno del tribunale di appartenenza:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, la distribuzione dei fascicoli all'interno delle diverse sezioni del tribunale di appartenenza.
2. **Visualizzazione della media mobile della durata delle fasi dei fascicoli della sezione di appartenenza:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, la media mobile della durata delle fasi dei fascicoli in carico alla sezione da lui presieduta.
3. **Confronto tra la media mobile della durata delle fasi dei fascicoli della sezione con quella delle sezioni del tribunale di appartenenza:**
  - a. Il presidente di sezione deve poter confrontare, mediante grafici, la media mobile della durata delle fasi dei fascicoli della sua sezione con la media mobile della durata delle fasi dei fascicoli delle altre sezioni del suo tribunale di appartenenza.
4. **Visualizzazione della media mobile della durata delle fasi dei fascicoli della sezione di appartenenza raggruppati per materia:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, la media mobile della durata delle fasi dei fascicoli della sezione da lui presieduta, raggruppati per materia.
5. **Visualizzazione della media mobile della durata dei fascicoli della sezione di appartenenza raggruppati per materia:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, la media mobile della durata dei fascicoli della sua sezione, raggruppati per materia.
6. **Visualizzazione del numero di fascicoli assegnati ad un giudice:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, il numero dei fascicoli assegnati a ciascun giudice della sua sezione di appartenenza.
7. **Visualizzazione dei fascicoli critici della sezione:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, i fascicoli critici della sua sezione, ossia quelli più vicini al raggiungimento della durata media e quindi con maggiore probabilità di sfornamento.
8. **Visualizzazione della media mobile della durata dello stato di quiescenza dei fascicoli della sezione di appartenenza:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, la media mobile della durata dello stato di quiescenza dei fascicoli in carico alla sezione da lui presieduta.
9. **Confronto tra la media mobile della durata dello stato di quiescenza dei fascicoli della sezione con quella delle sezioni del tribunale di appartenenza:**
  - a. Il presidente di sezione deve poter confrontare, mediante grafici, la media mobile della durata dello stato di quiescenza dei fascicoli della sua sezione con

la media mobile della durata dello stato di quiescenza dei fascicoli delle altre sezioni del suo tribunale di appartenenza.

### **Cruscotto: Carico di Giustizia**

- 1. Visualizzazione del Clearance Rate (semestrale e annuale) per i fascicoli della sezione:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, il clearance rate calcolato prendendo in esame i fascicoli gestiti nella sua sezione di appartenenza.
- 2. Confronto tra il Clearance Rate della sezione con quello delle altre sezioni del tribunale:**
  - a. Il presidente di sezione deve poter confrontare, mediante grafici, il clearance rate della sua sezione con il clearance rate delle altre sezioni del tribunale.
- 3. Visualizzazione del Disposition Time (semestrale e annuale) per i fascicoli della sezione:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, il disposition time calcolato prendendo in esame i fascicoli gestiti nella sua sezione di appartenenza.
- 4. Confronto tra il Disposition Time della sezione con quello delle altre sezioni del tribunale:**
  - a. Il presidente di sezione deve poter confrontare, mediante grafici, il disposition time della sua sezione con il clearance rate delle altre sezioni del tribunale.
- 5. Visualizzazione del tasso di smaltimento (semestrale e annuale):**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, il tasso di smaltimento dei fascicoli in carico alla sezione.

### **Utente: Presidente del Tribunale**

#### **Cruscotto: Tempestività del Giudizio**

- 1. Visualizzazione della media mobile della durata delle fasi dei fascicoli del tribunale:**
  - a. Il presidente del tribunale deve poter visualizzare, mediante grafici, la media mobile della durata delle fasi dei fascicoli trattati nel suo tribunale.
- 2. Confronto tra la media mobile della durata delle fasi dei fascicoli del tribunale con quella dei fascicoli degli altri tribunali del distretto:**
  - a. Il presidente del tribunale deve poter confrontare, mediante grafici, la media mobile della durata delle fasi dei fascicoli del tribunale da lui presieduto con la media mobile della durata delle fasi dei fascicoli degli altri tribunali del distretto.
- 3. Visualizzazione del numero dei fascicoli per sezione:**
  - a. Il presidente del tribunale deve poter visualizzare, mediante grafici, il numero dei fascicoli trattati da ciascuna sezione del suo tribunale.

4. **Confronto tra il numero dei fascicoli per sezione con il numero dei giudici di una sezione:**
  - a. Il presidente del tribunale deve poter confrontare, mediante grafici, il numero dei fascicoli trattati da una sezione del suo tribunale con il numero dei giudici in forza in quella sezione.
5. **Visualizzazione della media mobile della durata delle fasi dei fascicoli raggruppata per sezione:**
  - a. Il presidente del tribunale deve poter visualizzare, mediante grafici, la media mobile della durata delle fasi dei fascicoli per ciascuna delle sezioni del suo tribunale.
6. **Visualizzazione dei fascicoli critici del tribunale:**
  - a. Il presidente di sezione deve poter visualizzare, mediante grafici, i fascicoli critici del suo tribunale di appartenenza, ossia quelli più vicini al raggiungimento della durata media e quindi con maggiore probabilità di sfioramento.
7. **Visualizzazione della media mobile della durata dello stato di quiescenza dei fascicoli del tribunale:**
  - a. Il presidente del tribunale deve poter visualizzare, mediante grafici, la media mobile della durata dello stato di quiescenza dei fascicoli trattati nel suo tribunale.
8. **Confronto tra la media mobile della durata dello stato di quiescenza dei fascicoli del tribunale con quella dei fascicoli degli altri tribunali:**
  - a. Il presidente del tribunale deve poter confrontare, mediante grafici, la media mobile della durata dello stato di quiescenza dei fascicoli del tribunale da lui presieduto con la media mobile della durata dello stato di quiescenza dei fascicoli degli altri tribunali.

#### **Cruscotto: Carico di Giustizia**

1. **Visualizzazione del Clearance Rate (semestrale e annuale) per i fascicoli del tribunale:**
  - a. Il presidente del tribunale deve poter visualizzare, mediante grafici, il clearance rate calcolato prendendo in esame i fascicoli gestiti nel tribunale da lui presieduto.
2. **Confronto tra i Clearance Rate delle sezioni del tribunale:**
  - a. Il presidente del tribunale deve poter confrontare, mediante grafici, i clearance rate delle diverse sezioni del suo tribunale.
3. **Visualizzazione del Disposition Time (semestrale e annuale) per i fascicoli del tribunale:**
  - a. Il presidente del tribunale deve poter visualizzare, mediante grafici, il disposition time calcolato prendendo in esame i fascicoli gestiti nel tribunale da lui presieduto.
4. **Confronto tra i Disposition Time delle sezioni del tribunale:**
  - a. Il presidente del tribunale deve poter confrontare, mediante grafici, i disposition time delle diverse sezioni del suo tribunale.
5. **Visualizzazione del tasso di smaltimento (semestrale e annuale):**

- a. Il presidente del tribunale deve poter visualizzare, mediante grafici, il tasso di smaltimento dei fascicoli in carico all'intero tribunale.

## Generazione di report

### 1. Invio della notifica di disponibilità del report:

- a. Il sistema deve poter inviare, mediante diversi canali (es. SMS, email) un messaggio di notifica per informare l'utente interessato della disponibilità di un nuovo report mensile.

### 2. Generazione di un report mensile per il giudice:

- a. Il sistema deve generare, per il giudice, un report mensile in cui è riportata la situazione corrente del suo operato, ossia lo stato di avanzamento dei fascicoli trattati dallo specifico magistrato, con riferimento all'eventuale raggiungimento delle soglie prefissate o all'eventuale sfioramento dei limiti temporali superiori fissati per evitare di generare ritardo. La Tabella 1 riporta i campi contenuti nel report con relativa definizione, mentre la Tabella 2 contiene un esempio di report.

Tabella 1 Descrizione campi

<b>ID</b>	Codice identificativo del fascicolo
<b>Descrizione oggetto</b>	Descrizione dell'ambito relativo alla controversia
<b>Fase attuale</b>	Fase in cui si trova il fascicolo al momento corrente
<b>Periodo trascorso rispetto alla media</b>	Percentuale di avanzamento temporale della fase rispetto alla late threshold
<b>Messaggio</b>	Comunicazione informativa rispetto all'eventuale rischio di ritardo/ritardo generato

Tabella 2 Esempio di report periodico del magistrato

<b>ID</b>	<b>Descrizione oggetto</b>	<b>Fase attuale</b>	<b>Periodo trascorso rispetto alla media</b>	<b>Messaggio</b>
0105	Antitrust	Trattazione	> 25%	La fase sta procedendo regolarmente.
0072	Antitrust	Istruttoria	> 50%	<b>Warning!</b> La fase ha superato la early threshold. Rischio di ritardo elevato.

0060	Concorrenza sleale	Decisoria	> 100%	<b>Critical warning!</b> La fase ha superato la late threshold. Si sta generando ritardo.
0070	Diritto d'autore	Trattazione	> 100%	<b>Critical warning!</b> La fase ha superato la late threshold. Si sta generando ritardo.

### 3. Generazione di un report mensile al presidente di sezione:

- a. Il sistema deve generare, per presidente di sezione, un report mensile in cui è riportata la situazione corrente all'interno della sua sezione. Sono state ipotizzate due versioni di tale report:
  - i. Nella versione 1 è riportata la situazione corrente dell'operato dei giudici interni e quindi l'andamento delle controversie afferenti alle materie trattate nella specifica sezione effettuando **un'aggregazione per materia** e riportando poi, in un campo apposito, le fasi e i codici oggetto critici, relativi alla specifica materia, che generano più ritardo. Nella Tabella 3 sono riportati i campi contenuti nel report con relativa definizione, mentre la Tabella 4 contiene un esempio di report.
  - ii. Nella versione 2 è riportata la situazione corrente dell'operato dei giudici interni e quindi l'andamento delle controversie afferenti alle materie trattate nella specifica sezione effettuando **un'aggregazione per codici oggetto** e riportando poi le fasi più critiche. Nella Tabella 5 sono riportati i campi contenuti nel report con relativa definizione, mentre la Tabella 6 contiene un esempio di report.

Tabella 3 Descrizione campi

<b>Materia</b>	Denominazione della materia (vengono riportate tutte le materie che afferiscono alla specifica sezione)
<b>Fascicoli in ritardo (%)</b>	Percentuale dei fascicoli in ritardo per tale materia
<b>Fasi che generano ritardo</b>	Fasi critiche nella generazione del ritardo
<b>Codici oggetto che generano ritardo</b>	Codici oggetto critici nella generazione del ritardo

Tabella 4 Esempio di report periodico del Presidente di Sezione (Versione 1)

Materia	Fascicoli in ritardo (%)	Fasi che generano ritardo	Codici oggetto che generano ritardo
Industriale	20%	Istruttoria, Decisoria	172011
Locazione	0%	-	-
Assicurazione	30%	Decisoria	140051, 140052

Tabella 5 Descrizione campi

<b>Codice oggetto</b>	Codice identificativo dell'ambito relativo alla controversia
<b>Descrizione codice oggetto</b>	Descrizione dell'ambito relativo alla controversia
<b>Fascicoli in ritardo (%)</b>	Percentuale dei fascicoli in ritardo per tale codice oggetto
<b>Fasi che generano ritardo</b>	Fasi critiche nella generazione del ritardo

Tabella 6 Esempio di report periodico del Presidente di Sezione (Versione 2)

<b>Codice oggetto</b>	<b>Descrizione codice oggetto</b>	<b>Fascicoli in ritardo (%)</b>	<b>Fasi che generano ritardo</b>
172011	Concorrenza sleale	20%	Istruttoria, Decisoria
170004	Marchio	0%	-
140051	Assicurazione contro i danni	10%	Decisoria
140052	Assicurazione sulla vita	20%	Decisoria
140032	Locazione di beni mobili	0%	-
144021	Risoluzione del contratto di locazione per inadempimento uso abitativo	0%	-

#### 4. Generazione di un report mensile per presidente del tribunale:

- a. Il sistema deve generare, per il presidente del tribunale, un report mensile in cui è riportata la situazione corrente dell'intero Ufficio giudiziario, quindi lo stato di avanzamento di tutte le sezioni del tribunale, con l'eventuale situazione di ritardo facendo riferimento alle materie e ai codici oggetto più critici. Nella Tabella 7 sono riportati i campi contenuti nel report con relativa definizione, mentre la Tabella 8 contiene un esempio di report.

Tabella 7 Descrizione campi

<b>ID</b>	Codice identificativo della sezione (vengono riportate tutte le sezioni trattate dal Tribunale specifico) <b>N.B.</b> Le sezioni sono ordinate in base al quantitativo di ritardo generato
<b>Ritardo generato (gg)</b>	Percentuale dei fascicoli in ritardo per tale materia
<b>Materie che generano ritardo</b>	Materie critiche nella generazione del ritardo
<b>Codici oggetto che generano ritardo</b>	Codici oggetto critici nella generazione del ritardo

Tabella 8 Esempio di report periodico del Presidente di Tribunale

ID	Ritardo generato (gg)	Materie che generano ritardo	Codici oggetto che generano ritardo
LA	1480	Assicurazione, Industriale	140051, 140052, 172011
01	502	Società	151999, 153112
AP	0	-	-

#### 5. Accesso allo storico dei report:

- a. Ciascun utente deve poter accedere allo storico dei report personali prodotti dal sistema, in modo da poterli visualizzare e scaricare sotto forma di documento.

## Previsione degli andamenti

#### 1. Stima dell'andamento dei fascicoli:

- a. Il sistema, sulla base dei dati storici, deve essere in grado di stimare l'andamento dei fascicoli ancora aperti, prevedendo per singolo tribunale:
  - i. le medie mobili esponenziali delle durate dei fascicoli futuri del singolo giudice data una materia specifica;
  - ii. le medie mobili esponenziali delle durate dei fascicoli futuri relativi alla sezione data una materia specifica;
  - iii. le medie mobili esponenziali delle durate dei fascicoli futuri relativi ad una sezione specifica;
  - iv. le medie mobili esponenziali delle durate dei fascicoli futuri relativi all'intero tribunale;
  - v. il clearance rate;
  - vi. il disposition time.
- b. Per richiedere e visualizzare i risultati della predizione verranno estesi i seguenti widget:
  - i. "Durata media nel tempo dei fascicoli" (per il Giudice);
  - ii. "Durata media nel tempo" (per il Presidente di Sezione);
  - iii. "Durata media nel tempo" (per il Presidente di Tribunale).

## Requisiti di interfaccia esterna

1. **Autenticazione degli utenti:** l'interfaccia deve fornire un form intuitivo per consentire ai diversi utenti di effettuare l'accesso al sistema fornendo le proprie credenziali.
2. **Visualizzazione dei cruscotti e dei grafici:** l'interfaccia deve consentire, a ciascuna tipologia di utente, di visualizzare il proprio cruscotto contenente i grafici a lui destinati.
3. **Visualizzazione dei dati previsionali sugli andamenti:** l'interfaccia deve consentire agli utenti di poter visualizzare in maniera intuitiva le previsioni effettuate dal sistema circa gli andamenti delle fasi dei fascicoli aperti. La visualizzazione dei dati previsionali può essere integrata nei grafici dei cruscotti, assumendo un orizzonte temporale di predizione.
4. **Visualizzazione dei report periodici:** tra le interfacce esterne vi deve essere la possibilità di poter visualizzare i report periodici.

## Requisiti non funzionali

1. **Sicurezza:**
  - a. Il software deve garantire la sicurezza delle informazioni personali dei vari utenti e dei loro dati sensibili.
  - b. L'accesso al sistema deve essere controllato tramite autenticazione degli utenti e gestione dei ruoli.
2. **Usabilità:**
  - a. L'interfaccia utente deve essere intuitiva, user-friendly e facile da utilizzare.
  - b. Deve assicurare che l'utente possa svolgere le attività in modo efficiente e senza difficoltà.
  - c. Le funzionalità principali devono essere facilmente accessibili e comprensibili per l'utente.
3. **Performance:**
  - a. Il software deve essere efficiente e responsivo per garantire tempi di risposta bassi o comunque tollerabili.
  - b. Deve essere progettato per gestire grandi quantità di dati senza rallentamenti significativi.
4. **Manutenibilità:**
  - a. Il software deve essere facilmente manutenibile e pensato per integrare facilmente eventuali aggiornamenti o modifiche future.
  - b. Il codice sorgente deve essere ben strutturato, documentato e aderente alle migliori pratiche di sviluppo.
5. **Affidabilità:**
  - a. Il software deve essere affidabile, garantendo che le informazioni sugli andamenti temporali dei fascicoli siano accurate e aggiornate.

## Architettura software

In questa sezione viene descritta l'architettura di HYPERION IUS, iniziando da una panoramica sulle principali architetture di elaborazione dei dati, in particolare di Big data, per poi passare alla descrizione dell'architettura individuata per il sistema e all'analisi delle principali tecnologie usate.

### Paradigma di riferimento: le architetture per Big Data

Un'architettura per Big Data è progettata per gestire l'inserimento, l'elaborazione e l'analisi di dati troppo grandi o complessi per i sistemi di database tradizionali. Il criterio adottato dalle organizzazioni per individuare la soglia oltre la quale è opportuno preferire soluzioni per Big Data varia a seconda delle capacità degli utenti e degli strumenti a disposizione. Può trattarsi di una quantità di dati dell'ordine di migliaia di gigabyte, ma anche di centinaia di terabyte. Con lo sviluppo di strumenti sempre più avanzati per la gestione di set di Big Data, anche il significato di Big Data si è evoluto. Il termine viene sempre più usato per fare riferimento al valore che è possibile estrarre dai set di dati tramite l'analisi avanzata, anziché esclusivamente alle dimensioni dei dati, anche se in questi casi tendono a essere molto grandi.

Nel corso degli anni, il panorama dei dati ha subito varie trasformazioni. Il concetto di ciò che è possibile fare, o si prevede di poter fare, è cambiato. I costi di archiviazione sono diminuiti notevolmente, mentre i mezzi disponibili per la raccolta dei dati sono in continua espansione. Alcuni dati arrivano a un ritmo veloce, sempre pronti per essere raccolti e osservati. Altri arrivano invece a un ritmo più lento, ma in blocchi di dimensioni molto grandi, spesso sotto forma di dati cronologici relativi a decenni. Può quindi capitare di dover affrontare un problema che richiede una soluzione di analisi avanzata o di apprendimento automatico. Queste sono le sfide che tentano di risolvere le architetture per Big Data.

Le soluzioni per i Big Data implicano in genere uno o più dei seguenti tipi di carico di lavoro:

- L'elaborazione batch di origini di Big Data inattivi.
- L'elaborazione in tempo reale di Big Data in movimento.
- L'esplorazione interattiva di Big Data.
- L'analisi predittiva e il Machine Learning.

Il diagramma seguente mostra i componenti logici relativi a un'architettura per Big Data. È comunque possibile che le singole soluzioni non contengano tutti gli elementi illustrati nel diagramma.

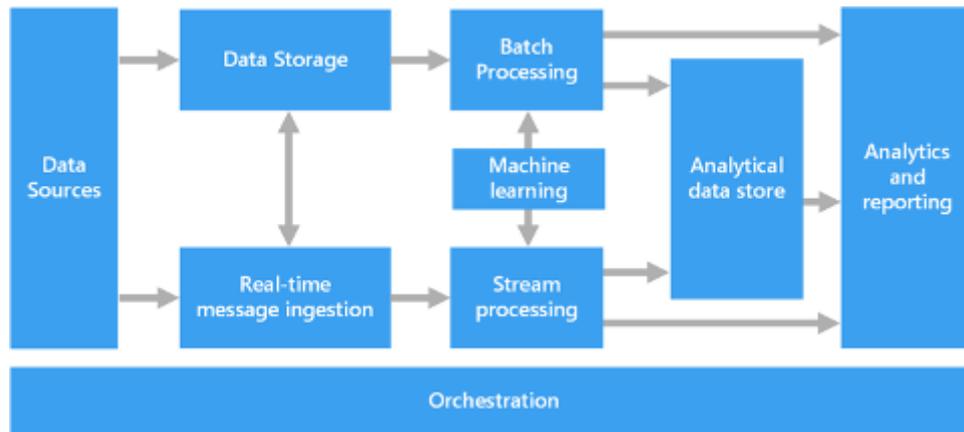


Figura 1 Componenti logici architettura per Big Data

La maggior parte delle architetture per i Big Data include alcuni o tutti i seguenti componenti:

- **Sorgenti dei dati.** Il punto di partenza di tutte le soluzioni per Big Data è costituito da una o più sorgenti dei dati. Tra gli esempi sono inclusi:
  - Archivi dati di applicazioni, ad esempio database relazionali.
  - File statici generati dalle applicazioni, ad esempio file di log di server Web.
  - Sorgenti di dati in tempo reale.
- **Archiviazione dei dati.** I dati per le operazioni di elaborazione batch vengono in genere inseriti in un archivio di file distribuito che può contenere volumi elevati di file di grandi dimensioni in vari formati. Questo tipo di archivio viene spesso chiamato data lake.
- **Elaborazione batch.** Poiché i set di dati hanno dimensioni considerevoli, una soluzione per Big Data deve spesso elaborare i file di dati mediante processi batch con esecuzione prolungata per filtrare, aggregare e preparare in altro modo i dati per l'analisi. In genere, questi processi prevedono la lettura dei file di origine, la relativa elaborazione, ad esempio mediante MapReduce, e la scrittura dell'output in nuovi file.
- **Inserimento di messaggi in tempo reale.** Se la soluzione include sorgenti di dati in tempo reale, l'architettura deve includere un modo per acquisire e archiviare i messaggi in tempo reale per l'elaborazione del flusso. Potrebbe trattarsi di un archivio dati semplice in cui i messaggi in ingresso vengono rilasciati in una cartella per l'elaborazione. Tuttavia, molte soluzioni richiedono che un archivio di inserimento dei messaggi funga da buffer per i messaggi e supporti l'elaborazione scale-out, il recapito affidabile e altri tipi di semantica di accodamento dei messaggi. Questa parte dell'architettura per l'elaborazione del flusso è spesso definita memorizzazione nel buffer del flusso.
- **Elaborazione del flusso.** Dopo avere acquisito i messaggi in tempo reale, la soluzione deve elaborarli filtrando, aggregando e preparando in altro modo i dati per l'analisi. I dati del flusso elaborati vengono quindi scritti in un sink di output.
- **Archivio dati analitici.** Numerose soluzioni per Big Data preparano i dati per l'analisi e quindi servono i dati elaborati in un formato strutturato su cui è possibile eseguire query con strumenti analitici. L'archivio dati analitici usato per rispondere a queste

query può essere un data warehouse relazionale in stile Kimball, come nella maggior parte delle soluzioni di business intelligence (BI) tradizionali. In alternativa, i dati possono essere presentati tramite una tecnologia NoSQL a bassa latenza come HBase o un database Hive interattivo che fornisce un'astrazione di metadati sui file di dati nell'archivio dati distribuito.

- **Analisi e creazione di report.** L'obiettivo della maggior parte delle soluzioni per Big Data è fornire informazioni dettagliate sui dati tramite strumenti di analisi e report.
- **Orchestrazione.** La maggior parte delle soluzioni per Big Data consiste in operazioni ripetute di elaborazione dei dati, incapsulate in flussi di lavoro, che trasformano i dati di origine, spostano i dati tra più origini e sink, caricano i dati elaborati in un archivio dati analitico o li inseriscono direttamente in un report o in una dashboard. Per automatizzare questi flussi di lavoro, è possibile usare una tecnologia di orchestrazione.

## Architettura del sistema HYPERION IUS

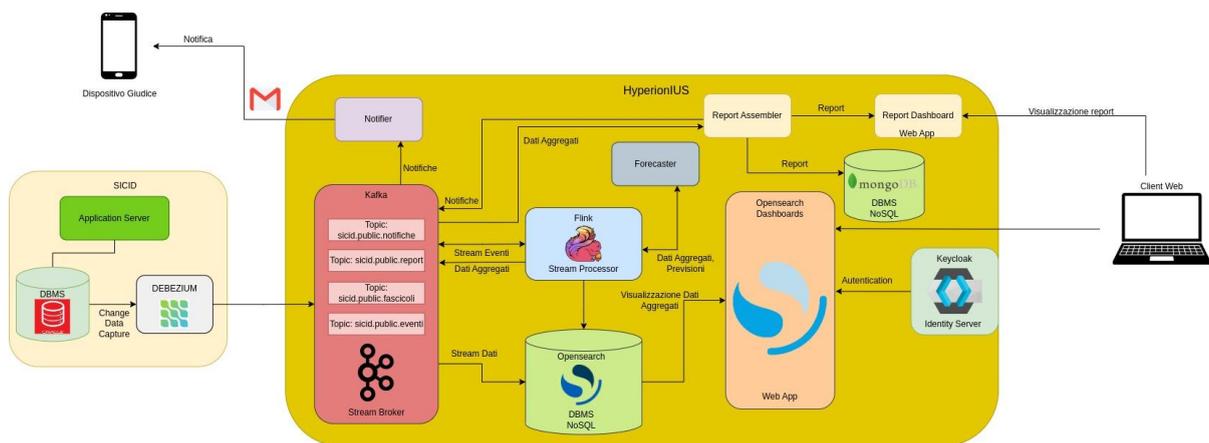


Figura 2 Architettura di HYPERION IUS

La Figura 2 mostra l'architettura di alto livello del sistema di monitoraggio e l'integrazione con il sistema SICID. La parte sinistra della figura riporta l'architettura di alto livello del sistema SICID. La parte destra della figura, invece, illustra l'architettura del sistema di monitoraggio. Come si vede, esso è composto da un *message broker* che ha il compito di instradare gli eventi generati all'interno del sistema SICID, mediante la cattura effettuata dal modulo di Change Data Capture (CDC) (si veda l'appendice per i dettagli), e di inoltrarli sia ad una base di dati per il loro salvataggio, sia ad un componente di stream processing che provvederà alla loro elaborazione (ad esempio per generare le aggregazioni necessarie alla valorizzazione degli indicatori proposti in precedenza) e al loro salvataggio nella base di dati. I dati prodotti, sia in forma base che in forma aggregata, saranno poi resi disponibili ai client che ne faranno richiesta, mediante un componente di accesso ai dati.

## Componenti del sistema

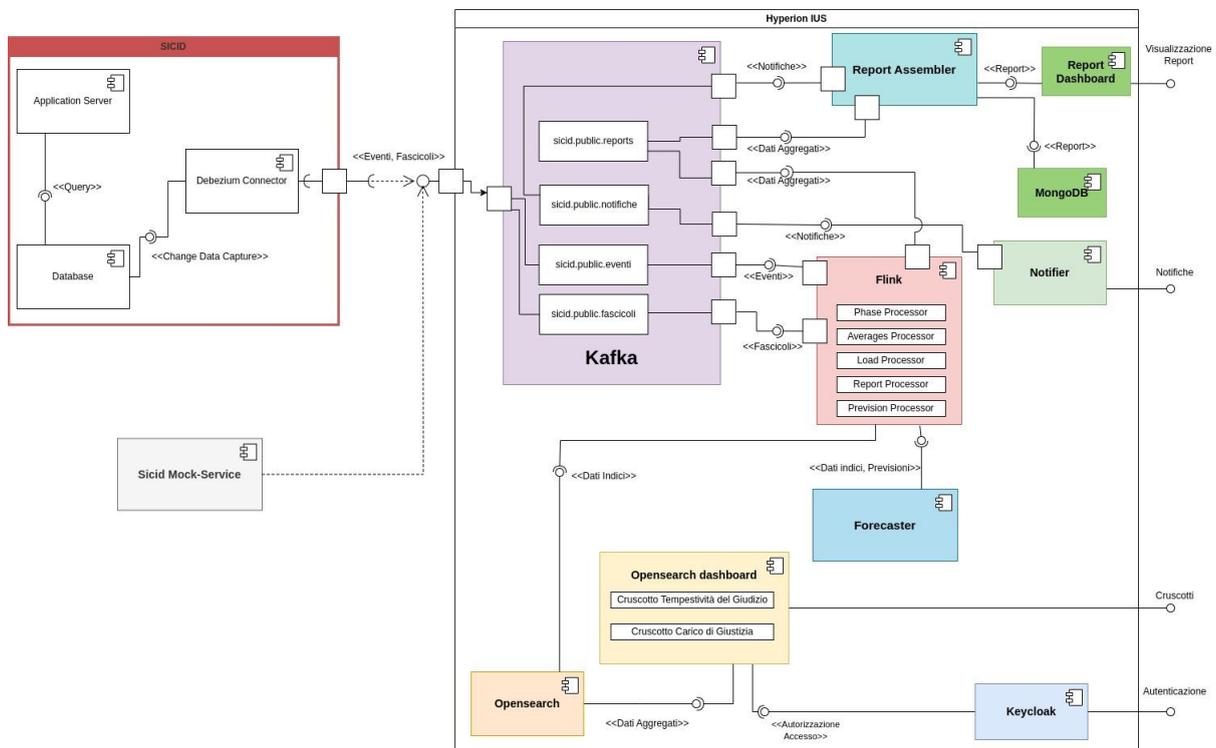


Figura 3 Diagramma dei componenti

Il sistema è composto da diversi componenti, le cui tecnologie, qui citate, verranno descritte nella sezione apposita di questo documento:

- **Change data capture:** permette ad HYPERION IUS di intercettare gli eventi generati dal SICID nel modo meno invasivo possibile, nello specifico mediante **Debezium** e un suo connettore da agganciare al DBMS del SICID;;
- **Stream Broker:** fornisce un sistema di smistamento messaggi all'interno di HYPERION IUS, gestendo code per le informazioni in ingresso al sistema ("sidic.public.eventi" e "sidic.public.notifiche"), ma anche code per le informazioni in uscita dal sistema ("sidic.public.notifiche"); è realizzato dallo stream broker **Kafka**;
- **Stream Processor:** recupera dallo stream broker le informazioni su fascicoli ed eventi di fascicoli, processando i dati ottenuti e inviando i risultati sia alla parte di visualizzazione del sistema, sia al sistema di notifica, ed è realizzato dallo stream processor **Flink**;
- **Report assembler:** utilizza i dati elaborati dallo stream processor per produrre i report periodici che saranno poi salvati e messi a disposizione mediante un'interfaccia Web;
- **Notifier:** è il componente che si occupa dell'invio delle notifiche, mediante più canali (es. SMS, email), sulla disponibilità di nuovi report;
- **Forecaster:** è il componente che si occupa di generare le previsioni sugli andamenti dei fascicoli, sfruttando il modello di previsione ARIMA;

- **DBMS NoSQL:** gestisce la conservazione e il recupero dei dati processati dallo stream processor e li mette a disposizione alla Web App per la visualizzazione. È realizzato con **Opensearch**;
- **Web App:** preleva i dati da mostrare dal DBMS NoSQL e li visualizza sui grafici relativi ai cruscotti di cui viene richiesta la visualizzazione dal Giudice che ha effettuato l'accesso al sistema con le sue credenziali. È realizzato con **Opensearch Dashboard**;
- **Identity & Access Manager:** gestisce gli accessi e i dati relativi agli account dei Giudici sul sistema di monitoraggio Hyperion. È realizzato da **Keycloak**.

Per garantire che il sistema di monitoraggio possa funzionare in qualsiasi fase del suo sviluppo, è stato utilizzato un ulteriore componente esterno, il **SICID mock-service**, che emula le funzionalità di interesse del SICID, operando quindi da driver e consentendo il testing incrementale dell'intera architettura, fornendo dati riguardanti fascicoli ed eventi in base alle esigenze di testing.

## Dispiegamento del Sistema

Il sistema è dispiegato in un ambiente virtualizzato, e containerizzato, ed è attualmente accessibile sia tramite VPN sia con indirizzo pubblico per finalità di testing pre-alpha anche da parte del personale dei tribunali. Ogni sottosistema costituente il sistema di monitoraggio HYPERION IUS è stato dispiegato in un proprio container, utilizzato come una macchina logica, assicurando l'utilizzo delle stesse tipologie di comunicazione che avverrebbero se i diversi sottosistemi fossero davvero dispiegati su macchine diverse.

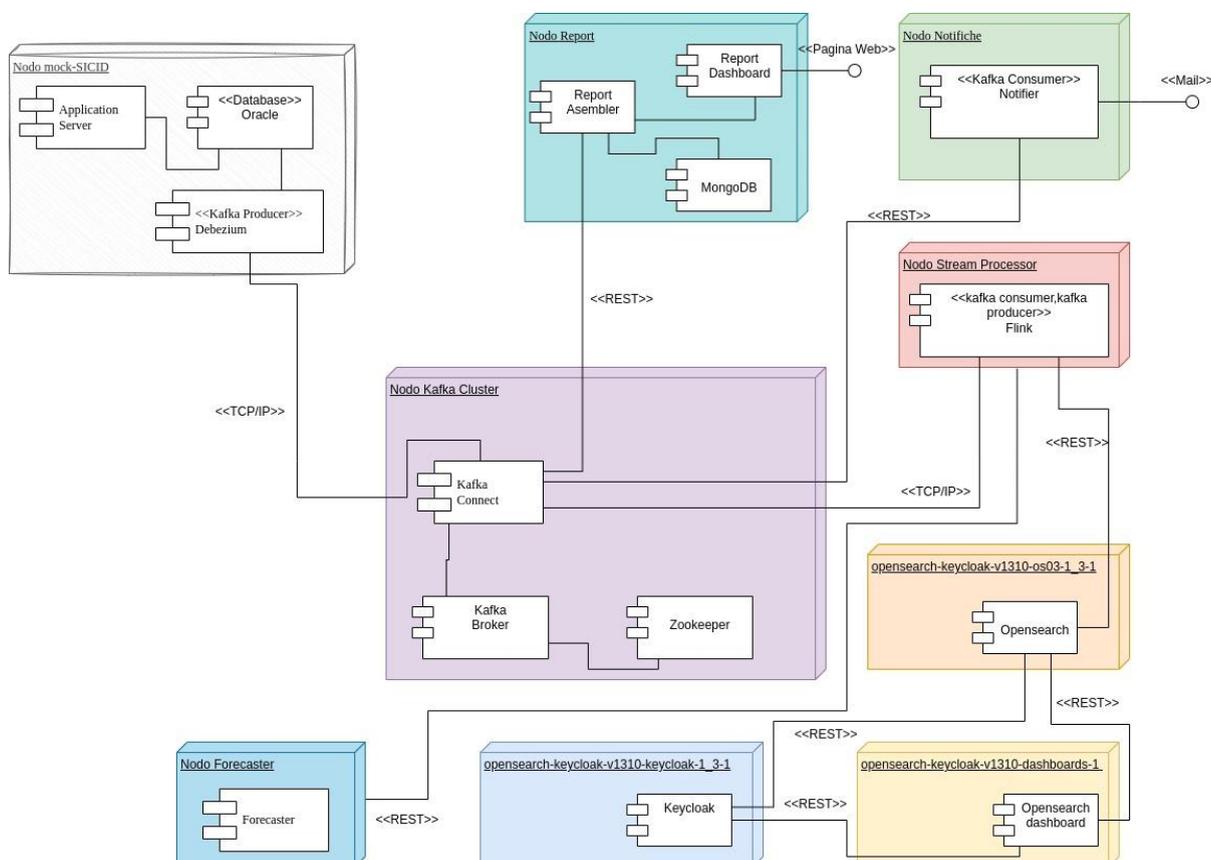


Figura 4 Diagramma di deploy

Nello schema di dispiegamento riportato in Figura 4, viene evidenziata la **containerizzazione in un nodo per ogni sottosistema** e vengono evidenziate le **tipologie di comunicazione** utilizzate tra i differenti sottosistemi di HYPERION IUS, durante il suo funzionamento.

## Flusso dei dati e funzionalità

I flussi di dati tra i componenti costituenti HYPERION IUS riguardano dati di diversa natura, necessari alla realizzazione delle diverse funzionalità fornite dal sistema.

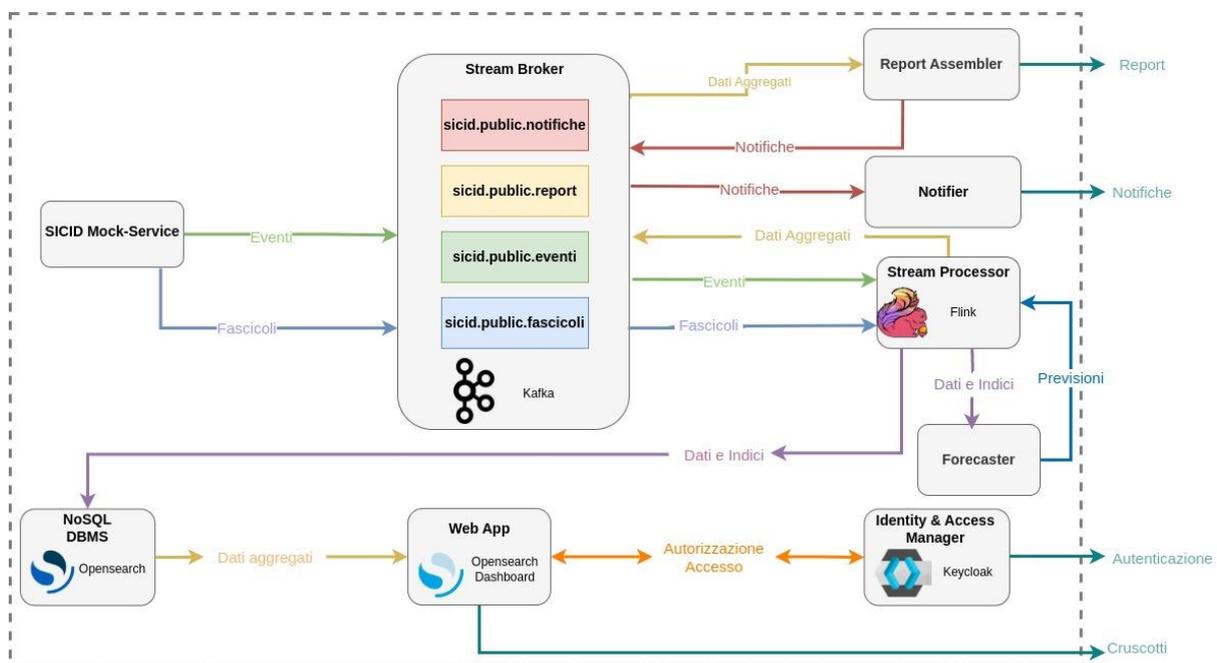


Figura 5 Flussi dati

I flussi di informazioni durante il funzionamento del sistema di monitoraggio prevedono i seguenti tipi di dati:

- **Fascicoli:** sono i dati principali di un fascicolo, che vengono definiti al momento della creazione del fascicolo e che cambiano poco durante il ciclo di vita di un processo. Essi comprendono nrg, oggetto, sezione, giudice e data di iscrizione a ruolo e sono rappresentati all'interno del sistema in formato json.
- **Eventi:** sono i dati che costituiscono la cronologia di azioni intraprese per una specifica controversia, conservate all'interno del database distrettuale a cui accede il SICID per recuperare i dati relativi ai fascicoli. Ogni evento è costituito da diverse coppie chiave-valore, tra cui data di avvenimento, data di registrazione, nome evento, descrizione evento e codice utente. Gli eventi di interesse per HYPERION IUS vengono rappresentati in formato json.
- **Notifiche:** sono i dati relativi ai report che vengono inviati via mail ai Giudici grazie al sottosistema Notifier di HYPERION IUS.
- **Dati e Indici:** sono i dati sulla base dei quali i grafici dei cruscotti vengono costruiti.

- **Aggregati:** sono i dati aggregati che vengono rappresentati dai grafici dei cruscotti.
- **Autorizzazione e Accesso:** sono i dati relativi all'autenticazione degli utenti.
- **Previsioni:** sono i dati relativi agli andamenti statistici predetti dal blocco Forecaster sulla base dei dati aggregati a sua disposizione.

L'intero flusso di dati segue un percorso ben preciso, che parte dagli eventi generati dal SICID, che per esigenze di implementazione è stato ricostruito in una versione semplificata identificata dal "SICID mock-service", per poi passare allo stream processor attraverso il message broker. A questo punto, i dati vengono elaborati e conservati all'interno del Database NoSQL di Opensearch, da cui verranno recuperati per essere visualizzati grazie ai grafici costituenti i cruscotti mediante Opensearch Dashboard.

Un ulteriore flusso dati riguarda lo scambio di informazioni tra lo stream processor e il modulo previsionale ARIMA (descritto nella sezione Modello ARIMA). Perché il modello possa effettuare una regressione necessita delle informazioni riguardanti le medie mobili esponenziali delle durate a) dei fascicoli relativi ad un singolo giudice e nell'ambito di una stessa materia b) dei fascicoli relativi ad una sezione data una materia specifica c) dei fascicoli relativi ad una sezione specifica. Tale scambio avviene mediante l'accesso condiviso a file CSV contenenti i dati di input. Le previsioni vengono prodotte periodicamente. Il risultato dell'elaborazione sarà poi salvato in un file CSV, contenente appunto le previsioni, e sarà letto dallo stream processor che provvede all'inoltro dei dati ad Opensearch.

## Eventi nel sistema SICID

Per poter elaborare correttamente gli eventi che si verificano durante il processo a cognizione ordinaria, in modo da sfruttarli per la produzione degli indicatori del sistema di monitoraggio in tempo reale visti in precedenza, occorre conoscere quali sono gli eventi gestiti dal sistema SICID. L'analisi del sistema effettuata a partire dalla fase di ricognizione e di un insieme di storici di eventi di fascicoli, forniti dal Tribunale di Benevento e di Avellino, tutti pre-riforma Cartabia, ha consentito di identificare una serie di tipologie di eventi che possono essere sfruttati per dividere i processi in fasi. Gli eventi individuati sono:

- Iscrizione a ruolo
- Assegnazione a sezione
- Assegnazione a Giudice
- Udienza
- Rinvio Udienza
- Differimento Udienza
- Deposito
- Costituzione (di attore o di convenuto)
- Apertura sub-processo
- Cambio di Giudice (per Surroga, per Ricusazione o per Sostituzione)
- Sospensione Fascicolo
- Riassunzione Fascicolo
- Rimessione Fascicolo
- Ritorno Fascicolo
- Trattazione Scritta

- Provvedimenti Fuori Udienza
- Nomina CTU
- Liquidazione CTU
- Revoca CTU
- Proroga
- Sentenza

Gli eventi di tipo Deposito sono ulteriormente divisi in:

- Deposito di memorie
- Deposito di atti
- Deposito di istanze di liquidazione CTU
- Deposito di richieste di comparizione di terzi
- Deposito di istanze di anticipazione o differimento udienza
- Deposito di atti di costituzione delle parti
- Deposito di consulenze CTU
- Deposito di verbali di udienza
- Deposito di consulenze
- Deposito di sentenze

Gli eventi sono stati quindi studiati con l'obiettivo di individuare quelli che operano di fatto come delimitatori delle fasi del processo ordinario e cioè:

- Fase Introduttiva
- Fase di Trattazione
- Fase Istruttoria
- Fase Decisoria

Di seguito vengono elencati gli eventi che determinano un effettivo passaggio da una di queste fasi alla successiva.

**Iscrizione a Ruolo:** indica l'inizio della fase introduttiva di un processo.

**Prima Udienza:** quando si tiene la prima udienza il processo entra nella fase di trattazione; dallo storico non viene generato un evento nel momento in cui viene tenuta la prima udienza di un processo, però viene generato un evento che specifica quando viene fissata la prima udienza per quel processo, comprendendo anche la data futura in cui questa dovrà avvenire.

**Udienza Istruttoria:** per quanto riguarda la fase istruttoria, può essere affrontata o meno a discrezione del Giudice. L'evento individuato per delineare l'inizio di tale fase è il primo evento presente nello storico che sia immediatamente successivo al deposito delle memorie (art. 183).

**Udienza di precisazione delle conclusioni:** questa udienza sancisce l'inizio della fase decisoria, ultima fase del processo. Come per la prima udienza, anche in questo caso non sono stati rilevati eventi specifici che indicano l'avvenimento di questa udienza, ma viene generato un evento quando viene fissata un'udienza di questo tipo, compreso di data futura in cui questa udienza dovrà avvenire.

Tutti gli altri eventi che non determinano un cambio di macrofase, ma che non sono stati scartati precedentemente, determinano un cambio di sottofase all'interno dei workflow delle macro-fasi, non considerato (per adesso) al fine del monitoraggio.

Per quanto riguarda, invece, gli eventi che delimitano una fase di quiescenza, sono stati presi in considerazione i seguenti eventi:

**Sospensione fascicolo:** indica l'inizio della fase di sospensione di un fascicolo.

**Riassunzione:** indica la ripresa delle attività su un fascicolo precedentemente sospeso.

## Stream Processing

Nei capitoli precedenti, è stato sottolineato che ai fini del recupero dei dati dal SICID è necessaria una sua estensione che si è deciso di effettuare mediante l'utilizzo della tecnica Change Data Capture, applicata ai dati contenuti nel database dell'applicativo. Tali dati confluiscono nel sistema di monitoraggio mediante un message broker e sono poi opportunamente elaborati per poter generare gli indicatori e le informazioni di interesse. Le elaborazioni prodotte sono illustrate nella Figura 6.

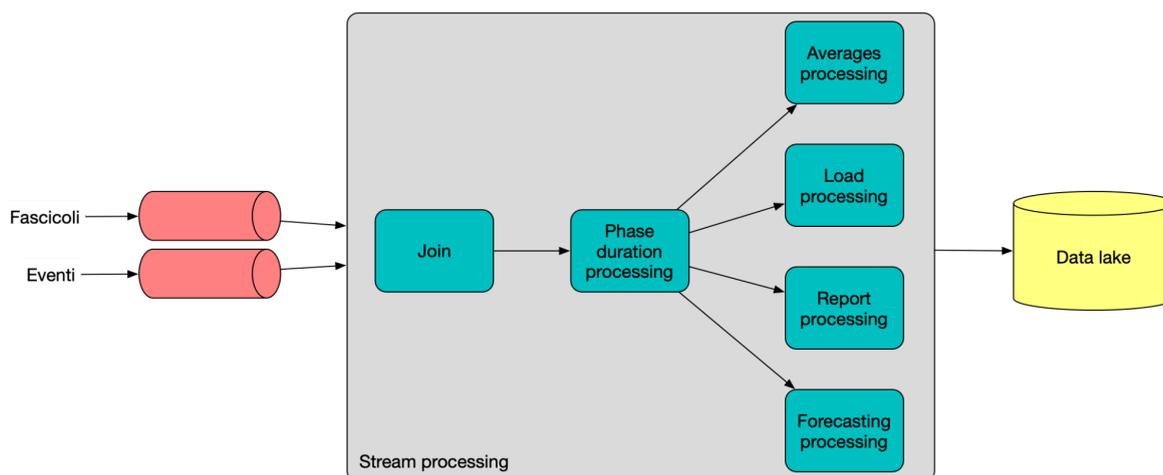


Figura 6 Task dello stream processing

La pipeline di elaborazione dei dati inizia con l'acquisizione dei flussi riguardanti i fascicoli e gli eventi. Tali flussi, come spiegato in precedenza, sono acquisiti mediante la tecnica del Change Data Capture (CDC) sui dati contenuti nel database. Nel caso in esame, le tabelle monitorate per il CDC sono quelle dei Fascicoli e degli Eventi (vedi paragrafo precedente).

I dati, una volta acquisiti, subiscono una serie di trasformazioni ad opera dello Stream Processor, quali:

- **Join:** rappresenta il primo task dello stream processing. Ha il compito di effettuare la join tra i flussi dei fascicoli e i flussi degli eventi;
- **Phase duration processing:** è il secondo task dello stream processing. Ha il compito di calcolare la durata delle fasi del processo (cfr. paragrafo Eventi nel sistema SICID). In particolare, per ciascuna delle fasi principali del processo, il task provvede a calcolarne la durata mediante specifici timer. La durata elaborata per ciascuna fase di ogni processo è memorizzata in un data lake e successivamente passata alla fase successiva;

- **Averages processing:** è il terzo task dello stream processing. Esso ha il compito di stimare la durata media di ciascuna fase dei processi. La durata media elaborata per ciascuna fase dei processi è memorizzata in un data lake;
- **Load processing:** è il quarto task dello stream processing. Esso provvede a calcolare gli indici di carico (es. Clearance Rate e Disposition Time). Le metriche di carico calcolate sono poi memorizzate in un data lake.
- **Report processing:** è il quarto task dello stream processing. Esso provvede a generare su base periodica i report per giudici, presidenti di sezione e presidenti di tribunale;
- **Forecasting processing:** è il task dello stream processing che si occupa di elaborare i dati provenienti dal modulo di Forecasting (vedere architettura) in modo da generare le previsioni per gli indici di Clearance Rate e Disposition Time a partire da quelle effettuate per gli andamenti delle fasi dei fascicoli.

Analizziamo nel dettaglio le varie fasi del processo di elaborazione. Come detto la prima fase è quella rappresentata dal task di **Join** dei flussi degli eventi provenienti dal sistema SICID. In particolare, mediante la tecnica del CDC, i flussi in arrivo allo stream processor sono così caratterizzati:



Figura 7 Modello dati dello stream di Fascicoli ed Eventi

Il flusso Fascicoli contiene i dati riguardanti i fascicoli inseriti nel SICID. In particolare, come si vede dalla Figura 7, ogni fascicolo è caratterizzato da:

- **id:** identificativo del fascicolo all'interno del SICID;
- **nrg:** il numero di ruolo del fascicolo all'interno del SICID;
- **ufficio:** l'ufficio che ha in carico il procedimento giudiziario;
- **sezione:** la sezione dell'ufficio giudiziario che gestisce il procedimento;
- **giudice:** il giudice che sta gestendo il procedimento;
- **oggetto:** la materia trattata dal fascicolo;
- **codiceOggetto:** il codice corrispondente alla materia.

Il flusso Eventi contiene, invece, i dati riguardanti i differenti eventi che vengono registrati per ciascun fascicolo. Ogni elemento all'interno del flusso è caratterizzato da:

- **id:** identificativo dell'evento all'interno del SICID;

- **data:** la data in cui si è verificato l'evento;
- **evento:** la descrizione testuale dell'evento verificatosi;
- **idFascicolo:** l'identificativo del fascicolo al quale l'evento è connesso.

La fase di Join provvede ad effettuare la fusione dei due stream, Fascicoli ed Eventi, in modo da poter correlare ciascun evento con il fascicolo relativo. L'output della fase di Join è visibile nella figura seguente:

EventoFascicolo
idFascicolo : Long
nrg : String
ufficio : String
sezione : String
giudice : String
oggetto : String
codiceOggetto : String
evento : String
data : Date

Figura 8 Dato prodotto dal task di Join

Come visibile dalla Figura 8, il dato in uscita è rappresentato dalla fusione (Join) dei due flussi di partenza.

La seconda fase dello stream processor è rappresentata dal task di **Phase duration processing**. Questo, come detto in precedenza, provvede a calcolare, per ciascun fascicolo, la durata di ciascuna fase del processo. Il calcolo viene effettuato analizzando il flusso con i dati EventoFascicolo in modo da selezionare solo gli eventi di interesse, ossia quelli che delimitano l'inizio e la fine di una particolare fase e calcolare la durata temporale (in giorni) di ciascuna di esse. Il dato prodotto in uscita dal task è il seguente:

DurataFase
idFascicolo : Long
nrg : String
ufficio : String
sezione : String
giudice : String
oggetto : String
codiceOggetto : String
fase : String
durata : Integer
dataInizio : Date
dataFine : Date

Figura 9 Dato prodotto dal task Phase Duration Processing

Come visibile dalla Figura 9, per ogni fascicolo e per ogni sua fase ne viene riportata la durata e la data di inizio e di fine. È bene precisare che tale conteggio viene effettuato in tempo

reale e dunque il valore della durata della fase è costantemente aggiornato (ogni 24h); il campo *dataFine* è, quindi, valorizzato soltanto alla chiusura di una fase.

Una volta ottenute le durate delle fasi dei singoli procedimenti, queste sono inviate agli altri task dello stream processor in modo tale da procedere alle altre elaborazioni.

La prima di queste elaborazioni è denominata **Averages Processing** ed ha il compito di calcolare le medie mobili delle durate delle fasi dei fascicoli secondo diversi raggruppamenti. Le medie mobili sono calcolate mediante una finestra mobile di ampiezza 180 giorni e scorrimento di 30 giorni. Nel dettaglio le medie mobili calcolate sono:

- Media mobile raggruppata per fase per il singolo giudice
- Media mobile raggruppata per fase e per materia per il singolo giudice
- Media mobile per sezione
- Media mobile raggruppata per materia per la singola sezione
- Media mobile raggruppata per fase per la singola sezione
- Media mobile raggruppata per fase e per materia per la singola sezione
- Media mobile per il tribunale
- Media mobile raggruppata per fase per il tribunale

La seconda elaborazione è denominata **Load Processing** ed ha il compito di calcolare gli indici di carico a partire dalle durate delle fasi dei fascicoli ricavate nella prima elaborazione. Gli indici sono calcolati mediante una finestra mobile di ampiezza 30 giorni. Nel dettaglio gli indici di carico calcolati sono:

- Numero totale dei fascicoli pendenti per singolo giudice
- Storico del disposition time per sezione
- Storico del disposition time per tribunale
- Storico del clearance rate per sezione
- Storico del clearance rate per tribunale
- Storico dell'indice di smaltimento per giudice
- Storico dell'indice di smaltimento per sezione
- Storico dell'indice di smaltimento per tribunale

La terza elaborazione è denominata **Report Processing** ed ha il compito di calcolare gli elementi che costituiscono i report per ciascun giudice, presidente di sezione e presidente di tribunale (crf. sezione Generazione dei Report), effettuando un confronto periodico tra le durate delle fasi dei singoli fascicoli e le durate medie calcolate nella elaborazione di **Average Processing**.

Infine, l'ultima elaborazione è denominata **Forecasting Processing** e, come detto in precedenza, ha il compito di elaborare i dati che arrivano dal modulo di Forecasting in modo da elaborare una previsione degli indici di Clearance Rate e Disposition Time. Questa previsione viene effettuata utilizzando come base di partenza la stima futura sulle durate dei singoli fascicoli, in modo da determinare, alla fine del periodo di previsione, quanti saranno i fascicoli ancora aperti e ricavare quindi il Clearance Rate e il Disposition Time futuri.

## Funzione di rischio

Per supervisionare l'operato dell'utente ed avvertirlo, eventualmente, del raggiungimento di una zona ad alto rischio di ritardo, è stata definita una **funzione di rischio** da integrare in un modello statistico e previsionale del ritardo.

La necessità di tale strumento nasce dalla considerazione che per mitigare il problema della formazione dell'arretrato è opportuno fornire ai giudici delle misure in tempo reale delle proprie performance, per poter correggere tempestivamente l'andamento assunto nella risoluzione delle controversie, cercando così di evitare situazioni di ritardo e conseguente accumulo di arretrato. Lo scopo è quello di ricondurre le performance dei giudici ad un **"ottimo locale"** e, nel tempo, attraverso l'utilizzo di modelli statistici e previsionali dinamici, ad un indicatore ideale (la ragionevole durata raccomandata per legge).

Al fine di definire una funzione di rischio è necessario individuare una serie di indici statistici e un modello previsionale.

## Individuazione della *late threshold*

Sebbene per le controversie sia stabilita una cosiddetta ragionevole durata per arrivare alla decisione, nella pratica per una serie di motivi questa durata viene disattesa. A supporto del miglioramento dei processi interni ai tribunali e per avvicinare le durate dei processi a quella ragionevole, sono stati definiti degli **indicatori dinamici** che catturano l'andamento corrente degli uffici e che supportino nel tempo le performance in continua evoluzione dei magistrati. Siccome l'obiettivo è quello di avvicinare la ragionevole durata, è opportuno partire dalla durata media complessiva dei fascicoli, da migliorare.

Si è scelto di impiegare la media, come misura statistica, ed in particolare la **media mobile esponenziale (MME)**, calcolata mensilmente, in cui la ponderazione è esponenzialmente più grande per i dati più recenti, mentre i dati più vecchi hanno un peso più basso:

$$MME_k = (P_i - MME_{k-1}) \frac{2}{n+1} + MME_{k-1}$$

$MME_k$  rappresenta la media mobile esponenziale al passo  $k$  (mensile),  $P_i$  è il peso attribuito ai dati correnti ed  $n$  è il numero di fascicoli a cui viene dato il peso  $P_i$ . La variazione della MME (la derivata delle varie medie nel tempo) dovrà risultare sempre **negativa** per rappresentare un miglioramento delle performance globali. In caso contrario sarebbe opportuno notificare la variazione positiva come peggioramento in vista di adottare misure correttive.

È importante la **scelta** della media mobile esponenziale per evitare le problematiche di cui potrebbe essere affetta la media statistica come, ad esempio:

- *l'impossibilità di attribuire dei pesi*, rendendo la variazione mensile impercettibile perché polarizzata dallo storico (grazie alla media mobile esponenziale è possibile far pesare di più le performance più recenti, rispetto a quelle passate);
- *il non riuscire a rappresentare correttamente la stazionarietà della media* nel caso in cui, nel contesto specifico, un magistrato in un certo mese non possedesse fascicoli su

cui lavorare. In tale situazione la media mobile esponenziale resterebbe stabile, pari a quella dell'ultimo mese utile.

Una volta individuata la late threshold, al fine di evitarne lo sfioramento da parte del magistrato, questo riceverà dei report periodici mirati a monitorare il proprio andamento e a far presente il raggiungimento di soglie prefissate (il template del report è riportato nella sezione *“Generazione di report”*). In particolare, le soglie individuate sono quattro, due riferite ad avvisi temporali e due ad avvisi di sfioramento. Tali soglie saranno definite nelle prossime sezioni.

## Avvisi temporali e threshold

Gli avvisi temporali sono progettati per guidare il magistrato rispetto allo stato di avanzamento ideale che il fascicolo in questione dovrebbe seguire. In particolare, sono considerate le seguenti threshold:

- Il **25%** della late threshold rappresenta la soglia superata la quale si fa presente al magistrato che il suo fascicolo dovrebbe essere completato per il 25% del totale.
- Il **50%** della late threshold, invece, rappresenta la soglia superata la quale si fa presente al magistrato che il suo fascicolo dovrebbe essere completato per il 50% del totale.

## Avvisi di sfioramento e threshold

Gli avvisi di sfioramento sono progettati per informare il magistrato che il fascicolo in questione si colloca in una zona ad elevato rischio di ritardo. In particolare, sono considerate le seguenti soglie.

La **early threshold** è definita come la mediana delle medie delle durate dei fascicoli per giudice, che rappresenta l'andamento peggiore tra le migliori performance dei giudici. Anche questo indice è rapportato alla situazione locale e quindi empiricamente raggiungibile ed ha bisogno, inoltre, come per la late threshold, di un **aggiornamento dinamico** mensile.

La **late threshold**, invece, è definita nella sezione *“Individuazione della late threshold”*.

Lo sfioramento della early threshold indica al magistrato di essere in una zona ad elevato rischio di ritardo (early warning), mentre lo sfioramento della seconda indica un ritardo certo (critical warning).

## Modello previsionale del ritardo

L'obiettivo di questo modello è quello di delineare l'andamento nel tempo delle controversie in ritardo per le varie fasi. Questo permette di prevedere, in relazione allo stato di avanzamento della fase del processo in un dato giorno  $x$  (giornalmente o settimanalmente), quale sarà l'ipotetica data di fine fase/processo e quindi stimarne eventuale ritardo/incremento dell'arretrato. La tecnica principale scelta per realizzare la componente di previsione, è il Modello ARIMA, anche se è possibile utilizzare ulteriori tecniche, come ad

esempio LSTM (Long Short-Term Memory). Le due tecniche differiscono in quanto la prima è a carattere statistico, la seconda, invece, rientra tra le reti neurali (è un tipo ottimizzato di rete neurale ricorrente - RNN) progettata per risolvere problemi di memorizzazione a lungo termine ed, eventualmente, adeguata a gestire dati a carattere sequenziale, come nel caso specifico, una serie temporale.

## Modello ARIMA

Per realizzare un modello previsionale, è possibile sfruttare le peculiarità dei modelli ARIMA (***Auto Regressive Integrated Moving Average***). Tali modelli sono ampiamente utilizzati, appunto, per la predizione delle serie temporali.

Nel dettaglio ARIMA è un particolare algoritmo di predizione basato sull'idea che le informazioni nei valori passati delle serie temporali possano essere utilizzate da sole per prevederne i valori futuri.

Un modello ARIMA è caratterizzato da tre parametri:

- **p**: l'ordine del parametro autoregressivo;
- **q**: l'ordine del parametro relativo alla media mobile;
- **d**: il numero di differenziazioni necessarie a rendere stazionaria la serie storica.

Tali parametri possono essere ricavati dall'analisi dei grafici di *autocorrelazione* (parametro d e q) e *autocorrelazione parziale* (parametro p), al fine di poter definire il modello e poter confrontare i risultati della predizione (output del modello) con quelli reali. Per capire la bontà del modello, inoltre, è opportuno effettuare dei test di cui controllare il p-value (che deve tenersi sotto allo 0.05, ovvero inferiore al livello di significatività).

Un'alternativa al calcolo dei tre parametri sopra citati, consiste nell'utilizzo della funzione *auto\_arima* che configura automaticamente i parametri ottimali. Tale soluzione è utilizzata nel modello implementato.

## Tecnologie utilizzate

Di seguito sono descritte brevemente le principali tecnologie adottate per la realizzazione di HYPERION IUS.

### Debezium

Debezium è una tecnologia open source e distribuita per il change data capture. Essa consente di acquisire le modifiche che si verificano in un database in modo che le applicazioni possano vedere tali modifiche e rispondere ad esse. Debezium registra tutte le modifiche che si verificano alle righe all'interno di ogni tabella del database e le trasmette in un flusso di eventi di modifica a cui le applicazioni possono accedere per vedere gli eventi di modifica nello stesso ordine in cui si sono verificati.

## Apache Kafka

Apache Kafka è una piattaforma di streaming di eventi distribuita e open source utilizzata da migliaia di aziende per pipeline di dati ad alte prestazioni, analisi di streaming, integrazione di dati e applicazioni mission-critical.

È sviluppata dalla Apache Software Foundation in Scala e Java. Il progetto mira a fornire una piattaforma unificata, ad alta velocità e bassa latenza per la gestione dei feed di dati in tempo reale. Kafka può connettersi a sistemi esterni (per l'importazione/esportazione di dati) tramite Kafka Connect e fornisce Kafka Streams, una libreria di elaborazione di flussi Java. Kafka utilizza un protocollo binario basato su TCP che è ottimizzato per l'efficienza e si basa su un'astrazione di "set di messaggi" che raggruppa naturalmente i messaggi per ridurre il sovraccarico del round trip di rete.

Kafka memorizza messaggi chiave-valore che provengono da molti processi arbitrariamente chiamati produttori. I dati possono essere partizionati in diverse "partizioni" all'interno di diversi "topic". All'interno di una partizione, i messaggi sono rigorosamente ordinati in base ai loro offset (la posizione di un messaggio all'interno di una partizione) e indicizzati e archiviati insieme a un timestamp. Altri processi chiamati "consumatori" possono leggere i messaggi dalle partizioni. Per l'elaborazione del flusso, Kafka offre l'API Streams che consente di scrivere applicazioni (es. in Java) che consumano dati da Kafka e riscrivono i risultati su Kafka. Apache Kafka funziona anche con sistemi di elaborazione del flusso esterni come Apache Apex, Apache Flink, Apache Spark, Apache Storm e Apache NiFi.

Kafka viene eseguito su un cluster di uno o più server (chiamati broker) e le partizioni di tutti i topic sono distribuite tra i nodi del cluster. Inoltre, le partizioni vengono replicate su più broker. Questa architettura consente a Kafka di fornire enormi flussi di messaggi in modo tollerante ai guasti e gli ha permesso di sostituire alcuni dei sistemi di messaggistica convenzionali come Java Message Service (JMS), Advanced Message Queuing Protocol (AMQP), ecc.

## Apache Flink

Apache Flink è un framework open-source unificato per lo stream-processing e il batch-processing sviluppato dalla Apache Software Foundation. Il core di Apache Flink è un motore di flusso di dati in streaming distribuito scritto in Java e Scala. Flink esegue programmi di flusso dati arbitrari in modalità parallela ai dati ed in pipeline (quindi task parallel). Il sistema di runtime pipeline di Flink consente l'esecuzione di programmi di elaborazione bulk/batch e stream. Inoltre, il runtime di Flink supporta l'esecuzione di algoritmi iterativi in modo nativo.

Flink fornisce un motore di streaming ad alta velocità e bassa latenza oltre al supporto per l'elaborazione degli eventi in tempo e la gestione dello stato. Le applicazioni Flink sono tolleranti ai guasti e supportano la semantica exactly-once. I programmi possono essere scritti in Java, Scala, Python, e SQL e vengono automaticamente compilati e ottimizzati in flusso di dati programmati ed eseguiti in un ambiente cluster o cloud.

Flink non fornisce un proprio sistema di archiviazione dei dati, ma fornisce connettori dati e sink a sistemi come Amazon Kinesis, Apache Kafka, HDFS, Apache Cassandra ed ElasticSearch.

## Opensearch

Opensearch è un database server, scritto in Java, che si occupa di memorizzare grandi quantità di dati in un formato ottimizzato per ricerche basate sul linguaggio.

Per comprendere meglio Opensearch e il suo utilizzo è utile avere una comprensione generale dei principali componenti di backend. Un nodo è un singolo server che fa parte di un cluster, memorizza i dati e partecipa alle funzionalità di indicizzazione e ricerca del cluster. Un cluster è una raccolta di uno o più nodi che tengono insieme tutti i dati e forniscono funzionalità di indicizzazione e ricerca federate. Possono esserci N nodi con lo stesso nome cluster. Opensearch funziona in un ambiente distribuito: con la replica tra cluster, un cluster secondario può entrare in azione come backup a caldo senza la necessità di far ripartire Opensearch.

L'indice è una raccolta di documenti con caratteristiche simili. Ad esempio, è possibile avere un indice per un cliente specifico, un altro per informazioni sul prodotto e un altro per una diversa tipologia di dati. Un indice è identificato da un nome univoco da usare come riferimento quando si eseguono operazioni di ricerca, aggiornamento ed eliminazione dell'indicizzazione. Un documento è un'unità di base di informazioni che può essere indicizzata. Ad esempio, è possibile avere un indice sul prodotto e quindi un documento per un singolo cliente. Questo documento è espresso in JSON (JavaScript Object Notation).

Opensearch offre la possibilità di suddividere il proprio indice in più pezzi chiamati frammenti (Shards). I frammenti sono importanti perché consentono di dividere orizzontalmente il volume di dati, parallelizzando l'accesso ai dati e aumentando così le prestazioni.

## Opensearch Dashboard

Opensearch Dashboard fornisce le funzionalità di visualizzazione del contenuto indicizzato su un cluster Opensearch. In particolare, consente di creare, in maniera semplice ed intuitiva, una grande varietà di tipologie di grafici (grafici a barre, a linee e a dispersione, grafici a torta e mappe, heatmap, ecc.) su grandi volumi di dati.

## Integrazione del prototipo nei sistemi IT della giustizia

Il prototipo di Hyperion sviluppato recupera i dati dei fascicoli da un SICID costruito ad-hoc, opportunamente semplificato nelle API e nello schema dati. Per poter rendere utilizzabile il software nell'ambiente reale occorre quindi procedere alla sua integrazione con il SICID vero e proprio, per quanto concerne il recupero dei fascicoli, e con il suo sistema di gestione degli utenti per quanto concerne, invece, le funzionalità di autenticazione e autorizzazione. In particolare, le fasi necessarie all'integrazione e alla messa in esercizio del sistema sono

elencate di seguito. Per ogni fase è riportata anche una stima di massima delle ore/uomo necessarie al completamento:

- Analisi del sistema SICID (API e schema dati) e del suo meccanismo di gestione delle utenze. Per questa fase sono stimate circa 300 ore/uomo;
- Integrazione del SICID e del suo schema dati in Hyperion mediante modifica degli eventi catturati con il Change Data Capture e adeguamento della fase di elaborazione degli stessi. Per questa fase sono stimate circa 450 ore/uomo;
- Integrazione del sistema di gestione degli utenti del SICID in Hyperion, mediante modulo di Keycloak o mediante mapping degli utenti. Per questa fase sono stimate circa 300 ore/uomo;
- Verifica e testing delle funzionalità di Hyperion, in un apposito ambiente di test con un sottoinsieme dei dati del SICID, sia per quanto riguarda la correttezza delle elaborazioni sia per quanto concerne gli aspetti non funzionali. Per questa fase sono stimate circa 600 ore/uomo;
- Fine tuning delle funzionalità a partire dai feedback raccolti dagli utenti. Per questa fase sono stimate circa 450 ore/uomo;
- Dispiegamento del sistema nell'ambiente finale e validazione finale. Per questa fase sono stimate circa 150 ore/uomo.

## Manuale d'uso

### Installazione

#### Requisiti di sistema

Le caratteristiche necessarie affinché sia possibile usufruire di prestazioni ottimali del software comprendono requisiti hardware e software da considerare come minimi poiché, a seconda dell'utenza e dei dati da elaborare, è necessario scalare e quindi è opportuno disporre di soluzioni che abilitino la scalabilità.

##### Requisiti hardware:

1. Processore: CPU multicore (6 o più) 2.5 Ghz;
2. Memoria (RAM): 32 GB;
3. Spazio su disco (ROM): almeno 50 GB di spazio libero sul disco rigido.

##### Requisiti software:

1. Sistema operativo: Linux Ubuntu 20.04.06;
2. Framework o librerie: Java SE Development Kit 11.0.19, Docker 20.10.24;
3. Browser web: Google Chrome, Microsoft Edge, Mozilla Firefox etc.

#### Installazione del software

Il sistema è composto da 5 componenti, dispiegate in container docker attraverso un docker compose. Nello specifico, sono previsti i seguenti container:

- 1 per Keycloak, utilizzato per la gestione degli utenti della piattaforma;
- 3 per OpenSearch (1 + 2 repliche);
- 1 per OpenSearch Dashboards.
- 3 per Apache Kafka (1 per Apache Zookeeper, 1 per il broker e 1 per Kafka Connect con Debezium)
- 2 per Apache Flink (1 per il task manager e 1 per il job manager)

Attualmente è usata la versione 1.3.9 di OpenSearch e OpenSearch Dashboards al fine di garantire la compatibilità con le diverse componenti, in particolare con il connettore di Apache Flink.

I container avranno dei volumi condivisi con l'host per la gestione delle configurazioni.

La comunicazione, secondo le specifiche di OpenSearch, con Keycloak, avviene esclusivamente tramite HTTPS, pertanto è necessario assegnare, previa eventuale generazione, dei certificati SSL.

Il docker-compose.yaml prevede già un mapping delle seguenti porte:

- 32311 porta HTTPS OpenSearch;
- 32411 porta HTTPS OpenSearch Dashboards;
- 32310/32410 porta HTTP/HTTPS Keycloak.

In generale, per l'installazione del software, è necessario richiamare il file docker-compose.yml da riga di comando. Tuttavia, si rimanda al file README.md, nel quale sono presenti anche gli script per la generazione dei certificati SSL.

NB: L'avvio dei container non può essere fatto in ordine casuale. L'ordine da seguire è il seguente:

1. KeyCloak
2. OpenSearch
3. OpenSearch Dashboards
4. Zookeeper
5. Kafka
6. Notifier
7. Kafka Connect
8. Flink
9. Modulo ARIMA

## Configurazione del software

Una volta effettuato il dispiegamento delle componenti sopra descritte, è necessario configurarle al fine di garantire l'interoperabilità tra le stesse.

- **Keycloak**

La console di amministrazione, raggiungibile all'indirizzo specificato in fase di dispiegamento, consente di configurare le funzionalità messe a disposizione dall'identity provider.

1. Accedere alla console di amministrazione;
2. Creare un *realm* dedicato alla gestione degli utenti, dei client e altre configurazioni;
3. Creare un client dedicato, il quale rappresenterà l'applicazione (Opensearch Dashboard) che farà richiesta del servizio di autenticazione. Per fare ciò accedere alla sezione "Clients" nell'interfaccia di amministrazione e aggiungere un nuovo client fornendo Client ID (es. dashboards), redirect URIs (post login e post logout) e infine abilitare la "Client authentication";
4. Creare i ruoli da assegnare agli utenti nella sezione "Realm roles", assegnando un nome identificativo per ogni tipologia di utente (es. giudice, presidente di sezione, presidente di tribunale);
5. Creare i profili utente nella sezione "Users", specificando l'username che sarà utilizzato in fase di autenticazione, il ruolo da assegnare (tra quelli creati al punto 4) e successivamente la password di accesso. Infine, è necessario definire delle informazioni aggiuntive e personali per ogni utente nell'apposito tab "Attributes", nel

quale specificare coppie chiave-valore del tipo *giudice: giudice01*, *sezione: 01*, oppure *tribunale: Benevento*. Queste informazioni risultano indispensabili per il filtraggio dei documenti che applica OpenSearch (Document level security);

6. Creare un mapping per ogni tipologia di attributo con un *claim*, il quale incapsula il valore della chiave mappata nel token di accesso. Per fare ciò, nella sezione “Clients” selezionare il client creato al punto 3 e poi il tab “Client scopes”. Nell’elenco che apparirà selezionare lo scope dedicato al client (il nome è del tipo *nomeClient-dedicated*), e successivamente aggiungere un mapper di tipo “User Attribute”. Aperta la finestra dedicata, specificare: il nome del mapper, la chiave dell’user attribute da mappare, il nome del claim che conterrà il valore associato alla chiave e infine aggiungere il claim al token di accesso abilitando lo specifico pulsante.
7. Nella stessa sezione creare un mapper per inserire in uno specifico claim all’interno del token di accesso la lista dei ruoli dell’utente che sarà indispensabile per OpenSearch Dashboards al fine di mappare i ruoli di Keycloak nei suoi ruoli interni che verranno definiti al punto successivo. Per fare ciò aggiungere un mapper di tipo “User Realm Role”, specificare il nome del claim che conterrà la lista dei ruoli (es. *roles*) e infine aggiungere il claim al token di accesso abilitando lo specifico pulsante.

- **Opensearch Dashboard**

Prima di realizzare le diverse viste è necessario configurare il componente per garantire la corretta autorizzazione alle risorse.

1. Accedere al componente Opensearch Dashboards;
2. Creare i *tenant* nella sezione “Security”, i quali sono dedicati per ogni tipologia di utente al fine di isolare e gestire in modo indipendente ogni spazio di lavoro;
3. Popolare gli indici di interesse mediante API consultabili dalla documentazione ufficiale;
4. Creare i ruoli da assegnare agli utenti che si autenticeranno mediante Keycloak nel tab “Roles” della sezione “Security”. Specificare il nome del ruolo, i permessi sul cluster e sui tenant e infine i permessi sugli indici creati al punto 3 con eventuali filtri sui relativi documenti. Questa azione è fondamentale per limitare l’accesso a un subset di documenti e si collega ai claim mappati al punto 6 di Keycloak. Opensearch Dashboard li preleva dal token di accesso e l’utente può inserirli come query (es. “*giudice*”: “*{attr.jwt.giudice}*”) nell’apposita sezione “Document level security”. In questo modo Opensearch filtrerà tutti i documenti che matchano il campo presente nella query configurata.
5. Dopo aver configurato i ruoli di interesse (*giudice*, *presidente di sezione* e *presidente di tribunale*) è necessario configurare delle regole che consentano di mappare i ruoli degli utenti definiti in Keycloak nei ruoli appena creati al punto 4. Per fare ciò, per ogni ruolo, accedere al relativo tab “Mapped users” e cliccare il bottone “Manage Mapping”. A questo punto inserire nel campo “Backend roles” il nome del ruolo definito nel servizio di autenticazione esterno (Keycloak al punto 4) che si vuole mappare nel ruolo scelto in Opensearch Dashboards.

- **config.yml**

Aggiungere il seguente frammento di codice che risulta indipendente dall'installazione ad eccezione del campo `openid_connect_url` il quale dipende dall'indirizzo della macchina su cui il servizio di autenticazione di Keycloak è stato dispiegato. In particolare:

1. Accedere alla console di amministrazione di Keycloak;
2. Nella sezione "Realm settings" accedere al link in cui è presente la configurazione per OpenId Connect;
3. Copiare l'URL presente nella barra di ricerca nello yaml.

`openid_auth_domain:`

```
http_enabled: true
transport_enabled: true
order: 1
http_authenticator:
  type: openid
  challenge: false
  config:
    openid_connect_idp:
      enable_ssl: true
      verify_hostnames: false
      pemtrustedcas_filepath: /usr/share/openssl/config/certificates/ca/ca.pem
      subject_key: preferred_username
      roles_key: roles
    openid_connect_url:
https://xxx.xxx.xxx.xxx:yyyyy/realms/master/.well-known/openid-configuration
  authentication_backend:
    type: noop
```

- **opensearch-dashboard.yml**

Aggiungere le seguenti righe di configurazione, di cui alcune da recuperare nella console di amministrazione di Keycloak. In particolare:

- `opensearch_security.multitenancy.tenants.enable_global: false`
- `opensearch_security.multitenancy.tenants.enable_private: false`

per disabilitare il tenant pubblico e il tenant privato, non funzionali ai fini del sistema di monitoraggio;

- `opensearch_security.readonly_mode.roles: [{"Ruolo"}, {"Ruolo"}, {"Ruolo"}]`

per limitare i permessi sulla piattaforma alla sola lettura per gli utenti con i ruoli specificati;

- `opensearch_security.auth.type: openid`

per abilitare la sola autenticazione mediante Keycloak con protocollo OpenID Connect;

- `opensearch_security.openid.connect_url:`  
**<https://xxx.xxx.xxx.xxx:yyyy/realms/master/.well-known/openid-configuration>**

per specificare l'URL in cui è presente la configurazione di OpenID Connect;

- `opensearch_security.openid.base_redirect_url:` **<https://xxx.xxx.xxx.xxx:yyyy/>**

per specificare l'URL di redirect dopo l'autenticazione;

- `opensearch_security.openid.client_id:` **`{clientID}`**

per specificare l'ID del client OpenID Connect configurato in Keycloak e dedicato a Opensearch Dashboards;

- `opensearch_security.openid.client_secret:` **`{clientSecret}`**

per specificare il client secret associato al client, che deve essere condiviso esclusivamente tra l'applicazione client e il server Keycloak;

Inoltre, Opensearch Dashboards offre la possibilità di brandizzare la piattaforma mediante label e logo custom da sostituire agli elementi di default. Per fare ciò è necessario inserire il logo custom nel folder `/assets` del container di Opensearch Dashboard e aggiungere le seguenti righe di configurazione nel file:

`opensearchDashboards.branding:`

`logo:`

`defaultUrl: "https://xxx.xxx.xxx.xxx:yyyyy/ui/assets/{nomefile}"`

`mark:`

`defaultUrl: "https://xxx.xxx.xxx.xxx:yyyyy/ui/assets/{nomefile}"`

`loadingLogo:`

`defaultUrl: "https://xxx.xxx.xxx.xxx:yyyyy/ui/assets/{nomefile}"`

`faviconUrl: "https://xxx.xxx.xxx.xxx:yyyyy/ui/assets/{nomefile}"`

`applicationTitle: "{nomePrototipo}"`

## Aggiornamento del software

Il software è ospitato su GitLab che ne assicura la gestione dal punto di vista dello sviluppo collaborativo e ne facilita gli sviluppi evolutivi.

## Disinstallazione del software

Per disinstallare il software è sufficiente eliminare i container docker dispiegati e, eventualmente le cartelle condivise con il file system host contenenti le configurazioni. Per disabilitarlo, invece, è sufficiente interrompere (stoppare) i container docker. Per riavviarlo è necessario fare attenzione e seguire l'ordine sopra citato.

## Guida all'uso

### Accesso al software

Hyperion IUS è un'applicazione web, pertanto l'accesso avviene mediante un browser (es. Google Chrome, Microsoft Edge, Mozilla Firefox etc.), inserendo nella barra di ricerca in alto il seguente indirizzo:

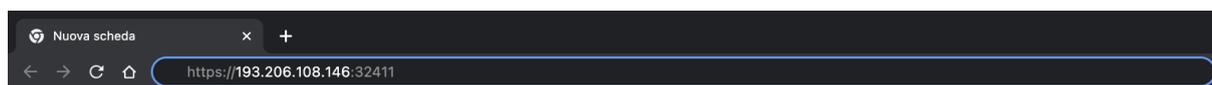


Figura 10 Indirizzo di Hyperion IUS

Effettuata la ricerca, dopo aver superato l>alert di sicurezza relativo ai certificati SSL autogenerati<sup>1</sup>, viene visualizzata la pagina di autenticazione nella quale inserire le credenziali di accesso fornite (username e password), e infine cliccare il pulsante "Sign in".

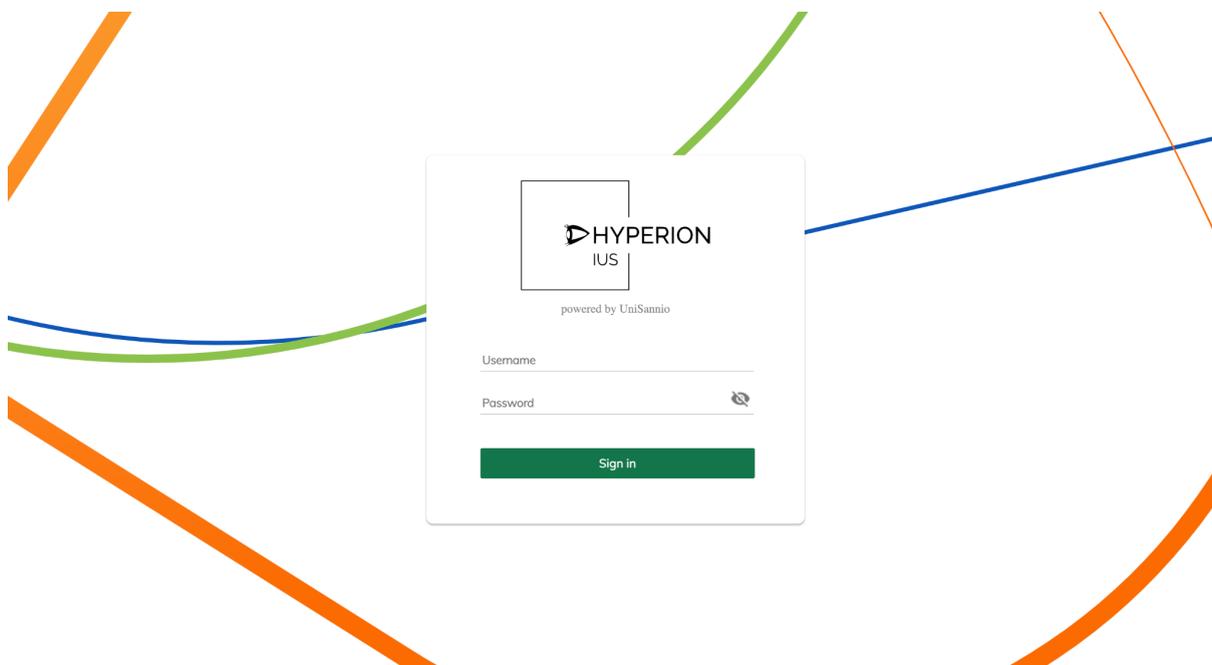


Figura 11 Pagina di autenticazione

Una volta verificato il corretto inserimento delle credenziali, viene visualizzata la pagina contenente l'elenco delle viste disponibili.

<sup>1</sup> I certificati saranno sostituiti nelle versioni successive del prototipo.

## Interfaccia utente

Completata la fase di login, l'utente potrà scegliere la vista desiderata. Nella pagina, visibile in Figura 12, è possibile selezionare la vista che si desidera cliccando sul titolo della stessa, evidenziato in rosso nell'immagine. Attualmente sono disponibili la vista "{Ruolo} - Tempestività di Giudizio", con gli indicatori relativi alla tempestività di giudizio, e la vista "{Ruolo} - Carico di Giustizia", con gli indicatori relativi al carico.

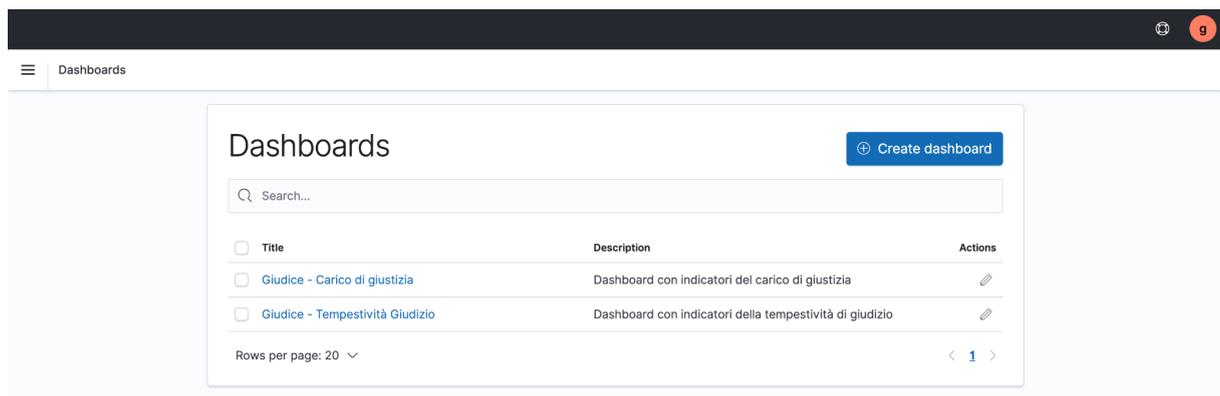


Figura 12 Pagina di selezione delle viste

## Funzionalità di base

Le viste sono diversificate a seconda della tipologia dell'utente autenticato.

- **Giudice - Tempestività Giudizio**

Il giudice ha a disposizione 6 grafici:

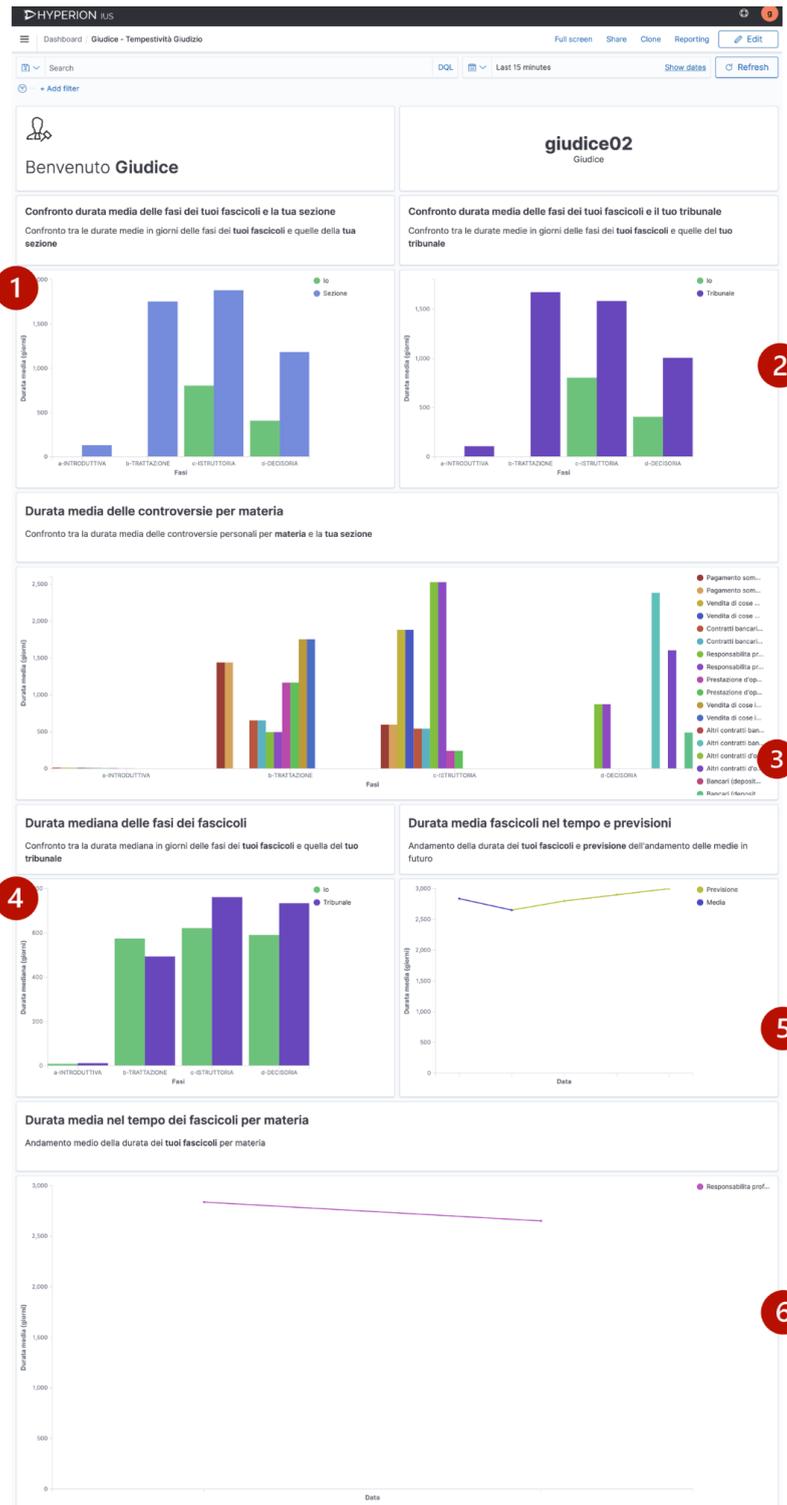


Figura 13 Grafici per il cruscotto Tempestività di giudizio (giudice)

1. Il grafico permette di confrontare la media mobile della durata delle fasi dei propri fascicoli con la media mobile della durata delle fasi dei fascicoli in carico alla sua sezione di appartenenza.
2. Il grafico permette di confrontare la media mobile della durata delle fasi dei propri fascicoli con la media mobile della durata delle fasi dei fascicoli in carico al suo tribunale di appartenenza.

3. Il grafico permette di confrontare la media mobile della durata delle fasi dei propri fascicoli, suddivisi per materia, con la media mobile della durata delle fasi dei fascicoli in carico alla sua sezione di appartenenza.
4. Il grafico permette di confrontare la durata mediana in giorni delle fasi dei propri fascicoli con la durata mediana delle fasi dei fascicoli in carico al suo tribunale di appartenenza.
5. Il grafico mostra l'andamento della media mobile della durata dei fascicoli nel tempo e la relativa previsione per una finestra di osservazione futura.
6. Il grafico mostra l'andamento della media mobile della durata dei fascicoli nel tempo, suddivisi per materia.

- **Giudice - Carico di Giustizia**

Il giudice ha a disposizione 3 grafici:

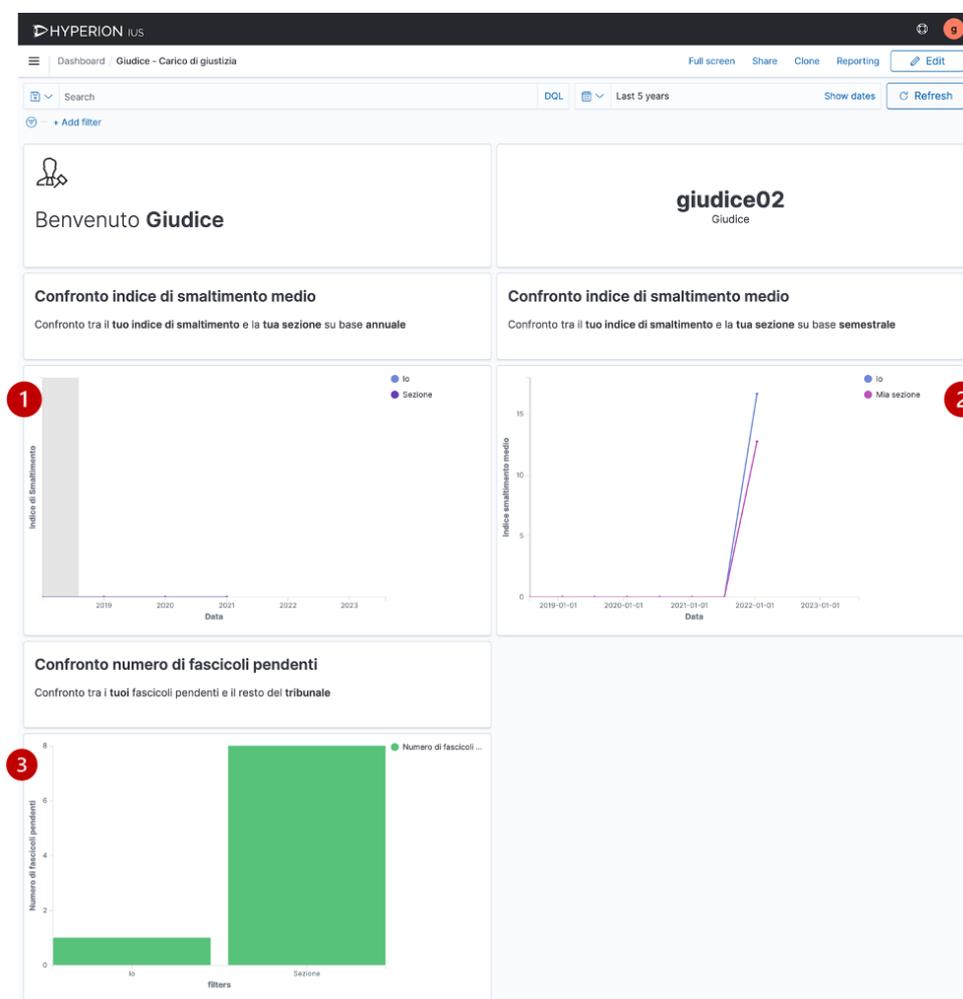


Figura 14 Grafici per il cruscotto Carico di giustizia (giudice)

1. Il grafico permette di confrontare l'indice di smaltimento medio rispetto allo stesso valutato per la sezione di appartenenza su base annuale.
2. Il grafico permette di confrontare l'indice di smaltimento medio rispetto allo stesso valutato per la sezione di appartenenza su base semestrale.

- Il grafico permette di confrontare il numero dei propri fascicoli pendenti con gli stessi relativi al tribunale di appartenenza.

- Presidente di Sezione - Tempestività Giudizio**

Il presidente di sezione ha a disposizione 8 grafici:

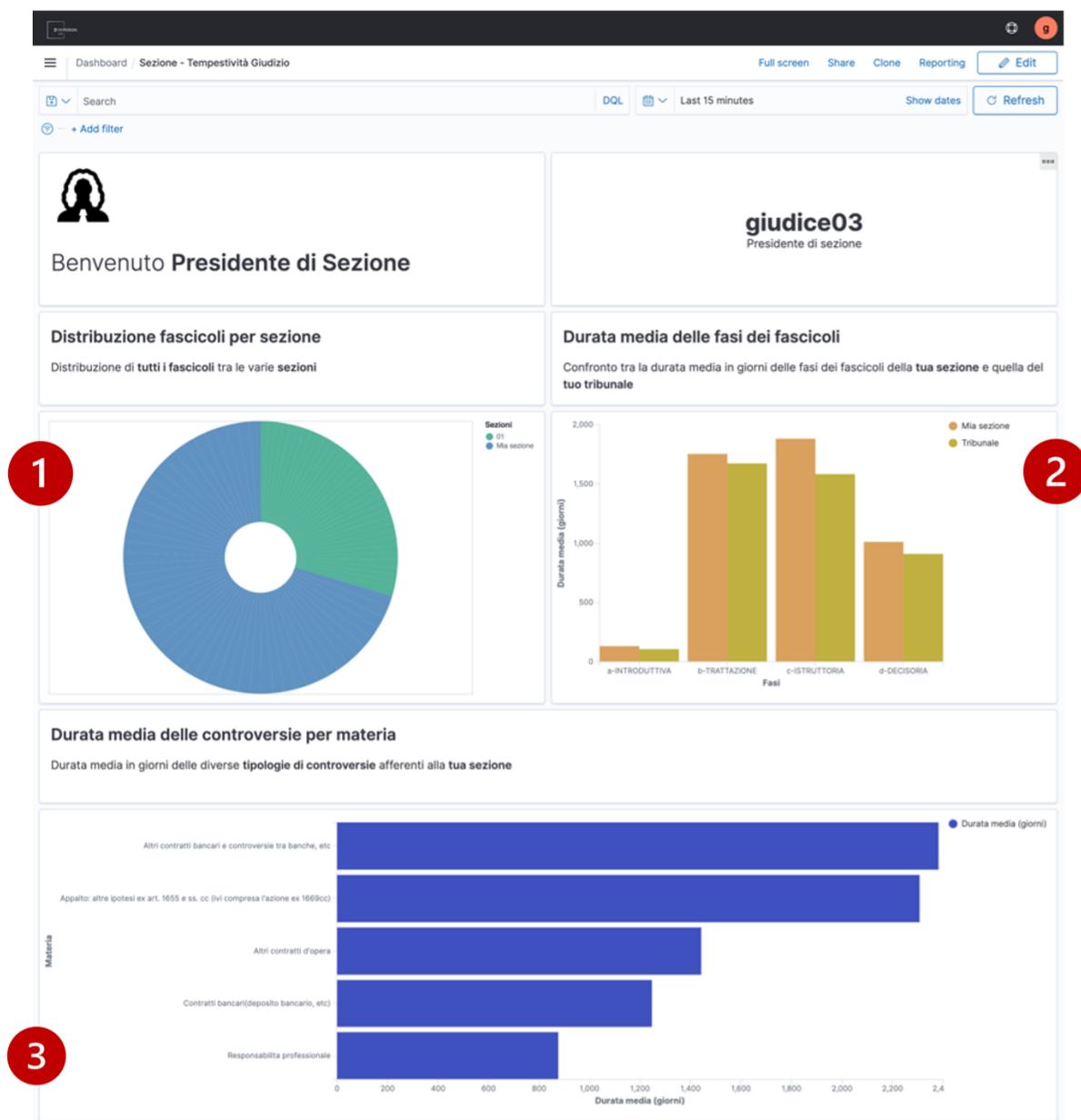


Figura 15 Grafici per il cruscotto Tempestività di giudizio (presidente di sezione)

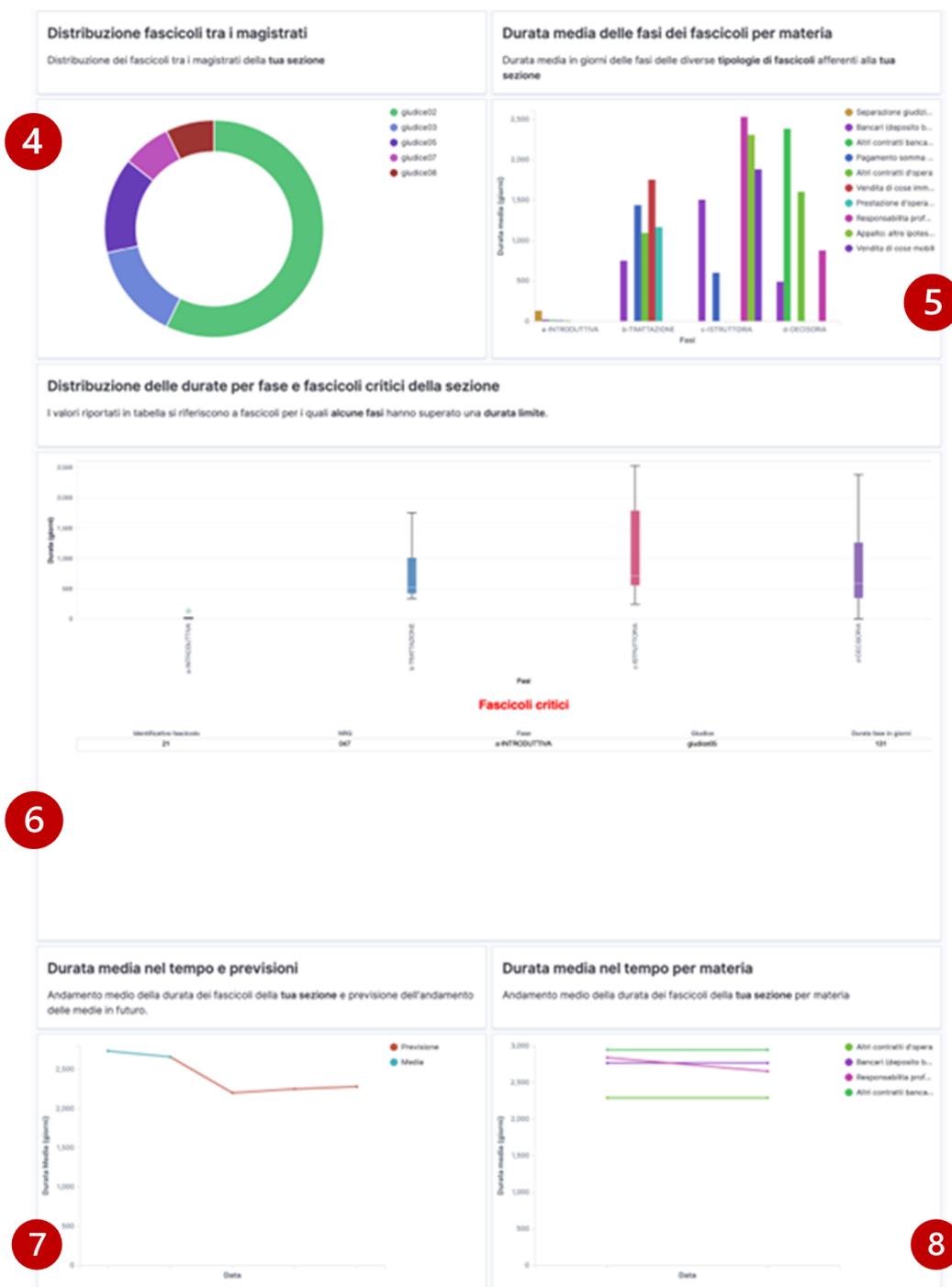


Figura 16 Grafici per il cruscotto Tempestività di giudizio (presidente di sezione)

1. Il grafico mostra la distribuzione dei fascicoli all'interno delle diverse sezioni del tribunale di appartenenza.
2. Il grafico permette di confrontare la media mobile della durata delle fasi dei fascicoli della sua sezione, con la media mobile della durata delle fasi dei fascicoli delle altre sezioni del suo tribunale di appartenenza.
3. Il grafico mostra la media mobile della durata delle diverse tipologie di controversie afferenti alla sezione presieduta.
4. Il grafico mostra la ripartizione dei fascicoli tra i magistrati afferenti alla sezione presieduta.

5. Il grafico mostra la media mobile della durata delle fasi delle diverse tipologie di controversie afferenti alla sezione presieduta.
6. Il grafico mostra i fascicoli critici della sezione, ossia quelli più vicini al raggiungimento della media mobile della durata e quindi con maggiore probabilità di sfioramento.
7. Il grafico mostra l'andamento della media mobile della durata dei fascicoli della sezione nel tempo e la relativa previsione per una finestra di osservazione futura.
8. Il grafico mostra l'andamento della media mobile della durata dei fascicoli della sezione nel tempo, suddivisi per materia.

- **Presidente di Sezione - Carico di Giustizia**

Il presidente di sezione ha a disposizione 7 grafici:

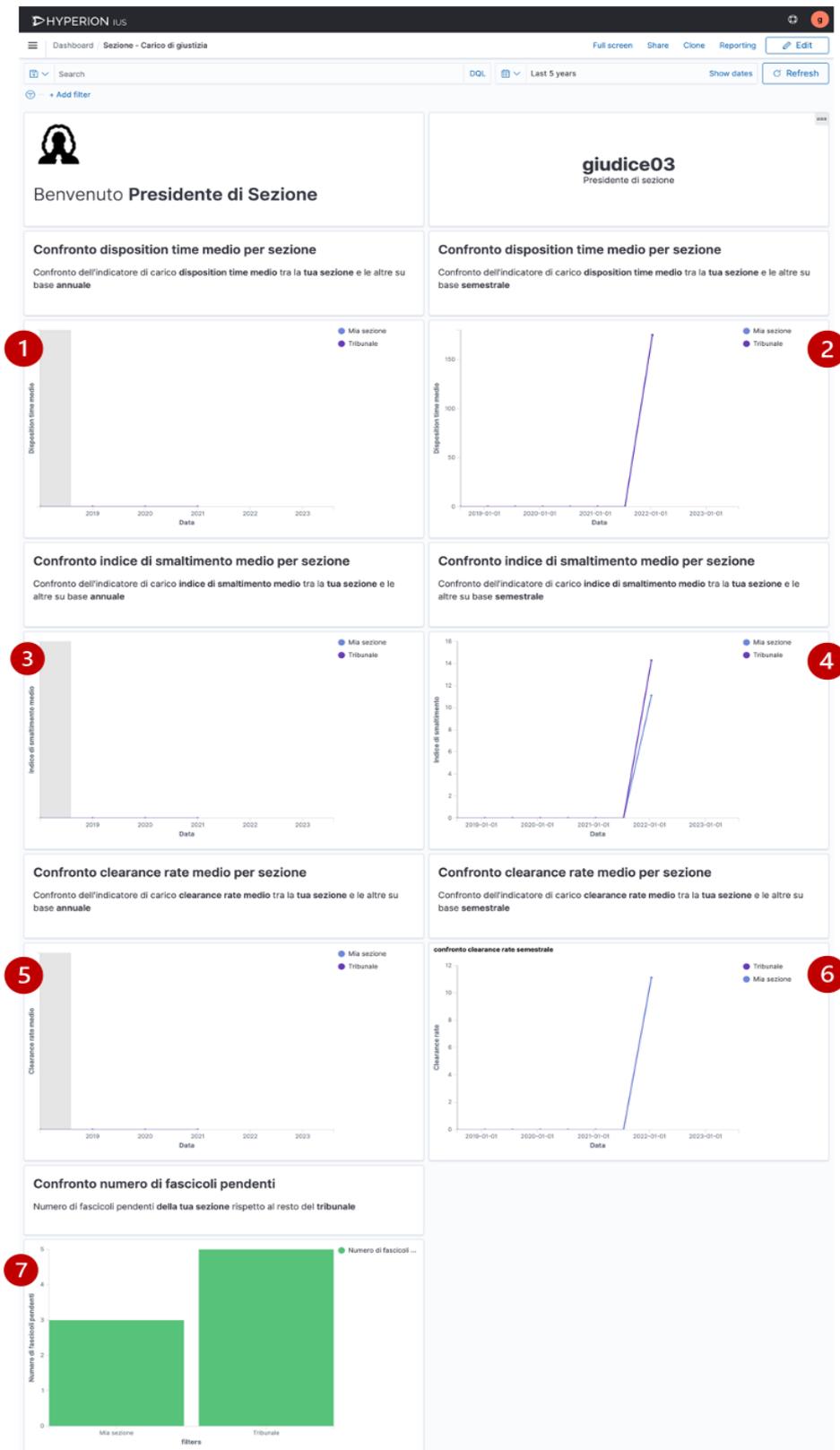


Figura 17 Grafici per il cruscotto Carico di giustizia (presidente di sezione)

1. Il grafico permette di confrontare il disposition time medio della propria sezione e lo stesso relativo al tribunale di appartenenza su base annuale.
2. Il grafico permette di confrontare il disposition time medio della propria sezione e lo stesso relativo al tribunale di appartenenza su base semestrale.
3. Il grafico permette di confrontare l'indice di smaltimento medio della propria sezione e lo stesso relativo al tribunale di appartenenza su base annuale.
4. Il grafico permette di confrontare l'indice di smaltimento medio della propria sezione e lo stesso relativo al tribunale di appartenenza su base semestrale.
5. Il grafico permette di confrontare il clearance rate medio della propria sezione e lo stesso relativo al tribunale di appartenenza su base annuale.
6. Il grafico permette di confrontare il clearance rate medio della propria sezione e lo stesso relativo al tribunale di appartenenza su base semestrale.
7. Il grafico permette di confrontare il numero dei fascicoli pendenti relativi alla propria sezione con gli stessi relativi al tribunale di appartenenza.

● **Presidente di Tribunale - Tempestività Giudizio**

Il presidente di tribunale ha a disposizione 6 grafici:

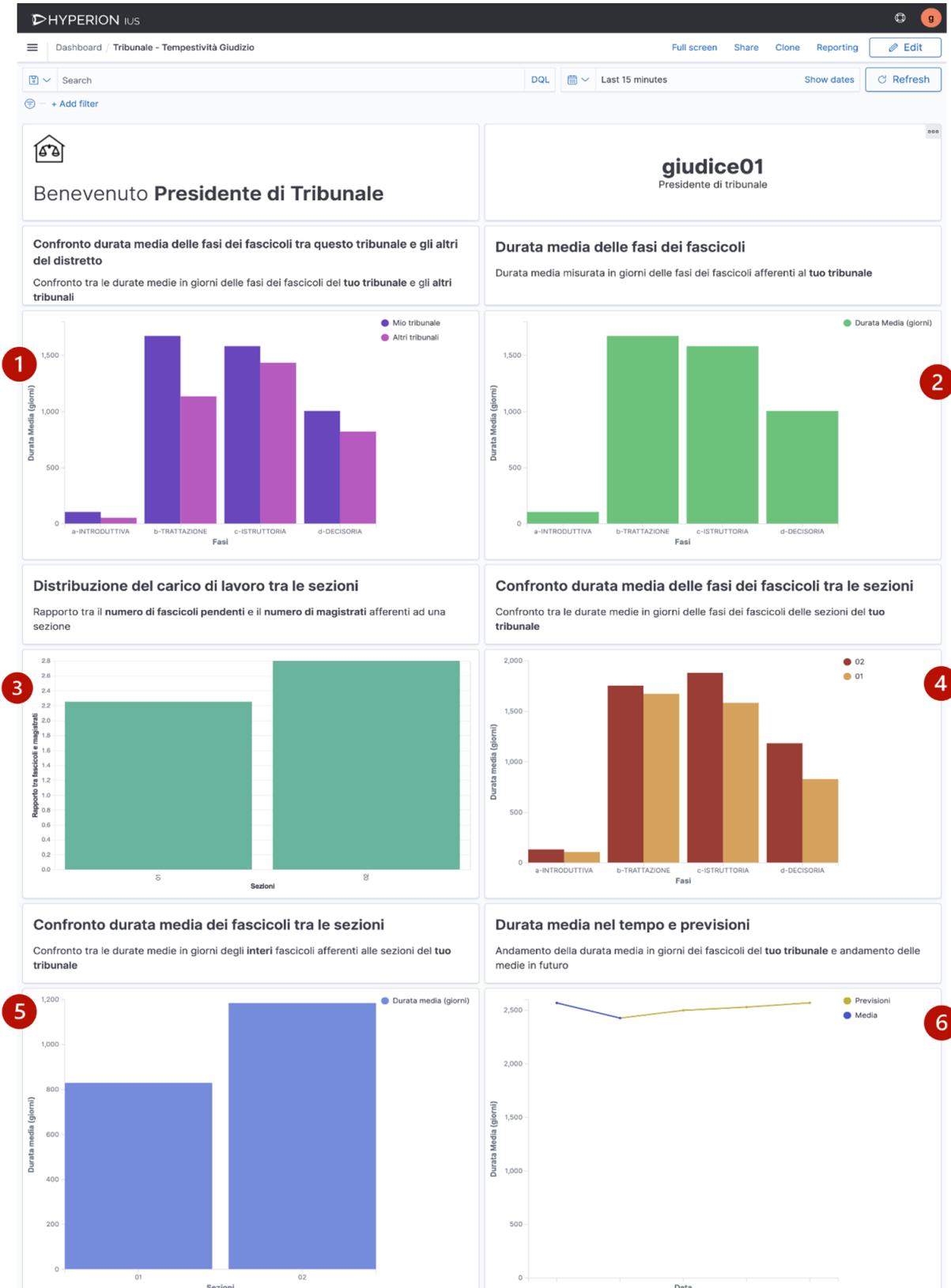


Figura 18 Grafici per il cruscotto Tempestività di giudizio (presidente tribunale)

1. Il grafico permette di confrontare la media mobile della durata delle fasi dei fascicoli del tribunale, con la media mobile della durata delle fasi dei fascicoli degli altri distretti.
2. Il grafico mostra la media mobile della durata delle fasi dei fascicoli del tribunale.
3. Il grafico permette di confrontare il numero di fascicoli pendenti e il numero di giudici, suddivisi per sezione.
4. Il grafico permette di confrontare la media mobile della durata delle fasi dei fascicoli di ogni sezione.
5. Il grafico permette di confrontare la media mobile della durata complessiva dei fascicoli di ogni sezione.
6. Il grafico mostra l'andamento della media mobile della durata dei fascicoli del tribunale nel tempo e la relativa previsione per una finestra di osservazione futura.

- **Presidente di Tribunale - Carico di Giustizia**

Il presidente di tribunale ha a disposizione 7 grafici:

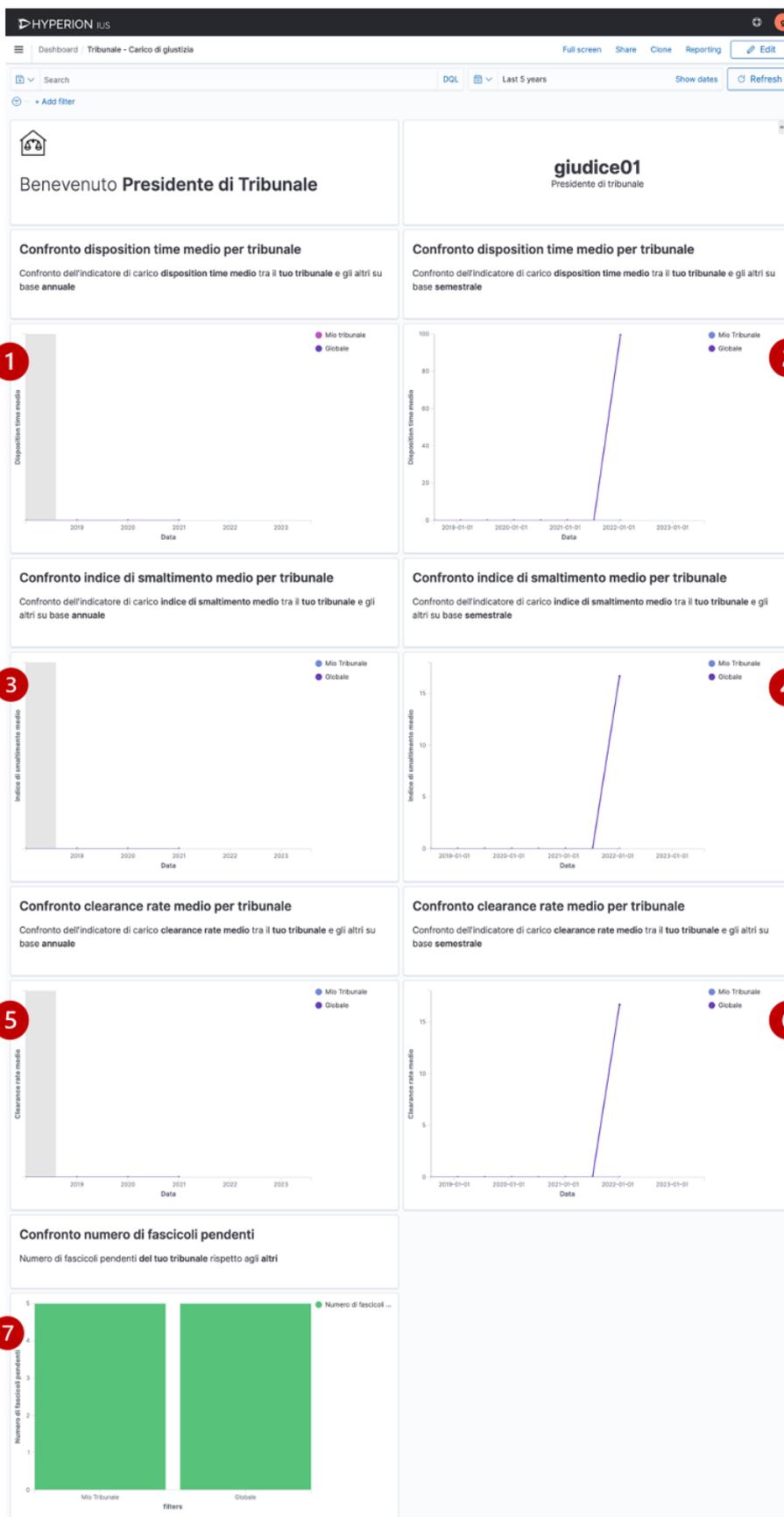


Figura 19 Grafici per il cruscotto Carico di giustizia (presidente tribunale)

1. Il grafico permette di confrontare il disposition time medio del proprio tribunale e lo stesso relativo agli altri tribunali su base annuale.
2. Il grafico permette di confrontare il disposition time medio del proprio tribunale e lo stesso relativo agli altri tribunali su base semestrale.
3. Il grafico permette di confrontare l'indice di smaltimento medio del proprio tribunale e lo stesso relativo agli altri tribunali su base annuale.
4. Il grafico permette di confrontare l'indice di smaltimento medio del proprio tribunale e lo stesso relativo agli altri tribunali su base semestrale.
5. Il grafico permette di confrontare il clearance rate medio del proprio tribunale e lo stesso relativo agli altri tribunali su base annuale.
6. Il grafico permette di confrontare il clearance rate medio del proprio tribunale e lo stesso relativo agli altri tribunali su base semestrale.
7. Il grafico permette di confrontare il numero dei fascicoli pendenti relativi al proprio tribunale e lo stesso relativo agli altri tribunali.

## Funzionalità avanzate

All'interno della piattaforma è possibile far uso di funzionalità avanzate per interagire con i grafici e coglierne dettagli fondamentali.

- **Focus**

Con riferimento al grafico 3 (bollino rosso) presente nella vista del giudice, che illustra un confronto tra la situazione personale e quella della sezione di appartenenza, posizionando il cursore su una qualsiasi barra, si vedrà comparire un box contenente le informazioni rappresentate dalla specifica barra come la fase, la materia e se il dato è relativo alla situazione personale (vedasi la dicitura "Io" evidenziata nella Figura 20) oppure alla sezione di appartenenza (vedasi la dicitura "Sezione" evidenziata nella Figura 21).

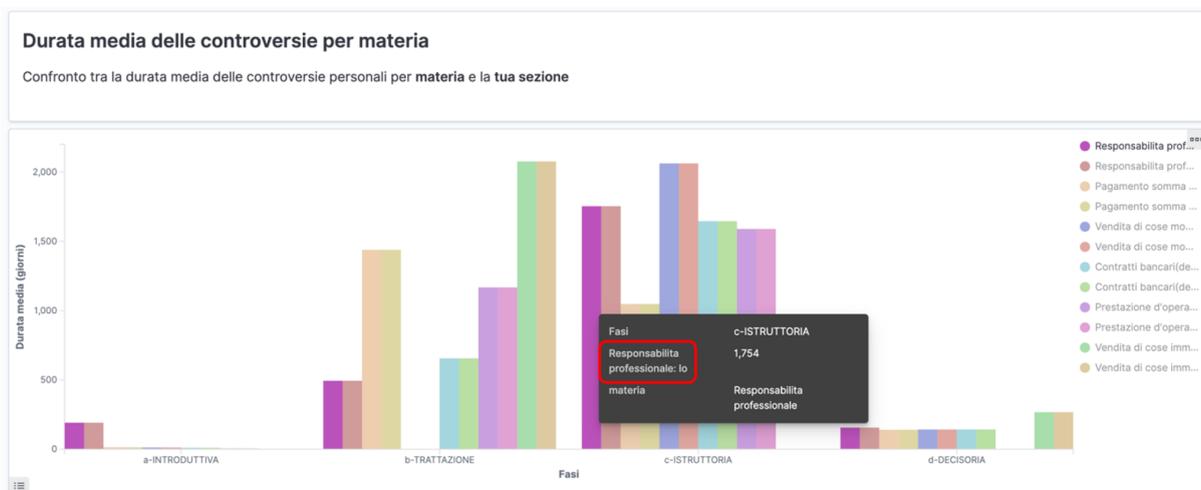


Figura 20 Esempio di informazioni avanzate



Figura 21 Esempio di informazioni avanzate

- **Modalità Full screen**

Per una visualizzazione più accurata e isolata rispetto agli elementi presenti nella schermata è possibile abilitare la modalità Full screen.

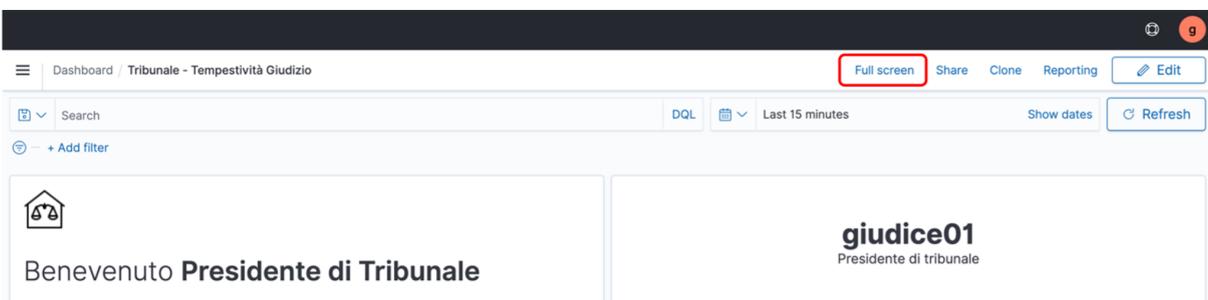


Figura 22 Accesso alla modalità full screen

- **Filtri**

È possibile isolare i dati di interesse cliccando su uno specifico grafico, in particolare sul dato di cui si vuole avere maggiori informazioni.

Con riferimento al grafico 3 (bollino rosso) presente nella vista del presidente di sezione, che illustra la ripartizione dei fascicoli tra i magistrati afferenti alla sezione presieduta, posizionando il cursore su una qualsiasi barra e cliccando in sua corrispondenza, si vedrà attivato un filtro (visibile in alto alla vista come in Figura 23) che isolerà quello specifico dato nascondendo tutti gli altri e che verrà applicato a tutti i grafici presenti nella vista analizzata.

Per rimuoverlo basta cliccare la x di fianco al filtro impostato.



Figura 23 Impostazione di un filtro

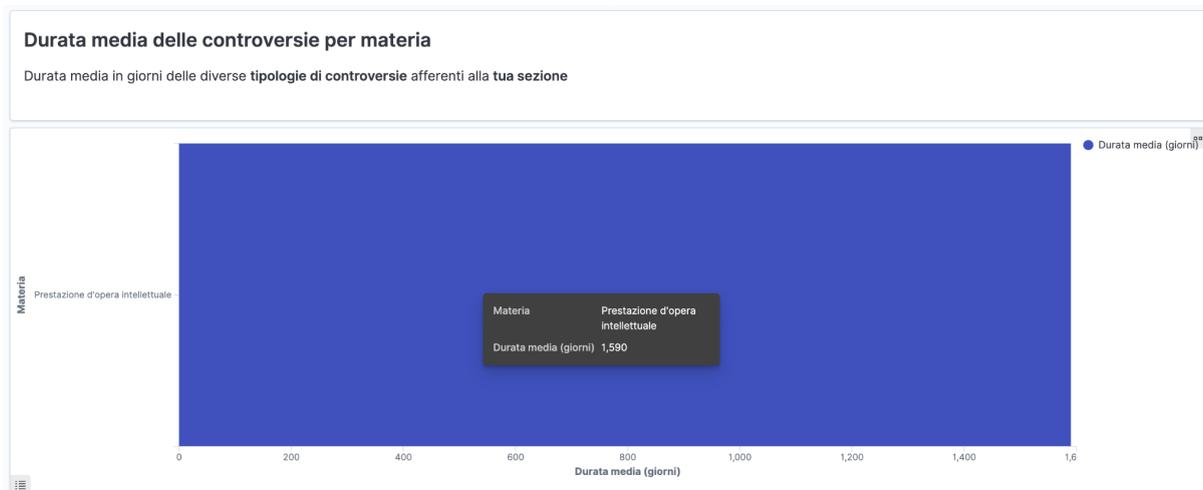


Figura 24 Risultato dell'applicazione del filtro

## Disconnessione dal software

Per disconnettersi dal sistema di monitoraggio cliccare sul pulsante in alto a destra come mostrato in Figura 25 e successivamente cliccare in corrispondenza di "Log out" come mostrato in Figura 26.

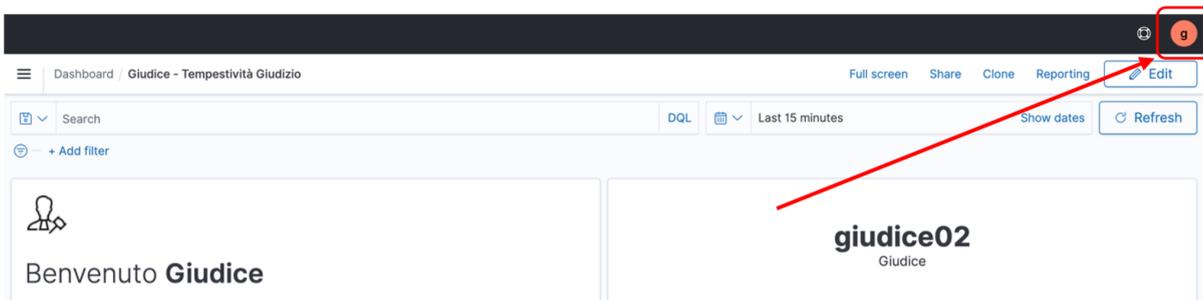


Figura 25 Esecuzione del log out

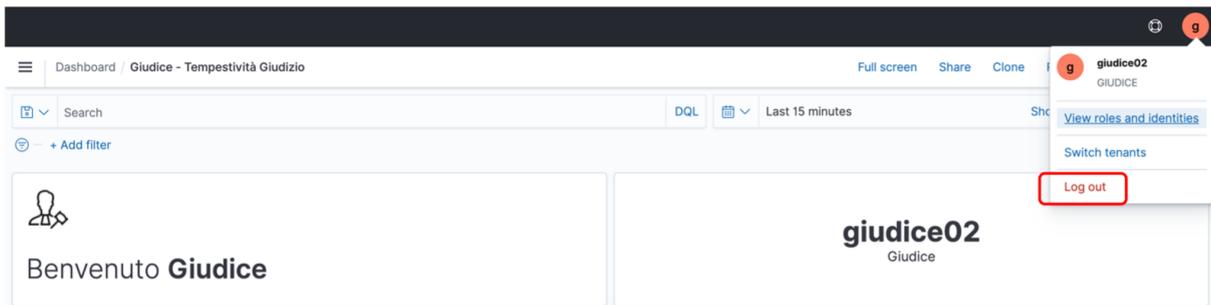


Figura 26 Esecuzione del log out

## Interfaccia utente - Visualizzazione dei report

A seguito della fase di login, come riportato in sezione [Accesso al software](#), si visualizzerà la schermata principale come in Figura 27.

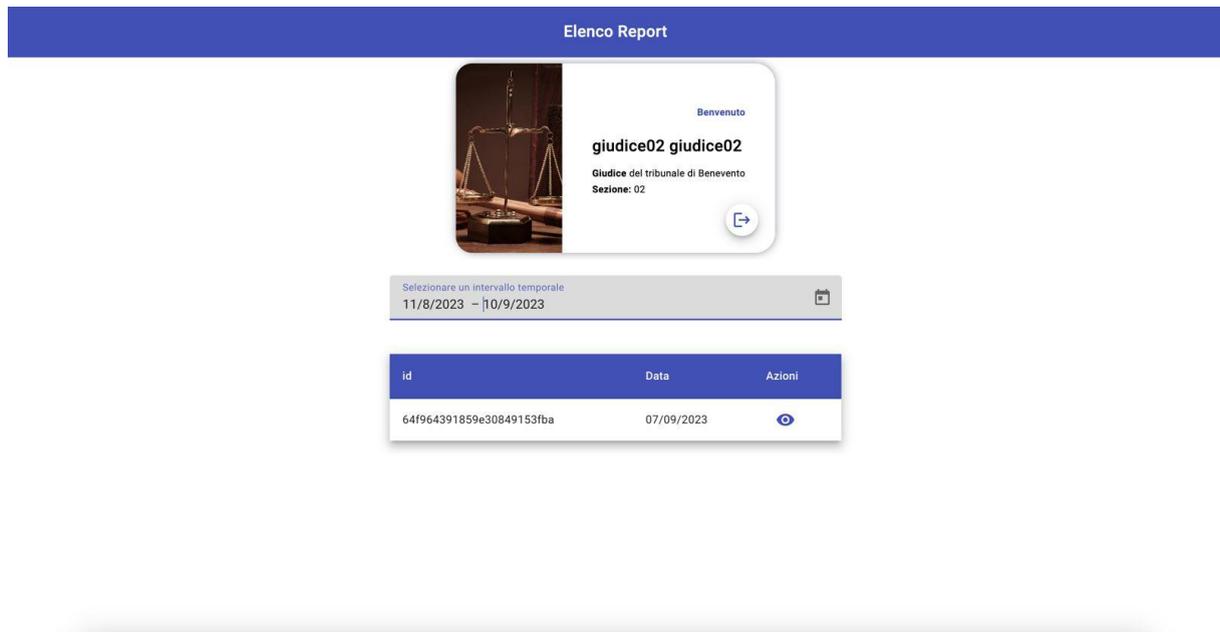


Figura 27 Schermata principale

A questo punto l'utente potrà specificare un preciso range temporale per la visualizzazione dei report disponibili, come mostrato in Figura 28.

Elenco Report

Benvenuto

**giudice02 giudice02**

Giudice del tribunale di Benevento  
Sezione: 02

Selezionare un intervallo temporale  
1/8/2023 -

AGO 2023

Azioni

D	L	M	M	G	V	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Figura 28 Selezione di uno specifico range temporale

Dalla schermata successiva verranno mostrati i report disponibili, come in Figura 29.

Benvenuto

**giudice02 giudice02**

Giudice del tribunale di Benevento  
Sezione: 02

Selezionare un intervallo temporale  
1/9/2023 - 10/9/2023

id	Data	Azioni
64f964391859e30849153fba	07/09/2023	

Dettaglio

Figura 29 Visualizzazione report disponibili

Di ogni report specifico (in base al ruolo dell'utente) è possibile visualizzare i dettagli, come riportato di seguito.

- **Vista dettagli report - Magistrato**

In Figura 30 è mostrato il report del Magistrato con i campi specifici.

**Elenco Report**

Benvenuto

**giudice02 giudice02**  
Giudice del tribunale di Benevento  
Sezione: 02

**Dettaglio Report**

ID	Fase attuale	Descrizione oggetto	Periodo trascorso rispetto alla media	Messaggio
026	a-INTRODUTTIVA	Responsabilita professionale	90	Warning! La fase ha superato la early threshold. Rischio di ritardo elevato.
023	b-TRATTAZIONE	Responsabilita professionale	29	La fase sta procedendo regolarmente.

651985401859e30849153fc2 29/05/2016

6519854d1859e30849153fc3 29/07/2016

Figura 30 Visualizzazione dettagli del report - Magistrato

- **Vista dettagli report - Presidente di Sezione**

In Figura 31 è mostrato il report del Presidente di Sezione con i campi specifici.

**Elenco Report**

Benvenuto

**giudice03 giudice03**  
Presidente di sezione del tribunale di Benevento  
Sezione: 02

**Dettaglio Report**

Codice oggetto	Descrizione codice oggetto	Fascicoli in ritardo	Fasi che generano ritardo
142002	Responsabilita professionale	30	b-TRATTAZIONE,c-ISTRUTTORIA
111002	Separazione giudiziale	10	b-TRATTAZIONE

6515195a1859e30849153fbd 29/05/2016

651519631859e30849153fbe 29/07/2016

Figura 31 Visualizzazione dettagli del report - Presidente di Sezione

- **Vista dettagli report - Presidente di Tribunale**

In Figura 32 è mostrato il report del Presidente del Tribunale con i campi specifici.

**Elenco Report**

Benvenuto

**Giudice01 Giudice01**  
Presidente di Tribunale del tribunale di Benevento  
Sezione: 01

**Dettaglio Report**

ID	Ritardo generato	Materie che generano ritardo	Codici oggetto che generano ritardo
01	20	Responsabilità professionale, Separazione giudiziale	142002,111002
02	30	Lesione personale, Appalto: altre ipotesi ex art. 1655 e ss. cc (ivi compresa l'azione ex 1669cc)	145002,140022

651519761859e30849153bf      27/09/2023

Figura 32 Visualizzazione dettagli del report - Presidente di Tribunale

### Disconnessione dal software - Visualizzazione dei report

Per effettuare il "Log out" è possibile cliccare sull'apposito pulsante, come evidenziato in Figura 33 (in rosso).

**Elenco Report**

Benvenuto

**giudice02 giudice02**  
Giudice del tribunale di Benevento  
Sezione: 02

11/8/2023 - 10/9/2023

id	Data	Azioni
64f964391859e30849153fba	07/09/2023	

Figura 33 Esecuzione del log out

## Appendice

### Architetture Lambda

Quando si lavora con set di dati di dimensioni molto grandi, può essere necessaria una notevole quantità di tempo per eseguire le query adatte ai requisiti dei client. Le query di questo tipo non possono essere eseguite in tempo reale e spesso richiedono algoritmi, come MapReduce, che operano in parallelo sull'intero set di dati. I risultati vengono quindi archiviati separatamente rispetto ai dati non elaborati e usati per ulteriori query.

Uno svantaggio di questo approccio è determinato dalla latenza. Se l'elaborazione richiede alcune ore, è possibile che una query restituisca risultati che risalgono a diverse ore prima. L'ideale sarebbe ottenere alcuni risultati in tempo reale, anche se con una perdita di precisione, e combinarli con quelli provenienti dall'analisi in batch.

L'**architettura lambda**, proposta per la prima volta da Nathan Marz, risolve questo problema creando due percorsi per il flusso di dati. Tutti i dati in ingresso nel sistema passano attraverso questi due percorsi:

- Un **livello di elaborazione batch** (percorso ad accesso sporadico) archivia tutti i dati in ingresso nella forma non elaborata ed esegue l'elaborazione batch sui dati. Il risultato di questa elaborazione viene archiviato come visualizzazione batch.
- Un **livello di elaborazione rapida** (percorso ad accesso frequente) analizza i dati in tempo reale. Questo livello è stato progettato per offrire una bassa latenza, a scapito della precisione.

Il livello di elaborazione batch confluisce in un **livello di gestione** che indicizza la visualizzazione batch per un'elaborazione efficiente delle query. Il livello di elaborazione rapida aggiorna in modo incrementale il livello di gestione in base ai dati più recenti.

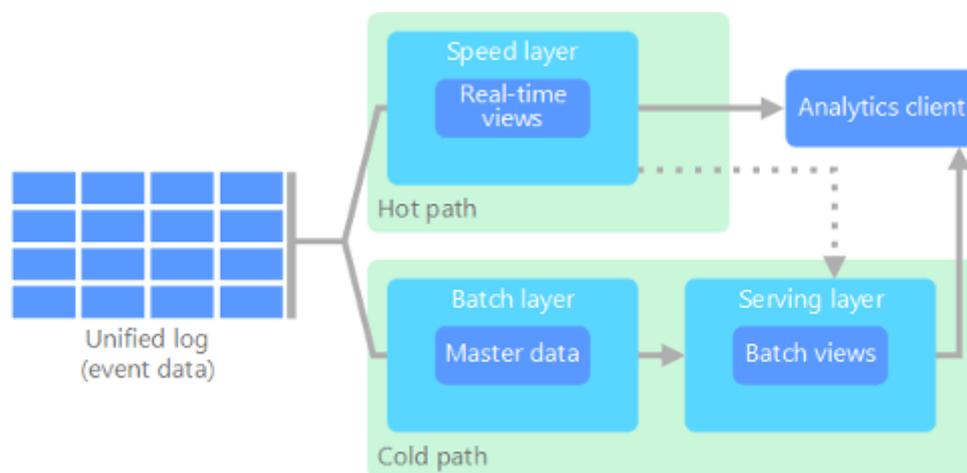


Figura 34 Architettura lambda

I dati che confluiscono nel percorso ad accesso frequente sono vincolati dai requisiti di latenza imposti dal livello di elaborazione rapida, in modo da essere elaborati il più velocemente possibile. È spesso necessario raggiungere un compromesso tra il livello di precisione dei dati e la velocità con cui risultano disponibili. Si consideri ad esempio uno scenario di Internet delle cose in cui un numero elevato di sensori di temperatura inviano dati di telemetria. Il livello di elaborazione rapida può essere usato per elaborare i dati in ingresso in una finestra temporale scorrevole.

Diversamente da questo flusso, i dati che confluiscono nel percorso ad accesso sporadico non sono soggetti agli stessi requisiti di bassa latenza. In questo modo si può eseguire anche un'elaborazione più precisa di grandi set di dati, che può però richiedere molto tempo.

I percorsi ad accesso frequente e sporadico convergono infine nell'applicazione client usata per l'analisi. Se il client deve visualizzare dati in tempo reale, e quindi immediatamente disponibili anche se meno precisi, acquisirà i risultati dal percorso ad accesso frequente. In caso contrario, selezionerà i risultati dal percorso ad accesso sporadico per visualizzare dati più precisi ma meno immediati. In altre parole, il percorso ad accesso frequente contiene dati per un intervallo di tempo relativamente breve, dopo il quale i risultati possono essere aggiornati con dati più precisi provenienti dal percorso ad accesso sporadico.

I dati non elaborati archiviati al livello di elaborazione batch non sono modificabili. I dati in ingresso vengono sempre aggiunti a quelli già esistenti, che non vengono mai sovrascritti. Le eventuali modifiche apportate al valore di un particolare dato vengono archiviate come un nuovo record di evento con timestamp. Ciò consente di eseguire in qualsiasi momento il ricalcolo sulla cronologia dei dati raccolti. La possibilità di ricalcolare la visualizzazione batch rispetto ai dati originali non elaborati è importante perché consente di creare nuove visualizzazioni di pari passo con l'evoluzione del sistema.

## Architetture Kappa

Uno svantaggio dell'architettura lambda è dato dalla sua complessità. La logica di elaborazione viene applicata a due percorsi diversi, ad accesso frequente e sporadico, usando framework distinti. La presenza di due percorsi ha l'effetto di duplicare la logica di calcolo e di rendere più complessa la gestione dell'architettura.

In alternativa all'architettura lambda, Jay Kreps ha proposto l'**architettura kappa**, con gli stessi obiettivi di base della precedente, ma una differenza importante. Tutti i flussi di dati passano attraverso un unico percorso, in base a un sistema di elaborazione di flussi.

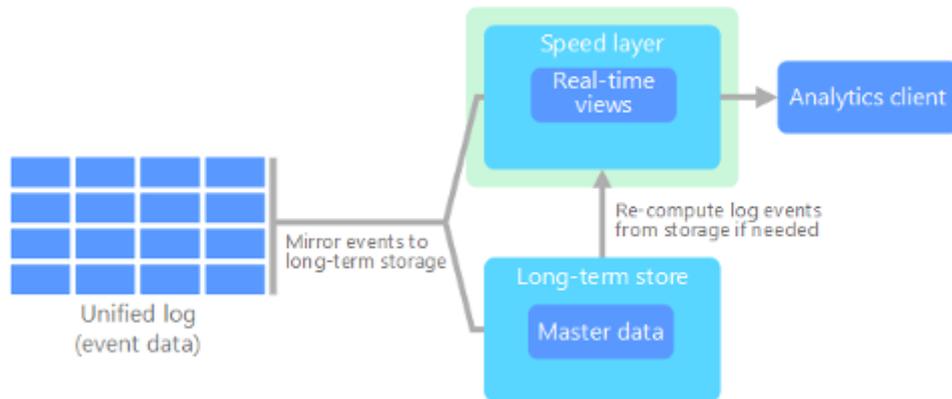


Figura 35 Architettura kappa

Questa architettura presenta alcune analogie con il livello di elaborazione batch dell'architettura lambda, in quanto i dati degli eventi non sono modificabili e vengono raccolti per intero, anziché in subset. I dati vengono inseriti come flusso di eventi in un unico log distribuito a tolleranza di errore. Gli eventi vengono ordinati e lo stato corrente di un evento viene modificato solo in caso di accadimento di un nuovo evento. In modo analogo al livello di elaborazione rapida dell'architettura lambda, l'intera elaborazione degli eventi viene eseguita nel flusso di input e salvata in modo persistente come visualizzazione in tempo reale.

Se è necessario ricalcolare l'intero set di dati, come avviene nel livello di elaborazione batch dell'architettura lambda, è sufficiente riprodurre il flusso. Per completare questo calcolo in modo rapido viene normalmente usato il parallelismo.

## Change Data Capture

Uno dei passaggi fondamentali per la creazione di un sistema di elaborazione Big Data è quello dell'estrazione, che consente di ottenere i dati dalle sorgenti. Una delle tecniche che è possibile utilizzare per effettuare questa fase è quella del Change Data Capture (CDC).

Con questo termine si intende il processo di identificazione e acquisizione delle modifiche apportate ai dati in un database e la consegna di tali modifiche in tempo reale a un processo o ad un sistema a valle. Tale processo consente, quindi, di acquisire ogni modifica dalle transazioni in un database di origine e di trasferirla in tempo reale ad una qualsiasi destinazione, mantenendo così i sistemi sincronizzati e fornendo una replica affidabile dei dati.

Vediamo adesso alcuni dei principali benefici del Change Data Capture. Questi possono essere riassunti nei seguenti punti:

- Eliminazione della necessità di trasferire i dati in maniera bulk in apposite finestre di sincronizzazione, consentendo di caricare i dati in maniera incrementale e in tempo reale;

- Limitazione dell'impatto sui sistemi esistenti, in quanto integrabile senza modifiche agli applicativi.

Esistono alcuni modi per implementare un sistema di Change Data Capture. Prima dell'introduzione delle funzionalità integrate per CDC nei principali DBMS, venivano utilizzate tecniche come la differenziazione delle tabelle, la selezione del valore di modifica e i trigger del database per acquisire le modifiche apportate a un database. Questi metodi, tuttavia, possono essere inefficienti o invadenti e tendono a creare un notevole sovraccarico sui server di origine. Questo è il motivo per cui i DBMS hanno adottato rapidamente funzionalità CDC integrate basate su log. Queste funzionalità utilizzano un processo in background per eseguire la scansione dei registri delle transazioni del database al fine di acquisire i dati modificati. Pertanto, le transazioni non sono interessate e l'impatto sulle prestazioni sui server di origine è ridotto al minimo. Vediamo la tecnica nel dettaglio.

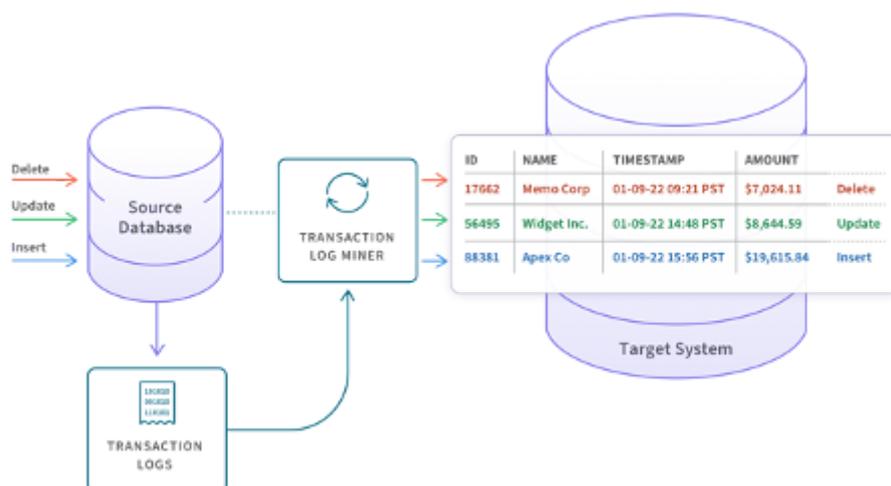


Figura 36 CDC con tecnica basata sui log

Come visibile nella Figura 36, quando una nuova transazione viene effettuata nel database, questa viene registrata in un file di registro (il log delle transazioni) senza alcun impatto sul sistema di origine. Questo log è poi continuamente analizzato da un processo di sincronizzazione che ha il compito di trasmettere le singole transazioni al sistema target.

## Considerazioni sull'impatto della riforma Cartabia sul sistema di monitoraggio

### Aspetti della riforma

La riforma Cartabia nasce dall'obiettivo concordato in sede europea di **avere un processo di "durata ragionevole"**. Dal rapporto della *Commissione Europea per l'Efficienza della Giustizia* (CEPEJ 2022), risulta che **il nostro Paese sia il più litigioso per cause civili**, detenendo i tristi primati del numero di processi e della loro **eccessiva durata**. Il rapporto CEPEJ evidenzia inoltre che il **numero degli avvocati in Italia rappresenta un quinto di quelli presenti in**

**Europa** (mentre la popolazione italiana è circa il 13% della popolazione complessiva della UE).

La **Riforma Cartabia** mira a **velocizzare i tempi del processo civile**, intervenendo:

- su alcuni aspetti dell'iter processuale;
- prevedendo un progressivo aumento della digitalizzazione dei processi.

La riforma Cartabia apporta modifiche alle c.d. **ADR (Alternative Dispute Resolution)**, **mediazione** e **negoziata assistita** (entrate in vigore dal **30 giugno 2023**). Alcuni aspetti della riforma hanno investito il processo civile **impattando sull'iter previsto dal Codice di procedura civile**.

Il **processo telematico** diventato realtà nel 2014, con la riforma Cartabia vedrà una **amplificazione della digitalizzazione**. Ciò porterà all'**accelerazione del processo civile**: il giuramento del consulente tecnico d'ufficio avverrà con la **firma digitale** e ci sarà la **notifica via PEC** per gli atti introduttivi del processo, che, in determinati casi, è ora obbligatoria. La passata oralità delle udienze in aula di tribunale è ora sostituita con l'obbligatorietà del **deposito in via telematica degli atti scritti, redatti in formato elettronico**. Anche le udienze orali in presenza sono state sostituite, in determinate condizioni, con l'**udienza da remoto, evitando così quei meri rinvii causati dall'impossibilità degli avvocati o giudici a presenziare all'udienza**.

Già ora, grazie al **processo telematico**, il **tasso di smaltimento dell'arretrato delle cause civili è positivo**. Attualmente, nello stesso anno, in tribunale e in corte d'appello **si chiudono con sentenza più cause rispetto alle nuove che vengono iscritte**. In cassazione, invece, resta superiore il numero di ingresso dei ricorsi, ma con la riforma si avrà una **riduzione dell'arretrato pendente**.

Per contrastare l'elevato numero delle controversie già dal 2010 è stata introdotta l'**obbligatorietà della mediazione: il cittadino prima di iniziare un processo**, ma solo in determinate cause, **è obbligato a tentare di risolvere la lite con una mediazione con l'altra parte in via stragiudiziale**, presso dei centri specializzati (ADR – *Alternativa alla Risoluzione delle Dispute*). Solo se il tentativo di accordo fallisce allora si può intentare la causa in tribunale.

## **Impatti della riforma Cartabia sul monitoraggio**

La riforma Cartabia verte principalmente sull'**abbreviamento del procedimento con la semplificazione del rito**. Per accorciare i tempi del processo si sono sintetizzate le fasi processuali chiedendo agli avvocati di giungere già alla prima udienza con la presentazione delle prove. È stato unificato il rito per separazioni e divorzi e, infine, sono stati **accorciati i tempi di pubblicazione della sentenza**.

L'elemento di maggiore interesse, dal punto di vista del monitoraggio dei procedimenti, consiste nel nuovo procedimento di primo grado, le **memorie di trattazione ed istruttorie** – che sino all'entrata in vigore della riforma erano successive alla prima udienza – **dovranno essere depositate anteriormente alla prima udienza**, con l'obiettivo di far sì che il Giudice, avendo già chiaro e compiutamente definito l'oggetto del procedimento, possa regolarne l'andamento nel modo più rapido ed efficace possibile, in relazione alla complessità della causa e dell'eventuale fase di assunzione dei mezzi di prova richiesti dalle parti. Tale modifica

influisce sulla prima udienza, la quale pertanto è stata collocata “a valle” di tale scambio degli atti introduttivi.

In relazione alla costituzione delle parti, la riforma modifica le tempistiche. Al fine di adeguare la fase introduttiva al nuovo impianto normativo, si è quindi stabilito che tra la data di notifica della citazione e la data dell’udienza debbano intercorrere termini liberi non inferiori a centoventi giorni (centocinquanta se la notifica deve essere effettuata all’estero), mentre prima della riforma il termine a comparire era di novanta giorni. Resta, invece, di dieci giorni dall’ultima notifica il termine per la costituzione dell’attore mediante iscrizione a ruolo della causa. Il convenuto, per contro, non si deve più costituire entro venti giorni dalla prima udienza, bensì settanta giorni prima dell’udienza di comparizione indicata nell’atto di citazione.

All’esito dello scambio delle memorie, il giudice:

- decide **sull’istanza di chiamata** in causa del terzo formulata dall’attore nella prima memoria integrativa e, in caso di ammissione, fissa nuova prima udienza per consentire la comparizione del terzo;
- ove ritenga la causa matura per la decisione **senza necessità di istruttoria**, anche solo su questioni pregiudiziali, la rimette avanti al Collegio o la trattiene avanti a sé per la fase decisoria;
- qualora ne reputi sussistenti i presupposti, dispone **con ordinanza non impugnabile** che il giudizio prosegua con le **forme del rito semplificato** ex art. 281 *decies* ss., C.p.c.;
- laddove la causa richieda **l’espletamento di attività istruttoria**, decide in udienza (o con ordinanza resa entro i trenta giorni successivi) sulle istanze istruttorie delle parti e **stabilisce il calendario delle udienze successive** con i relativi incumbenti, tenendo conto della natura, dell’urgenza e della complessità della causa, **ma fissando comunque la prima udienza per l’assunzione di mezzi di prova entro novanta giorni**.

## **Adeguamenti per il monitoraggio dei processi aderenti alla riforma**

Onde tenere in conto gli impatti individuati occorre ridisegnare il workflow esplicito del Processo a Cognizione Piena con particolare riferimento alla prima fase per tenere conto delle modifiche alla costituzione delle parti, che può essere ridefinita come di **Instaurazione**. A seguito in tale fase si entra in fase di **Trattazione**, denominata come in precedenza ma con delle modifiche da apportare nella necessità di anticipazione del deposito delle memorie, in posizione precedente alla prima udienza. Il processo può terminare, in questa fase se il giudice ha tutti gli elementi necessari alla decisione.

Se, invece, è necessario acquisire prove, documentali o testimoniali, si pianifica un calendario delle udienze e si entra in fase **Istruttoria**. Questa fase mantiene la struttura e le tempistiche già in essere, con una accelerazione delle attività conseguenti una maggiore enfasi della digitalizzazione, ad esempio il giuramento del consulente tecnico d’ufficio avverrà con la **firma digitale**, in modo più rapido. Inoltre, la passata oralità delle udienze in aula di tribunale è ora sostituita con l’obbligatorietà del deposito in via telematica degli atti scritti, redatti in formato elettronico. Anche le udienze orali in presenza sono state sostituite, in determinate

condizioni, con l'udienza da remoto, evitando così rinvii causati dall'impossibilità degli avvocati o giudici a presenziare all'udienza.

Il workflow termina con la fase **Decisoria**, che mantiene la sua struttura attuale ma per la quale si modificano i tempi limite previsti per i diversi modelli decisionali contemplati.

In sintesi, il workflow va adeguato alle seguenti fasi:

1. **Instaurazione** (sostituisce la precedente introduttiva e va riorganizzata per quanto riguarda le tempistiche).
2. **Trattazione** (va riorganizzata anticipando il deposito delle memorie rispetto alla prima udienza). In analogia alla precedente può portare alla fase Istruttoria (che rimane opzionale), entrare in fase decisoria, passare ad una procedura di rito abbreviato oppure terminare con una conciliazione, oltre a tutti gli altri esiti già previsti dal workflow attuale.
3. **Istruttoria**. Rimane praticamente inalterata.
4. **Decisoria**. La struttura rimane inalterata cambiano i limiti temporali.