



**Università degli Studi di Roma "Tor Vergata"**  
**Dipartimento di Fisica**

**Sviluppo di un sistema di alerting e  
monitoring per lo SpaceWeather  
nell'ambito della missione  
CSES-Limadou**

**Tesi di Laurea in Fisica**

Anno Accademico 2020/2021

**Relatore:**

Prof.ssa Roberta Sparvoli

**Correlatori:**

Dott. Matteo Mergè

Dott. Alessandro Sotgiu

**Candidato:**

Roberto Lulli

---

## Introduzione

La presente tesi descrive la mia attività nell'ambito del Esperimento *China Seismo-Electromagnetic Satellite* (CSES) ed in particolare nella progettazione e realizzazione di un sistema di monitoraggio per i dati provenienti da uno dei suoi payload, il rivelatore di particelle ad alta energia HEPD (*High Energy Particle Detector*).

La missione spaziale CSES nasce da una collaborazione internazionale tra la Cina e l'Italia, in particolare tra la CNSA (*China National Space Administration*) e l'ASI (*Agenzia Spaziale Italiana*), e mira al monitoraggio delle perturbazioni nella ionosfera, la magnetosfera e le fasce di radiazione di Van Allen originate da sorgenti elettromagnetiche interne ed esterne alla cavità geomagnetica, da raggi cosmici e da eventi solari e allo studio di possibili correlazioni con eventi sismici.

Per questi scopi, il satellite cinese ospita diversi strumenti per la misura delle componenti del campo elettrico e magnetico, la misura dei disturbi del plasma e la misura dei flussi energetici di particelle e spettri. HEPD è l'unico strumento scientifico non cinese ed è stato sviluppato dalla componente italiana della missione CSES denominata *Limadou* (che include le Sezioni INFN di Bologna, Napoli, Perugia, Roma Tor Vergata e Frascati, oltre alle Università di Bologna, Trento, Roma Tor Vergata e agli Istituti INAF-IAPS e INGV).

Il rivelatore HEPD, descritto nel dettaglio nel prossimo capitolo, è composto da una serie di sotto-rivelatori progettati per misurare i flussi di elettroni, protoni e nuclei leggeri cosa che congiunta con l'elevato angolo di inclinazione della sua orbita gli consente di discriminare la componente galattica dalla componente intrappolata dei raggi cosmici o dalle SEPs mediante una precisa ricostruzione della direzione della particella incidente. A questo scopo i sotto-rivelatori che compongono HEPD sono un tracker al silicio, un piano di trigger, un calorimetro elettromagnetico e un sistema Veto. Lo strumento è ottimizzato per studiare i flussi di elettroni e protoni nell'intervallo di energia rispettivamente da 3 MeV a 100 MeV e da 30 MeV a 250 MeV. Il suo obiettivo scientifico è quello di studiare i raggi cosmici di bassa energia, la stabilità delle fasce di Van Allen e fenomeni Solari come CME e SEPs nonché il monitoraggio della modulazione solare dei raggi cosmici. Nel corso del tempo il rivelatore HEPD si sta dimostrando un valido strumento anche per lo studio dello Space Weather. Proprio in quest'ultimo ambito poter disporre di una piattaforma *operativa* di accesso ai dati che permetta rapidamente: interrogazione, visualizzazione, allarmistica e monitoraggio costante è un passaggio fondamentale per l'acquisizione di informazioni sui

---

fenomeni oggetto di studio così come sull'individuazione dei data set più interessanti su cui concentrare un'analisi approfondita.

## Contributo dell'autore

Abbiamo già accumulato una quantità notevole di dati a terra in questi quattro anni. L'analisi efficiente ed efficace dei dati è, in generale, il cuore di una missione. È il fattore che permette secondo la piramide *DIKW* (Data, Information, Knowledge, Wisdom) di passare dal *dato grezzo* all'*informazione* e da questa alla *conoscenza*. È fondamentale quindi possedere gli strumenti adeguati per un attento monitoraggio che possano mettere in luce in modo veloce i dataset più interessanti, le correlazioni fra i dati, andamenti e anomalie. Così come la possibilità di confrontare i dati con quelli provenienti da altre sorgenti in modo da sfruttare a pieno la possibilità di integrarli con diversi studi o missioni ed aumentare il livello di comprensione dei fenomeni ad essi associati.

In particolare oggi i dati di livello L3<sup>1</sup>4.1 di HEPD sono largamente sottoutilizzati. Spesso sono consultati con una ricerca manuale solo a seguito di una segnalazione esterna su una eventuale anomalia. Non esiste una piattaforma di visualizzazione integrata dei dati né un sistema automatico di individuazione dei dataset potenzialmente interessanti. Così come non esiste un sistema di alert su anomalie e non c'è la possibilità di condividere agevolmente la visualizzazione a seguito di una prima analisi.

Il mio personale contributo è stata la realizzazione di un sistema di monitoraggio ed alert per i dati di livello L3 di HEPD in grado di rispondere a questi requisiti. Non di meno il sistema è in grado di collezionare facilmente dati provenienti da altri payload di CSES, come il rivelatore HEPP (*High Energy Particle Package*) e, in generale, di poter essere collegato, senza un effort eccessivo, a qualsiasi sorgente di dati di tipo *time series*.

Il sistema è costituito da una *pipeline* di tre componenti:

- un codice scritto in Python che si occupa di estrarre i dati dai nuovi dataset, formattarli opportunamente preparandoli per l'inserimento nel componente successivo;

---

<sup>1</sup>I dati del rivelatore subiscono diverse fasi di processamento a partire dal dato grezzo e sono quindi disponibili in differenti formati, da L0 ad L3. In particolare nel livello L3 troviamo le particelle nettamente separate in protoni ed elettroni e un'integrazione su intervalli di un secondo dei conteggi e degli altri valori associati (coordinate, campo magneti, etc...).

- 
- una *time series platform* basata su *InfluxDB*[1] ottimizzata per gestire dati di tipo *time series* che incorpora delle API complete per la gestione dei dati e un potente linguaggio di query tramite il quale è possibile effettuare operazioni e trasformazione anche complesse sui dati stessi.
  - un'interfaccia di visualizzazione di alto livello basata su *Grafana*[2] in grado di comunicare via API con *InfluxDB* e a sua volta anche di operare trasformazioni sui dati tramite filtri opportuni. Tutto poi viene rappresentato in *Grafana* attraverso la costruzione personalizzata di *dashboard* e *pannelli* di diversa natura (grafici, statistiche, mappe etc...)

I dati della missione CSES costituiscono anche un servizio di condivisione osservativa per la comunità scientifica. Il mio lavoro e il mio contributo vanno proprio in questo verso.

## Outline della tesi

Nei primi due capitoli di questa tesi introduco la missione CSES.

Il capitolo 1 illustra gli obiettivi scientifici della missione, il contributo della componente italiana, la descrizione generale del satellite con riferimento a tutti i payloads e la descrizione delle sue orbite.

Il capitolo 2 è dedicato interamente al rivelatore HEPD. Ne descrivo gli obiettivi scientifici specifici fornendo anche un'introduzione all'environment nel quale lo strumento opera riferendomi alla fisica terrestre, solare e allo space weather. Affronto poi una descrizione dettagliata del rivelatore HEPD e di tutti i sotto-rivelatori, le caratteristiche fisiche e le configurazioni nelle quali può lavorare.

Nel capitolo 3 fornisco una introduzione ai sistemi di monitoraggio illustrando le varie tipologie di sistemi e le loro complessità. Introduco il processo che porta dal dato grezzo alla conoscenza e i concetti base delle *time series*.

Nel capitolo 4 si trova la parte fondamentale del mio contributo: la progettazione e l'implementazione del sistema di monitoraggio.

Parto dai requisiti del sistema introducendo la sua architettura e la scelta conseguente del software. Nella sezione successiva illustro il codice sviluppato da me per l'estrazione e la formattazione dei dati, descrivo poi nel dettaglio il funzionamento del software, il linguaggio di query e la costruzione delle dashboard, dei pannelli e delle funzioni per il monitoring. Nell'ultima sezione introduco due esempi di eventi anomali mettendo in luce dei casi d'uso che mostrano l'efficacia del sistema.

---

Il capitolo 5 è dedicato alle potenzialità di questo sistema e alle sue caratteristiche orientate alla collaboratività, la flessibilità e scalabilità. Dedico una sezione agli sviluppi nel breve e medio termine e una alle prossime collaborazioni. Introduco poi brevemente la missione CSES-Limadou 2 e chiudo il capitolo descrivendo la natura cloud di questo servizio.



# Indice

<b>1</b>	<b>La missione CSES</b>	<b>15</b>
1.1	Panoramica . . . . .	15
1.2	Obiettivi scientifici della missione . . . . .	18
1.3	Il contributo italiano: CSES-Limadou . . . . .	23
1.4	Il Satellite CSES-01 . . . . .	24
1.4.1	Strumenti scientifici a bordo . . . . .	25
<b>2</b>	<b>Il rivelatore HEPD</b>	<b>29</b>
2.1	Gli obiettivi scientifici di HEPD . . . . .	29
2.1.1	Raggi Cosmici e Fisica Solare . . . . .	30
2.1.2	Studi sulla Magnetosfera . . . . .	32
2.1.3	Precursori a breve termine di terremoti e burst di particelle . . . . .	32
2.1.4	Space Weather . . . . .	35
2.1.4.1	Brillamenti (flares) . . . . .	35
2.1.4.2	Eruzioni di massa coronale (CME) . . . . .	37
2.2	Descrizione fisica del rivelatore . . . . .	39
2.3	Il Tracker . . . . .	41
2.4	Il sistema di TRIGGER . . . . .	42
2.5	Il Calorimetro . . . . .	44
2.5.1	Gli Scintillatori Plastici . . . . .	45
2.5.2	La matrice LYSO . . . . .	46
2.5.3	I tubi fotomoltiplicatori . . . . .	47
2.6	Il sistema VETO . . . . .	47
2.7	Configurazioni di trigger . . . . .	48
<b>3</b>	<b>I sistemi di monitoraggio</b>	<b>51</b>
3.1	Introduzione . . . . .	51
3.2	Tipologie e complessità . . . . .	52
3.3	Time series . . . . .	54

---

<b>4</b>	<b>Realizzazione del sistema</b>	<b>59</b>
4.1	Identificazione dei Requisiti . . . . .	59
4.2	Introduzione alla pipeline . . . . .	61
4.3	Infrastruttura . . . . .	63
4.3.1	Autenticazione ed autorizzazione . . . . .	65
4.4	Scelta del software . . . . .	66
4.5	Python agent . . . . .	67
4.6	TSDB . . . . .	70
4.7	InfluxDB . . . . .	71
4.7.1	Raccolta e scrittura dei dati . . . . .	72
4.7.2	Il linguaggio di Query: Flux . . . . .	73
4.8	Grafana . . . . .	78
4.8.1	Configurazione . . . . .	80
4.8.2	Dashboard e Panels . . . . .	81
4.8.3	Sistema di Alerting . . . . .	100
4.9	Use case . . . . .	105
<b>5</b>	<b>Roadmap e casi d'uso</b>	<b>111</b>
5.1	Integrazioni e miglioramenti a breve termine . . . . .	111
5.2	Collaborazioni . . . . .	112
5.2.1	CSES-02 . . . . .	113
5.3	Sistema o servizio? - SaaS-MaaS . . . . .	114
	<b>Conclusioni</b>	<b>115</b>
	<b>Codice</b>	<b>117</b>
	<b>Bibliografia</b>	<b>123</b>

# Elenco delle figure

1.1	<i>Il moto delle particelle intrappolate è la sovrapposizione di tre moti: il moto di rotazione attorno alle linee di campo magnetico [a], il moto di oscillazione lungo le linee di campo [b] e il moto di deriva intorno alla Terra (quella indicata è relativa agli elettroni, per i protoni la direzione è opposta) [c].</i>	22
1.2	<i>Loss cone: Se il pitch angle della particella si trova all'interno del loss cone (<math>\alpha &lt; \alpha_{LC}</math>) il suo punto di mirror risulterà troppo interno e la particella verrà assorbita dalle collisioni con gli atomi dell'atmosfera.</i>	22
1.3	<i>Schema che illustra la precipitazione di particelle intrappolate dovuta all'interazione con EMEs di origine sismica. Tratto da Sgrigna 2001 [6]</i>	23
1.4	<i>Schema del satellite CSES e del posizionamento dei payload</i>	25
1.5	<i>ground track del satellite: la distanza tra due tracce vicine nell'arco di un solo giorno è di circa 2650 km, ma nel periodo di 5 giorni si riduce a 530 km.</i>	26
1.6	<i>Comparazione tra i flussi di protoni (sopra) ed elettroni (sotto) misurati da PAMELA. Le regioni in blu si riferiscono ai range di energia di HEPD.</i>	27
2.1	<i>HEPD montato a bordo di CSES. La finestra di HEPD è coperta da uno strato di Pyralux (rame e kapton) per assicurare un buon isolamento termico</i>	30
2.2	<i>Istogrammi della differenza di tempo <math>\Delta T</math> tra il tempo del terremoto selezionato e il tempo del burst di particelle ottenuto dalle missioni MARIA, ELECTRON, GAMMA-1 e SAMPEX. Un valore positivo del picco indica che il burst precede nel tempo il terremoto. I grafici sono da [29].</i>	34
2.3	<i>Rilascio energetico durante una flare. Si possono vedere le 4 fasi (preheating, impulsive, explosive e hot).</i>	37
2.4	<i>Rappresentazione della curvatura delle linee di campo magnetico del Sole a causa della sua rotazione differenziale. Si può vedere che in prossimità dell'equatore solare si forma una regione di cambiamento della polarità del campo magnetico chiamata «heliospheric current sheet».</i>	37
2.5	<i>Una eruzione di massa coronale (SOHO/LASCO)</i>	39
2.6	<i>Sviluppo di una CME. La riconnessione magnetica crea una protuberanza che si separa dal campo magnetico solare permettendo il rilascio del plasma contenuto all'interno.</i>	39

2.7	<i>L'involucro in alluminio del rivelatore HEPD. Sono mostrati anche i connettori per l'interfaccia del satellite.</i>	41
2.8	<i>Rappresentazione schematica dell'interno del box di HEPD (a) e foto presa all'incirca dalla stessa prospettiva (b)</i>	41
2.9	<i>Schema che mostra il posizionamento del tracker (a). Fotografia dei piani di silicio del tracker assemblati (b)</i>	43
2.10	<i>Schema del sensore a micro-strip di silicio doppia faccia. Una particella carica attraversa il sensore e genera coppie <math>e^-</math> /lacuna che migrano verso direzioni opposte.</i>	43
2.11	<i>Disegni meccanici del ladder in silicio di HEPD: lato S (a) e lato K (b). Si possono vedere i due sensori e il circuito ibrido di lettura.</i>	43
2.12	<i>Schema che mostra il posizionamento del trigger plane (a). Immagine del piano di trigger (b).</i>	44
2.13	<i>Schema meccanico della torre di scintillatori che contiene i piani di scintillatore plastico, i tubi fotomoltiplicatori agli angoli e la struttura meccanica (a). Fotografia della torre assemblata (b).</i>	45
2.14	<i>Immagine di un piano scintillatore della torre del calorimetro.</i>	46
2.15	<i>Schema che mostra il posizionamento della matrice LYSO (a). Immagine del piano della matrice LYSO (b).</i>	47
2.16	<i>Schema che mostra il posizionamento del sistema veto (a). Schema del pannello inferiore e due di quelli laterali (b).</i>	48
3.1	<i>Dashboard di un sistema di monitoraggio complesso in tempo reale</i>	54
3.2	<i>Esempio di istogramma - cross-sectional</i>	56
3.3	<i>Esempio di grafico - time series</i>	57
4.1	<i>Schema semplificato della pipeline del sistema. I dati vengono prelevati dall'agent e inseriti nel database (InfluxDB). Vengono poi resi disponibili per l'elaborazione grafica (Grafana) tramite API e presentati via web all'utente.</i>	63
4.2	<i>InfluxDB capacity</i>	64
4.3	<i>Schema del meccanismo di autenticazione ed autorizzazione al servizio</i>	66
4.4	<i>HDF5 file, organizzazione gerarchica in gruppi e dataset</i>	68
4.5	<i>Un file hdf5 del rivelatore HEPD. A sinistra è visualizzato un gruppo con all'interno i vari dataset. A destra i dati all'interno del dataset e i metadati che lo descrivono.</i>	69
4.6	<i>Diagramma di flusso del python agent</i>	70
4.7	<i>InfluxDB Big Picture</i>	74
4.8	<i>Lo stream di tabelle risultato della query</i>	77
4.9	<i>configurazione su InfluxDB degli elementi di base per la connessione di Grafana ad InfluxDB. Il token assegnato a grafana per la connessione verso i bucket è in sola lettura.</i>	80

4.10	Configurazione su Grafana di una data source di tipo InfluxDB con linguaggio Flux, autenticazione tramite l'utente grafana al bucket hepd_data tramite il relativo token in sola lettura.	81
4.11	Creazione di una nuova dashboard e di un pannello (a dx) collegato alla data source InfluxDB. Nell'editor una query sul bucket hepd_data che filtra il solo conteggio dei protoni.	82
4.12	Dashboard variable che può assumere il valore degli ultimi 100 dataset risultante dalla query.	83
4.13	Il Pannello mostra il conteggio dei protoni una volta selezionato il file dal menu in alto a sinistra.	83
4.14	Costruzione del pannello con i valori massimi in un dataset	84
4.15	Pannello con la sovrapposizione dei grafici dei tre dataset.	84
4.16	Due pannelli con la sovrapposizione dei grafici di latitudine (sopra) e longitudine (sotto) su quelli di conteggio.	85
4.17	«Latest datasets» dashboard dove sono presenti tutti i pannelli descritti sopra.	86
4.18	Top ten protoni	87
4.19	Top ten protoni fuori dalla SAA	88
4.20	Top ten dashboard	89
4.21	Creazione del data link (sopra), cliccando su uno specifico data set sul grafico compare la visualizzazione del link (sotto).	90
4.22	Serie storica del conteggio degli elettroni per i data set relativi agli ultimi 15 giorni con riduzione dei dati attraverso le funzioni aggregateWindow() e mean().	92
4.23	Dynamic aggregate data dashboard.	93
4.24	Grafico del conteggio dei protoni correlato a quello del valore di latitudine.	94
4.25	Grafico sovrapposto Protoni-Elettroni	95
4.26	Pannelli del rate di conteggio dei protoni globale (sopra) e escludendo la SAA (sotto)	95
4.27	Flat dashboard. In questo caso i pannelli sono collegati ad un bucket con punti privi di tag set.	96
4.28	Configurazione di un dashboard link.	98
4.29	Geo Map relativa ad uno specifico dataset richiamata direttamente dalla dashboard «Latest datasets». In rosso il ground track del satellite. Spostandosi con il mouse sul tracciato è possibile vedere il dettaglio dei punti.	99
4.30	Heat Map della distribuzione geografica del conteggio degli elettroni di più orbite sommate.	99
4.31	Home dashboard.	101
4.32	Configurazione di una soglia visibile sul grafico agendo direttamente sulla configurazione del pannello.	102
4.33	Interfaccia di gestione centralizzata delle regole di alert.	102
4.34	Configurazione alert rule 1-2	104

---

4.35	Configurazione alert rule 3-4 . . . . .	104
4.36	Solar flare di classe M4 registrato il 2 aprile 2022. Dati <a href="https://spaceweather.com">spaceweather.com</a> . . . . .	106
4.37	Pannello Top Ten per il conteggio di protoni fuori dalla SAA (sopra). Dal link in alto a dx si può accedere alla Flat dashboard (sotto). . . . .	106
4.38	Correlazione tra i picchi del flare e quelli del valore di latitudine. . . . .	107
4.39	Sovrapposizione dei grafici elettroni-protoni per l'evento del 2 aprile 2022. . . . .	107
4.40	Plot e immagine del flare di classe X1 del 20/10/2022. Dati <a href="https://SpaceWeatherLive.com">SpaceWeatherLive.com</a> . . . . .	108
4.41	Top ten panel del conteggio di protoni nel periodo relativo al flare di classe X1. . . . .	108
4.42	Visualizzazione della correlazione protoni-elettroni nel periodo relativo al flare di classe X1. . . . .	109
4.43	Grafico del rate di conteggio dei protoni fuori dalla zona dell'Anomalia del Sud Atlantico. . . . .	109

# Elenco dei listati

4.1	<i>Esempio di una semplice query</i>	75
4.2	<i>Query di creazione del pannello in Fig. 4.15</i>	84
4.3	<i>highestMax()</i>	87
4.4	<i>Query per la creazione di una top ten dei file che presentano il più alto conteggio/s di protoni.</i>	87
4.5	<i>Top ten fuori dalla zona SAA</i>	89
4.6	<i>Query di creazione di un pannello con la media dei conteggi degli elettroni calcolata all'interno di una finestra di tempo dinamica.</i>	92
4.7	<i>Query di creazione del pannello di Fig. 4.30</i>	99
4.8	<i>Esempio di Alert query sul conteggio dei protoni con esclusione della zona SAA</i>	105
5.1	<i>Python agent</i>	117
5.2	<i>docker-compose.yml</i>	121



# 1 La missione CSES

La missione CSES (China Seismo-Electromagnetic Satellite) è un programma scientifico dedicato al monitoraggio del campo e delle onde elettromagnetiche, delle perturbazioni del plasma e delle particelle dell'atmosfera, ionosfera e magnetosfera indotte da sorgenti naturali ed emettitori antropogenici, al fine di studiarne le correlazioni con il verificarsi di fenomeni sismici. Più in generale, la missione CSES indaga sulla struttura e le dinamiche della ionosfera, i meccanismi di accoppiamento con gli strati di plasma inferiori e superiori e le variazioni temporali del campo geomagnetico, in condizioni di quiete e di disturbo. I dati raccolti dalla missione permettono quindi di studiare anche le interazioni sole-terra e i fenomeni della fisica solare, in particolare le *eruzioni di massa coronale* (CME - eruzione di plasma e campo magnetico emessi dalla corona solare), i *brillamenti solari* (*Flares* - improvvisi burst di particelle osservati sulla superficie del Sole) e la *modulazione solare dei raggi cosmici*. In effetti grazie alla suite di strumenti a bordo e alla precisione delle misure, CSES sta lavorando come ottimo strumento in ambito *space weather* [3].

La missione prevede il lancio di una costellazione di satelliti nei prossimi anni che avranno il compito di monitorare in modo sempre più completo e accurato l'ambiente del *vicino spazio* terrestre. Il lancio del primo di questi satelliti, CSES-01, è avvenuto il 2 febbraio 2018 dal Jiuquan Satellite Launch Centro situato nel deserto del Gobi, Mongolia Interna, per mezzo del veicolo cinese per lanci orbitali *Long March 2D*. Il secondo satellite, CSES-02 dovrebbe partire entro dicembre 2022.

Il primo capitolo di questa tesi presenta le caratteristiche generali del satellite CSES-01 e una panoramica della missione CSES, con particolare attenzione al rivelatore italiano HEPD e ai suoi obiettivi scientifici.

## 1.1 Panoramica

La missione CSES fa parte di un programma di collaborazione tra la *China National Space Administration* (CNSA) e l'*Agenzia Spaziale Italiana* (ASI), e sviluppato dalla Chi-

na *Earthquake Administration* (CEA) e l'*Istituto Nazionale di Fisica Nucleare* (INFN), insieme a diverse università e istituti di ricerca cinesi e italiani. CSES è il primo satellite di un sistema di monitoraggio spaziale, proposto per indagare la ionosfera con le tecniche e le apparecchiature più avanzate e progettato per raccogliere dati su scala globale dell'ambiente elettromagnetico vicino alla Terra.

L'accoppiamento litosfera-atmosfera-ionosfera è un argomento complesso che coinvolge molti effetti fisici e interazioni che si verificano dalla superficie terrestre fino alla magnetosfera. L'indagine di tali meccanismi di accoppiamento, ed in particolare del comportamento parzialmente sconosciuto della regione di transizione iono-magnetosfera, è di fondamentale importanza per il telerilevamento terrestre, il monitoraggio dell'ambiente elettromagnetico vicino alla Terra e lo studio dei rischi naturali. Gran parte di questi effetti è causata da emissioni elettromagnetiche naturali non sismiche e antropiche, ma uno degli obiettivi della missione è anche lo studio delle perturbazioni elettromagnetiche associate all'attività sismica che possono produrre perturbazioni ionosferiche e precipitazioni di particelle dalle *Fasce di Van Allen*, osservate prima, durante e dopo terremoti di media e forte magnitudo. Questi fenomeni devono essere distinti da quelli indotti dalle sorgenti esterne alla cavità geomagnetica e dagli eventi atmosferici. In effetti, un ruolo importante nel controllare la dinamica della ionosfera superiore è giocato dal Sole che genera variazioni (quasi-regolari e irregolari) dei parametri della litosfera-ionosfera-magnetosfera attraverso eventi impulsivi come espulsioni di massa coronale solare e brillamenti solari, nonché dall'attività troposferica (ad esempio fulmini). Sono state eseguite molte osservazioni sperimentali e proposti modelli teorici per analizzare e discriminare i disturbi ionosferici causati da eventi naturali terrestri, come terremoti ed eruzioni vulcaniche, e da attività antropiche.

Lo studio delle condizioni fisiche che danno origine a un terremoto, nonché dei processi che precedono una rottura sismica, costituisce la base degli approcci di previsione dei terremoti. Un modo per studiare i precursori sismici aumentando notevolmente la copertura delle zone sismiche e la statistica su questi eventi è quello di effettuare misure delle anomalie ionosferiche mediante strumenti installati a bordo dei satelliti. Attualmente, solo la missione francese DEMETER (2004-2010) dedicata proprio a questo scopo, ha raccolto dati dettagliati; altri risultati precedenti sono stati ottenuti analizzando le osservazioni di alcuni esperimenti spaziali non specificamente dedicati a questo.

L'obiettivo principale della missione CSES è lo studio delle perturbazioni ionosfe-

riche indotte dall'attività sismica e dai meccanismi di preparazione dei terremoti. Si basa su misurazioni sistematiche e dettagliate a lungo termine eseguite da rivelatori appositamente progettati. In particolare, la missione mira ad analizzare la correlazione temporale tra eventi sismici e il verificarsi di perturbazioni elettromagnetiche nella ionosfera superiore e precipitazioni di particelle dalle fasce di Van Allen. Inoltre, CSES-01, fornisce importanti informazioni sui parametri della ionosfera e sul comportamento sconosciuto della regione di transizione ionosfera-magnetosfera, al fine di sviluppare modelli fisici del meccanismo di accoppiamento litosfera-atmosfera-ionosfera.

La missione CSES sta monitorando anche l'attività solare impulsiva e la modulazione solare dei raggi cosmici, rilevando flussi di protoni ed elettroni da pochi MeV a centinaia di MeV. Infatti l'attività solare varia fortemente nel tempo (con il suo periodo di attività di circa 11 anni) da un livello minimo, quando il sole è calmo, a un massimo di attività. I raggi cosmici di origine galattica rispondono a questa attività, che ne modifica significativamente gli spettri differenziali a seconda dell'energia, della specie e del segno di carica. L'effetto di modulazione solare risultante è evidente nei dati di monitoraggio dei neutroni, mostrando una chiara anti-correlazione tra intensità del flusso di particelle e attività solare. In particolare per particelle con rigidità ( $R = pc/Ze$  con  $p =$  impulso e  $Ze =$  carica della particella) sotto i 30 GV l'effetto diventa progressivamente maggiore al diminuire della rigidità.

L'attività solare è anche caratterizzata da un numero di fenomeni transitori come i *brillamenti solari* ed *eruzioni di massa coronale*. Le particelle espulse possono essere accelerate a un'energia che varia da poche decine di keV a pochi GeV (*Solar Energetic Particles - SEP*); trasportate attraverso l'eliosfera possono raggiungere la Terra provocando tempeste geomagnetiche interagendo con gli strati più esterni della magnetosfera. Le misurazioni di CSES forniscono un'estensione ad energie più basse degli spettri di particelle osservati dagli esperimenti PAMELA e AMS, nel corso 24° ciclo solare. È anche possibile confrontare gli spettri misurati con quelli di altre missioni spaziali, come GOES e ACE.

Come detto in precedenza i dati contribuiscono inoltre a fornire un servizio di condivisione osservativa per la comunità scientifica.

## 1.2 Obiettivi scientifici della missione

I terremoti sono una delle catastrofi naturali più pericolose. Terremoti di grande magnitudo (7 o superiore) si verificano circa 20 volte l'anno in tutto il mondo. L'energia rilasciata da tali terremoti in pochi minuti è paragonabile all'energia prodotta da una grande esplosione nucleare, quello di Amatrice ad esempio ha liberato un'energia totale paragonabile a 300 Megatoni (300 mln di tonnellate di tritolo). Un terremoto è un evento di deformazione, frattura e trasformazione che rilascia improvvisamente una grande quantità di energia elastica immagazzinata nel mezzo (*litosfera terrestre*) ed è accompagnata da una frazione sostanziale di energia irradiata come onde sismiche. Per quantificare l'entità del terremoto i sismologi utilizzano i concetti di magnitudo ( $M$ ) e rilascio di energia ( $E$ ). La relazione tra la Magnitudo  $M_s$ , determinata dalle onde sismiche di superficie, e l'energia (in erg) fu trovata da Gutenberg e Richter [?, ?, ?]:

$$\log E = 1.56M_s + 11.8$$

Gli stessi autori proposero anche la relazione tra la dimensione del terremoto e la frequenza di occorrenza, nella forma:

$$\log N = a - bM_s$$

dove  $N$  è il numero di terremoti con magnitudo in un intervallo fisso intorno alla magnitudo  $M_s$  e  $a, b$  sono costanti.

Ci sono molti modelli e interpretazioni sul problema di come avvengono i terremoti. Tutti suppongono che la fratturazione della roccia sia preceduta dall'accumulo di deformazione nella zona di faglia. L'aumento della deformazione è accompagnato da altri fenomeni naturali in prossimità della frattura che si sta preparando, che sono i cosiddetti *precursori* dei terremoti. Questi precursori includono le emissioni elettromagnetiche di origine sismica che sono uno dei precursori dei terremoti più significativi e possono essere studiati sia con esperimenti a terra che su veicoli spaziali.

Le EME (*ElettroMagnetic Emission*) di origine sismica sono presumibilmente legate a terremoti litosferici (cioè eventi sismici che si verificano nella litosfera elastica). I modelli più affidabili della descrizione dei meccanismi di generazione di EME richiamano effetti come piezoelettricità, piezomagnetismo ed effetti elettrocinetici.

La piezoelettricità è una proprietà che deriva dall'anisotropia cristallina del materiale. Una sostanza piezoelettrica mostra una risposta elettromagnetica alle stimolazioni meccaniche con una polarizzazione elettrica causata dalla deformazione dei reticoli cristallini. Simile a piezoelettricità, il piezomagnetismo consiste nella magnetizzazione di un materiale ottenuto applicando uno stress al cristallo. Inoltre nelle fasi di preparazione al terremoto, si verificano deformazioni cristalline e processi di fratturazione che provocano un cambiamento di porosità, questo a sua volta determina una variazione del moto del fluido e della corrente elettrica ad esso associata. I fenomeni di elettrocinetica sono legati all'esistenza di un doppio strato che si forma nell'interfaccia liquido/solido costituito da ioni legati al solido e uno strato mobile di ioni di carica opposta che si estende nel liquido. La differenza di potenziale elettrico tra l'interfaccia solido-liquido e la massa del liquido formano il cosiddetto *potenziale elettrocinetico*. [7]

Si pensa che durante la loro propagazione attraverso la crosta solida, le frequenze più elevate delle onde sismiche EME vengano attenuate e solo le onde ULF/ELF (Ultra Low Frequency/ Extremely Low Frequency) possano raggiungere la superficie terrestre e propagarsi attraverso la ionosfera nel *vicino spazio* terrestre. In particolare, se l'ipocentro è a una profondità di 10 km dalla superficie terrestre, solo le ULF ( $\omega \simeq 10\text{Hz}$ ) possono raggiungere la superficie terrestre con un'attenuazione inferiore a 10 dB. Importanti sorgenti di background sono però le perturbazioni nella ionosfera causate dalle EME di origine antropica. Ad esempio vengono utilizzati trasmettitori a terra nei 10-20 kHz (frequenze VLF) per la radionavigazione e nelle comunicazioni e la loro regione di interazione più forte è in prossimità dell'equatore geomagnetico che è anche la regione nella quale si verificano i terremoti più intensi. Va posta, quindi, ancora maggior attenzione nell'individuazione dell'origine della perturbazione. Anche le emittenti, a frequenze HF, utilizzano potenti trasmettitori che possono riscaldare la ionosfera e modificarne la temperatura e la densità.

L'interazione di queste EME con l'ambiente ionosferico può causare diversi tipi di perturbazioni, sia nella composizione del plasma, come nel campo elettromagnetico e sulle particelle intrappolate nelle fasce di radiazione di Van Allen. Queste sono zone di particelle energeticamente cariche, la maggior parte delle quali provengono dal vento solare, che vengono catturate dal campo magnetico terrestre. La Terra ha due di queste fasce (interna ed esterna) che si estendono da un'altitudine di circa 500 km fino a 58000 km sopra la superficie. Sono composte principalmente da elettroni energetici ( $1\text{MeV} < E < 50\text{MeV}$ ) e protoni ( $10\text{eV} < E < 1\text{GeV}$ ) che si muovono lungo le linee di campo e oscillano avanti e indietro. Il meccanismo di cattura nasce

dall'interazione tra le particelle energetiche cariche in movimento e il campo geomagnetico statico. Possiamo assumere il movimento delle particelle intrappolate come una sovrapposizione di tre moti periodici: una rotazione attorno alle linee di campo magnetico locale, un'oscillazione lungo le linee di campo tra punti di *mirror* coniugati degli emisferi magnetici nord e sud, e un moto di deriva intorno alla Terra. Gli invarianti adiabatici, che si conservano in condizioni di piccole variazioni del campo magnetico durante il periodo del moto, e in assenza di perdita di energia, scattering e diffusione radiale, sono associati a ogni tipo di movimento che la particella può eseguire. Il primo invariante adiabatico è il momento magnetico  $\mu$  associato al moto del ciclotrone delle particelle attorno al campo geomagnetico (vedi Fig. 1.1 (a)). La frequenza giroscopica, o frequenza di ciclotrone  $\omega_B$ , di una particella carica  $q$  con massa  $m$  in un campo magnetico di densità di flusso  $B$  e il corrispondente raggio di rotazione (*gyroradius*)  $r_B$  sono definiti come:

$$\omega_B = \frac{qB}{m}$$

e

$$r_B = \frac{mv_{\perp}}{qB}$$

dove  $v_{\perp}$  è la velocità della particella normale alla linea di campo.

Il secondo invariante adiabatico, detto invariante integrale, è associato al moto di oscillazione con una certa frequenza di oscillazione (vedi Fig. 1.1 (b)) delle particelle tra i punti specchio e lungo il campo magnetico. Poiché la forza di Lorentz è perpendicolare alla velocità, non può cambiare l'energia di una particella carica che si muove al suo interno. Così l'energia cinetica della particella, e quindi il modulo della velocità totale, rimane costante. Nell'approssimazione di dipolo magnetico della Terra, l'ampiezza del campo è massima vicino ai poli magnetici e minima vicino all'equatore magnetico. A causa della costanza del momento magnetico definito come:

$$\mu = \frac{mv_{\perp}^2}{2B},$$

quando la particella si muove verso i poli ( $B$  crescente), la componente di velocità trasversale al campo aumenta. Di conseguenza, la componente di velocità parallela al campo locale deve diminuire. I punti di mirror sono le posizioni in cui il vettore della

velocità della particella nella direzione del campo raggiunge lo zero e le particelle invertono il loro moto.

Se le particelle colpiscono la parte superiore dell'atmosfera durante questo movimento di oscillazione, possono essere deviate dalle collisioni con gli atomi nell'aria, perdere energia ed essere perse dalla fascia di radiazione. Per una data particella, la posizione del punto di mirror è determinata dal suo *angolo di inclinazione equatoriale* (*pitch angle*) definito come segue:

$$\alpha_{eq} = \tan^{-1} \frac{v_{\perp}}{v_{\parallel}}.$$

Questo è l'angolo tra la direzione della velocità della particella e la linea del campo magnetico, calcolato nel il piano equatoriale (vedi Fig. 1.2). È possibile definire un *loss-cone* entro il quale tutti i valori del *pitch angle* corrispondono a punti di mirror che si trovano troppo in profondità nell'atmosfera e la corrispondente particella andrà persa.

Il terzo tipo di movimento è la deriva delle particelle attorno alla Terra. Questo è causato da un gradiente nell'intensità del campo geomagnetico che provoca un cambiamento nel raggio di rotazione, dando luogo ad uno spostamento laterale dell'orbita con direzione opposta per particelle positivo e negative (vedi Fig. 1.1 (c)).

I flussi di particelle anomali, rilevati da diversi esperimenti spaziali in coincidenza con eventi sismici, si pensa che siano dovuti all'interazione ULF/ELF EME che può causare la precipitazione delle particelle intrappolate nelle fasce di radiazione di Van Allen. Un possibile meccanismo di diffusione è l'interazione risonante tra le onde di Alfvén ULF di origine sismica e le particelle che hanno frequenze di oscillazione tra i punti di mirror nella banda ULF (cioè elettroni con  $E > 1\text{MeV}$  e protoni con  $E$  dell'ordine di alcune decine di MeV). Quando la frequenza di oscillazione e la frequenza dell'onda corrispondono, la particella sperimenta l'onda di campo elettrico  $E$  ad ogni passaggio nella zona perturbata. Questo produce una variazione  $\Delta\alpha$  nel pitch angle delle particelle che può causare un abbassamento dei punti mirror e, come conseguenza, la precipitazione delle particelle colpite. Una stima di  $\Delta\alpha$  è data da Aleshina et al.[4]. Una rappresentazione qualitativa della fenomenologia è descritta in Fig. 1.3.

La missione CSES mira a stabilire un sistema di osservazione spaziale per rilevare anomalie elettromagnetiche e perturbazioni della ionosfera. A questo scopo, diversi strumenti scientifici utili sono installati a bordo del satellite, come ad esempio due rilevatori di particelle sensibili a *burst* di particelle (un improvviso aumento del flusso

concentrato in un breve intervallo di tempo). Il grande vantaggio degli esperimenti satellitari, rispetto agli esperimenti a terra, è la possibilità di coprire la maggior parte delle zone sismiche della Terra, aumentando le statistiche a causa del numero molto maggiore di eventi che possono essere registrati.

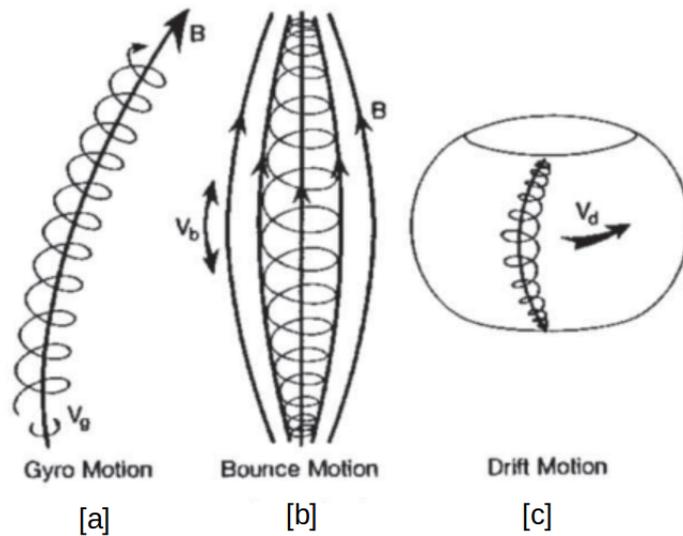


Figura 1.1: Il moto delle particelle intrappolate è la sovrapposizione di tre moti: il moto di rotazione attorno alle linee di campo magnetico [a], il moto di oscillazione lungo le linee di campo [b] e il moto di deriva intorno alla Terra (quella indicata è relativa agli elettroni, per i protoni la direzione è opposta) [c].

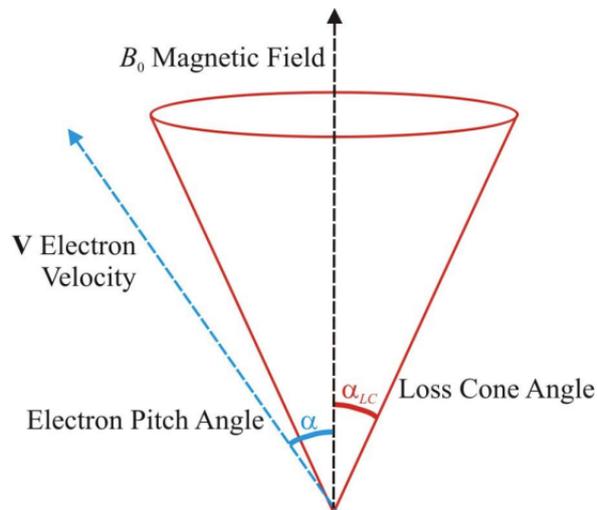


Figura 1.2: Loss cone: Se il pitch angle della particella si trova all'interno del loss cone ( $\alpha < \alpha_{LC}$ ) il suo punto di mirror risulterà troppo interno e la particella verrà assorbita dalle collisioni con gli atomi dell'atmosfera.

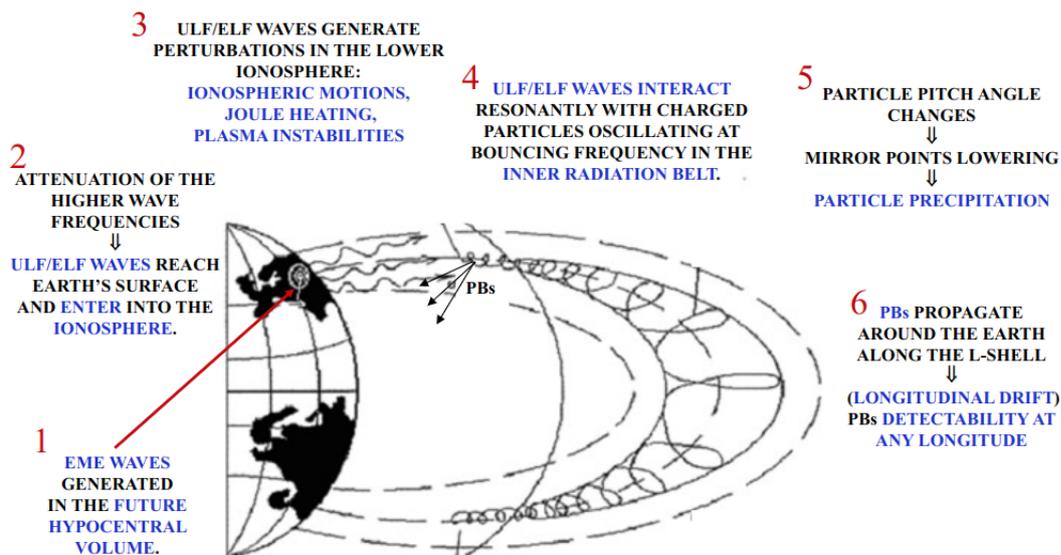


Figura 1.3: Schema che illustra la precipitazione di particelle intrappolate dovuta all'interazione con EMEs di origine sismica. Tratto da Sgrigna 2001 [6]

### 1.3 Il contributo italiano: CSES-Limadou

L'Italia partecipa alla missione satellitare CSES con il progetto Limadou<sup>1</sup>, finanziato dall'ASI (Agenzia Spaziale Italiana), in collaborazione con diverse Sezioni (Bologna, Napoli, Roma Tor Vergata e Trento) dell'INFN (Istituto Nazionale Italiano di Fisica Nucleare), l'INGV (Istituto Nazionale di Geofisica e Vulcanologia), INAF-IAPS (Istituto Italiano Nazionale di Astrofisica e Planetologia) e diverse Università italiane. Il contributo italiano alla missione è stato nella realizzazione del High Energy Particle Detector (HEPD), in quattro diversi modelli, e nel supporto allo sviluppo del Electric Field Detector (EFD). La "Camera al plasma" dell'INAF-IAPS di Roma è stata utilizzata anche per testare diversi payload cinesi al fine di studiare la risposta dei sensori e verificarne la compatibilità con il plasma ionosferico, in condizioni ambientali simili a quelli incontrate dal satellite in orbita.

<sup>1</sup>Il nome è stato scelto in onore del missionario italiano Matteo Ricci (riprendendo il suo nome cinese)

## 1.4 Il Satellite CSES-01

Il satellite CSES-01 è basato sulla piattaforma cinese CAST2000 (vedi Fig. 1.4). È un Satellite stabilizzato con assetto a 3 assi che ha un'orbita polare eliosincrona con un'altitudine di 507 km e un'inclinazione di  $97^\circ$ . Ha un periodo ricorsivo di 5 giorni, in questo modo il *ground track* (Fig. 1.5) viene ripetuto rigorosamente dopo 5 giorni. Il corpo principale del satellite, nella configurazione di lancio, ha una dimensione di 145 cm (Y)×144 cm (Z)×143 cm (X) che aumenta dopo il dispiegamento del pannello solare e le aste con i rilevatori posti sulla punta. Il satellite è composto da diversi payload scientifici e sottosistemi. Questi sottosistemi includono: *Attitude and Orbit Control* (AOC), *On-Board Data Handling* (OBDH), *Tracking, Telemetry and Command* (TTC), l'alimentatore (composto da una batteria agli ioni di litio da 80 Ah e dai pannelli solari) e i sottosistemi di controllo termico. La Tab. 1.1 riassume le principali specifiche di CSES.

Il sottosistema AOC utilizza la stabilizzazione a 3 assi orientata verso la terra e i sensori di assetto (3 star-sensor, 2 gruppi di giroscopi e 1 sensore solare digitale) vengono utilizzati per misurare l'assetto, la ruota di reazione e la coppia magnetica per mantenere il controllo del momento a zero. I dati di *housekeeping* tra i payload e i sottosistemi vengono scambiati tramite il CAN bus, dove il sistema OBDH è l'*host* e tutti gli altri apparati sono i *guest*.

L'asse  $X$  è diretto lungo la direzione del moto, l'asse  $Z$  è diretto verso il centro della terra e l'asse  $Y$  completa il sistema di riferimento ortonormale. Un pannello solare si trova su un lato del satellite e può ruotare attorno all'asse  $Y$ . Per ridurre la possibilità di interferenza per gli strumenti scientifici indotti dalla rotazione dei pannelli solari o dalle regolazioni AOC, il satellite definiva originariamente due zone di lavoro: la zona di lavoro di payload con latitudini comprese tra  $-65^\circ$  e  $+65^\circ$  e la zona di regolazione della piattaforma a latitudine  $> 65^\circ$  o latitudine  $< -65^\circ$ , dove gli strumenti scientifici smettono di funzionare e vengono effettuate operazioni di ottimizzazione o correzioni orbitali. A partire da novembre 2019 la zona di lavoro è stata ampliata a  $\pm 70^\circ$ .

ITEM	Specification/Parameter	Specification/Parameter
Mass	$\approx 700$ kg	
Data Transmission	Band	X
	Downlink Rate	120 Mbps
	Mass Memory Size	160 Gbit
Life Span	$\geq 5$ years	
Orbit Type	Sun-Synchronous Orbit	
Orbit Altitude	507 km	
Orbit Inclination	97°	
Orbit Period	94 min	
Circels per Day 1	15 +1/5	
Recursive Period	5 Days	

Tabella 1.1: Specifiche e parametri orbitali di CSES

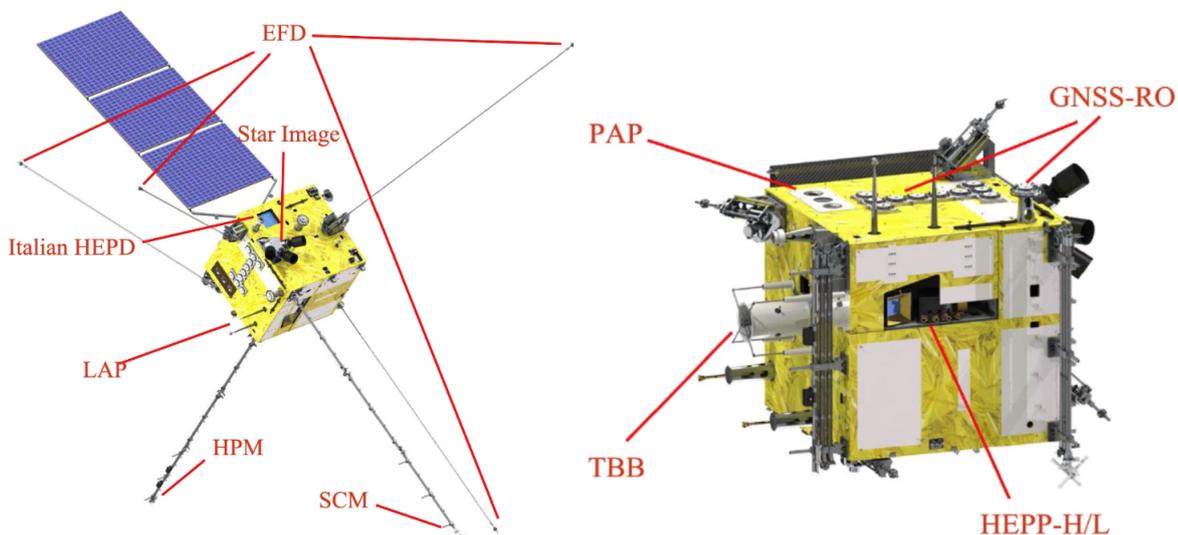


Figura 1.4: Schema del satellite CSES e del posizionamento dei payload

### 1.4.1 Strumenti scientifici a bordo

Di seguito una lista degli strumenti scientifici ospitati a bordo del satellite CSES (Fig. 1.4):

- posizionati sulle punte dei bracci si trovano due megnetometri (un *Search-Coil Magnetometer* - SCM e un *High Precision Magnetometer* - HPM) e un rivelatore di campo elettrico (*Electric Field Detector* - EFD) per la misurazione delle tre componenti dei campi magnetico ed elettrico;

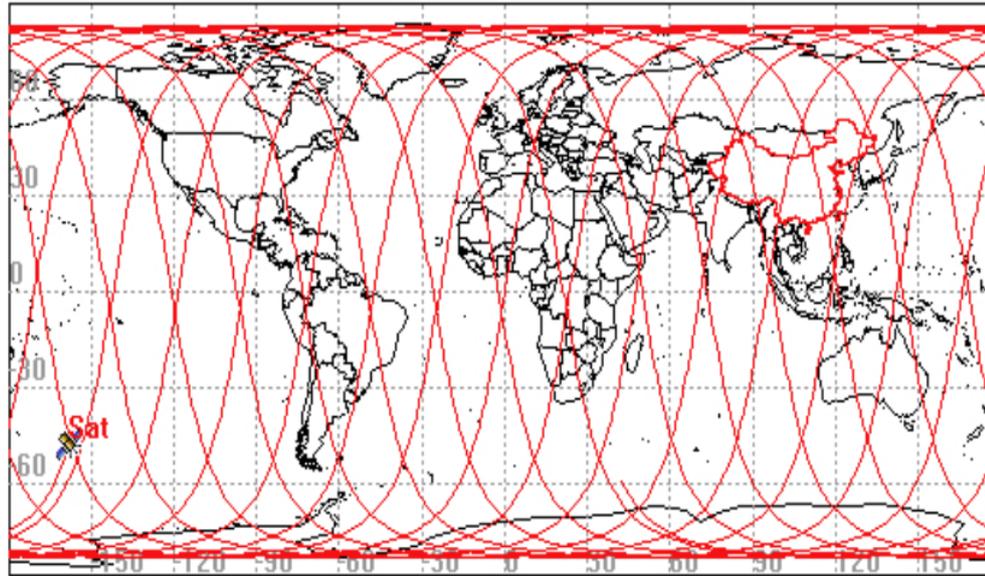


Figura 1.5: ground track del satellite: la distanza tra due tracce vicine nell'arco di un solo giorno è di circa 2650 km, ma nel periodo di 5 giorni si riduce a 530 km.

- un analizzatore di plasma (*Plasma Analyzer Package* - PAP) per la misurazione dei disturbi locali mediante l'analisi dei parametri del plasma, tra cui la densità, la temperatura, la velocità di deriva e la composizione degli ioni;
- una coppia di Langmuir Probes (LP) per la misura della densità elettronica e della temperatura nel plasma;
- un Ricevitore di Occultazione GNSS e un Trasmettitore a tre frequenze (VHF/UHF) per lo studio dei disturbi del profilo del plasma;
- l'High Energy Particle Package (HEPP) e l'Italiano High Energy Particle Detector (HEPD) per la misura di flussi e spettri energetici di particelle con particolare attenzione a quelle che precipitano dalle fasce di radiazione di Van Allen (con un valore del pitch angle all'interno del cono di perdita).

Grazie alla varietà di strumenti, CSES è in grado di affrontare diversi argomenti scientifici, come il monitoraggio dell'ambiente spaziale elettromagnetico vicino alla Terra, lo studio delle perturbazioni del plasma nella ionosfera, la misura del flusso di elettroni e protoni, lo studio dell'ambiente di radiazione attorno alla Terra anche durante fenomeni di fisica solare. CSES è complementare al raggio di azione di altre missioni spaziali nella misurazione dei raggi cosmici come PAMELA e AMS estendendone

lo spettro a range di bassa energia, fino a pochi MeV (Fig. 1.6). Come anticipato, per la sua natura specifica, CSES è quindi un valido strumento per lo Speace Weather soprattutto tendendo conto dell'avvicinarsi del prossimo massimo di attività solare.

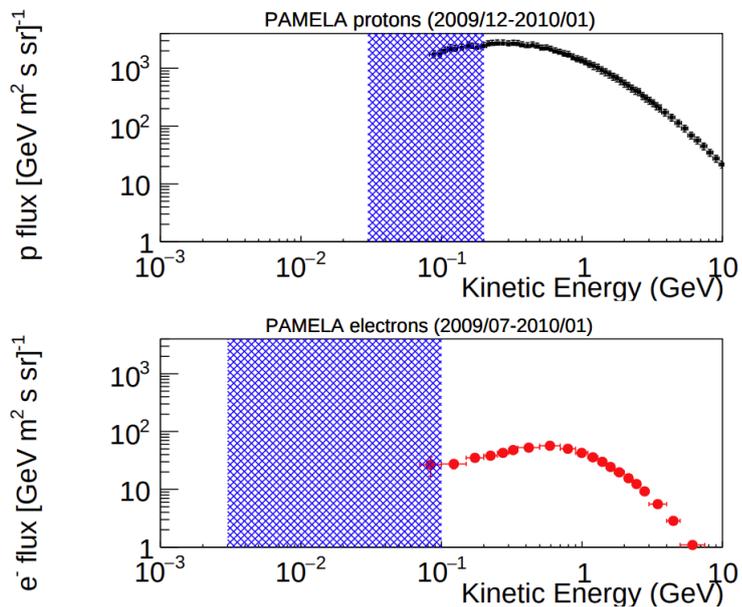


Figura 1.6: Comparazione tra i flussi di protoni (sopra) ed elettroni (sotto) misurati da PAMELA. Le regioni in blu si riferiscono ai range di energia di HEPD.



## 2 Il rivelatore HEPD

Come accennato il rivelatore di particelle ad alta energia HEPD è stato interamente realizzato dalla componente italiana CSES-Limadou.

HEPD è in grado di rivelare elettroni nella gamma di energia tra 3 MeV e 100 MeV e protoni nell'intervallo di energia compreso tra 30 MeV e 250 MeV, nonché nuclei leggeri.

Come vedremo in dettaglio nelle prossime sezioni di questo capitolo, lo strumento è composto da diversi rivelatori. Due piani di sensori a micro-strip di silicio doppia faccia, posti sulla parte superiore dello strumento, forniscono la direzione della particella incidente. Un sottile strato segmentato di scintillatore plastico è posto appena sotto il tracker al silicio e fornisce il trigger. Questo piano di trigger è seguito da un calorimetro, costituito da altri 16 piani scintillatori e uno strato di cristalli LYSO. Un sistema di Veto completa lo strumento. L'alimentatore e l'elettronica sono inseriti in una scatola posizionata su un lato del rivelatore. HEPD è contenuto in una scatola a nido d'ape di alluminio installato sul satellite CSES e puntato verso lo zenit (vedi Fig. 1.4).

Quattro diversi modelli di HEPD sono stati realizzati e completamente testati: l'*Electrical Model* (EM), lo *Structural and Thermal Model* (STM), il *Qualification Model* (QM) e il *Flight Model* (FM). Quest'ultimo è stato quindi installato a bordo del satellite dopo quasi quattro anni di lavoro (Fig. 2.1)

### 2.1 Gli obiettivi scientifici di HEPD

Il rivelatore ha diversi obiettivi, compreso lo studio della stabilità temporale delle fasce di Van Allen, l'indagine della precipitazione delle particelle intrappolate indotta da EMEs ionosferiche e troposferiche o lo studio del meccanismo di accelerazione delle particelle energetiche solari.

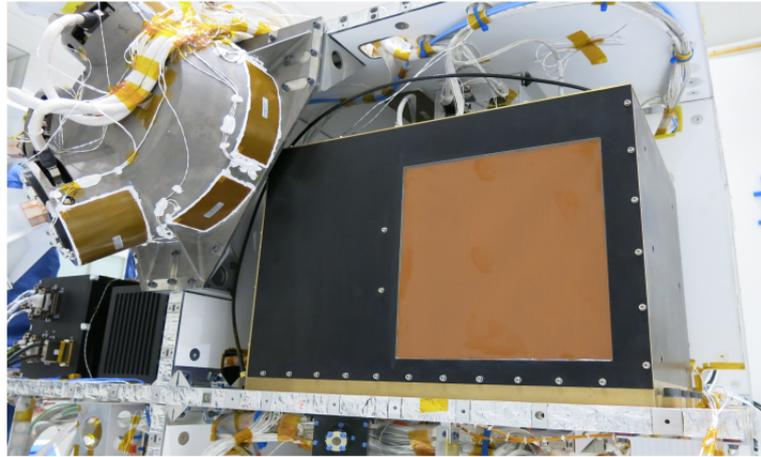


Figura 2.1: HEPD montato a bordo di CSES. La finestra di HEPD è coperta da uno strato di Pyralux (rame e kaptan) per assicurare un buon isolamento termico

### 2.1.1 Raggi Cosmici e Fisica Solare

Per missioni spaziali di lunga durata come CSES, le misurazioni delle particelle di raggi cosmici devono essere correlate all'ambiente in continua evoluzione in cui opera il rivelatore. L'eliosfera, ovvero la regione modellata dalla presenza del vento solare che fuoriesce dall'atmosfera superiore (la corona) del Sole, presenta diversi livelli di variabilità, alcuni dei quali ancora poco capiti. Questa variabilità del vento solare porta all'insorgenza di fenomeni dinamici a tutte le scale spaziali e temporali [9].

Dal momento del lancio, HEPD è stato operativo durante il minimo solare del 25° ciclo iniziato nel dicembre del 2019, offrendo le condizioni ideali per studiare i meccanismi di propagazione all'interno dell'eliosfera. Inoltre, tali condizioni di tranquillità, hanno aiutato a ottenere una descrizione indisturbata dei transienti solari come *solar particle events* (SPE); questi eventi sono rari durante le fasi di attività minima del Sole, ma l'accelerazione e la propagazione delle *solar energetic particles* (SEPs) può essere studiata con minori incertezze causate invece da un'eliosfera turbolenta (come avviene durante le fasi massime del ciclo di attività del Sole). Il potente evento di classe X del 6 dicembre 2006[10] insieme al *Ground Level Enhancement* (GLE) del 13-14 dicembre 2006 ne rappresentano un esempio perfetto [11]. Durante la massima attività, le SEPs del Sole diventano più frequenti. Entrando quindi in questa nuova fase di massimo del 25° ciclo, potrà essere accumulata un'ampia varietà di dati e confrontata, raggruppando eventi diversi in base ad alcune loro caratteristiche come durata, indice spettrale o intensità. Si spera che HEPD a bordo di CSES possa registrare un ampio

numero di eventi solari, in un intervallo di energia compreso tra le osservazioni in situ di ACE, STEREO, GOES ecc. [12] e i dati ad alta energia provenienti da AMS02 [13] e monitor di neutroni [14] a terra, riempiendo così il vuoto lasciato da PAMELA [15, 16].

La soglia  $> 30\text{MeV}$  per i protoni, insieme all'orbita polare, consente il rilevamento in regioni in cui il cutoff geomagnetico è considerevolmente basso e possono essere misurate anche particelle energetiche originate da deboli eventi solari. Lo studio della relazione tra l'indice spettrale e l'energia di rollover di un evento SEP potrebbe anche spiegare il processo di accelerazione avvenuto; in effetti se il Sole accelera le particelle a bassa quota attraverso la riconnessione magnetica oppure nelle regioni più elevate della corona attraverso shock guidati da espulsione di massa coronale (*CME-driven shocks*) è ancora oggetto di studio a causa di alcune modificazioni che avvengono durante il trasporto all'interno dello spazio interplanetario [17]. Un altro possibile argomento di interesse riguarda la diminuzione del flusso di particelle galattiche in seguito al passaggio di una CME diretta verso la Terra (effetto Forbush<sup>1</sup> [20] dal nome del primo scienziato che l'ha osservato). Il range di energia di HEPD è adatto per seguire parzialmente l'evoluzione di questi fenomeni transienti ottenendo informazioni sull'entità della diminuzione stessa o la sua durata [18].

Su scale temporali più lunghe, le osservazioni degli spettri energetici di elettroni e protoni e la loro variazione durante uno o più cicli solari saranno estremamente utili per comprendere i processi che dominano il trasporto all'interno dell'eliosfera. I dati a bassa energia di HEPD potrebbero essere utilizzati per stabilire un modello numerico tridimensionale completo [21] basato sulla soluzione numerica dell'equazione di trasporto di Parker [22]. Ogni spettro di particella possiede peculiari caratteristiche relative ai meccanismi di trasporto; inoltre molti modelli di deriva [23] prevedono una chiara dipendenza dal segno della carica per la modulazione dei raggi cosmici, per cui una misurazione congiunta elettrone/protone è importante. Il ruolo di queste osservazioni potrebbe anche acquisire valore se il 25° minimo solare sarà insolito come il 23°, offrendo un'opportunità unica di studio tale meccanismi in una situazione non convenzionale.

---

<sup>1</sup>L'effetto forbush consiste in una rapida diminuzione dell'intensità della componente galattica dei raggi cosmici che segue una CME. Questa trasportando plasma e campo magnetico costituisce uno schermo per i raggi cosmici diminuendone l'intensità.

### 2.1.2 Studi sulla Magnetosfera

La magnetosfera terrestre, creata dal suo campo magnetico, non è un sistema chiuso. La sua forma e composizione è continuamente influenzata dal vento solare che scorre provenendo dal Sole e in alcuni casi eventi solari estremi possono modificare temporaneamente e pesantemente la sua struttura [19].

Le fasce di radiazione di Van Allen esterne ed interne hanno una composizione estremamente variabile e questa variabilità è enfatizzata quando potenti SEPs incontrano le linee del campo magnetico. Questo tipo di interazioni [24] tra radiazione cosmica e magnetosfera origina particelle chiamate albedo (di direzione ascendente) che possono essere successivamente identificate come rientranti se la loro traiettoria viene piegata dal campo geomagnetico permettendo loro di rimanere intrappolate con una direzione discendente e splash albedo se riescono a sfuggire alla magnetosfera [25]. La prima famiglia comprende particelle quasi-intrappolate e non-intrappolate, a seconda che siano confinate o meno nella regione equatoriale al di sotto della fascia di Van Allen interna [26, 27]. Una nuova e accurata misurazione della radiazione cosmica ad alta energia ( $> 70\text{MeV}$ ) a bassa orbita terrestre è stata riportata in [28] in funzione dell'energia e dell'angolo tra il campo magnetico e la direzione della particella in arrivo. Il sistema di tracciamento al silicio di HEPD (che fornisce la direzione della particella incidente) può misurare i protoni rientranti dell'albedo fino a  $\sim 200\text{MeV}$  confrontando i risultati con gli esperimenti precedenti.

Grazie all'orbita di CSES attraverso l'Anomalia del Sud Atlantico e la grande accettazione angolare di HEPD (sensibile a diversi valori di pitch angle), unitamente alla precisa misurazione della direzione della particella fornita dal tracker di silicio, HEPD consente anche l'osservazione di particelle geomagneticamente intrappolate dalla fascia interna di Van Allen e lo studio della variazione della loro composizione.

### 2.1.3 Precursori a breve termine di terremoti e burst di particelle

Per l'accoppiamento tra litosfera e magnetosfera inferiore, abbiamo visto che i vettori energetici più considerati sono le emissioni elettromagnetiche sismiche sotto forma di onde a bassa frequenza. Il telerilevamento dallo spazio è il metodo più efficiente, se non l'unico, per ricercare questi accoppiamenti, perché sono necessari il monitoraggio di aree di grandi dimensioni e misurazioni simultanee di molte osservabili.

Correlazioni tra variazioni a breve termine (burst) di flussi di particelle cariche ad alta energia nello spazio vicino alla Terra e l'attività sismica sono stati inizialmente

evidenziati sulla base dei risultati ottenuti da diverse misurazioni satellitari. Flussi di particelle cariche ad alta energia sono stati ottenuti a bordo di vari veicoli spaziali: stazione orbitale MIR ( $20\text{MeV} < E_e < 200\text{MeV}$ ), METEOR-3 ( $E_e \leq 30\text{MeV}$ ), GAMMA ( $E_e \geq 50\text{MeV}$ ) e SAMPEX ( $E_e \leq 15\text{MeV}$ ) con diversi strumenti e sono stati elaborati e analizzati con l'obiettivo di ricercare la correlazione temporale e spaziale tra *burst* di particelle e forti terremoti[29], [30]. Come già accennato, la spiegazione di questo fenomeno è basata sul disturbo locale del flusso di particelle della fascia di radiazione causato dalle emissioni elettromagnetiche a ultra bassa frequenza (ULF) di origine sismica. Queste EMEs possono essere generate nella zona del terremoto diverse ore prima della scossa principale e possono propagarsi attraverso la ionosfera, venire catturate dal campo geomagnetico e quindi essere propagate come onde di Alfvén lungo le linee del campo geomagnetico. L'onda Alfvén interagisce con le particelle intrappolate nella fascia interna, provocandone le precipitazioni. La caratteristica principale di questo processo è che, a causa della deriva delle particelle intrappolate attorno alla Terra, il burst di particelle di origine sismica può essere osservato non solo al di sopra dell'epicentro ma a qualsiasi longitudine in cui il satellite incrocia l' L-shell<sup>2</sup> disturbato.

Aleksandrin et al. [29] hanno eseguito un'analisi sui dati provenienti da diversi strumenti. Hanno utilizzato gli spettrometri magnetici con gli scintillatori a tempo di volo MARIA e MARIA-2[31] installati a bordo delle stazioni orbitali SALYUT-7 e MIR (inclinazione  $51^\circ$ , altitudine  $400\text{km}$ ), lo strumento ELECTRON (una pila di rivelatori a scintillatore e un contatore Cherenkov) installato a bordo del satellite METEOR-3 (inclinazione  $82^\circ$ , altitudine  $1250\text{km}$ ), il telescopio per raggi gamma GAMMA-1 [32] a bordo della stazione astrofisica GAMMA (inclinazione  $51^\circ$ , altitudine  $350\text{km}$ ), e il rivelatore al silicio PET [33] a bordo del satellite SAMPEX (inclinazione  $82^\circ$ , altitudine  $600\text{km}$ ). La Fig. 2.2 mostra la correlazione temporale che gli autori hanno ottenuto, tra burst di particelle e un insieme di terremoti con  $M > 4$ , dopo aver selezionato posizioni satellitari con L-shell minore di 2 (la parte principale dei terremoti è situata nella regione vicino equatoriale) ed esclusa la regione dell'Anomalia Sud-Atlantica. Il valore di  $\Delta T$  è stato definito come:

$$\Delta T = T_{EQ} - T_{PB} \quad (2.1)$$

dove  $T_{EQ}$  e  $T_{PB}$  sono il tempo in cui si osserva il terremoto e quello del burst di parti-

<sup>2</sup>Il parametro L-shell identifica il set di linee di campo magnetico che attraversano l'equatore geomagnetico ad una distanza dalla terra pari appunto al suo valore espresso in raggi terrestri.

celle rispettivamente. Il valore positivo di tutti i picchi (2-5 ore) significa che il burst di particelle può essere un precursore di terremoti a breve termine. Nell'analisi, forti incrementi a breve termine del rate di conteggio delle particelle, da decine di secondi a pochi minuti, vengono identificati come burst di particelle se il rate di conteggio attuale supera 4 deviazioni standard dal valore medio di fondo. Gli autori studiano anche la correlazione spaziale tra il burst e il terremoto studiando la variazione delle distribuzioni temporali utilizzando

$$\Delta L = L_{EQ} - L_{PB}$$

come parametro aggiuntivo. Il termine  $L_{EQ}$  è l' $L$ -shell del terremoto (cioè la coordinata  $L$  del punto ad una certa quota sopra l'epicentro, che coincide con l'altitudine della regione da cui l'EME di origine sismica viene catturata dalle linee di campo geomagnetico) e  $L_{PB}$  è l' $L$ -shell dei satelliti in corrispondenza dei burst di particelle. L'analisi mostra che le distribuzioni  $\Delta T$  non presentano picchi quando  $\Delta L > 0,5$ , ovvero quando il  $L$ -shell del terremoto differisce significativamente dall' $L$ -shell del burst di particelle, supportando che la precipitazione delle particelle e i terremoti sono correlati.

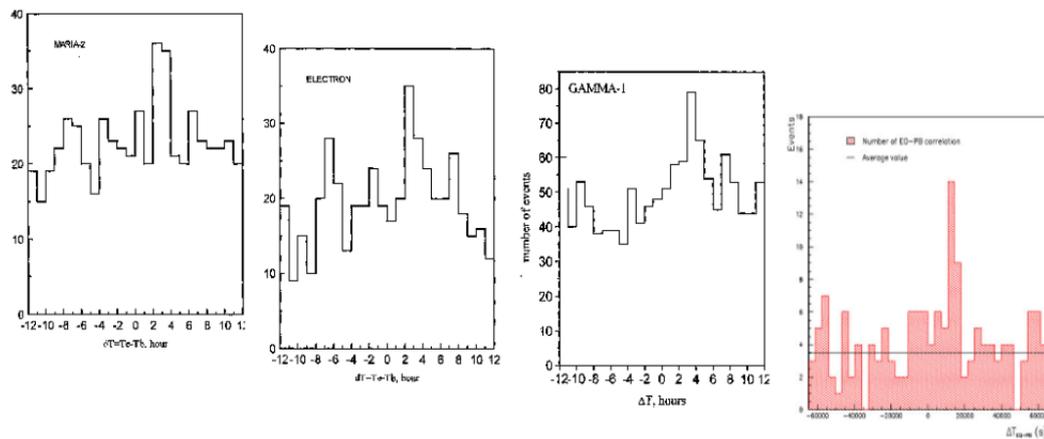


Figura 2.2: Istogrammi della differenza di tempo  $\Delta T$  tra il tempo del terremoto selezionato e il tempo del burst di particelle ottenuto dalle missioni MARIA, ELECTRON, GAMMA-1 e SAMPEX. Un valore positivo del picco indica che il burst precede nel tempo il terremoto. I grafici sono da [29].

### 2.1.4 Space Weather

La missione CSES opera in un periodo in cui le problematiche associate allo Space Weather vengono riconosciute come urgenti, quindi assume anche un ruolo di monitor del vicino spazio terrestre e aiuta a sviluppare ulteriori contromisure a possibili fenomeni pericolosi provenienti in futuro dal Sole.

Lo Space Weather è lo studio dei cambiamenti nel *vicino spazio terrestre* e gli effetti che questi possono causare alle attività antropiche. Questi cambiamenti sono principalmente originati dall'attività solare. L'aumento dell'energia del vento solare e della radiazione solare si traduce, ad esempio, in forti correnti elettriche che scorrono nella magnetosfera e nella ionosfera e in significative strutturazioni della densità del plasma o alterazioni della componente neutra, densità e venti nell'alta atmosfera[35]. Tali fenomeni (tempeste geomagnetiche) provocano rapide variazioni del campo geomagnetico fino a diverse centinaia di nT in pochi minuti. Queste forti variazioni nella forza e nella direzione del campo geomagnetico possono disturbare le attività di navigazione basate su mappe magnetiche precise o navigazione satellitare, ovvero GPS o GNSS; tali disturbi provocano errori di navigazione che non sono inclusi nei modelli di correzione ionosferica implementati nei ricevitori GPS commerciali e possono causare interruzioni del segnale e scintillazioni ionosferiche[37].

La radiazione proveniente dal Sole può essere riassunta in due tipi di particelle:

- le particelle costituenti il vento solare con energie del keV;
- le particelle energetiche solari (SEPs), manifestazioni di sole attivo (energie fino a 10 GeV)

Le SEPs sono originate da due fenomeni, già più volte citati, distinti ma profondamente legati tra loro come i *flares* e le *eruzioni di massa coronale*.

#### 2.1.4.1 Brillamenti (flares)

I brillamenti (o flares) sono la più potente manifestazione di rilascio di energia tra tutti i processi dell'attività solare. L'energia di un grande flare può arrivare fino a  $3 \times 10^{32}$  erg ossia 100 volte più grande dall'energia ottenuta dalla combustione di carbone sulla Terra in un anno. Considerando che questa enorme quantità di energia è rilasciata mediamente in pochi ( $\simeq 10$ ) minuti, la potenza media che si ottiene è di  $3 \times 10^{29}$  erg/s. I flares, inoltre, emettono radiazione che copre l'intero spettro elettromagnetico dalle onde radio fino ai raggi gamma e che può essere rivelata da stazioni terrestri o satelliti.

Allo stesso tempo anche le particelle accelerate nel flare o l'*ejecta* di plasma che viene prodotto dallo stesso possono essere misurate direttamente.

Il processo di rilascio di energia di un flare può essere convenzionalmente diviso in quattro parti (Fig. 2.3). Il violento rilascio è preceduto inizialmente da una fase chiamata *preheating* (PH). Successivamente c'è una fase in cui il plasma viene accelerato e le linee di campo magnetico cambiano struttura. In questa fase, chiamata *explosive phase* (EP) è compresa anche la sequenza di brevi ma potenti esplosioni a volte indicata come *impulsive phase* (IP) di durata più breve dell'intera EP. Durante queste esplosioni elettroni e ioni vengono accelerati fino alle alte energie. In ultimo è presente la *hot phase* (HP) o *main phase* del flare. È chiamata *hot* perchè si manifesta principalmente con un'elevata emissione di raggi X e *main* perchè durante questa fase avviene il rilascio di gran parte dell'energia totale del flare. Questa energia ha la forma di un intenso flusso di calore.

I flares sono classificati come A, B, C, M o X a seconda della loro luminosità nei raggi X vicino alla Terra, misurata in  $Watt/m^2$ . Ogni classe è dieci volte più potente di quella precedente, con X (la più elevata) pari a  $10^{12}W/m^2$ , ed inoltre è divisa linearmente in sottoclassi che vanno da 1 a 9 (esclusa la X che non è limitata). Di conseguenza un brillamento X2 è quattro volte più potente di uno M5. L'attività solare si trova normalmente compresa tra le classi A e C. I brillamenti C hanno pochi effetti sulla Terra, mentre i più potenti M e X possono causare danni. Il 16 agosto 1999 e il 2 aprile 2001 vennero misurati dei brillamenti X20, ma essi furono superati da quello del 4 novembre 2003, che fu stimato a X45 ed è il più potente mai registrato.

Dunque una parte significativa dell'energia totale rilasciata da un flare si manifesta nella forma di un *ejecta* di plasma che raggiunge velocità fino a  $1000km/s$  mentre un'altra parte si manifesta come un flusso di particelle altamente energetiche come elettroni, protoni o nuclei leggeri.

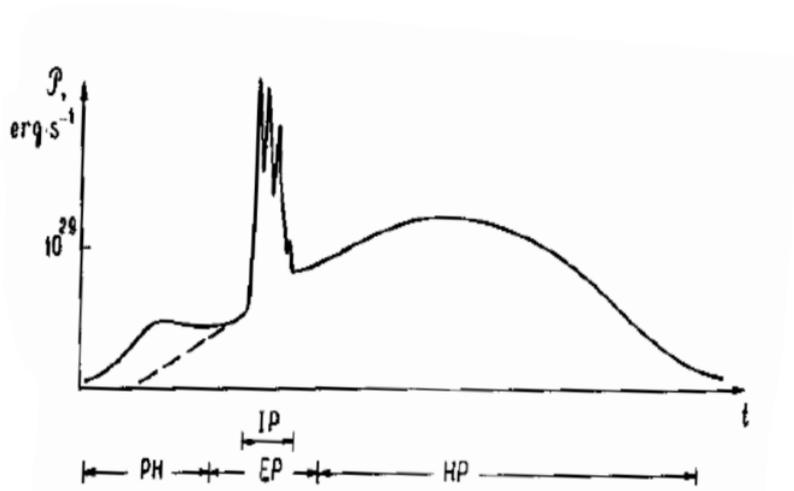


Figura 2.3: Rilascio energetico durante una flare. Si possono vedere le 4 fasi (preheating, impulsive, explosive e hot).

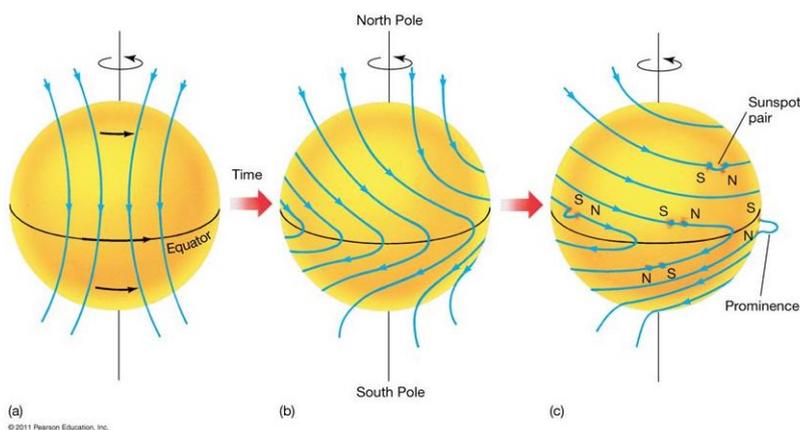


Figura 2.4: Rappresentazione della curvatura delle linee di campo magnetico del Sole a causa della sua rotazione differenziale. Si può vedere che in prossimità dell'equatore solare si forma una regione di cambiamento della polarità del campo magnetico chiamata «heliospheric current sheet».

### 2.1.4.2 Eruzioni di massa coronale (CME)

Le CME sono delle grandi eruzioni di plasma e campo magnetico dal Sole che possono contenere masse fino a  $10^{13}kg$  e possono raggiungere velocità di diverse migliaia di  $km/s$ .

All'inizio degli studi di fisica solare si pensava che i flares e le CMEs fossero lo stesso fenomeno o comunque che le seconde fossero una conseguenza dei primi. Seppur

profondamente collegate oggi si sa che i due fenomeni sono ben distinti. Infatti, nonostante una buona parte dei flares più intensi possa produrre un'eruzione di massa coronale, è possibile anche avere una CME senza un flare che la generi. Un'immagine di una CME è riportata in Fig. 2.5 osservata grazie al coronografo LASCO a bordo del satellite SOHO[36]. Le CME possono eruttare da ogni regione della corona ma sono più probabili in regioni di basse latitudini come nell'*heliospheric current sheet* (Fig. 2.4). La loro frequenza è di circa una al giorno durante i minimi solari e 4-5 al giorno durante periodi di massimo solare. Soltanto una piccola percentuale di queste raggiunge la Terra.

Il meccanismo che porta all'eruzione di una CME è correlato, come per i flares, alla riconnessione magnetica. Come il sole evolve attraverso il suo ciclo, a causa della sua rotazione differenziale, le linee di campo assumono una forma sempre più complessa finché il livello di complessità non raggiunge una soglia in cui è più conveniente il rilascio dell'energia magnetica piuttosto che il sostentamento della stessa. Le arcate del campo magnetico prima si sollevano, poi si comprimono e si riconnettono bruscamente al di sotto dell'apice permettendo così al materiale confinato nel tubo di flusso di propagarsi nello spazio (Fig. 2.6).

Le CMEs sono inoltre responsabili dell'effetto Forbush di cui abbiamo già parlato. Se definiamo la pressione magnetica come  $p_B = B^2/2\mu$  possiamo definire il rapporto  $\beta = p/p_B$  dove  $p$  è la pressione termodinamica del plasma. Il valore di  $\beta$  maggiore o minore di 1 determina il comportamento del plasma e del campo magnetico. Nella fotosfera (dove la pressione termodinamica è dominante  $\beta > 1$ ) sono i moti convettivi a trasportare il campo magnetico. Nella corona, invece, si ha  $\beta < 1$  e sono i campi magnetici a regolare il moto del plasma. Nel momento in cui una CME si distacca dalla corona la pressione termodinamica torna a essere rilevante e quindi è il plasma che trasporta con sé il campo magnetico che genera un schermo per i raggi cosmici galattici portando ad un decremento.

Le osservazioni della missione CSES e della sua costellazione di satelliti, effettuando misurazioni simultanee dell'ambiente magnetico, elettrico, del plasma e della componente neutra nel vicino spazio terrestre, hanno, e avranno in futuro, un grande valore nello studio e nel monitoraggio di questi fenomeni.

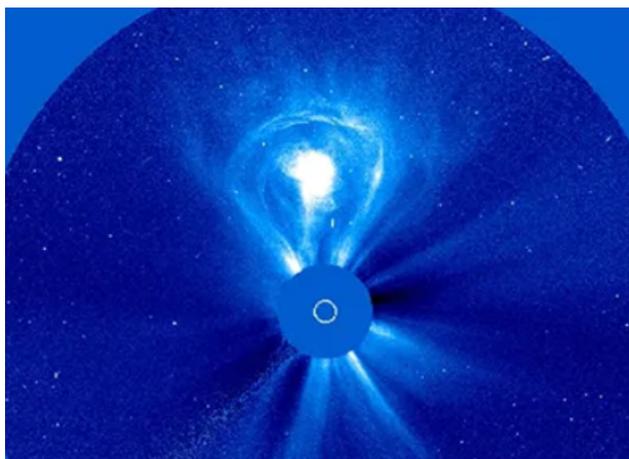


Figura 2.5: Una eruzione di massa coronale (SOHO/LASCO)

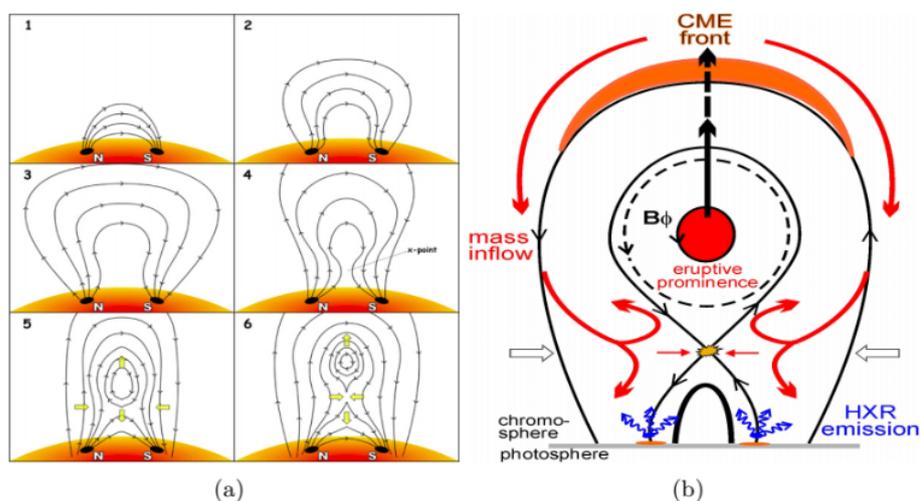


Figura 2.6: Sviluppo di una CME. La riconnessione magnetica crea una protuberanza che si separa dal campo magnetico solare permettendo il rilascio del plasma contenuto all'interno.

## 2.2 Descrizione fisica del rivelatore

Il rivelatore HEPD è contenuto in una scatola di alluminio con dimensioni  $40 \times 53 \times 28\text{cm}^3$  come mostrato in Fig. 2.7. Le pareti e la piastra di base sono realizzate con piastre in alluminio con struttura a nido d'ape, con la superficie esterna ricoperta da uno strato di poliammide alluminata per assicurare un buon isolamento termico. Il rivelatore stesso, insieme all'alimentatore e alla scatola dell'elettronica, si fissa alla

piastra di base che viene montata sul satellite mediante 8 viti. La Fig. 2.8 mostra uno schema dell'apparato all'interno della scatola, dove i pannelli laterali e superiori sono stati rimossi. Nella figura è mostrato anche il sistema di riferimento di HEPD, con il assi orientati come segue: l'asse Z è diretto lungo la dimensione longitudinale del calorimetro, l'asse X è diretto lungo la segmentazione del rivelatore al silicio e l'asse Y è ortogonale ai due precedenti. La rivelazione di elettroni nella gamma di energia tra 3 e 100 MeV e protoni tra 30 e 200 MeV, così come il rilevamento di nuclei leggeri, si ottiene per mezzo di una serie di rivelatori specifici:

- Un *TRACKER* composto di due piani di microstrip di silicio, distanti 1 cm. Ogni piano, di dimensioni 213.2mm × 214.8mm, è diviso in tre sezioni identiche chiamate *ladders*. È utilizzato per ricostruire la direzione di incidenza della particella utilizzando il segnale raccolto su entrambi i piani (le strisce di questi sono poste ortogonalmente le une rispetto alle altre per fornire coordinate X e Y);
- Un *TRIGGER* costituito da un piano di scintillatore plastico (20 cm × 18 cm × 0.5 cm) diviso in sei *paddles*. Localizzato subito sotto il Tracking System, quando è attraversato da una particella genera un segnale di trigger che dà inizio all'acquisizione dei dati. Tale piano ottiene anche informazioni aggiuntive sulla particella incidente riguardanti la posizione, la perdita di energia e, in caso, la presenza di eventi multiparticella che vanno rigettati;
- Un *calorimetro*, per la misura dell'energia depositata dalla particella incidente. Esso consiste di due sezioni, una superiore e una inferiore. La prima sezione, detta TOWER, è composta da 16 piani (15cm × 15cm × 1cm) di scintillatore plastico che diventa luminescente se colpito da particelle cariche. Ciascun piano è letto da due *Photo Multiplier Tubes* (PMTs) posizionati agli angoli opposti dello scintillatore. La seconda sezione invece consta di una matrice 3 × 3 di scintillatori inorganici LYSO (Ortosilicato di Lutezio e Ittrio), per una dimensione totale 15 cm × 15 cm × 4 cm. Ogni cristallo di LYSO è letto a sua volta da un PMT posto al di sotto della sua superficie inferiore. La matrice di LYSO permette di estendere il range energetico rivelabile dal detector: aumentando con la sua presenza lo spessore e la densità del materiale incontrato dalla particella incidente, permette di rivelare anche particelle più energetiche;
- Un sistema di anti-coincidenza (*VETO*) che circonda completamente il calorimetro e che include 5 piani di scintillatori plastici (4 laterali e uno inferiore),

letti anch'essi da due PMTs. Questi piani sono circondati a loro volta da piastre in fibra di carbonio che prevengono qualsiasi possibile problema generato da shock, urti o stress applicati al rivelatore. Inoltre, il VETO è utile anche per rigettare particelle non interamente contenute nel calorimetro o che non rispettano le condizioni di trigger, come ad esempio particelle entranti lateralmente o particelle secondarie prodotte all'interno del calorimetro.

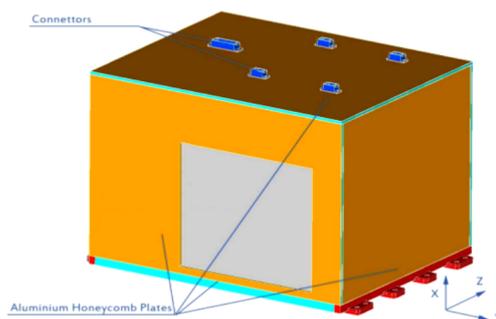


Figura 2.7: L'involucro in alluminio del rivelatore HEPD. Sono mostrati anche i connettori per l'interfaccia del satellite.

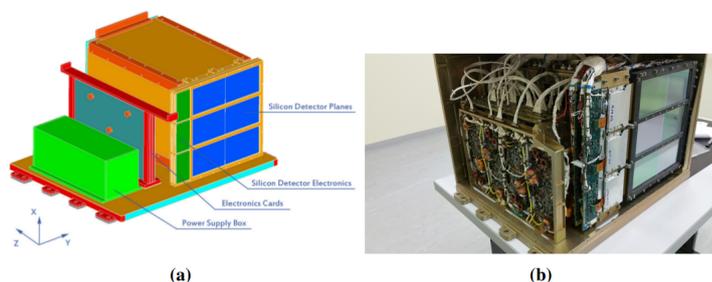


Figura 2.8: Rappresentazione schematica dell'interno del box di HEPD (a) e foto presa all'incirca dalla stessa prospettiva (b)

## 2.3 Il Tracker

Il sistema di tracciamento dell'HEPD è composto da 2 piani (denominati esterno e interno) di rivelatori a micro-strip di silicio a doppia faccia, posizionati nella parte superiore dell'apparato e con una distanza di 1 cm tra di loro. La caratteristica principale di questi sensori è la presenza di uno strato sensibile sul lato superiore e inferiore del

wafer di silicio, con le strisce su un lato installate ortogonalmente a quelle dell'altro, al fine di realizzare la misurazione delle coordinate X e Y della particella ionizzante incidente.

L'uso di un rivelatore a micro-strip a doppia faccia (vedi Fig. 2.10), invece di due strati a singola faccia, offre diversi vantaggi: in particolare il minor spessore totale di materiale che può essere ottenuto porta ad una riduzione degli effetti indesiderati come lo scattering multiplo e semplifica la struttura meccanica. Lo spessore del sensore è un fattore importante che deve essere preso in considerazione, poiché è richiesto un compromesso tra la minimizzazione delle interazioni secondarie, o dello scattering coulombiano, e la necessità di produrre un numero sufficientemente grande di coppie elettrone-lacuna, al fine di fornire un buon rapporto segnale/rumore. Il valore di  $300\mu\text{m}$  è generalmente considerato un valore adeguato per lo spessore [34] ed è utilizzato anche per i sensori in silicio di HEPD.

Quando una particella ionizzante attraversa i sensori, le coppie elettrone/lacuna create nella regione di svuotamento, si spostano in direzioni opposte: le lacune vengono raccolte dalle strisce  $p$  sul lato X e gli elettroni sono raccolti dalle strisce  $n$  sul lato Y. Dalla distribuzione della carica ionizzante raccolta dalle strisce, è possibile ricostruire la coordinata del punto di intersezione normale alle strisce.

Nel rivelatore *tracker* di HEPD, ogni piano di silicio, con una dimensione di  $213,2 \times 214,8\text{mm}^2$  esclusa la meccanica, è suddivisa in 3 sezioni identiche indipendenti (*ladders*) lungo l'asse X denominate *top*, *central* e *bottom*, dove il *top* è quello con il valore più alto della coordinata X.

Ogni *ladder* è composto da 2 moduli (o sensori). Il singolo sensore, prodotto da FBK[38], ha una dimensione di  $109,63\text{mm} \times 77,58\text{mm}$  ma con un'area attiva più piccola di  $106,63\text{mm} \times 71,58\text{mm}$  per evitare interferenze causate dalla struttura di supporto al bordo (2.11).

## 2.4 Il sistema di TRIGGER

Il sistema di *trigger* è costituito da un sottile strato di scintillatore plastico ( $20 \times 18 \times 0,5\text{cm}^3$ ) diviso in 6 segmenti (*paddles*) di dimensione  $20 \times 3 \times 0,5\text{cm}^3$ , ciascuno letto da due Tubi fotomoltiplicatori Hamamatsu (PMT). Il piano è posizionato sotto il *tracker* e assolve ai seguenti compiti:

- una rapida generazione del segnale di trigger per avviare e sincronizzare l'acquisizione dei dati per l'intero apparato;

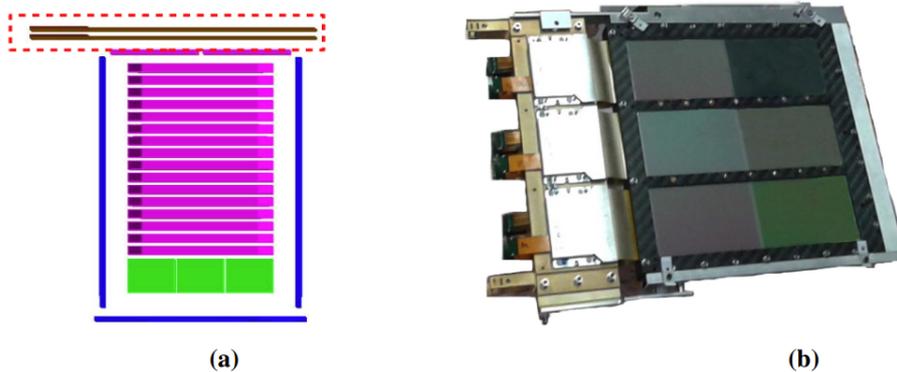


Figura 2.9: Schema che mostra il posizionamento del tracker (a). Fotografia dei piani di silicio del tracker assemblati (b)

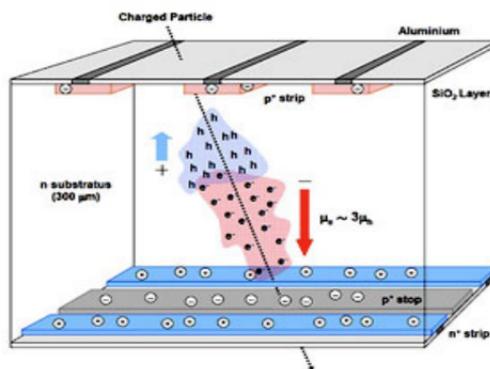


Figura 2.10: Schema del sensore a micro-strip di silicio doppia faccia. Una particella carica attraversa il sensore e genera coppie  $e^-$  /lacuna che migrano verso direzioni opposte.

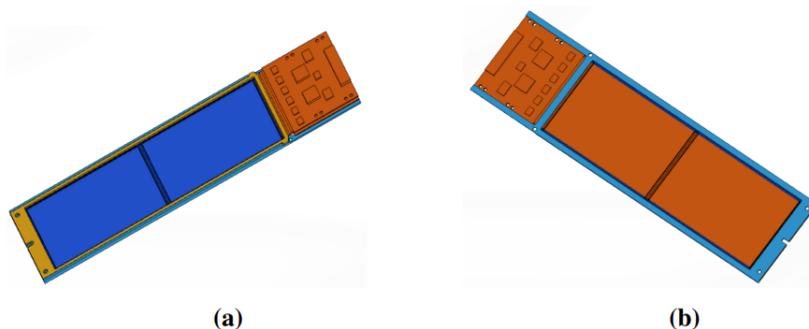


Figura 2.11: Disegni meccanici del ladder in silicio di HEPD: lato S (a) e lato K (b). Si possono vedere i due sensori e il circuito ibrido di lettura.

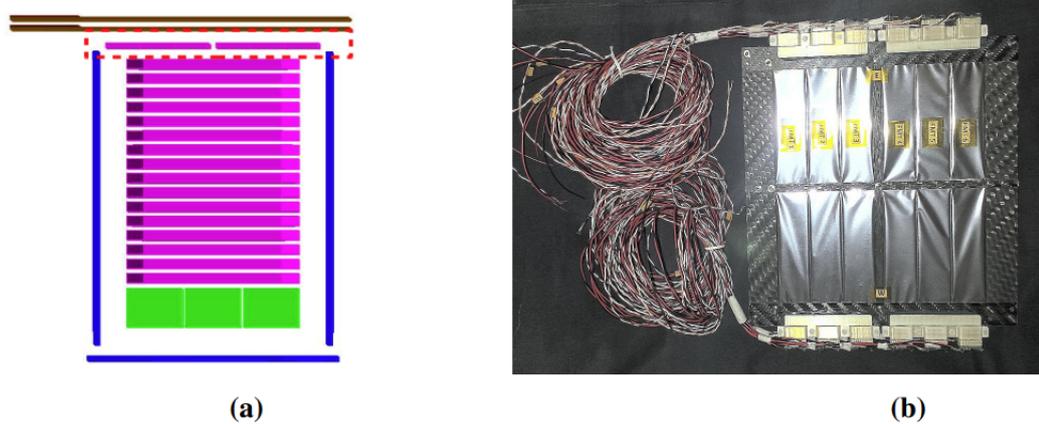


Figura 2.12: Schema che mostra il posizionamento del trigger plane (a). Immagine del piano di trigger (b).

- la rimozione dell'ambiguità sulla ricostruzione della posizione nella coordinata Y introdotta dal sistema tracker;
- un'eventuale determinazione secondaria della perdita di energia di ionizzazione prodotta nello strato, oltre alla misurazione più precisa ottenuta con il tracker;
- la possibilità di rifiutare eventi di incidenza con più di un paddle (eventi multi-particella).

Per generare il segnale di trigger, è necessaria la coincidenza tra un segnale su almeno uno dei paddle e un segnale su almeno un piano aggiuntivo del calorimetrico. Un'immagine del piano di innesco può essere vista in Fig. 2.12 (b).

## 2.5 Il Calorimetro

Il calorimetro del rivelatore *HEPD* è costituito da due differenti parti. La prima, in alto e immediatamente sotto il piano del Trigger, è composta da 16 piani di scintillatore plastico. Ogni piano ha una dimensione di  $15 \times 15 \times 1 \text{ cm}^3$  ed è letto da due PMT, posizionati ai due angoli opposti del piano. La parte inferiore del calorimetro è costituita di una matrice  $3 \times 3$  di scintillatori a cristalli inorganici LYSO, per un piano con dimensione totale di  $15 \times 15 \times 4 \text{ cm}^3$ .

La stabilità del sistema è assicurata da un'adeguata struttura meccanica, che mantiene il rivelatore in posizione e previene eventuali danni durante la fase di lancio. Gli

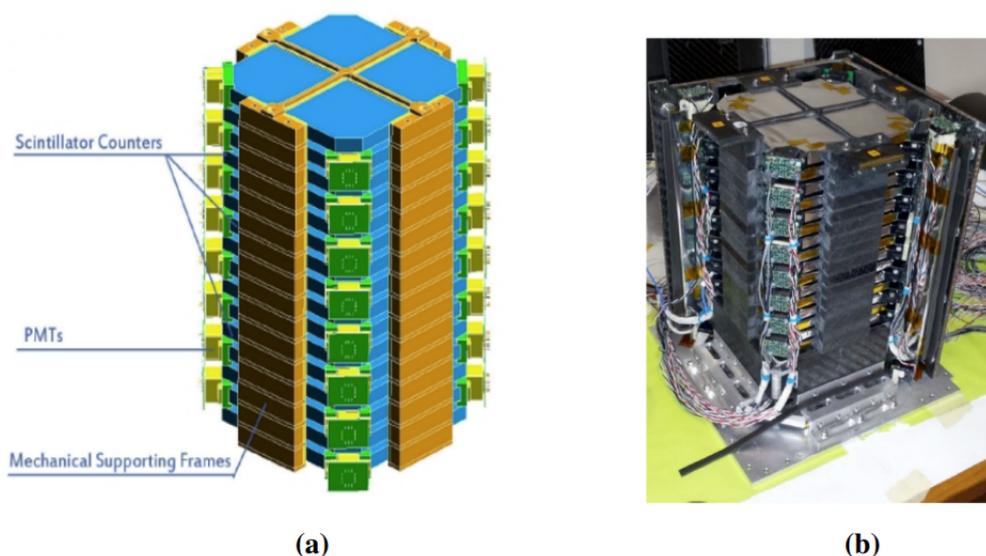


Figura 2.13: Schema meccanico della torre di scintillatori che contiene i piani di scintillatore plastico, i tubi fotomoltiplicatori agli angoli e la struttura meccanica (a). Fotografia della torre assemblata (b).

scintillatori sono disposti in una pila verticale e tenuti in posizione da un telaio in fibra di carbonio, montati uno sopra l'altro, in modo tale da formare una torre rigida dove ogni piano scintillatore è separato da quelli adiacenti (vedi schema e figura in Fig 2.13). Strati di materiale relativamente morbido, Poron a celle aperte (la struttura gialla nello schema meccanico in Fig. 2.13 (a)), posto tra i piani scintillatori e la fibra di carbonio, prevengono qualsiasi sollecitazione o urto che possa essere trasferito agli scintillatori stessi, in modo da evitare danni. L'elevata rigidità del sistema offre quindi un ampio margine di sicurezza contro le sollecitazioni e le vibrazioni.

Gli scopi del calorimetro sono la misurazione dell'energia depositata e il range delle particelle incidenti. La presenza dello strato inferiore LYSO fornisce un incremento del range di energia operativa attraverso uno spessore maggiore di materiale con alta densità.

### 2.5.1 Gli Scintillatori Plastici

Uno scintillatore in materiale plastico, caratterizzato da una risposta rapida e da una luce di uscita relativamente elevata, è stato scelto per la realizzazione del calorimetro superiore. Uno dei più grandi vantaggi degli scintillatori plastici è la loro capacità di essere modellati in quasi tutte le forme desiderate così come l'elevata durata.



Figura 2.14: Immagine di una piano scintillatore della torre del calorimetro.

Un piano scintillatore del calorimetro può essere visto in Fig. 2.14. I quattro angoli sono stati tagliati al fine di creare lo spazio per posizionare i PMT. Ogni piano contiene due PMT, posizionati a due angoli opposti. Per motivi meccanici, due piani consecutivi hanno i PMT posti su coppie differenti di angoli, come si può vedere in Fig. 2.13.

### 2.5.2 La matrice LYSO

La parte inferiore del calorimetro è costituita da uno scintillatore inorganico LYSO (Lutetium Itrium ortosilicato con cerio drogato). Questo è un materiale ad alta densità ( $d = 7,3g/cm^3$ ) che, unito ad uno spessore di  $4cm$ , permette di aumentare il range di energia operativa del rivelatore. Lo strato LYSO è formato da una matrice  $3 \times 3$  (vedi Fig. 2.15), dove ogni cristallo ha dimensioni  $4,8 \times 4,8 \times 4cm^3$  ed è letto da un unico PMT posto nella parte inferiore (il lato opposto al piano di Trigger). Questi cristalli hanno i vantaggi di un'elevata resa luminosa, tempo di decadimento rapido, eccellente risoluzione energetica e basso costo. Il picco dello spettro di emissione è a  $428nm$ ; questo rende possibile l'utilizzo, per la lettura, degli stessi tubi Fotomoltiplicatori adottati per i piani di scintillatore plastico.

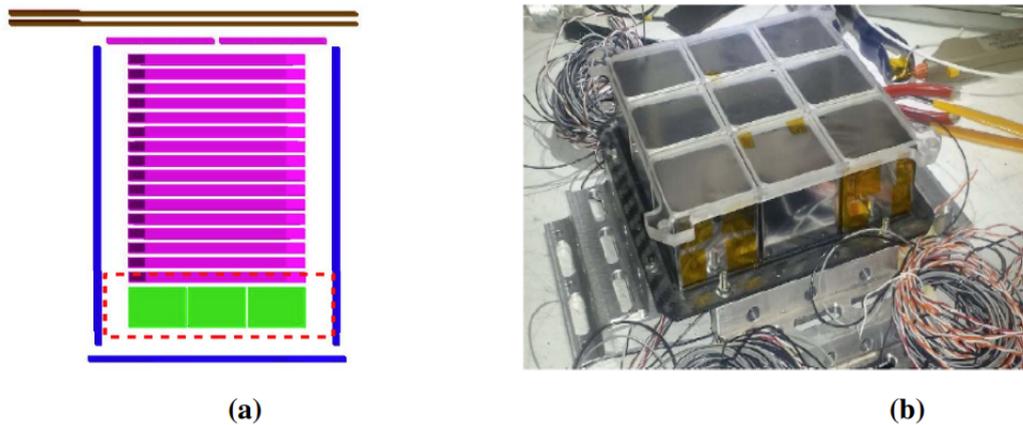


Figura 2.15: Schema che mostra il posizionamento della matrice LYSO (a). Immagine del piano della matrice LYSO (b).

### 2.5.3 I tubi fotomoltiplicatori

Un tubo fotomoltiplicatore è composto da uno strato fotosensibile (foto-catodo), che emette elettroni quando colpito dalla luce in uno specifico intervallo di lunghezze d'onda, un insieme di elettrodi intermedi (*dinodi*), e un anodo finale, mantenuto ad alte tensioni rispetto al fotocatodo. L'intera struttura si trova in una scatola metallica sottovuoto con una finestra di vetro per la luce entrante, in corrispondenza del fotocatodo. Gli  $e^-$  emessi dal fotocatodo, sono accelerati dal forte campo elettrico verso il primo stadio *dinodico* e provocano l'emissione di un numero maggiore di elettroni secondari. Quindi questi vengono accelerati verso il dinodo successivo fino a raggiungere l'anodo formando in tal modo un impulso di corrente.

## 2.6 Il sistema VETO

Il sistema di *veto* di cui è dotato HEPD comprende cinque sottili scintillatori plastici che circondano completamente il calorimetro. I contatori del veto sono posti all'interno di un volume creato tra struttura del calorimetro e le piastre a nido d'ape in fibra di carbonio posizionate esternamente. Uno spessore di Poron, posto tutto intorno agli scintillatori, previene qualsiasi stress e shock da trasferire ai contatori stessi.

Il rivelatore veto è fondamentale per rifiutare qualsiasi tipo di background causato da eventi fuori dall'accettazione. In particolare, il sistema è progettato per:

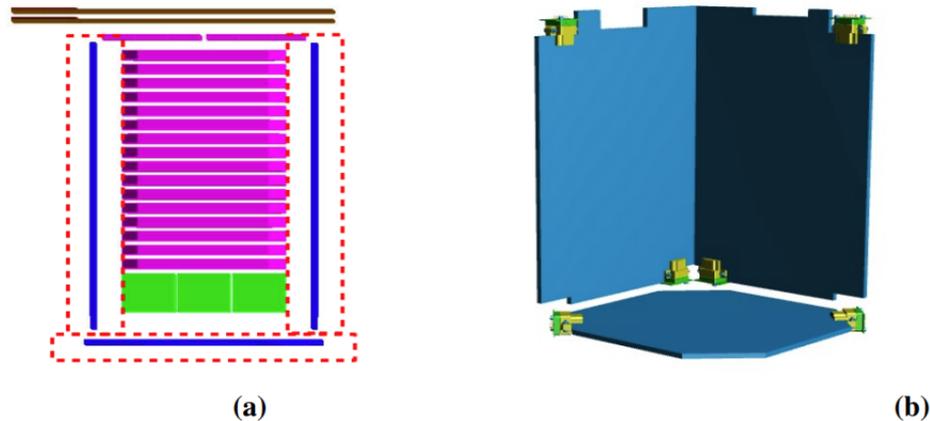


Figura 2.16: Schema che mostra il posizionamento del sistema veto (a). Schema del pannello inferiore e due di quelli laterali (b).

- l'identificazione e il rifiuto di particelle che non attraversano i due piani di tracking ma generano comunque un segnale di trigger (ad esempio entrando lateralmente nell'apparato);
- il rifiuto di particelle secondarie prodotte all'interno del calorimetro;
- il rifiuto di eventi non del tutto contenuti.

Tutti i piani del sistema di veto hanno uno spessore di 5 mm e sono letti dagli stessi due PMT. Quattro piani si trovano ai lati del calorimetro e il quinto è al di sotto dello strato LYSO, per respingere particelle che non sono completamente contenute all'interno del calorimetro o particelle ascendenti (Fig. 2.16).

## 2.7 Configurazioni di trigger

Introduco brevemente le configurazioni di trigger possibili in quanto per ognuna di queste il rivelatore è in grado di produrre dei *rate meter*, ovvero delle integrazioni su finestre temporali di 1s dei conteggi degli eventi che tornano molto utili in ambito space weather. Infatti la configurazione *online* di trigger è la «T & P1 & P2» (vedi sotto), ma i dati di rate meter vengono acquisiti per ognuna delle configurazioni e contengono informazioni sui rate di conteggio anche a soglie più basse di energia (usando ad esempio il rate meter della configurazione con il solo piano di trigger).

Una particella che incide su HEPD è considerata un evento valido se, attraversando la strumentazione, dà il via all'acquisizione dei dati. Tale processo è definito dalla *configurazione di trigger*, la quale richiede un rilascio di energia da parte della particella in uno o più piani della strumentazione che costituisce il rivelatore, e solo se ciò avviene inizia il processamento ed il download a terra delle informazioni. Le configurazioni di trigger disponibili per HEPD sono:

- T
- T & P1
- T & (P1 || P2)
- (T3 || T4) & (P1 || P2)
- T & P1 & P2
- T & P1 & P2 & P3
- T & (P1 || P2) & (P15 || P16)
- T & (P1 || P2) & L

Le paddles del piano di trigger e i piani della TOWER vengono considerati colpiti se almeno uno dei PMT che li leggono genera un segnale, proporzionale all'energia depositata dalla particella incidente, che superi una certa soglia. Pertanto, ad esempio, si indica con T l'OR dei dodici PMT corrispondenti alle sei paddles del piano di trigger; con T3 l'OR dei due PMT che leggono la paddle T3 del trigger; con P1 l'OR dei due PMT del primo piano del calorimetro (analogamente per gli altri piani Pn). Basterà quindi che almeno uno dei due PMT del piano definito nella configurazione rilasci un segnale affinché questo piano venga considerato attraversato dalla particella, e si dia il via all'acquisizione dei dati sull'evento. Normalmente la configurazione «online» è T & P1 & P2, ovvero un evento è valido solo se rivelato dal piano di trigger e dai primi due piani del Calorimetro.



## 3 I sistemi di monitoraggio

In questo capitolo introduco sinteticamente alcuni concetti fondamentali dei sistemi di monitoraggio e del processo che dal dato grezzo porta alla conoscenza secondo il paradigma della piramide DIKW. Dedico una sezione alle tipologie di monitoraggio e alla complessità dei sistemi in funzione del fenomeno che si vuole studiare, del numero di variabili da monitorare e del contesto in cui il sistema si inserisce.

L'ultima sezione di questo capitolo è dedicata alle serie storiche (*time series*) e alla loro utilità nell'ambito dello studio di alcuni fenomeni e del loro andamento nel tempo. Questo infatti sarà il contesto in cui si inserisce la progettazione del sistema oggetto di questa tesi.

### 3.1 Introduzione

Il concetto più generale di *monitoraggio* si riferisce al controllo della variazione di una o più grandezze in funzione di altre. Gli scopi di un tale sistema possono essere molteplici come lo sono gli ambiti della sua applicazione.

Molto spesso un sistema di monitoraggio è associato ad una programmazione in un ciclo continuo di miglioramento di tipo PDCA (*Plan, Do, Check, Act*) in cui si desidera migliorare le prestazioni di un sistema o di un servizio. Si comincia appunto dalla sua *pianificazione*, si passa poi al *funzionamento*, si continua con il *monitoraggio*, si procede in fine ad *applicare* eventuali miglioramenti o ad eliminare malfunzionamenti per poi ricominciare il ciclo. La realizzazione dell'ultimo passaggio nasconde in se uno dei requisiti fondamentali di un sistema di monitoraggio: la predisposizione o la conoscenza di valori di riferimento (o di *soglia*) rispetto ai quali controllare l'andamento delle grandezze misurate.

Altre volte, spesso in ambito scientifico, si desidera raccogliere informazioni riguardanti fenomeni poco noti dei quali si vuole capire o studiare il funzionamento. In questo caso siamo fuori dal paradigma a *ciclo continuo*, ma rimane di fondamentale importanza la conoscenza e la predisposizione delle *soglie di riferimento*.

È chiaro inoltre che un «sistema di monitoraggio» non consiste esclusivamente nella visualizzazione del dato, ma è parte integrante di questo il meccanismo di raccolta dei dati, il loro formato, la frequenza di osservazione e il tipo di grandezza che si vuole osservare. Nell'ambito di questo lavoro, ad esempio, possiamo considerare come il sensore di raccolta dati del sistema di monitoraggio il rivelatore HEPD stesso, che raccoglie dati quali ad esempio conteggio, flusso, energia, direzione di particelle. Queste sono le grandezze che andiamo a monitorare, che confrontate con valori di soglia noti, ci danno indicazioni sui fenomeni fisici ad esse associati.

È altresì vero, seguendo la piramide DIKW (*Data, Information, Knowledge, Wisdom*), che il dato grezzo non porta con sé l'*informazione*, è solo un numero e va elaborato, contestualizzato e correlato con altri dati per poter ottenere informazioni significative. Informazioni che poi organizzate e messe in pratica portano alla *conoscenza*. Diventa quindi determinante avere lo strumento per accedere ai dati in modo veloce e chiaro, per poterli contestualizzare e correlare efficacemente con altri dati e per poterli confrontare con le soglie di riferimento. La realizzazione di questo strumento è lo scopo del mio lavoro.

Sorvolo infine sulla *saggezza*, intangibile modalità con cui si fa uso della propria conoscenza e della propria esperienza.

## 3.2 Tipologie e complessità

Come anticipato nell'introduzione, la progettazione di un sistema di monitoraggio non può prescindere dal suo scopo, dal tipo di dato che si vuole trattare e dalle *informazioni* che si vogliono ottenere ovvero dal fenomeno che è oggetto di studio. Da questo derivano scenari e, di conseguenza, sistemi differenti.

Si possono considerare sistemi semplici, per scopi non critici e a bassa frequenza come ad esempio la valutazione della qualità dell'aria e della temperatura di un ambiente, il cui scopo potrebbe essere raccogliere statistiche per risparmio energetico. Ben diverso è il monitoraggio in tempo reale dell'infrastruttura IT di una grande azienda a scopi di sicurezza informatica.

Nel secondo caso dovremo monitorare, in tempo reale, una quantità innumerevole di grandezze differenti (accessi ai servizi e alle macchine, signature di virus nel traffico di rete, stato dei sistemi etc...) e realizzare un sistema di alert immediato allo scopo di reagire il prima possibile ad un'anomalia. La quantità, la tipologia e la frequenza dei

dati raccolti così come lo scopo per il quale vengono raccolti, determinano i requisiti in base ai quali il sistema viene progettato e implementato.

In relazione a differenti caratteristiche possiamo brevemente catalogare i sistemi come segue:

- per numero di grandezze e complessità delle correlazioni:

semplici: monitoraggio di un basso numero di grandezze, ad esempio monitoraggio della temperatura di una sala macchine di un CED;

complessi: alto numero di grandezze da monitorare come nel caso appunto dell'infrastruttura IT di un'azienda, elevata quantità di dati, correlazioni multiple e complesse.

- in base alla frequenza di processamento dei dati:

tempo-reale: questo contesto prevede la necessità di monitorare una o più grandezze in tempo reale, i rilevatori sono in grado di fare acquisizione *continua* dei dati che vengono spesso rappresentati come uno o più flussi di eventi organizzati in base al tempo (*time series*); (Sez. 3.3)

Alta-frequenza: quando il monitoraggio non è continuo ma la frequenza di acquisizione e di processamento dei dati è molto alta in relazione all'andamento dell'evento da osservare;

Bassa-frequenza: la raccolta dati, anche di sistemi complessi, può avvenire in intervalli di tempo relativamente lunghi (giorni, settimane) come avviene nel caso di indagini statistiche di vario genere (es. demoscopiche o ambientali).

- in base al contesto:

critico: sicurezza informatica, apparecchiature mediche in strutture ospedaliere, automotive sono ad esempio contesti in cui la risposta ad un evento deve essere molto rapida ed efficace o di importanza vitale, spesso sono sistemi *complessi e real-time*;

non-critico: nel caso in cui i dati trattati e le procedure di risposta possono non essere critiche o il monitoraggio è mirato allo studio dei fenomeni ad essi associati. (Fig. 3.1)

Nello specifico il sistema di monitoraggio oggetto di questa tesi può essere catalogato come *complesso, a medio-bassa frequenza e non critico* come vedremo meglio nel capitolo successivo.

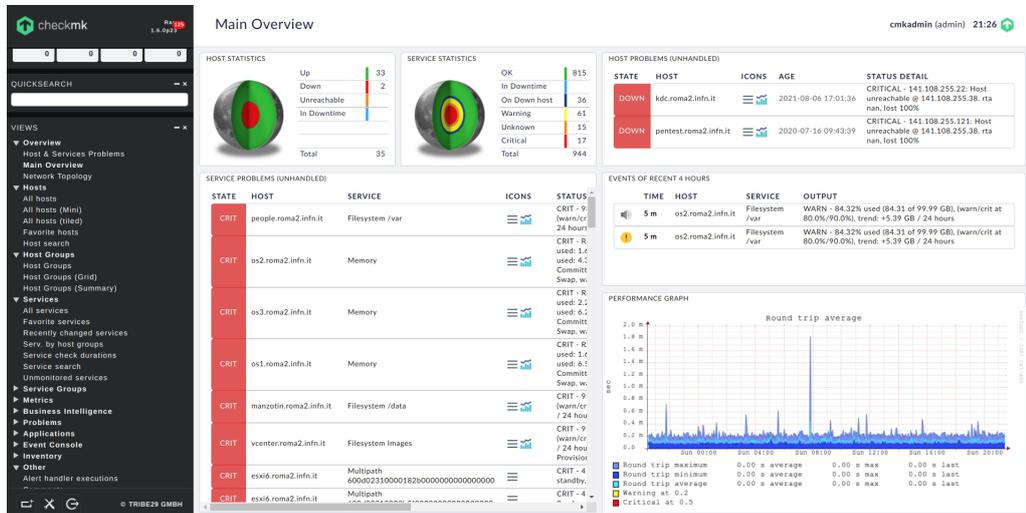


Figura 3.1: Dashboard di un sistema di monitoraggio complesso in tempo reale

### 3.3 Time series

Se volessimo osservare l'insorgenza di una malattia in funzione della regione di appartenenza della popolazione staremmo osservando un fenomeno in funzione di una *variabile qualitativa*, che è appunto la regione sul campione considerato. Il fenomeno oggetto di studio può essere indipendente dal tempo per lo scopo della nostra analisi, come in questo caso, e rappresentare una fotografia in un momento specifico. Questo approccio tipico si chiama trasversale (*cross-sectional*) il cui scopo non è studiare l'andamento temporale del fenomeno. Dunque un sistema di monitoraggio ben progettato dovrebbe considerare molti aspetti, da un campione statisticamente rappresentativo ad una raccolta dati accurata fino ad una presentazione del dato chiara ed efficace, ad esempio con una visualizzazione a barre mostrando bene le diverse regioni e il numero di unità affette dalla malattia. (Fig. 3.2)

Quando il carattere qualitativo rispetto al quale vogliamo studiare un fenomeno è invece il tempo, si parla serie storica o temporale (*time series*). In particolare si parla di time series quando si osserva una variabile in funzione di intervalli regolari di tempo.

Ci sono molti esempi di time series in ogni ambito: la misura della temperatura ambientale, il numero di macchie solari, l'occupazione di memoria di un computer, l'andamento di un titolo quotato in borsa e molti altri. (Fig. 3.3)

L'assunzione di base dell'approccio moderno è quella di eseguire molte osservazioni di altrettante variabili aleatorie non necessariamente indipendenti. Per cui un approccio stocastico per le time series prevede una correlazione maggiore tra due osservazioni vicine nel tempo rispetto a due osservazioni molto lontane.

Un approccio più tradizionale all'analisi delle time series è di tipo più deterministico in cui si descrive la serie come divisa in una parte *sistematica* ed una parte *stocastica*:

$$F_t = f(t) + n_t$$

dove  $f(t)$  è la parte sistematica (deterministica) e  $n_t$  la componente stocastica costituita da una serie di *variabili aleatorie*. Inoltre questo approccio prevede di scomporre la parte sistematica in tre componenti:

- *tendenziale (trend)*: l'andamento di fondo del fenomeno osservato (es. crescente o decrescente);
- *ciclica*: l'alternanza di fluttuazione regolari o meno rispetto al trend;
- *stagionale*: la ripetizione di fluttuazioni di intensità simile a intervalli regolari di tempo (es. giorni, settimane, mesi).

Ci sono poi:

- una componente casuale (*accidentale*);
- una componente occasionale o anomala (*outliers*): più rara e con valori distanti da gli altri valori.

Un aspetto fondamentale delle serie temporali è la dipendenza *ordinata* dal tempo, cambiare l'ordine di due osservazioni cambia radicalmente il significato della serie.

I motivi dell'analisi delle *time series* sono molteplici:

- risalire al meccanismo che genera il fenomeno osservato;
- prevedere l'andamento futuro del fenomeno attraverso *time series forecasting*;
- filtrare, ovvero rintracciare le componenti citate sopra;

- descrivere efficacemente l'andamento di un fenomeno, soprattutto mediante strumenti visuali come i grafici della serie, dando la possibilità di evidenziare rapidamente trend e anomalie.

I metodi matematici di analisi delle time series vanno oltre lo scopo di questa tesi che verte invece sulla progettazione di un sistema di monitoraggio in grado di assolvere completamente all'ultimo degli scopi citati sopra, evidenziando in tempi brevi sia fenomeni evidenti legati allo space weather, che i dataset di maggiore interesse per l'analisi relativa ad eventi sismici e agli altri ambiti di ricerca abbracciati dalla missione CSES. Tuttavia, come vedremo nel prossimo capitolo, con la piattaforma scelta per la gestione dei dati, sono riuscito certamente ad utilizzare una serie di funzioni tipiche nella gestione delle serie che permettono di effettuare anche un'analisi e una trasformazione notevole sui dati prima della generazione del grafico.

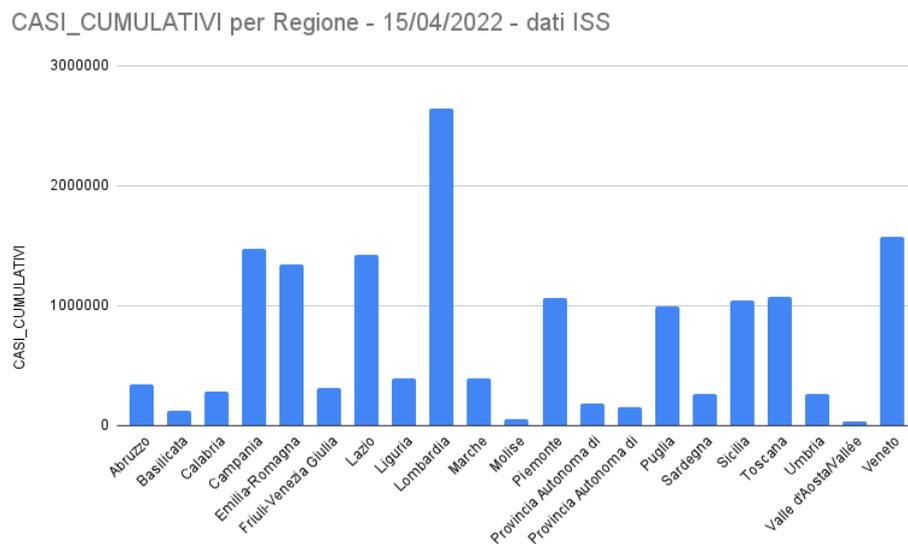


Figura 3.2: Esempio di istogramma - cross-sectional

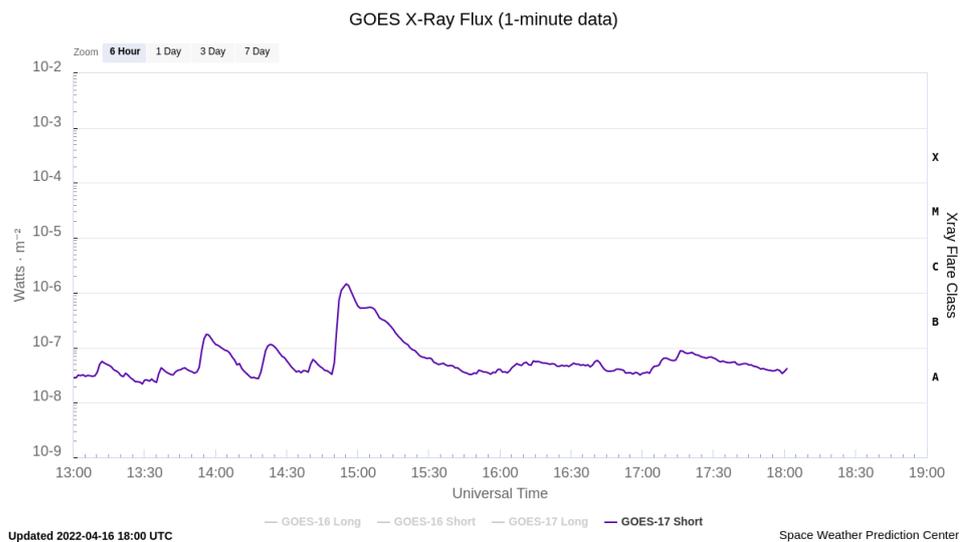


Figura 3.3: Esempio di grafico - time series



## 4 Realizzazione del sistema

Nel presente capitolo affronto la parte fondamentale del mio lavoro. La progettazione e la realizzazione di un sistema di alert e monitoraggio dei dati acquisiti dal rivelatore HEPD a bordo del satellite CSES. Tale sistema sarà in grado in futuro di raccogliere dati provenienti anche da altri payload (es. HEPP o EFD), da altri esperimenti o anche Open Data resi disponibili dalla comunità scientifica.

Illustro i requisiti del sistema e il contesto nel quale lavorerò, ovvero la possibilità di mostrare efficacemente grafici e statistiche delle serie temporali dei conteggi di elettroni e protoni nell'ambito dello space weather, nel quale il rivelatore si sta mostrando molto utile nel corso di questi quattro anni.

Il capitolo segue nel suo svolgimento il workflow dei dati dal momento in cui vengono scaricati e messi a nostra disposizione, passando per la descrizione dell'agent da me sviluppato in python che si occupa dell'estrazione dei dati e la formattazione dei *point* (così vengono definiti i singoli record da immagazzinare in InfluxDB) da inserire nel database. Per finire con l'interrogazione, attraverso opportune query e filtri, dei dati stessi che porteranno alla creazione dei *pannelli* più significativi e all'organizzazione delle *dashboard* e delle soglie di alert di Grafana.

Dedico in fine una sezione a vari casi d'uso in cui il sistema si mostra efficace.

### 4.1 Identificazione dei Requisiti

Il mio approccio alla progettazione di questo sistema è stato, e sarà, di tipo *agile*<sup>1</sup>, sono partito quindi con la necessaria individuazione dei requisiti per la realizzazione di un primo prototipo funzionante per poi iterare sottoponendo di volta in volta il sistema ai ricercatori che lo utilizzeranno.

---

<sup>1</sup>Nella progettazione del software il modello *agile*, contrapposto al modello *a cascata* (*waterfall*), si riferisce ad un approccio meno strutturato che mira alla realizzazione e la consegna di un sistema comunque funzionante e di qualità anche se non completo di tutte le sue funzionalità. Queste andranno implementate per successive iterazioni (ognuna delle quali è un piccolo progetto i cui requisiti possono cambiare durante il percorso) in un processo che coinvolge l'utilizzatore finale.

I dati acquisiti dal rivelatore HEPD attraversano vari stadi di processamento che danno luogo ad una ricostruzione dei singoli eventi con un grado di astrazione sempre maggiore. Da ciò derivano diversi formati dei dati relativi appunto ai diversi livelli: L0, L1 e L2. A questi si aggiunge un livello L3-h5 (anche L3) per uniformare il prodotto finale di HEPD a quello degli altri payload di CSES.

Da L0 ad L2 i dati sono registrati sulla base del singolo evento passando da L1, dove sono presenti i segnali ADC (*Analogic/Digital Converter*) dei vari sensori (micro-strip di silicio, scintillatori del trigger e del calorimetro, quelli della matrice LYSO e del sistema VETO) fino ad L2 dove troviamo informazioni di alto livello come:

- la stima dell'energia persa nel calorimetro;
- la direzione di arrivo stimata;
- indicatori sull'identificazione della particella;
- le coordinate del satellite al momento dell'evento;
- altre informazioni derivate dalla posizione del satellite come il valore di L-shell, di campo magnetico etc...

Nell'L2 sono anche presenti i *rate meters*: ad ogni configurazione di trigger è associato cioè un contatore che somma gli eventi registrati in una finestra di un secondo per quella specifica configurazione.

Questi livelli sono tutti in formato ROOT.

Il formato L3 contiene i dati derivati direttamente dal formato L2 con la differenza che anziché registrare i dati sulla base del singolo evento è presente una integrazione per ogni secondo di acquisizione. Quindi i valori relativi ai conteggi sono il risultato della somma all'interno della finestra di acquisizione di un secondo. Queste informazioni sono accompagnate dai valori medi (all'interno della medesima finestra) di altitudine del satellite, campo magnetico, coordinate geografiche e geomagnetiche, i pitch angle delle particelle, i valori di L-shell e il tempo in formato UTC. I dati sono poi aggregati per semi-orbita.

I dati di livello L3, sebbene contengano informazioni meno significative per la ricostruzione esatta di un evento, si prestano quindi particolarmente bene allo scopo di fare un monitoraggio di tipo *time series* sul conteggio di elettroni e protoni. Correlando questi dati con altre informazioni, quali ad esempio latitudine e longitudine, si può ottenere una visione immediata dell'andamento nel tempo dei fenomeni associati alla produzione di queste particelle e all'individuazione dei dataset e dei periodi

maggiormente interessanti da approfondire con successive analisi sul livello L2. I dati L3 quindi, se trattati e correlati adeguatamente, sono ideali per avere una visione qualitativa veloce (*quick look*) dei fenomeni possibili ad essi associati.

Da questo è nata l'esigenza di realizzare un sistema di monitoraggio che avesse i requisiti seguenti:

1. capacità di accogliere e gestire l'intera mole di dati L3 prodotta fin'ora dall'inizio della missione nonché quelli futuri;
2. possibilità di raccogliere dati remoti senza doverli spostare localmente;
3. fortemente orientato alle time series con possibilità di effettuare trasformazioni dei dati applicando funzioni di analisi;
4. flessibile e veloce nell'interrogazione dei dati, il che comporta la possibilità di realizzare filtri e query personalizzate;
5. visualizzazione dei dati chiara ed efficace attraverso la creazione di dashboard personalizzate e dinamiche;
6. possibilità di impostare soglie di riferimento e un sistema di alert basato sui sistemi più comuni di messaggistica (posta, Teams , Telegram etc...).
7. la possibilità di condivisione e collaborazione di gruppo;
8. la capacità di raccogliere dati anche da diverse sorgenti (altri payload di CSES, altri esperimenti, formati o database) in modo da rendere una visione integrata ed aumentare notevolmente le potenzialità di confronto e comprensione dei fenomeni oggetto di studio.
9. Non ultimo rendere un servizio di condivisione alla comunità scientifica.

## 4.2 Introduzione alla pipeline

I requisiti elencati sopra mi hanno portato a progettare un sistema flessibile costituito da tre componenti indipendenti con separazione e «basso accoppiamento» dei servizi. La possibilità di separare completamente lo strato di acquisizione e formattazione dei dati da quello di immagazzinamento e gestione e poi da quello di visualizzazione è stata la soluzione per poter rispondere ai requisiti 2. e 8. ad esempio. Così come la

scelta e la configurazione di software appropriato è stata necessaria per rispondere al resto dei requisiti.

Uno schema semplificato che rappresenta l'intera pipeline di gestione dei dati è quello riportato in Fig. 4.1. I dati vengono scaricati dal *Ground Segment* cinese con cadenza giornaliera e salvati su volumi che, a seconda del payload, risiedono nelle sedi dell'ASI oppure dell'INAF. Questi due volumi saranno possibilmente integrati attraverso un progetto a breve termine nel quale sono personalmente coinvolto. Nel caso dei dati HEPD il volume di riferimento si trova nella sede dell'ASI dove risiede il *Ground Segment* italiano della missione.

Come accennato un agent scritto da me, integrato nella pipeline di processamento dei dati (L0->L3), si occupa di rilevare automaticamente la presenza di nuovi file, scorrere le cartelle contenenti i dati, estrarli dal loro originale formato, formattare i *punti* e, attraverso una chiamata alle API, inserirli nella piattaforma di gestione e immagazzinamento dei dati *InfluxDB*. Da qui, sempre attraverso API vengono messi a disposizione del sistema di visualizzazione (*Grafana*).

L'utente del sistema, in fine, interagisce con Grafana via http/https potendo controllare e gestire completamente i dati attraverso l'interfaccia di query integrata. Insieme ai dati, infatti, InfluxDB mette a disposizione via API un completo linguaggio di query che permette non solo l'interrogazione e la visualizzazione, ma anche l'applicazione di filtri e la trasformazione dei dati e delle time series adattandoli alle esigenze specifiche di ogni caso d'uso possibile.

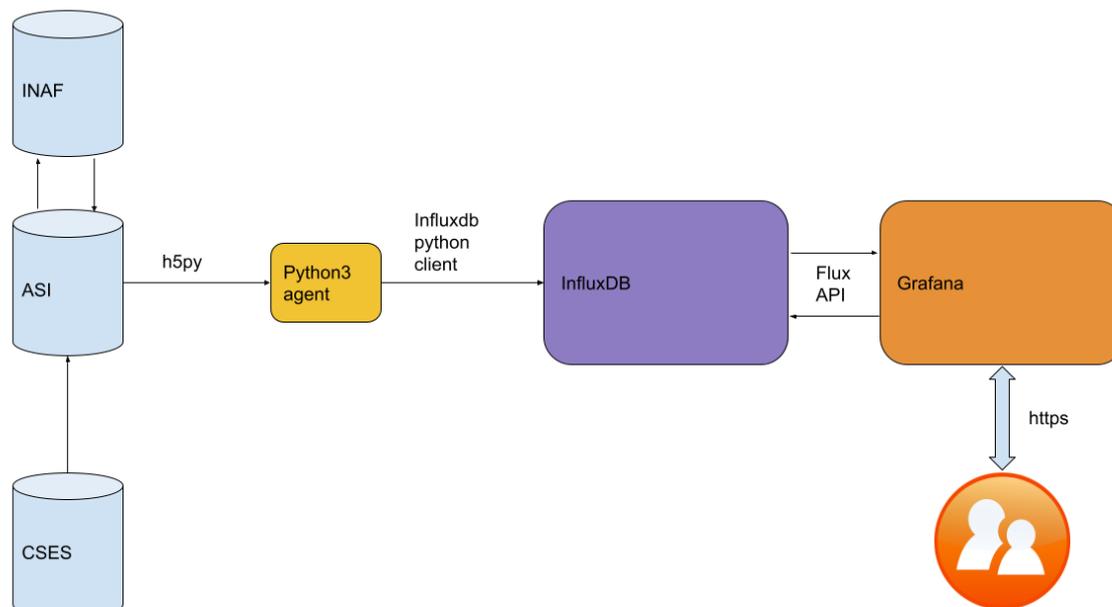


Figura 4.1: Schema semplificato della pipeline del sistema. I dati vengono prelevati dall'agent e inseriti nel database (InfluxDB). Vengono poi resi disponibili per l'elaborazione grafica (Grafana) tramite API e presentati via web all'utente.

### 4.3 Infrastruttura

Per la realizzazione dell'infrastruttura ho scelto di utilizzare una macchina virtuale su uno dei cluster VMware del Centro di Calcolo del Dipartimento di Fisica e della Sezione INFN di Roma Tor Vergata in quanto il servizio è pensato come «aperto» agli utenti per essere uno strumento collaborativo tra gruppi di ricerca. La scelta offre alta affidabilità e disponibilità della risorsa, possibilità di creare delle snapshot e di effettuare aggiornamenti, modifiche di configurazione e roll back istantanei in caso di malfunzionamento. Inoltre il sistema in questo modo può essere scalato verticalmente a caldo andando ad aumentare RAM, CPU o storage secondo le esigenze.

La scelta dell'infrastruttura, come quella del software, nasce da un'analisi dei requisiti, dei volumi di dati da trattare, delle scritture/s e delle query/s stimate, dal contesto di criticità del sistema e molti altri fattori che hanno portato ad un *capacity plan*. Riassumendo i fattori principali:

- per ora il sistema accoglie solo i dati del rivelatore HEPD;

- I dati vengono immessi nel database in modo totalmente asincrono, non parliamo di un sistema *real time*;
- non c'è necessità di avere quindi un elevato numero di write/s (che al momento è ampiamente sotto i 200k/s);
- il sistema è in un contesto *non critico*;
- il sistema è *complesso* in relazione al tipo e al numero dei dati da correlare, ma non richiede un elevato numero di query/s;
- la struttura dei *punti* ci porta ad una *cardinalità* (Sez. 4.7), e quindi un numero di serie *uniche* ampiamente sotto il milione;

anche considerando l'ipotesi che il sistema dovesse accogliere nel prossimo futuro i dati di HEPP, di altri esperimenti o Open Data della comunità scientifica, saremmo entro le capacità del sistema scalandolo verticalmente senza necessità di scalarlo anche orizzontalmente (clustering), cosa comunque realizzabile. Facendo riferimento alla Fig. 4.2 possiamo considerare il sistema attuale ampiamente dentro la seconda fascia.

vCPU or CPU	RAM	IOPS	Writes per second	Queries* per second	Unique series
2-4 cores	2-4 GB	500	< 5,000	< 5	< 100,000
4-6 cores	8-32 GB	500-1000	< 250,000	< 25	< 1,000,000
8+ cores	32+ GB	1000+	> 250,000	> 25	> 1,000,000

Figura 4.2: InfluxDB capacity

Per quanto concerne i servizi ospitati dalla VM (virtual machine), ovvero InfluxDB e Grafana, la mia scelta è ricaduta su docker [42]. I servizi girano quindi su container docker e sono *orchestrati* per mezzo di *docker compose*, un efficace sistema di gestione per applicazioni multi-container che permette di creare dipendenze e link di comunicazione tra più container, così come di gestire volumi persistenti di dati e molto altro attraverso la configurazione di un unico file in formato yaml (Listato 5.2). Questo permette di avviare e rendere operativa l'intera applicazione, costituita da più servizi (più container), con un unico comando. Ci offre quindi uno strumento unico di gestione per la nostra applicazione in grado di:

- avviare, fermare ed eseguire il rebuild dei servizi;
- visualizzare lo stato dei servizi;
- visualizzare i log dei servizi;
- effettuare operazioni di fermo sul singolo servizio.

Il collegamento di rete tra i due *servizi* Grafana e InfluxDB è gestito internamente da docker compose attraverso una bridget network alla quale sono collegati entrambi i docker (Listato 5.2)

La scelta dei container per i servizi ci offre una maggiore flessibilità in termini di portabilità. L'applicazione è auto-consistente e può essere spostata e dislocata facilmente su qualsiasi piattaforma e su server differenti senza essere legati a processi di installazione e riconfigurazione dei software utilizzati. Inoltre ogni componente dei servizi, database, dashboard, pannelli etc... è facilmente esportabile/importabile da e verso nuove installazioni di InfluxDB e Grafana stessi.

### 4.3.1 Autenticazione ed autorizzazione

Per l'autenticazione e l'autorizzazione all'utilizzo del servizio di monitoraggio Grafana permette un'ottima integrazione con il protocollo LDAP (*Lightweight Directory Access Protocol*). Tale protocollo è appositamente sviluppato per la gestione di servizi di *Directory*, ovvero di gestione di informazioni strutturate in modo gerarchico come elenchi di nomi, email, ruoli, gruppi etc... e permette l'autorizzazione di accesso a servizi sulla base di queste informazioni. Questo mi ha naturalmente portato verso tale scelta. Nella nostra struttura (Dipartimento di fisica e Sez. INFN), infatti, ogni utente è registrato su un sistema di gestione delle identità che fa uso di LDAP e questo mi ha consentito di organizzare l'accesso e l'autorizzazione creando degli attributi specifici per ogni utente che avrà necessità di utilizzare il servizio. Ho creato e configurato gli attributi attraverso un'interfaccia al sistema di gestione delle identità IAM (*Identity and Access Management*) dell'INFN (GODiVA [43]) sviluppato in parte in seno al gruppo INFN-AAI (*Authenticatio Authorization Infrastructure*) [44] del quale faccio parte da molti anni e che ho contribuito a fondare. L'interfaccia permette la creazione di *gruppi logici* che vengono riportati dentro l'architettura LDAP (anch'essa sviluppata dal gruppo INFN-AAI) come *attributi* [45]. L'applicazione, in questo caso il sistema di monitoraggio nella sua componente Grafana, verificherà la presenza di tali attributi sia allo scopo di permettere l'autenticazione, sia per l'autorizzazione alla

eventuale gestione in lettura e modifica, di dashboard e pannelli. A questo scopo ho opportunamente configurato un mapping tra l'attributo LDAP e il *Team* di Grafana (Sez. 4.8 e Fig. 4.3).

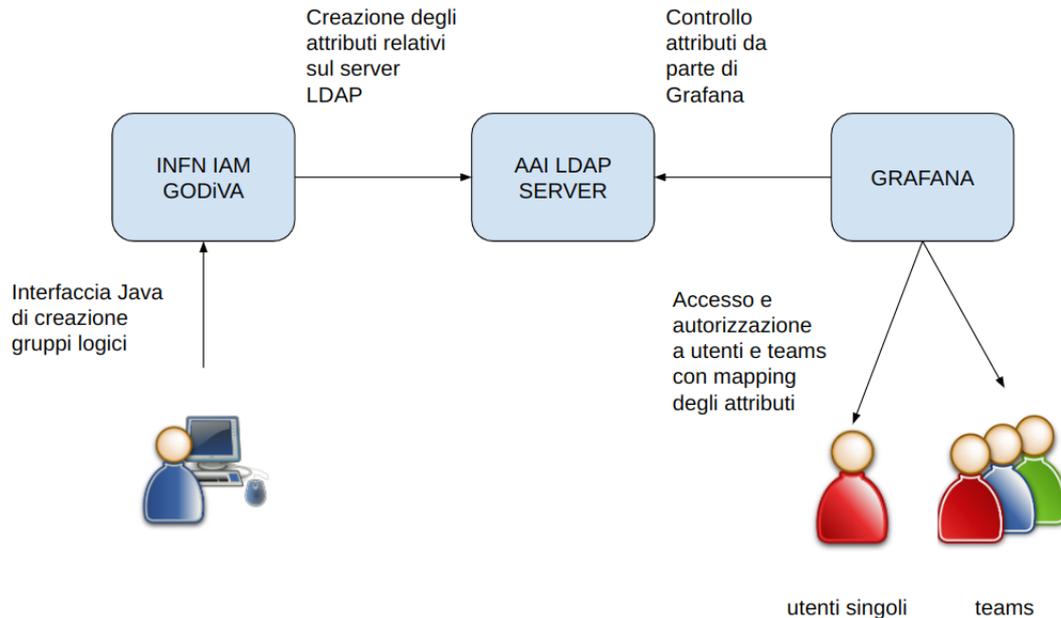


Figura 4.3: Schema del meccanismo di autenticazione ed autorizzazione al servizio

## 4.4 Scelta del software

La scelta del software per la realizzazione del sistema, come anticipato, è necessariamente legata al tipo di monitoraggio che si vuole ottenere e al tipo di dati che si dovranno trattare. Nel nostro caso vogliamo studiare la variazione nel tempo dei conteggi di elettroni e protoni provenienti dal rivelatore HEPD, questo è un tipico caso di *time series*.

Come illustrato nel primo capitolo, il satellite compie un'orbita eliosincrona con presa dati tra latitudini di  $-70^\circ$  e  $+70^\circ$ . Possiamo quindi considerare la presa dati durante un'orbita come divisa in due semi-orbite, una ascendente e una discendente. La presa dati delle semi-orbite ha una durata di 40 min circa intervallata da circa 10 min di passaggio ai poli durante i quali il rivelatore viene messo in stand-by e non è in grado di acquisire eventi. Considerando questo e il tempo di integrazione di 1s per il con-

teggio delle particelle, oggi abbiamo all'attivo circa 40mila semi-orbite e 70/80 milioni di record nel nostro DB...e la missione non è finita.

Se si vuole gestire in modo agile e dinamico una quantità di punti simile non si può utilizzare un database relazionale classico, è necessario invece orientarsi verso un TSDB (*Time Series Data Base*) che introdurrò sinteticamente nella Sez. 4.6. Ci sono molti database di questo tipo; la mia scelta è ricaduta su *InfluxDB OSS* (l'edizione Open Source di InfluxDB) per alcuni aspetti fondamentali che ritengo indispensabili nell'ambito in cui il sistema di monitoraggio opera e per la compatibilità con i requisiti sopra elencati. Solo per citarne alcuni: il completo linguaggio di query fornito e l'orientazione all'ambito scientifico con la presenza nativa della maggior parte delle funzioni di analisi di serie temporali. Ma una trattazione più approfondita su *InfluxDB* verrà affrontata nella Sez. 4.7.

Scelto il «cuore» del sistema sono passato ad esaminare i software possibili per lo strato di visualizzazione e trasformazione dei dati. Ancora una volta per rispettare i requisiti era necessario orientarsi verso un sistema flessibile, perfettamente integrato con *InfluxDB*, che permettesse una gestione di autenticazione e autorizzazione orientata ai gruppi e alla condivisione, la creazione di dashboard dinamiche, la gestione degli alert e molto altro. Su questo *Grafana* ha pochi rivali come vedremo meglio nella Sez. 4.8.

Infine i dati di tipo L3 sono racchiusi in formato *HDF5* [40] per il quale, sebbene sia un formato spesso utilizzato in ambito scientifico, non esistono plugin pronti per l'inserimento in *InfluxDB* o direttamente in *Grafana*. Questo mi ha portato alla scelta della scrittura dell'agent in Python con cui i dati vengono prelevati e formattati. Questa scelta è stata dettata dal fatto che esiste un'ottima libreria Python per gestire file in *hdf5* e che *InfluxDB* fornisce un'interfaccia completa con librerie e un client Python per l'inserimento, la rimozione e le query dei dati.

## 4.5 Python agent

Come anticipato i dati di tipo L3 sono in formato *HDF5* [40]. Utilizzato spesso in ambito scientifico non è un semplice formato. Oltre a questo in realtà è un *data model* che comprende gli elementi della struttura di organizzazione logica dei dati e di accesso ad essi tramite applicazione, più una collezione di strumenti e librerie per permettere di lavorare con questo formato. Dunque un file *HDF5* è in realtà un oggetto (o

gruppo) che può contenere in se un ampia varietà di altri dati (o anche set di dati) di qualsiasi natura (grafici, tabelle, documenti, numeri etc...).

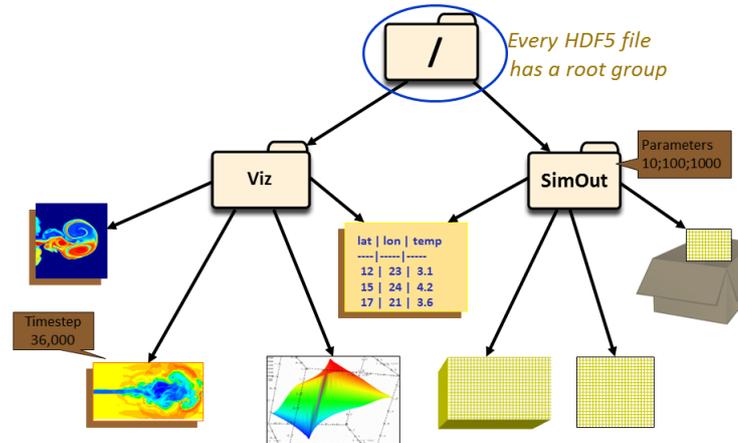


Figura 4.4: HDF5 file, organizzazione gerarchica in gruppi e dataset

Gli oggetti all'interno di un file HDF5 sono organizzati in modo molto simile a come lo sono i file e le cartelle in un filesystem UNIX. Gli elementi fondamentali sono i *Gruppi* e i *Dataset* (Fig. 4.4). I primi organizzano tutti gli oggetti in un file, contengono altri gruppi o dataset e possono anche essere collegati tramite link ad altri oggetti in file differenti. I dataset contengono e organizzano i dati grezzi e anche i metadati che li descrivono. Tipicamente un dataset può contenere un array multidimensionale di dati le cui proprietà vengono descritte appunto dai metadati.

Ho scritto il codice dell'agent utilizzando, da un lato, la libreria *h5py* [41] creata ad hoc per la gestione di dati in file HDF5, dall'altro utilizzando le librerie Python di scrittura e query verso il database InfluxDB. Come si può notare (Listato 5.1) è costituito principalmente da una funzione (*h5py.File*) che si occupa appunto dell'apertura dei file in formato hdf5. Il concetto di base è che la libreria *h5py* tratta i gruppi come *dizionari* python, ovvero degli insiemi non ordinati di elementi caratterizzati da una *chiave* e da un *valore*. I dataset vengono invece trattati come *array multidimensionali* di elementi omogenei, caratterizzati da alcuni metadati di cui almeno la *shape* (che descrive il numero e la grandezza di ogni dimensione) e il *data type*. Nel nostro caso abbiamo nel gruppo una serie di elementi (le *chiavi*) associati a dei dataset (i *valori*) costituiti ognuno da un array (di una o più dimensioni) contenenti i valori della misura effettuata. In Fig. 4.5 il gruppo «CSES\_HEP\_DDD\_0003661...» (ovvero il file corrispondente alla relativa semi-orbita) contiene diversi dataset tra i quali *HEPD\_ele\_counts* (relativo al conteggio degli elettroni) è un array unidimensionale di 2075 elementi ognuno

corrispondente all'intervallo di tempo di 1s all'interno del quale viene conteggiata la somma degli elettroni rilevati.

L'algoritmo dell'agent è riassunto schematicamente nel diagramma di flusso in Fig. 4.6

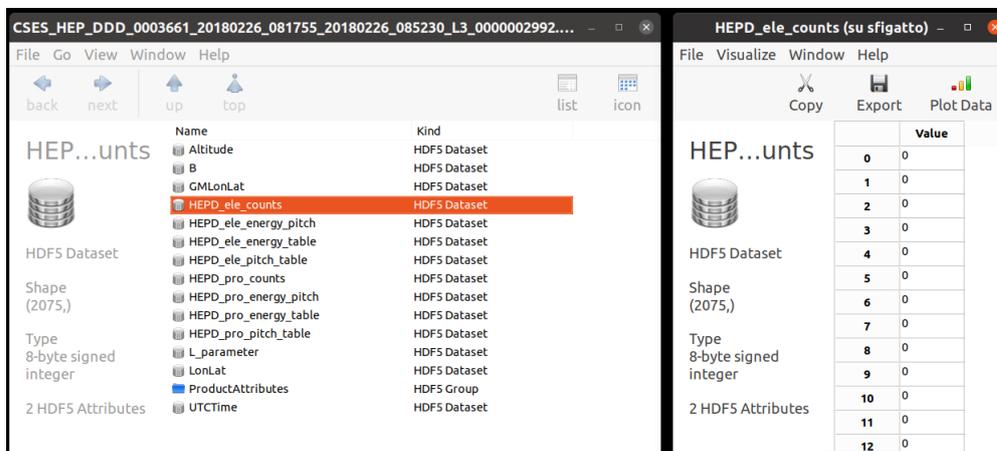


Figura 4.5: Un file *hdf5* del rivelatore HEPD. A sinistra è visualizzato un gruppo con all'interno i vari dataset. A destra i dati all'interno del dataset e i metadati che lo descrivono.

La funzione `_main` accede alla struttura di directory che contiene i file, seleziona solo i file in formato `.h5` e ne controlla l'unicità. Può capitare, infatti, che per problemi tecnici relativi al processamento dati da parte del Ground Segment Cinese, più file facciano riferimento alla stessa semi-orbita. In questo caso si seleziona il file con dimensioni maggiori. Alla fine di questo processo viene costruita una lista di file «buoni» e per ognuno di questi viene invocata la funzione `h5py` che li apre in lettura ed estrae i dataset. Sui dataset di interesse per il monitoraggio vengono di fatto costruiti i *punti* da inserire nel database. Questi vengono incapsulati in formato json (5.1 righe 86-100) e inseriti in un array. Una volta completo, l'array di punti del file in oggetto, viene scritto nel database.

A completamento dell'automatismo dell'architettura del sistema di monitoraggio, come accennato precedentemente, l'agent risiede «vicino» ai volumi di dati da estrarre e viene attivato dalla presenza di nuovi file. In questo modo è soddisfatto il requisito 2 di Sez. 4.1.

Per finire con qualche numero, questo agent si è dimostrato capace di importare l'intera mole di dati alla velocità di circa 2000 punti/s. Circa in un secondo viene processato un file. Nelle performance va tenuto conto del fatto che l'inserimento di un punto non equivale ad una singola scrittura bensì ad un numero di scritture pari

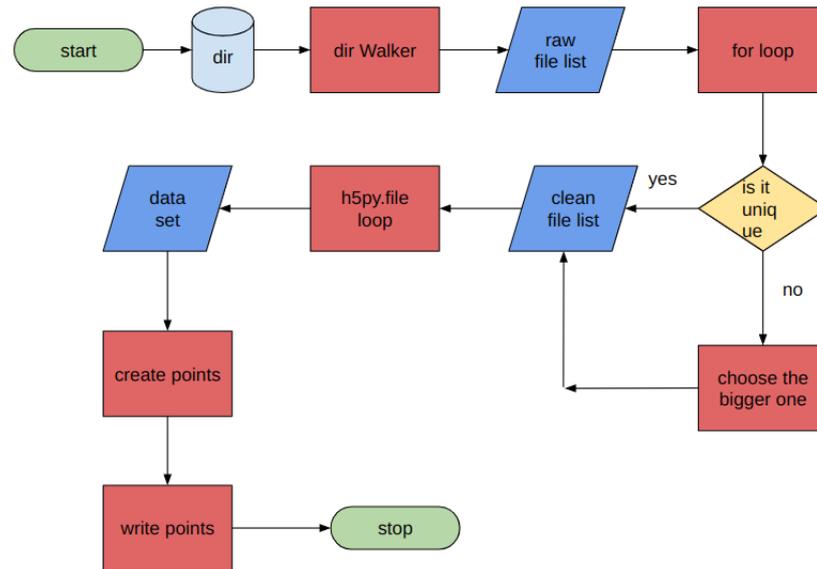


Figura 4.6: Diagramma di flusso del python agent

all'insieme degli elementi che lo costituiscono. Nel nostro caso ogni punto è costituito da un'insieme di più valori come descritto nella Sez. 4.7.1

## 4.6 TSDB

Negli ultimi anni l'aumento dei device IoT (*Internet of Things*, è riferito alla possibilità di connettere in rete sempre più oggetti fornendo servizi agli utenti) in ogni ambiente e l'informatizzazione sempre maggiore di qualsiasi processo hanno portato ad un aumento dei dati e un conseguente aumento della necessità di monitoraggio e di controllo. I time series database sono ottimizzati per gestire serie di dati caratterizzati da un time-stamp, ovvero misure che variano nel tempo. Esempi caratteristici sono metriche di performance di server (es. cpu, memoria), dati di sensori ambientali, ma anche andamenti di mercato e molti altri. Tra i requisiti che si cercano in questi casi c'è la capacità di gestire enormi volumi di dati, eseguire task tipici come la gestione del ciclo di vita dei dati stessi, l'applicazione di specifiche funzioni per l'analisi dei dati, la creazione di report anche su larghi periodi di tempo e query veloci.

L'ottimizzazione consiste nel disegnare da zero l'architettura del database in modo differente da un database relazionale classico. Alcuni concetti chiave sono:

- i dati trattati sono caratterizzati da un time-stamp;

- storage dei time-stamp: il fatto che il tempo tra una misura e l'altra cambi di pochissimo permette di gestirne l'indicizzazione con metodi differenziali (si utilizza il *delta* col time-stamp precedente, o anche il delta-del-delta, che ad intervalli regolari è nullo) che permettono un aumento delle performance ed un notevole risparmio di spazio;
- il db non è pensato per la modifica dei dati, nelle time series i dati vengono quasi sempre solo «appesi» e raramente modificati;
- non è specificamente orientato alle *relazioni* tra tabelle, non si fa un ampio uso degli indici, anzi i dati non sono indicizzati ma possono essere accompagnati da tag indicizzati;

un esempio tipico di un come viene costruito e memorizzato un *punto* è questo:

```
misura_ambiente località=Tor_Vergata,città=Roma temperatura=23,umidità=60 1650360333s
(measurement---|-----tag-set-----|-----field-set-----|-time-stamp)
```

I campi sono ad esempio organizzati sequenzialmente in colonne per blocchi di tempo, il che rende molto veloce l'aggregazione di dati su un solo campo e anche la compressione che sarà variabile in base alla precisione richiesta. Alcuni database possono lavorare con time-stamp dell'ordine dei micro o nano secondi, il che li rende utilizzabili anche in ambito scientifico.

Tutto questo porta ad un drastico aumento delle performance e dell'usabilità rispetto ad un database relazionale classico.

## 4.7 InfluxDB

Come accennato nella Sez. 4.4, *InfluxDB* è molto più che un Time Series DataBase. In realtà si tratta di una vera e propria piattaforma di gestione per le serie temporali. È disponibile in differenti versioni, *self-hosted*, *cloud* ed *enterprise* (l'ultima a pagamento) e per le nostre esigenze ho scelto la *self-hosted* (*InfluxDB OSS*) completamente gratuita e installabile «in casa».

Solo alcune delle sue caratteristiche principali sono:

- uno *storage engine* in grado di gestire volumi enormi di dati, possibilità di caricarli in batch o in streaming;

- RESTful APIs (*Application Program Interface*) e librerie client per i maggiori linguaggi di programmazione, compreso il Python utilizzato per il mio agent, per collezionare, trasformare e visualizzare i dati;
- un agent open source, *Telegraf*, basato su plugin (più di 300 al momento) per collezionare dati da qualsiasi sorgente;
- un linguaggio *funzionale* di scripting e query, *Flux*, pensato specificamente per analisi e gestione di serie temporali;
- una UI (*User Interface*) molto completa che permette la visualizzazione ma anche la trasformazione dei dati attraverso uno script editor incluso, con supporto delle funzioni di analisi, la possibilità di costruire, modificare e condividere dashboard...e molto altro.

Una «Big Picture» che illustra le funzionalità complete si trova in Fig. 4.7 Tra le caratteristiche c'è anche quella di presentare dei *connettori* verso strumenti di gestione e visualizzazione esterni come *Grafana*.

L'installazione e la prima configurazione del servizio sono state descritte nella Sez. 4.3.

### 4.7.1 Raccolta e scrittura dei dati

In InfluxDB c'è anche la possibilità di differenziare il metodo di raccolta e scrittura dei dati:

- Usando *Telegraf* e i suoi numerosi plugin per interfacciarsi a sorgenti dati di quasi ogni tipo;
- usando il metodo di *Scrape data* che permette di acquisire dati di un certo formato (*Prometheus data format*) direttamente via http;
- via *influx CLI* (*Command Line Interface*) un apposito client a riga di comando;
- oppure, come nel nostro caso, via API e librerie client (come spiegato in dettaglio nella Sez. 4.5).

Una volta collezionati, i dati vanno inseriti nel DB appositamente formattati. Il formato con cui InfluxDB scrive i *punti* è il *Line protocol*. Alcuni degli elementi base della struttura di un *punto* secondo questo protocollo sono:

Elemento	Tipo	Tipo di dato	Necessario	Descrizione	Es. nel mio sistema
Measurement	singolo	stringa	SI	il nome della misura	hepd_counts
Tag set	chiave/valore	stringa/stringa	opzionale	metadati del punto	Filename, Orbit_id
Field set	chiave/valore	stringa/Float   Integer   UInteger   String   Boolean	SI	nome e valore del punto	proto_count
Timestamp	singolo	Unix timestamp	opzionale	il valore di time-stamp per il punto	UTC file dataset

Tabella 4.1: Gli elementi fondamentali del punto secondo il Line protocol

Un punto è univocamente identificato dalla Misura, il Tag set e il time-stamp. La gestione dei punti duplicati è molto semplice: se si inserisce un punto con stessa Misura, Tag set e time-stamp ma Field set differente (ad esempio decidiamo di aggiungere un nuovo campo), il Field set diventa l'unione del vecchio con il nuovo Field set dove ogni conflitto viene risolto a favore del nuovo.

Assimilato il protocollo la scrittura dei dati va fatta inserendoli in quello che in InfluxDB si chiama *Bucket*, ovvero un contenitore virtuale dove scrivere i *punti* di una serie via via che vengono inseriti. Come già spiegato nella descrizione del python agent, ho scelto di incapsulare i punti in formato json nel quale vengono definiti tutti gli elementi descritti nella tabella 4.1. Per facilitare l'analisi, nel mio caso, ho deciso di creare due Buckets, uno (*hepd\_data*) dove è presente un Tag set (*Filename* e *Orbit*) e l'altro (*flat\_hepd\_data*) privo di Tag set. Il sistema è perfettamente funzionante utilizzando esclusivamente il primo, ma il secondo può tornare utile per query più veloci su lunghissimi intervalli di tempo.

In definitiva nei due buckets ogni singolo punto viene scritto in Line protocol così :

```
hepd_counts,Filename=CSES_HEP_DDD_0003661...,Orbit=0003661
ele_count=3,proto_count=5,L_shell=3,lat=65,lon=135 1650360333s
```

nel bucket *hepd\_data*, e come segue:

```
hepd_counts ele_count=3,proto_count=5,L_shell=3,lat=65,lon=135 1650360333s
```

nel bucket *flat\_hepd\_data*.

## 4.7.2 Il linguaggio di Query: Flux

Una volta popolati i buckets non resta che cominciare a fare query e analisi sulle time series. Fino alla versione 1.8 InfluxDB consentiva le query solo con linguaggio InfluxQL, un linguaggio SQL-like con chiamate molto simili a quelle verso un SQL database.

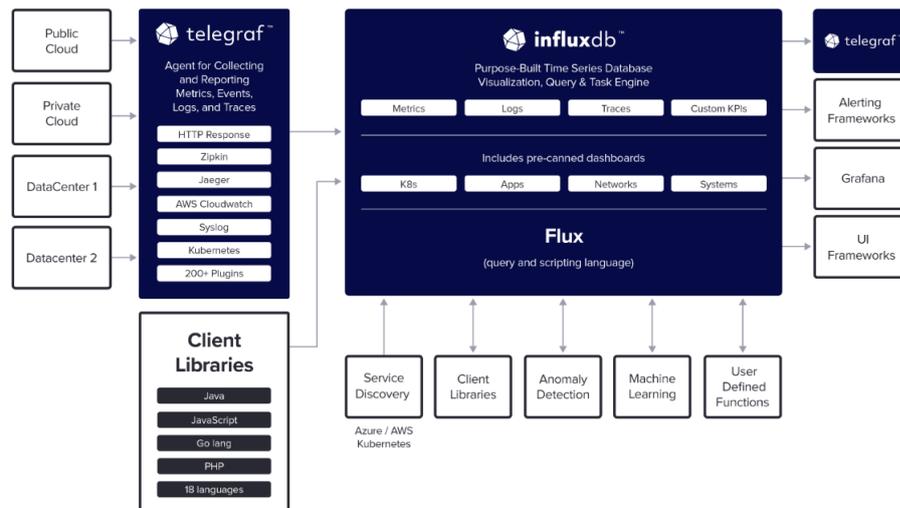


Figura 4.7: InfluxDB Big Picture

Questo linguaggio presentava però dei grossi limiti che sono stati superati con le versioni successive introducendo un nuovo linguaggio, *Flux*, completamente riscritto e che con la versione attuale, la 2.2, ha raggiunto un'ottima maturità.

Questa introduzione a Flux è molto importante perché il sistema di monitoraggio presentato in questo lavoro non è un oggetto statico. Anzi, il paradigma è quello di considerare il monitoraggio come un *processo* dinamico composto da molte fasi. La storia che i dati ci raccontano dipende da come vengono trattati e se vogliamo leggere storie interessanti dobbiamo saper *chiedere* le cose giuste.

In quest'ottica le persone che utilizzeranno questo sistema avranno accesso non solo alle dashboard e ai pannelli preconfezionati da me sulla base ovviamente di esigenze esplicite, ma anche all'interfaccia amministrativa con la quale crearne di nuovi e specifici per ogni nuovo caso d'uso si dovesse presentare, in piena autonomia.

Flux è un *linguaggio funzionale* di scripting pensato specificamente per analisi, query e trasformazione di dati. Il concetto di base è che i dati sono trattati come un flusso, dalla sorgente all'utilizzatore finale, attraverso il passaggio di uno o più stadi di filtraggio e trasformazione. Un'analogia pertinente è quella sistema di trattamento dell'acqua potabile: viene prelevata in base alla richiesta da una sorgente, attraverso vari stadi di filtraggio e purificazione e viene poi consegnata all'utente finale come *potabile*. Allo stesso modo Flux tratta i dati:

1. li preleva da una o più sorgenti;
2. li filtra in base al tempo, ai campi o ai valori delle colonne;

3. li trasforma in funzione del risultato atteso;
4. mostra il risultato.

Quindi quello di cui abbiamo bisogno nella costruzione di una query con Flux è:

- una sorgente di dati
- un intervallo di tempo
- e dei filtri da applicare ai dati

Nel caso specifico la sorgente è uno dei bucket finora creati per questo sistema, l'intervallo di tempo può essere definito come assoluto o relativo e i filtri da applicare in base a ciò che vogliamo vedere.

Nell'esempio che segue una semplice query che preleva i dati degli ultimi 4 giorni dal bucket *hepd\_data*, seleziona la misura di nome *hepd\_counts* ed esclusivamente il campo *proto\_count* relativo al rate di conteggio dei protoni. Viene poi calcolata la media e in fine mostrato il risultato.

Listato 4.1: Esempio di una semplice query

```
1 from(bucket: "hepd_data")
2   |> range(start: -4d)
3   |> filter(fn: (r) => r._measurement == "hepd_counts" and
4     r._field == "proto_count")
5   |> mean()
6   |> yield(name: "_result")
```

Si può notare come, dopo prelevati i dati, si passa da uno stadio all'altro mediante l'operatore *pipe-forward* «|>» che nella pipe inserisce appunto l'output di uno stadio nell'input dell'altro. È evidente anche la natura *funzionale* di questo linguaggio di scripting, le funzioni sono un elemento base e tutto il flusso viene gestito attraverso il passaggio da una funzione (e dai suoi parametri) all'altra.

Analizzandole nel dettaglio abbiamo le funzioni:

`from()`: che definisce la sorgente di dati e richiede necessariamente un parametro, il bucket.

- `range()`: definisce l'intervallo di tempo per la query. È necessario e non vengono accettate query senza che sia definito perché le query non confinate nel tempo possono essere molto «resource-intensive» e impattare in modo drammatico sul sistema. La funzione accetta due parametri *start* e *stop* dato che il tempo può essere definito in modo relativo (come nell'esempio) o assoluto, indicando inizio e fine.
- `filter()`: accetta un parametro «fn» che prevede che un'altra funzione (*predicate function*) valuti le righe in base ai valori della colonna. Itera quindi su ogni riga di input e la trasforma in un Flux record (una lista di oggetti chiave:valore) che viene passato come parametro «r» attraverso l'operatore «=>» alla *predicate function* la quale usando operatori di comparazione o logici restituisce *true* o *false*.
- `mean()`: esegua la media sui valori del campo *proto\_count*
- `yield()`: la funzione `yield()` ci mostra la media come risultato. In realtà questa funzione è implicitamente inserita alla fine di ogni script.

Allo scopo di interagire correttamente con i dati ed ottenere risultati significativi è fondamentale però comprendere bene il modo in cui Flux struttura i dati restituiti dal database. Gli elementi di base del modello di dati Flux sono:

**Flusso\_di\_tabelle:** è costituito da un insieme di zero o più tabelle.

**Tabella:** è un insieme di colonne divise per chiave di gruppo

**Colonna:** è un insieme di valori dello stesso tipo che contiene un valore per ogni riga

**Riga:** è un insieme di valori di colonna associati

**Chiave\_di\_gruppo:** è una lista di elementi [chiave:valore] che definisce quali colonne e quali valori specifici di colonna inserire in una tabella. Tutte le righe in una tabella hanno lo stesso valore nelle colonne relative alla chiave di gruppo. Quindi ogni tabella in un flusso di tabelle ha una chiave di gruppo univoca.

Considerando ancora la query 4.1, bisogna considerare che in input abbiamo un flusso di tabelle e non un'unica tabella. Infatti tornando a come sono formattati i punti in Line protocol (4.7.1) le chiavi di gruppo sono:

```
[_measurement: «hepd_counts», Filename: «...», Orbit: «...», _field: «ele_count»,
_field: «proto_count», _field: «L_shell», _field: «lat», _field: «lon», _start: «now»,
stop: «-4d»]
```

Gli unici valori che cambiano sono quelli del tag set, ovvero Filename e Orbit, che sono univoci per ogni file (cioè per ogni semi-orbita del satellite). Quindi avremo un numero di tabelle nel flusso in input pari al numero di file generati nei quattro giorni precedenti, in questo caso 12. Per ognuna di queste tabelle, Flux opera isolando il solo campo *proto\_count* calcolandone la media dei valori. Avremo quindi in output un gruppo di 12 tabelle caratterizzate da queste chiavi di gruppo:

```
[_measurement: «hepd_counts, Filename: «...», Orbit: «...», _field: «proto_count»,
_start: «-4d», stop: «now»]
```

e da una sola riga con il valore medio del conteggio dei protoni per ogni semi-orbita (Fig. 4.8)

	_measurement	_field	_value	_start	_stop	Filename	Orbit
1	hepd_counts	proto_count	13.379.541.108.986.600	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0233981_	233981
2	hepd_counts	proto_count	25.812.332.439.678.200	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0233990_	233990
3	hepd_counts	proto_count	19.801.354.401.805.800	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0233991_	233991
4	hepd_counts	proto_count	34.943.699.731.903.400	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234000_	234000
5	hepd_counts	proto_count	18.639.593.908.629.400	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234001_	234001
6	hepd_counts	proto_count	24.410.571.024.067.900	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234010_	234010
7	hepd_counts	proto_count	2.236.689.814.814.810	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234011_	234011
8	hepd_counts	proto_count	2.287.628.405.538.180	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234020_	234020
9	hepd_counts	proto_count	2.142.361.111.111.110	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234021_	234021
10	hepd_counts	proto_count	16.478.999.106.344.900	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234030_	234030
11	hepd_counts	proto_count	17.890.695.573.622.400	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234031_	234031
12	hepd_counts	proto_count	3.621.900.826.446.280	2022-04-21T14:15:14.826Z	2022-04-25T14:15:14.826Z	CSES_HEP_DDD_0234040_	234040

Figura 4.8: Lo stream di tabelle risultato della query

Uno dei grandi vantaggi di *Flux* rispetto al precedente linguaggio, *InfluxQL*, oltre al cambio di paradigma e all'efficienza, è quello di aver introdotto delle funzioni con le quali ottenere risultati prima impossibili o molto complicati. Solo per citarne alcune:

- join() potente funzione che rende possibile all'interno della stessa query unire ed eseguire trasformazioni su flussi di dati provenienti da più bucket, measurement e su qualsiasi colonna;
- sort() rende possibile l'ordinamento su qualsiasi colonna (*InfluxQL* poteva ordinare solo sul tempo)
- histogram() in continuo sviluppo permette di generare nativamente istogrammi di vario tipo

`covariance()` permette di calcolare semplicemente la covarianza tra due colonne (`cov()` lo fa tra due data stream)

e moltissime altre funzioni e librerie che permettono all'interno della query di manipolare e trasformare stringhe o arricchire i dati prendendoli direttamente da sorgenti esterne in molti formati differenti (CSV, PostgreSQL, MySQL, Google BigTable e molti altri).

## 4.8 Grafana

Nella sezione precedente abbiamo visto quanto sia potente e completa la piattaforma InfluxDB nella gestione delle time series. Malgrado essa metta a disposizione anche una completa interfaccia grafica, un sistema di dashboard e di alert, ho scelto di sfruttare la sua perfetta integrazione con Grafana perché ritengo che questo sia uno strumento ancora più potente e flessibile nella fase di visualizzazione dei dati.

Grafana è un software open source pensato e scritto in modo specifico per *osservazione* e *analisi* di dati e metriche in generale. In particolare permette di effettuare query, visualizzare, impostare allarmi e gestire molte tipologie di dati indipendentemente da dove essi risiedono. Supporta moltissimi storage backend (*data source*) per le serie di dati e possiede una gamma molto completa di plugin con i quali estendere le sue funzionalità non solo in termini di data source. Permette inoltre la creazione e la condivisione fra i team di dashboard e pannelli evoluti trasformando in grafici, statistiche, istogrammi, mappe e molto altro qualsiasi serie di dati. Si presta quindi perfettamente allo scopo per cui è pensato questo sistema di monitoraggio e ai requisiti elencati nella Sez. 4.1. Alcune tra le caratteristiche più importanti sono:

- la presenza di molti plugin con cui agganciare rapidamente sorgenti dati senza necessità di migrazione o inserimenti manuali, estendere le sue funzionalità;
- creazione di dashboard flessibili e dinamiche che permettono di coprire ogni esigenza nella visualizzazione dei dati;
- interfaccia centralizzata di gestione degli alert sotto ogni aspetto (rules, policy, contact point e molto altro...);
- condivisione di dashboard e pannelli tra diversi team nell'ambito della stessa organizzazione;

- generazione automatica di report pdf e schedulazione automatica di invio via email;
- granulare controllo degli accessi con definizione di progetti, ruoli e permessi.

Alcuni concetti chiave di Grafana per la comprensione della prossima sezione sono:

**Organizzazione** è l'entità di livello più alto, tipicamente usata per distinguere progetti o ambiti differenti. Ogni Organizzazione ha un suo amministratore, dashboard, pannelli e configurazioni proprie che non possono in genere essere condivise con altre Organizzazioni.

**Team** è un gruppo di utenti all'interno di un'Organizzazione. Utilizzato per assegnare più semplicemente ruoli e permessi sul Team e sui Folder anziché sui singoli utenti.

**Folder** utilizzato per insiemi di dashboard da assegnare a diversi Team organizzandone meglio ruoli e permessi.

**Pannello** è l'elemento base di visualizzazione in Grafana. Ogni pannello corrisponde ad una specifica query ed è caratterizzato dall'editor di query relativo alla data source a cui il pannello è collegato.

**Visualizzazione** è un elemento caratterizzante di un Pannello e stabilisce il tipo di «vista» e di organizzazione dei dati in output dalla relativa query (Time Sries, istogramma, grafico a barre etc...).

**Trasformazione** è una funzionalità del query editor con la quale si eseguono le trasformazioni più comuni sui dati prima che il sistema applichi la visualizzazione.

**Dashboard** è un insieme di uno o più Pannelli organizzati in una o più «righe». La varietà dei Pannelli e delle Visualizzazioni insieme alla possibilità di aggiustarne dimensioni e posizione permette la costruzione di dashboard in modo efficace ed efficiente.

**Data\_source** è lo storage back end per i dati. Ogni *data source* ha un suo specifico *query editor* che sfrutta le funzione che la particolare sorgente dati mette a disposizione.

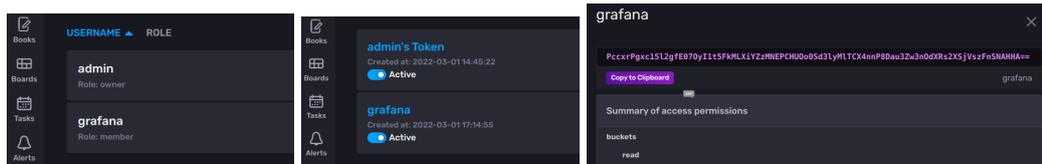


Figura 4.9: configurazione su InfluxDB degli elementi di base per la connessione di Grafana ad InfluxDB. Il token assegnato a grafana per la connessione verso i bucket è in sola lettura.

**Alert** è una notifica circa un evento di particolare interesse. Un alert è fatto di quattro componenti: *Rule*, *Contact point*, *Notification policy*, *Silences* necessari rispettivamente per la definizione della regola di «trigger», i contatti da avvisare e i metodi di avviso, le caratteristiche dell'avviso (contatto di default, tempi di avviso etc...), criteri e tempi per il «silenziamento» delle notifiche.

### 4.8.1 Configurazione

La configurazione di Grafana è la fase determinante del sistema (o processo) di monitoraggio, durante la quale mi sono occupato della scrittura delle specifiche query e della configurazione dei pannelli e delle dashboard necessarie al nostro scopo.

Prima di procedere con questo è stato necessario accedere ai dati, questo ha richiesto una configurazione di base di alcuni elementi. Su InfluxDB ho creato un utente (*grafana*) per l'accesso da Grafana sui vari *bucket*. Grazie all'integrazione nativa di InfluxDB, in Grafana ho creato una nuova *data source* di tipo InfluxBD, cosa che richiede la configurazione di alcuni parametri come: il nome della sorgente di dati, il database al quale connettersi, l'utente e il tipo di autenticazione per la connessione, il nome del bucket al quale accedere e il token di accesso creato in InfluxDB. Il token è stato creato con permessi di sola lettura (Fig. 4.10). Questo garantisce l'accesso autenticato e in sola lettura al bucket in modo da garantire l'integrità dei dati durante le operazioni di query e trasformazione effettuate su Grafana (Fig. 4.9).

La sorgente di dati di tipo InfluxDB supporta due tipi di linguaggio, InfluxQL e Flux, scegliendo il quale si ha a disposizione la specifica interfaccia di creazione delle query (*query editor*). Come ampiamente spiegato in Sez. 4.7.2 la mia scelta è ricaduta su Flux.

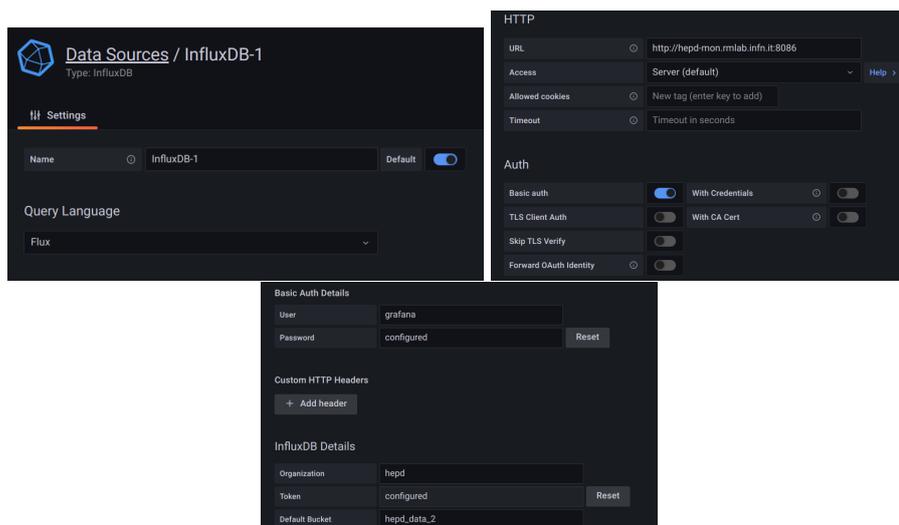


Figura 4.10: Configurazione su Grafana di una data source di tipo InfluxDB con linguaggio Flux, autenticazione tramite l'utente grafana al bucket `hepd_data` tramite il relativo token in sola lettura.

## 4.8.2 Dashboard e Panels

In questa sezione mostrerò ampiamente la costruzione del sistema realizzato finora illustrando tutti i suoi componenti, pannelli, query e dashboard. Mi soffermerò sugli aspetti più interessanti o significativi chiudendo la sezione con la configurazione degli alert. Procederò in modo sequenziale in un processo che porterà alla creazione di pannelli via via sempre più significativi scientificamente e aderenti al fenomeno che vogliamo monitorare.

Riprendendo quanto detto nella Sez. 4.7.1 riguardo la scrittura dei dati mi concentrerò, per ora, sul bucket `hepd_data` i cui dati sono stati inseriti completi di tag set. Questo bucket è utile perché, considerando che stiamo monitorando degli eventi in modo asincrono, tale sarà anche la nostra analisi. In questo scenario poter identificare le serie non solo come un unico flusso di dati ma anche come raggruppati in semi-orbite è una estensione naturale di come vengono acquisite nella realtà e torna utile nella gestione che i ricercatori usano applicare su queste serie.

**Latest datasets dashboard** Il primo approccio è stato quello di gestire al meglio le time series non real-time. In questo caso infatti la sincronizzazione dei dati avviene su base giornaliera e non real time, il che significa che i grafici di un pannello, che di default mostrano i dati dell'ultima ora, spesso risultano vuoti, non bellissimo per un sistema di monitoraggio che fa della visualizzazione il suo punto forte. Per ovviare

al problema ho lavorato in due modi: da una parte ho impostato un range di default molto più ampio (15gg nel caso specifico) per il pannello, dall'altro ho creato una variabile per la dashboard che permettesse di visualizzare e scorrere rapidamente le semi-orbite degli ultimi due giorni. Vediamo come:

Prima di tutto ho creato una nuova dashboard e aggiunto un nuovo pannello. Nella Fig. 4.11 si può notare, all'interno del query editor, una semplice query che isola il conteggio dei protoni. L'intervallo temporale della query è in questo caso stabilito attraverso il timepicker (in alto a destra nel pannello) il cui valore è letto, all'interno della query, dalle variabili `v.timeRangeStart` e `v.timeRangeStop`. Essendo i dati di questo bucket dotati di tag set, è visibile il modo in cui i conteggi risultano separati da diverse colorazioni associate al differente tag «Filename» (ovvero differenti semi-orbite).

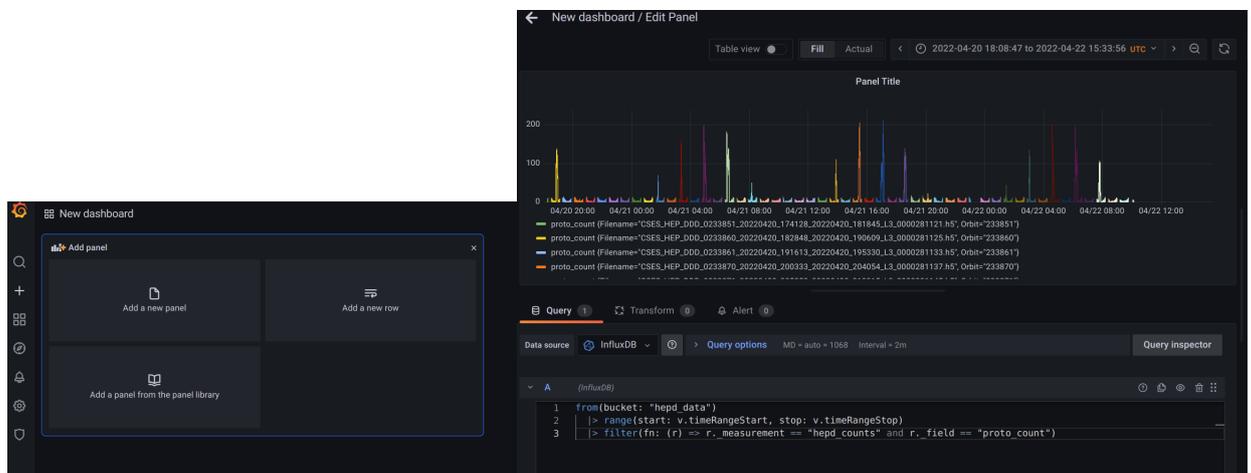


Figura 4.11: Creazione di una nuova dashboard e di un pannello (a dx) collegato alla data source InfluxDB. Nell'editor una query sul bucket `hepd_data` che filtra il solo conteggio dei protoni.

Ho creato poi una variabile per la dashboard, ovvero un «segna posto» dove inserire i valori risultanti da una specifica query.

questo mi ha dato la possibilità di integrarla nel codice del pannello e di creare un menù a tendina da cui scegliere i file relativi ad una delle ultime  $n$  semi-orbite (nel caso di Fig. 4.12 la variabile ne rende disponibili 100). Il risultato lo troviamo in Fig.4.13, nel query editor è visibile il modo in cui ho integrato la variabile: nella riga 3 della query, dove oltre a filtrare sulla «misura» e sul campo «proto\_count», metto in *and* anche il parametro `r.FileName == «${query0}»`. Nella figura il risultato della creazione automatica da parte di Grafana del menù a tendina (in alto a destra) dove è possibile appunto selezionare uno degli ultimi 100 file.

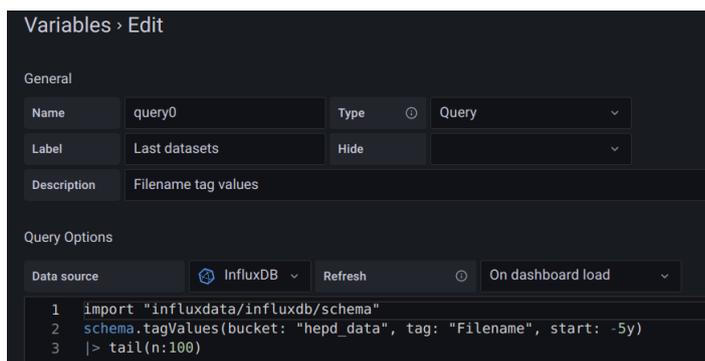


Figura 4.12: Dashboard variable che può assumere il valore degli ultimi 100 dataset risultante dalla query.

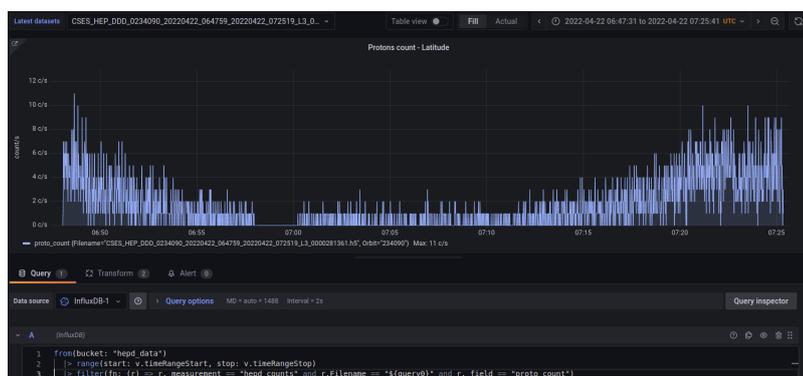


Figura 4.13: Il Pannello mostra il conteggio dei protoni una volta selezionato il file dal menu in alto a sinistra.

Questo tipo di approccio consente di concentrarsi su un solo dataset per studiarne nel dettaglio l'andamento, è quindi uno strumento utile di approfondimento nel caso risultasse evidente, da altre visualizzazioni o direttamente dagli alert, un evento anomalo o interessante.

Ho deciso quindi di creare una dashboard «Latest datasets» riproducendo la stessa query per gli altri dataset di interesse, ovvero il conteggio degli elettroni e il valore di  $L_{shell}$ . Per aumentare il livello di visibilità di alcuni valori, ad esempio il massimo conteggio/s ottenuto in un dataset, ho utilizzato poi una diversa Visualizzazione. In particolare ho costruito un pannello che potesse mettere subito in evidenza i valori massimi per tutti i dataset. Questo l'ho ottenuto scegliendo una Visualizzazione di tipo *Stat* per il pannello che ho inserito poi nella stessa dashboard (Fig. 4.14). Per completarla ho poi deciso di realizzare un pannello riassuntivo con i grafici dei tre dataset riuniti aggiungendo un nuovo asse y a destra con differenti unità di misura per il valore di  $L_{shell}$  (Fig. 4.15). La query è nel Listato 4.2



Figura 4.14: Costruzione del pannello con i valori massimi in un dataset

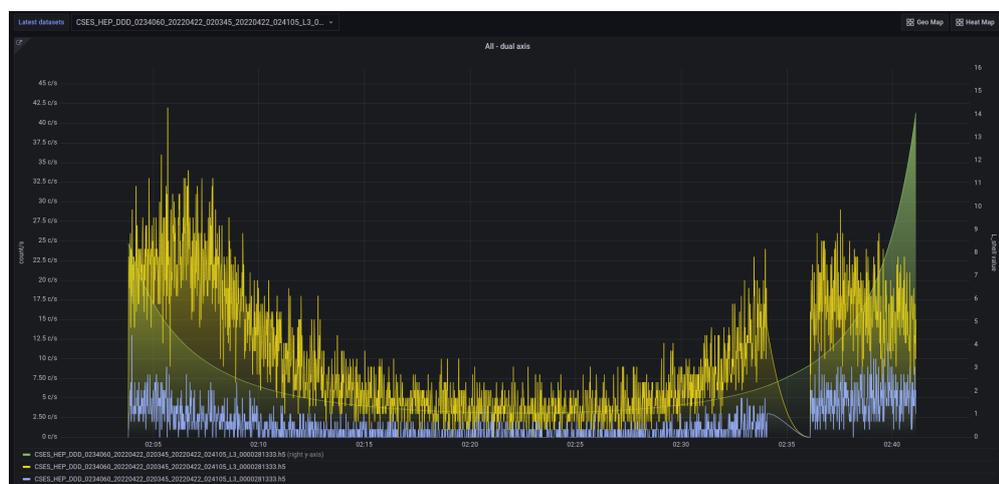


Figura 4.15: Pannello con la sovrapposizione dei grafici dei tre dataset.

Listato 4.2: Query di creazione del pannello in Fig. 4.15

```

1 from(bucket: "hepd_data")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r._measurement == "hepd_counts" and
4     r.FileName == "${query0}")
5   |> filter(fn: (r) => r._field == "ele_count" or
6     r._field == "proto_count" or r._field == "L_shell")

```

Tutti questi pannelli portano sicuramente ad una visualizzazione chiara ed efficace sull'andamento e sui conteggi, ma l'informazione che si può costruire con questi dati può essere maggiore. Soprattutto il conteggio dei protoni, per esempio, è fortemente e periodicamente legato al passaggio del satellite nella zona della SAA (*South Atlantic Anomaly*). Questa cosa è evidente anche in Fig. 4.11 dove si vede chiaramente

l'andamento periodico del conteggio dei protoni. La cosa può essere monitorata aggiungendo dei dati. Nella struttura dei punti inseriti in InfluxDB ci sono infatti i dati relativi alle coordinate geografiche del satellite. Sfruttando un'altra funzionalità importante di Grafana, le *Trasformazioni*, ho realizzato una sovrapposizione dei grafici di longitudine e latitudine su quelli di conteggio. Questo aiuta immediatamente a capire se un picco di protoni è in zona SAA (come in questo caso) oppure no.

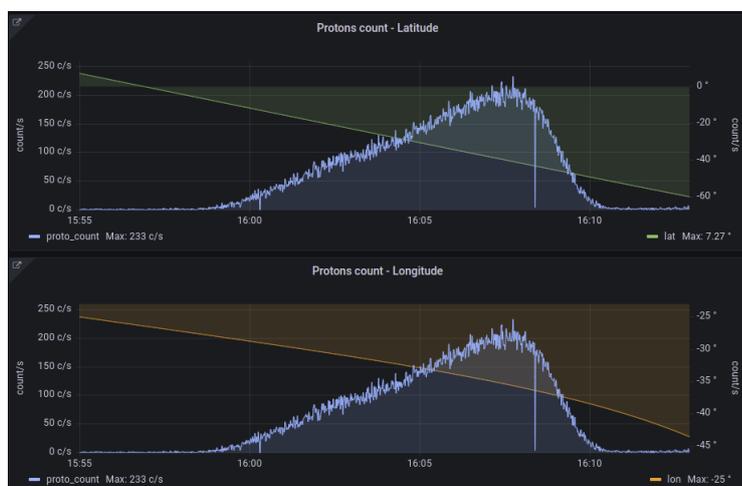


Figura 4.16: Due pannelli con la sovrapposizione dei grafici di latitudine (sopra) e longitudine (sotto) su quelli di conteggio.

Finora ho quindi ho creato pannelli costruiti con l'idea di esaminare serie relative ad una semi-orbita effettuata nell'arco degli ultimi tre giorni. Ho quindi inserito questi Panels in una dashboard denominata «Latest datasets» (Fig. 4.17). Tutto questo però non ci consente «a prima vista» di capire se è presente nelle serie un'anomalia. Dovremmo infatti scorrere via via tutti i file agendo sul menù e controllare i picchi nei vari grafici.



Figura 4.17: «Latest datasets» dashboard dove sono presenti tutti i pannelli descritti sopra.

**Top ten dashboard** Per rendere più veloce l'individuazione di eventuali anomalie ho deciso di creare un'altra dashboard che ho chiamato *Top Ten*. Per la creazione dei pannelli ho utilizzato una apposita funzione del linguaggio Flux: `highestMax()` (Listato 4.3). Questa funzione seleziona il record con valore più alto in ogni tabella dello stream di tabelle in input. Ricordo infatti, come spigato in Sez. 4.7.2, che anche selezionando il solo campo `proto_count` in input abbiamo un flusso di tabelle separate perché abbiamo differenti `Filename`. La funzione restituisce in output un'unica tabella contenente i primi  $n$  record ordinata sul valore di una colonna a scelta e nel range di tempo selezionato dal timepicker. Con una query come quella del Listato 4.4 ho ottenuto la visualizzazione di Fig. 4.18.

Anche in questo caso però l'informazione che se ne trae è influenzata dal fatto che non possiamo distinguere quali semi-orbite si riferiscano a zone che sono fuori dalla SAA. Per ovviare a questo ho riscritto la query escludendo la SAA sfruttando le possibilità funzionali del linguaggio Flux (Listato 4.5). Questa restituisce la visualiz-

Listato 4.3: *highestMax()*

```

1 highestMax(
2     n:10,
3     column: "_value",
4     groupColumns: [],
5 )

```

Listato 4.4: Query per la creazione di una top ten dei file che presentano il più alto conteggio/s di protoni.

```

1 from(bucket: "hepd_data")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r["_measurement"] == "hepd_counts")
4   |> filter(fn: (r) => r["_field"] == "proto_count")
5   |> highestMax(n: 10, groupColumns: ["Filename"])

```

zazione in Fig. 4.19. Confrontando le due figure risulta evidente come la prima query restituisca certamente un risultato corretto, ovvero i primi 10 file che presentano il più alto rate di conteggio/s dei protoni, ma forse privo di informazione se siamo in cerca ad esempio di un *Flare solare*. In questo caso infatti ci aspetteremmo un elevato numero di conteggi ad alte latitudini e, dalla seconda query, risulta evidente che nel range selezionato, fuori dalla zona SAA, i conteggi sono molto bassi. Questo è un tipico esempio di come bisogna chiedere ai dati le cose «giuste» se vogliamo farci raccontare una storia interessante.

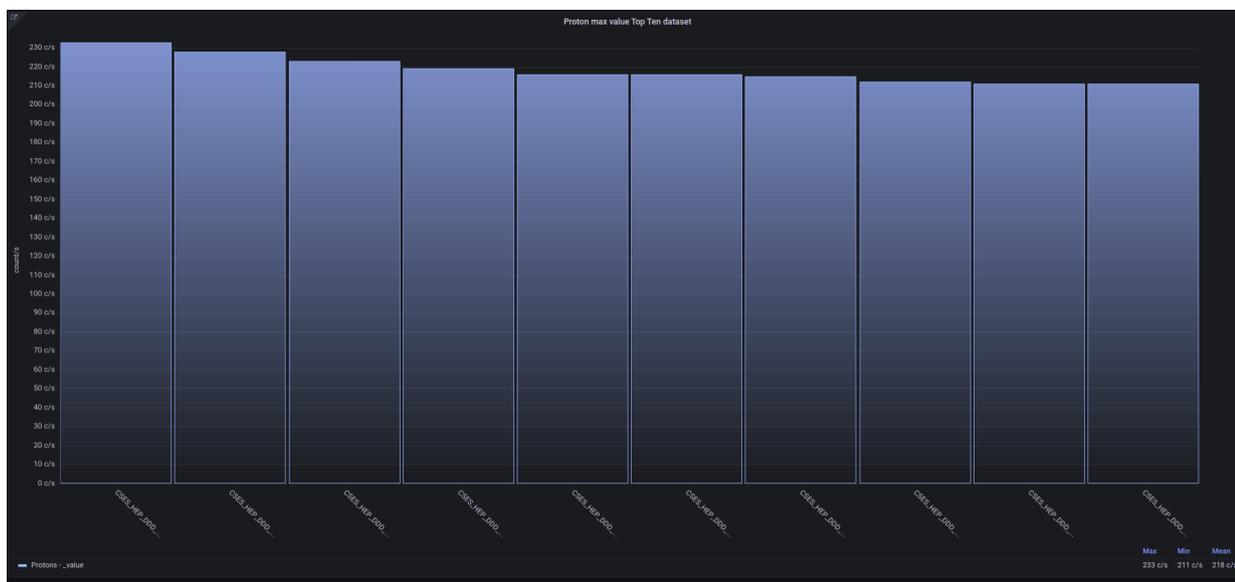


Figura 4.18: Top ten protoni

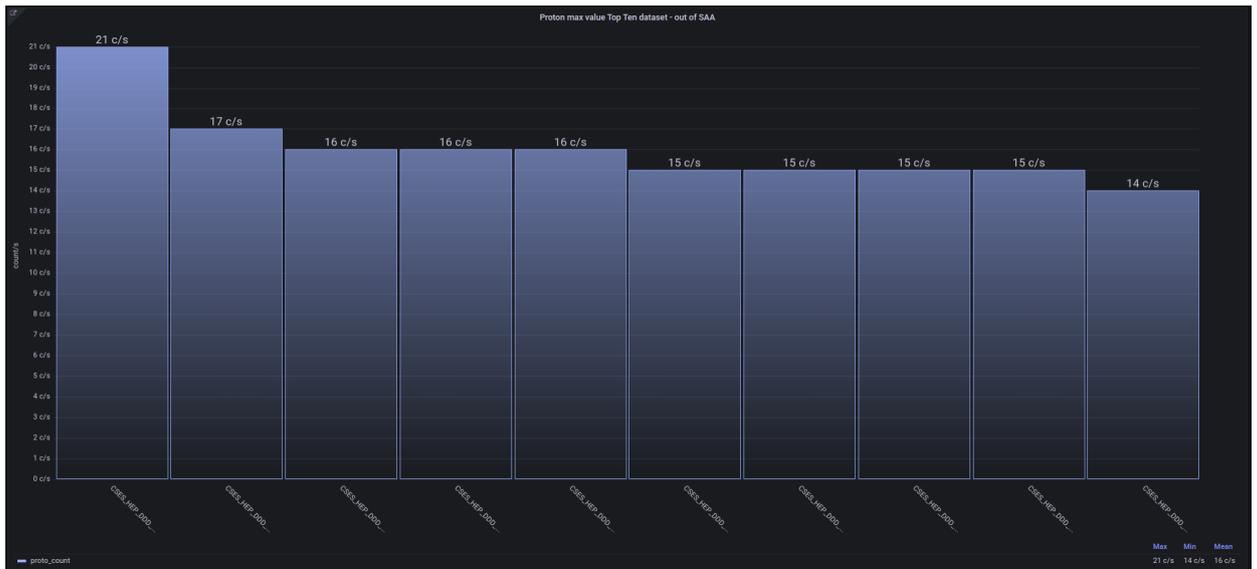


Figura 4.19: Top ten protoni fuori dalla SAA

Nella dashboard delle top ten ho quindi deciso di inserire, non solo le top ten assolute ma anche quelle fuori dalla zona SAA. Inoltre, sempre per aggiungere informazioni ho deciso di inserire le relative top ten dei file calcolate non sul conteggio massimo in un 1s, ma sul valore medio dei conteggi/s. Per completezza ho inserito anche gli stessi pannelli per i valori di  $L_{shell}$  (Fig. 4.20).



Figura 4.20: Top ten dashboard

Listato 4.5: Top ten fuori dalla zona SAA

```

1 from(bucket: "hepd_data")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r["_measurement"] == "hepd_counts")
4   |> filter(fn: (r) => r["_field"] == "proto_count" or r._field == "lat")
5   |> pivot(rowKey:["_time"], columnKey: ["_field"], valueColumn: "_value")
6   |> filter(
7     fn: (r) => if r.lat >= -55 and r.lat <= 0 and r.lon >= -93 and r.lon <= 40 then
8       r._field == "proto_count" and r._field == "Filename"
9     else
10      r._measurement != ""
11   )
12 |> highestMax(n: 10, column: "proto_count", groupColumns: ["Filename"])

```

Un'altra importante funzionalità che ho sfruttato in Grafana è quella dei *link*. Proprio nel contesto della dashboard appena descritta, ad esempio, sarebbe interessante, una volta individuato un file con alto conteggio di protoni poter passare subito ad una visualizzazione più dettagliata della specifica semi-orbita. Ho creato quindi un *data link*. Grafana supporta tre tipi di link: *dashboard link*, *panel link* e *data link*. Quest'ultimo è utilizzato nel caso si voglia scendere in un contesto più «granulare» rispetto alla visualizzazione corrente. In particolare per un grafico come quello della top ten dei

protoni ho aggiunto una data link che per ogni specifico file rimanda alla dashboard «Latest datasets». Ho configurato il link aggiungendo al suo interno delle variabili (Grafana global variables) in modo tale da essere rimandato alla dashboard contestualizzata in modo «dettagliato» trasportando, appunto, insieme al link anche il «Filename» ed il «time range». Cliccando quindi su uno dei data set del grafico ho creato la possibilità di essere rimandati allo specifico file e time range nella visualizzazione dettagliata della dashboard Latest datasets (Fig. 4.21).

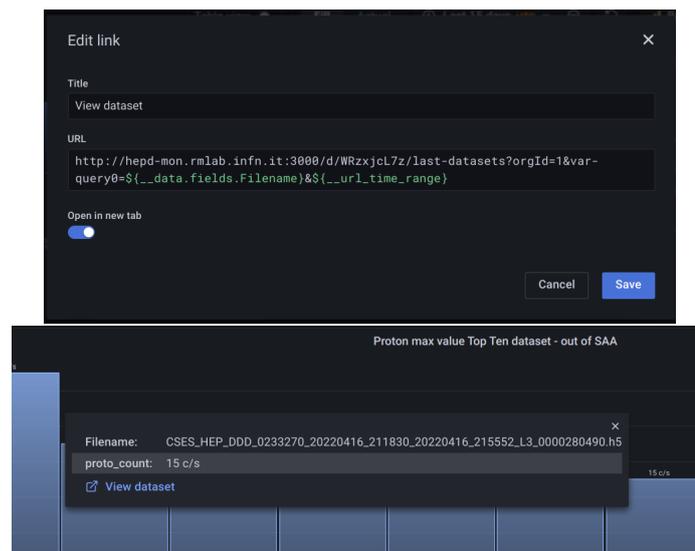


Figura 4.21: Creazione del data link (sopra), cliccando su uno specifico data set sul grafico compare la visualizzazione del link (sotto).

**Dynamic aggregate data** È interessante poter accedere anche ad una visualizzazione su un range di tempo più ampio di quello del singolo file proprio per avere un'idea rapida della serie storica di dati. Ampliando troppo l'intervallo di tempo si rischia però di inserire nella visualizzazione un numero eccessivo di punti fino a che sia il processamento della query, lato server, sia il lavoro fatto dal browser diventano eccessivi e si va incontro a dei crash di uno dei sistemi. Nel nostro caso specifico un intervallo di tempo di 15gg può contenere circa 1 milione di punti. Per evitare questa cosa nella gestione delle time series si usano tipicamente delle funzioni di *aggregazione* allo scopo di effettuare una riduzione dei dati prima della fase di visualizzazione. Anche in questo InfluxDB si è rivelato molto potente con molte *aggregate funtions* disponibili nella libreria standard. Una delle più utilizzate è certamente la funzione:

```
mean(column: "_value")
```

che calcola la media dei valori in una colonna.

Ma la funzione di aggregazione è solo un pezzetto del processo di riduzione dei dati. Ricordo infatti che in input abbiamo un flusso di tabelle. Il processo di riduzione deve, per ogni tabella cioè ogni chiave di gruppo univoca (Sez 4.7.2), suddividere la tabella in sotto tabelle col numero di righe contenute nell'intervallo temporale su cui vogliamo calcolare la media dei valori (quello che chiamerò «window»), calcolare la media dei valori su una sotto tabella e quindi restituire un flusso di tabelle contenenti una sola riga (quella del valor medio) per poi finalmente ricomporre le sotto tabelle (quello che si chiama «unwindow») in un'unica tabella ricostruendo la colonna con i valori di tempo tipicamente impostati a quello coincidente con la fine dell'intervallo della finestra temporale. tutto questo come detto viene effettuato per tutte le tabelle del flusso in input. È possibile effettuare questo processo per step applicando le singole funzioni, oppure sfruttare una funzione che si occupa dell'intero processo, la funzione `aggregateWindow()`:

```
aggregateWindow(  
  every: 1m,  
  period: 1m,  
  fn: mean,  
  column: "_value",  
  timeSrc: "_stop",  
  timeDst: "_time",  
  location: "UTC",  
  createEmpty: true,  
)
```

Come si vede la funzione accetta diversi parametri tra i quali `_start` e `_stop` assolutamente necessari così come `every` e `fn`. I primi due sono l'intervallo temporale scelto per la visualizzazione, `every` è la larghezza della finestra temporale nella quale eseguire la media e `fn` è la funzione di aggregazione da applicare. Il problema è che la larghezza della finestra su cui calcolare la media, a meno di non considerare una visualizzazione statica (ad esempio con time range di un giorno), sarebbe comodo poterla variare al variare appunto dell'intervallo di tempo scelto per la visualizzazione. Scegliendo di vedere un grafico con un time range relativo agli ultimi 15 giorni si avrà bisogno di impostare un valore maggiore per `every` rispetto ad una visualizzazione di un solo giorno. Questo l'ho ottenuto impostando i parametri `_start`, `_stop` e `every` dinamicamente attraverso delle variabili. Con una query come questa:

Listato 4.6: Query di creazione di un pannello con la media dei conteggi degli elettroni calcolata all'interno di una finestra di tempo dinamica.

```

1 from(bucket: "hepd_data")
2 |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3 |> filter(fn: (r) => r["_measurement"] == "hepd_counts")
4 |> filter(fn: (r) => r["_field"] == "ele_count")
5 |> aggregateWindow(every: v.windowPeriod, fn: mean, createEmpty: false)

```

la larghezza della finestra temporale su cui calcolare la media varia dinamicamente in base la time range scelto sul timepicker. In Fig.4.22 si può vedere il risultato della query che mostra il conteggio degli elettroni per i dataset degli ultimi 15 giorni. Applicando uno zoom-in con il mouse direttamente sul grafico la finestra temporale si riduce direttamente adattandosi al nuovo intervallo di tempo scelto fino ad arrivare ad un livello di zoom per cui la finestra si riduce sotto 1s e non viene più applicata la media, si ritorna cioè ad una visualizzazione completa dei dati.



Figura 4.22: Serie storica del conteggio degli elettroni per i data set relativi agli ultimi 15 giorni con riduzione dei dati attraverso le funzioni `aggregateWindow()` e `mean()`.

La visualizzazione di periodi lunghi e la conseguente riduzione dei dati introducono ovviamente dei possibili errori di valutazione nell'osservare la serie. Per mitigare questo ho introdotto a fianco del precedente un altro pannello generato con la stessa query ma applicando la funzione di aggregazione `stddev()` al posto della `mean()`, che calcola la deviazione standard. In questo modo è possibile farsi un'idea di dove i va-

lori visualizzati nel primo pannello possano essere meno accurati e verificare tramite uno semplice zoom.

Applicando tutto ai data set di interesse ho creato la dashboard «Dynamic aggregate data» (Fig. 4.23).

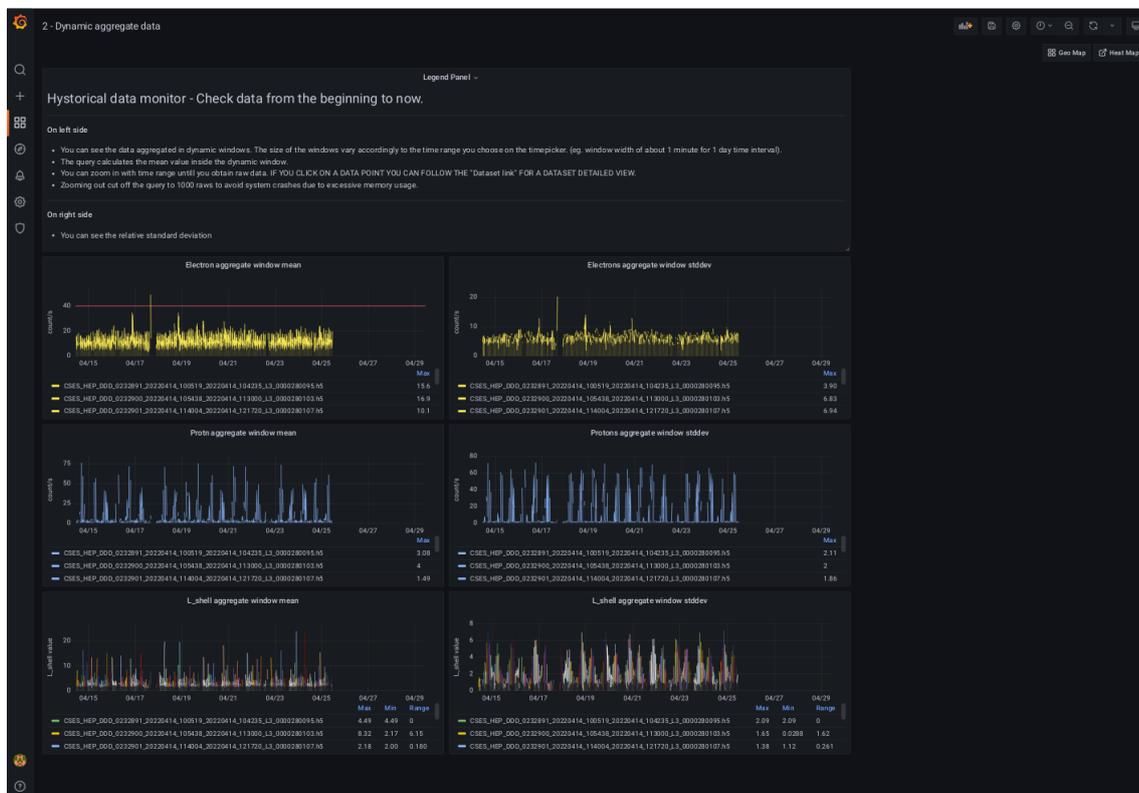


Figura 4.23: Dynamic aggregate data dashboard.

**Flat dashboard** L'applicazione di queste query per periodi molto lunghi può essere resa ancora più veloce operando su un database ad hoc dove i dati vengono salvati senza tag set. Questo riduce il numero di tabelle nel flusso in input e semplifica la query. Ho voluto quindi realizzare una dashboard dedicata a query veloci dove non ci sia necessità di risalire al filename o all'id dell'orbita, ma si voglia semplicemente analizzare l'andamento dei dati nel tempo delle varie grandezze. Anche in questo caso, oltre ai semplici pannelli con l'andamento di protoni ed elettroni ho deciso di creare quelli relativi anche ai valori di latitudine come in Fig. 4.24 dalla quale risulta (in questo specifico caso) subito chiaro che i picchi di conteggio dei protoni non sono coincidenti con quelli di latitudine. Ho inoltre inserito un pannello con la sovrapposizione dei grafici di protoni ed elettroni per provare visivamente la qualità della

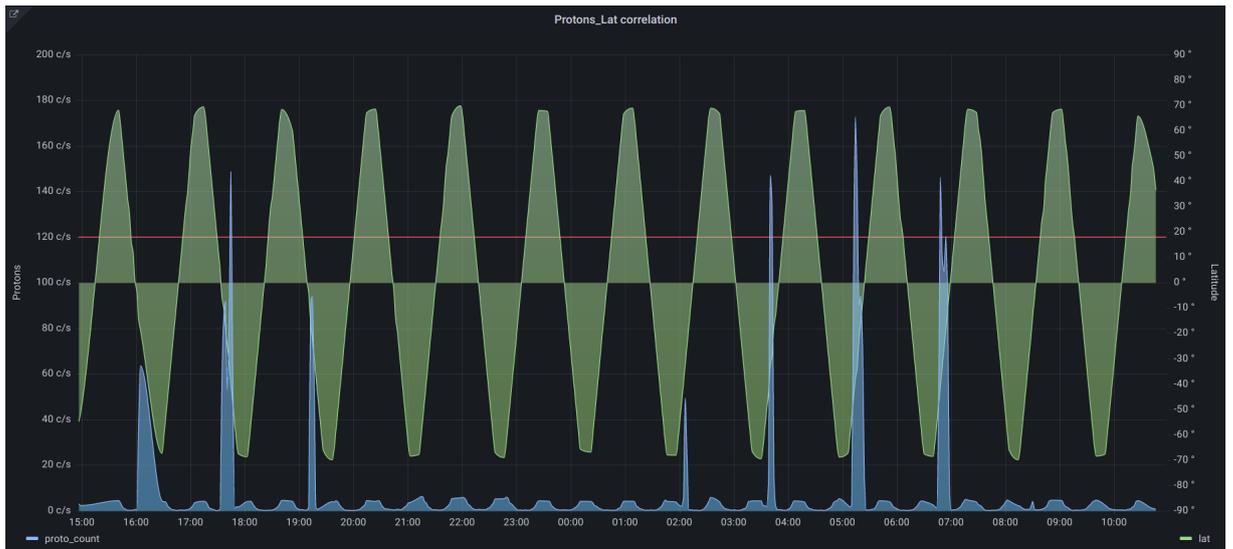


Figura 4.24: Grafico del conteggio dei protoni correlato a quello del valore di latitudine.

separazione, che avviene con il processamento da L2 ad L3, delle due particelle (Fig. 4.25). Ma i pannelli più significativi di questa dashboard sono quelli che presentano il rate di conteggio di protoni ed elettroni selezionati fuori dalla SAA. Come si può vedere dalla Fig. 4.26 questo tipo di visualizzazione (sotto) mette immediatamente in evidenza fenomeni anche piccoli molto probabilmente legati a transienti solari che andrebbero persi in una visualizzazione più generale (sopra).

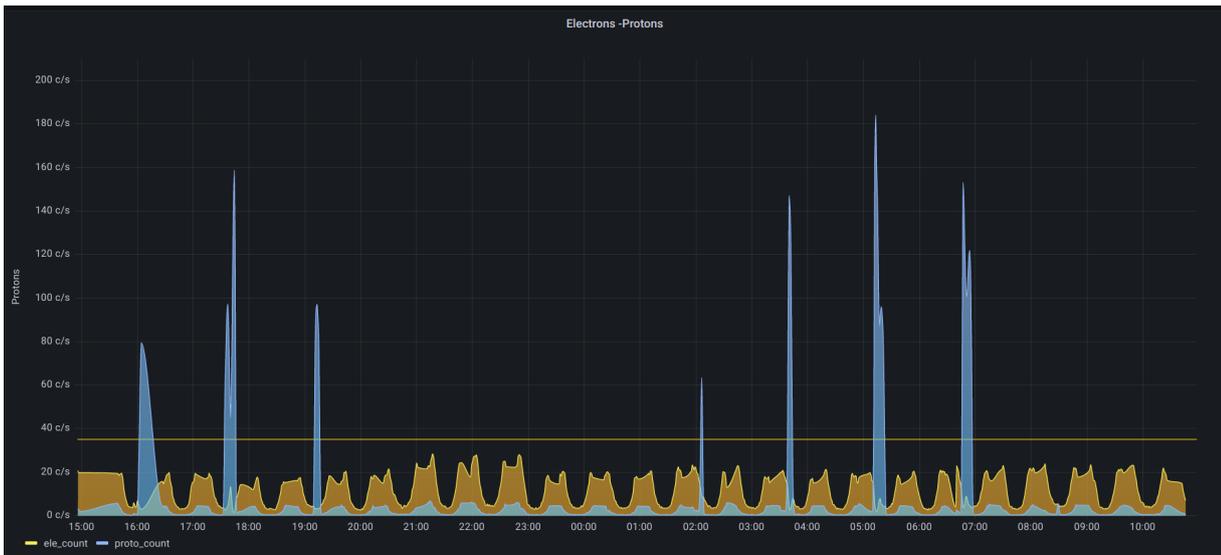


Figura 4.25: Grafico sovrapposto Protoni-Elettroni



Figura 4.26: Pannelli del rate di conteggio dei protoni globale (sopra) e escludendo la SAA (sotto)



Figura 4.27: Flat dashboard. In questo caso i pannelli sono collegati ad un bucket con punti privi di tag set.

La Flat dashboard è stata quindi realizzata collegando i pannelli ad un bucket differente di InfluxDB creato ad hoc, dove i dati vengono inseriti da un agent leggermente modificato nella parte riguardante il json di formattazione dei punti Flux. Confrontando il listato qui sotto con le righe 86-100 del Listato 5.1, si può vedere come i punti vengano formattati in questo caso con il solo field set e senza alcuna tag.

```
point_body = {
  "measurement": "hepd_counts",
  "time": ft,
  "fields": {
    "ele_count": elecounts_data[i],
    "proto_count": procounts_data[i],
    "L_shell": l_data[i],
    "lon": lonlat_data[i][0],
    "lat": lonlat_data[i][1]
  }
}
```

**Geo Map e Heat Map** Gli ultimi esempi che voglio riportare riguardano due dashboard create sfruttando una delle tante *Visualizzazioni* presenti per per la configurazione dei Pannelli, si tratta della visualizzazione *Geomap*. Questa consente di visualizzare dei dati «geospaziali» riportandoli sullo sfondo di una mappa terrestre. La mappa è configurabile scegliendola tra diverse tipologie (ArcGIS Map Server, Open Street Map, CARTO reference map etc...) e supporta data layer multipli. Ciò consente volendo di sovrapporre sulla mappa strati di dati differenti e gli strati possono essere di tre tipi a seconda di come il dato viene riportato sulla mappa:

**Marker** contrassegna il dato con un marchio (es. un cerchio) configurabile in dimensione e colore;

**Heatmap** mostra una heatmap dei dati calcolata sul valore di un campo specifico;

**GeoJSON** riporta dati statici formattati in json.

Per la localizzazione spaziale la visualizzazione si aspetta una sorgente di dati geografici che io ho inserito nei punti registrati nel DB. Quando Geomap trova dei nomi di colonna «lat» o «lon» è in grado automaticamente di riconoscerli ed utilizzarli come tali. Questo può creare un problema perché nel flusso di tabelle in input non esistono

«colonne» con tali nomi che invece sono i valori della colonna `_field`. Ho risolto questo problema utilizzando un'altra funzione di Flux, la funzione `pivot()`:

```
pivot(rowKey:["_time"], columnKey: ["_field"], valueColumn: "_value")
```

Come indicato dal nome questa funzione è in grado di collezionare i valori memorizzati verticalmente e ruotarli riorganizzandoli orizzontalmente in raggruppamenti logici. Nel Listato 4.7 è riportata la query con cui ho costruito il pannello in Fig. 4.30. Dopo aver selezionato solo i campi `lon`, `lat` ed `ele_count` la riga 7 ruota i valori creando tabelle con questi nomi di colonna.

Nel mio caso specifico ho deciso di realizzare due dashboard separate, una per le heat map l'altra che indicasse il *ground track* del satellite per uno specifico dataset o porzione di questo. Ho inoltre utilizzato i *dashboard link* per poterle richiamare da qualsiasi altra dashboard o pannello durante la visualizzazione dei dati. Nella Fig. 4.17 ad esempio, durante l'osservazione di un dataset è possibile dai link in alto a destra nella dashboard richiamare la Heat Map o la Geo Map già contestualizzate sullo quello specifico data set (Fig. 4.29). I link della dashboard, come i data link (Fig. 4.21), infatti permettono di trasportare verso la nuova visualizzazione alcune variabili come il time range (Fig. 4.28).

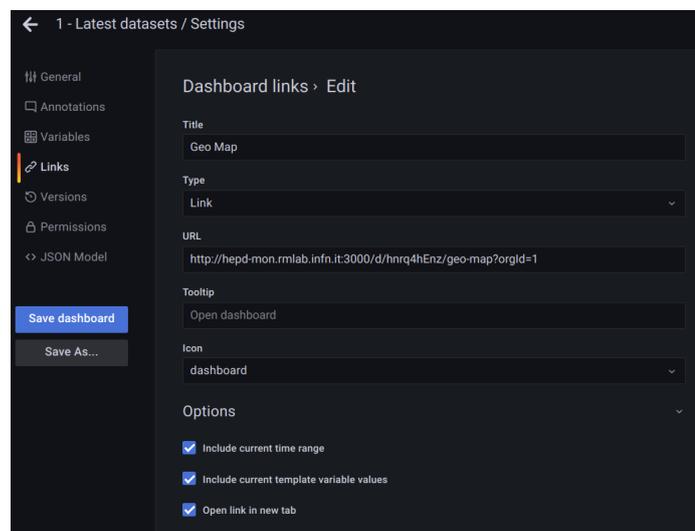


Figura 4.28: Configurazione di un dashboard link.

Così come è possibile visualizzare una Heat Map di un insieme di semi-orbite per avere subito una visione d'impatto della distribuzione geografica dei conteggi degli elettroni (Fig. 4.30).



Figura 4.29: Geo Map relativa ad uno specifico dataset richiamata direttamente dalla dashborad «Latest datasets». In rosso il ground track del satellite. Spostandosi con il mouse sul tracciato è possibile vedere il dettaglio dei punti.

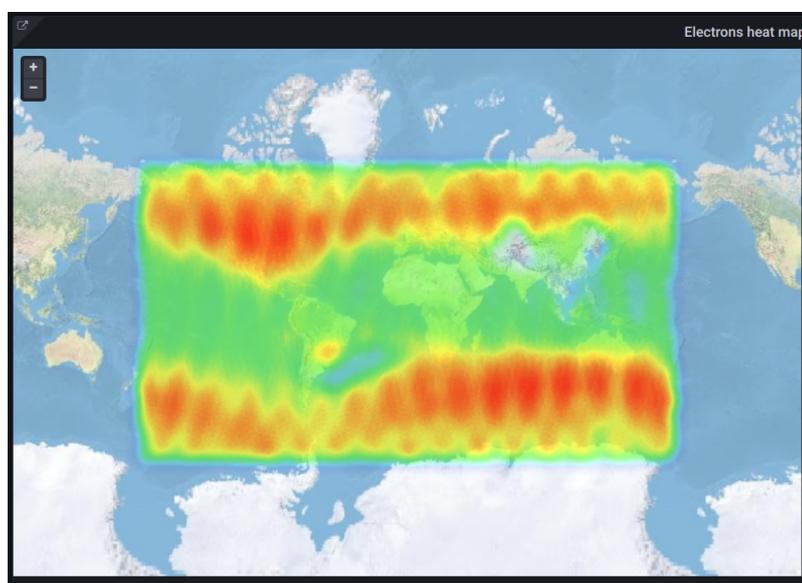


Figura 4.30: Heat Map della distribuzione geografica del conteggio degli elettroni di più orbite sommate.

Listato 4.7: Query di creazione del pannello di Fig. 4.30

```
1 from(bucket: "hepd_data")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r["_measurement"] == "hepd_counts")
4   |> filter(fn: (r) => r["_field"] == "lat" or r["_field"] == "lon" or
5     r._field == "ele_count")
6   |> aggregateWindow(every: ${window_size}, fn: mean, createEmpty: false)
7   |> pivot(rowKey:["_time"], columnKey: ["_field"], valueColumn: "_value")
```

**Home dashboard** Il sistema permette ovviamente di costruire una dashboard partendo da pannelli già creati e presenti in altre. Ho costruito quindi una dashboard «Home» impostandola come «default dashboard» ovvero quella che verrà visualizzata quando ci si connette via web al sistema di monitoraggio. È utile inserire nella Home i pannelli più significativi in base all'interesse verso le serie da monitorare. Al momento la Home dashboard (Fig. 4.31) presenta a partire dall'alto: i rate di conteggio di protoni ed elettroni e le Top Ten (elettroni e protoni) delle semi-orbite che mostrano i rate di conteggio più alti fuori dalla SAA, la lista delle dashboard preferite ed un pannello di ricerca globale (su tutto lo storico) per tag (Filename o Orbit), le Heat map del rate di conteggio degli elettroni e dei protoni fuori dalla zona SAA e le Top Ten globali (inclusa la SAA) di protoni ed elettroni. Il time range impostato di default è di 15 giorni. Tutto è ovviamente modificabile in modo molto semplice in qualsiasi momento e per qualsiasi componente o variabile.

### 4.8.3 Sistema di Alerting

Come già detto un sistema di monitoraggio di una grandezza difficilmente può essere pensato senza il concetto di una soglia o di un valore di confronto e senza l'implementazione di un meccanismo di alerting. Su Grafana ci sono diversi modi per segnalare anomalie. Uno è semplicemente «visivo». Nel singolo pannello si possono configurare soglie su qualsiasi campo e visualizzazioni di diverso tipo. Nella visualizzazione riportata in Fig. 4.32 ho configurato una soglia di 120 c/s per il rate conteggio dei protoni impostata di colore rosso e mostrata come linea semplice.

Passando alla configurazione degli alert veri e propri, l'ultima versione di Grafana ha migliorato la gestione degli avvisi creando una vera interfaccia centralizzata per la gestione di ogni componente del un singolo *alert* (Fig. 4.33).

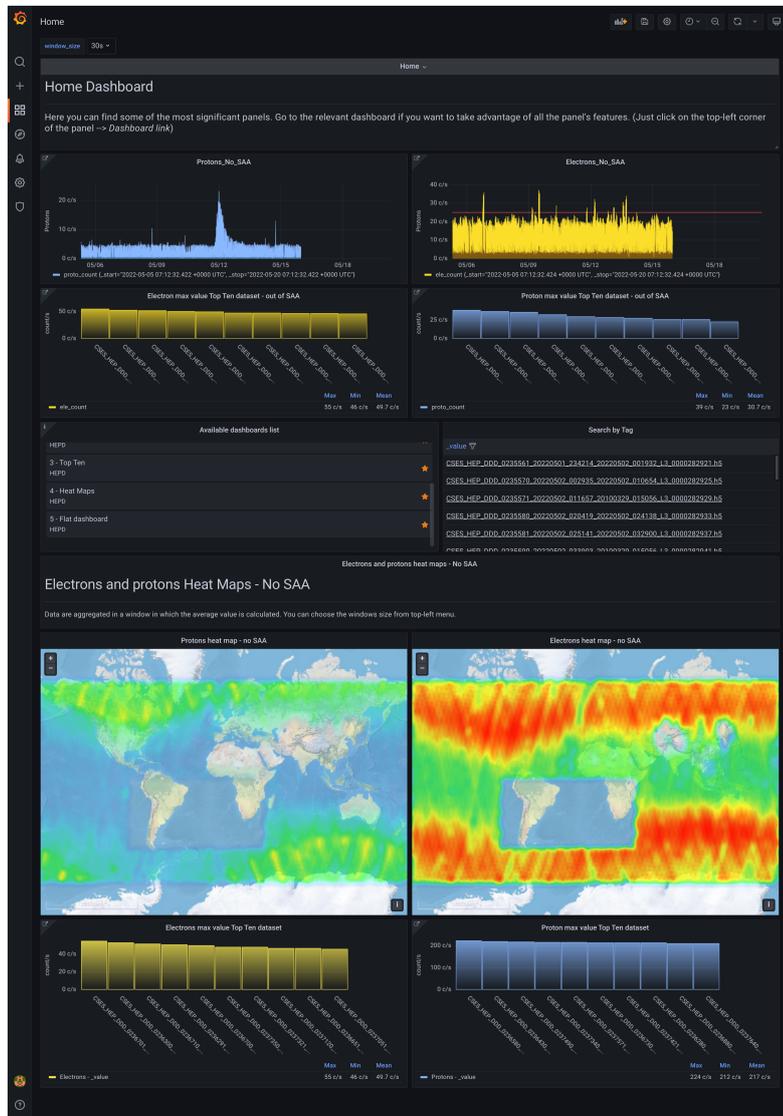


Figura 4.31: Home dashboard.

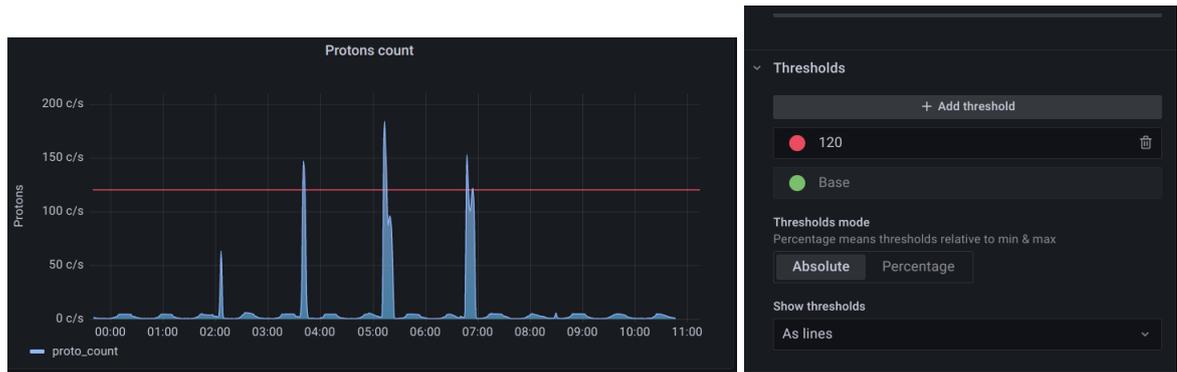


Figura 4.32: Configurazione di una soglia visibile sul grafico agendo direttamente sulla configurazione del pannello.

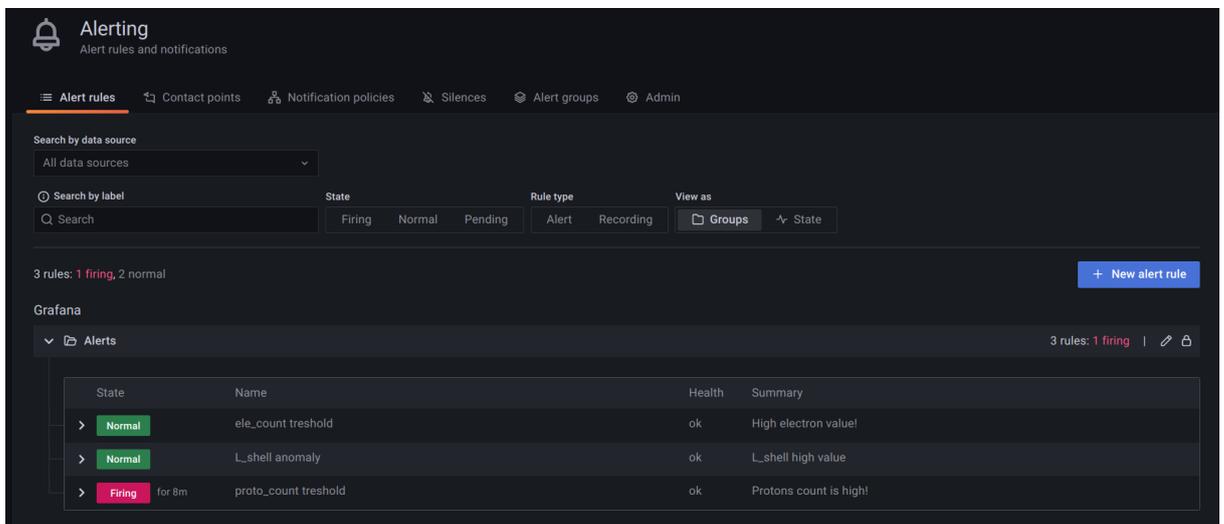


Figura 4.33: Interfaccia di gestione centralizzata delle regole di alert.

Questi componenti sono:

`Alert_rules` è costituita dall'insieme dei criteri di valutazione delle condizioni che determinano l'attivazione di un alert, ovvero almeno una query (ma anche più di una), una condizione sul risultato della query, la frequenza di valutazione e la durata durante la quale la condizione è soddisfatta;

`Contact_point` è un canale di comunicazione e un contatto a cui inviare la notifica in caso di alert;

`Notification_policy` una serie di criteri e regole per rendere le notifiche più specifiche;

Silences regole e periodi di tempo per gestire interruzioni di notifiche.

Una delle regole che ho creato, ad esempio, è proprio quella relativa al conteggio di protoni fuori dalla zona della SAA. La creazione della regola passa per i punti descritti sopra ovvero:

1. ho configurato prima di tutto il nome, il tipo e la cartella (Fig. [4.34](#))
2. ho creato poi un query che dovrà essere valutata contro una condizione che ho opportunamente impostato (Fig. [4.34](#))
3. ho scelto la frequenza di valutazione e il tempo di durata dell'evento valutato (Fig. [4.35](#))
4. ho completato la configurazione con la personalizzazione del messaggio da inviare (Fig. [4.35](#))

In particolare la query che ho creato per la regola è quella del Listato [4.8](#) che viene valutata con una condizione semplice per la quale se il valore massimo supera 80 c/s si attiva un alert che viene comunicato via email al contact point.

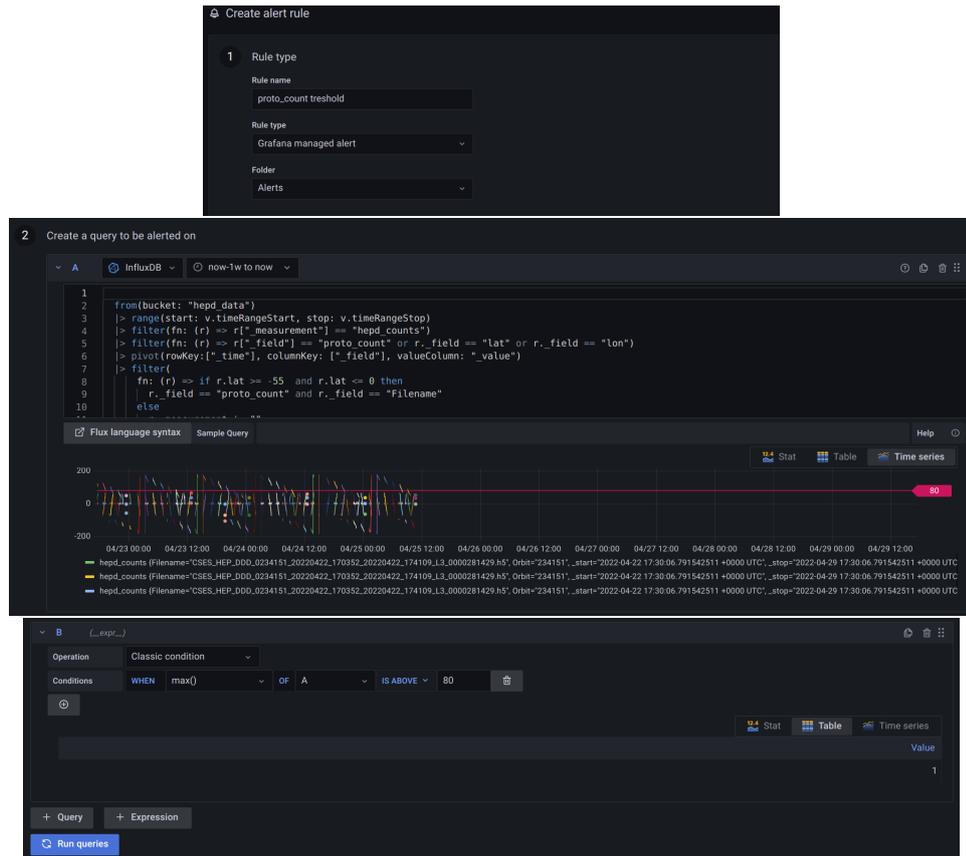


Figura 4.34: Configurazione alert rule 1-2

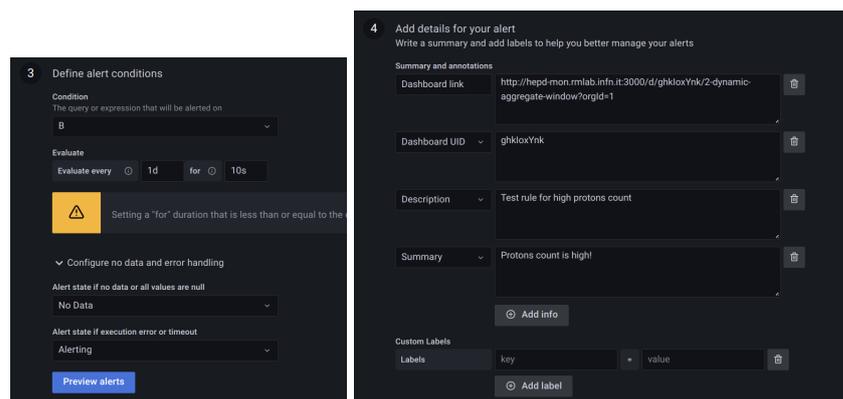


Figura 4.35: Configurazione alert rule 3-4

Listato 4.8: Esempio di Alert query sul conteggio dei protoni con esclusione della zona SAA

```
1 from(bucket: "hepd_data")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r["_measurement"] == "hepd_counts")
4   |> filter(fn: (r) => r["_field"] == "proto_count" or r._field == "lat" or
5     r._field == "lon")
6   |> pivot(rowKey:["_time"], columnKey: ["_field"], valueColumn: "_value")
7   |> filter(
8     fn: (r) => if r.lat >= -55 and r.lat <= 0 and r.lon >= -93 and r.lon <= 40 then
9       r._field == "proto_count" and r._field == "Filename"
10      else
11        r._measurement != ""
12    )
13   |> drop(columns: ["lat", "lon", "Orbit"])
```

## 4.9 Use case

Chiudo questo capitolo con un paio di esempi del funzionamento del sistema nella visualizzazione di eventi interessanti. Il 2 aprile l'esplosione di un Flare Solare di lunga durata di classe M4 (Fig. 4.36) è stato registrato dal Solar Dynamics Observatory della NASA.

Dopo aver ricevuto la mail di avviso per l'attivazione dell>alert (Listato 4.8), mi sono collegato sulla Home del sistema di monitoraggio impostando un time range compatibile con quanto riportato nella mail di avviso. In effetti il pannello della Top Ten sui protoni fuori dalla zona SAA ha mostrato subito un alto numero di conteggi/s (Fig. 4.37, sopra), dallo stesso pannello c'è la possibilità, tramite il link configurato precedentemente, di spostarsi sulla «Flat dashboard» trasportando il time range e visualizzando immediatamente l'intero evento (Fig. 4.37, sotto). Si può notare subito, oltre al solito andamento periodico dei picchi della SAA, la «portante» del flare con inizio alla 13 circa e lo smorzamento fino al giorno seguente. Non è visibile un flusso continuo nel grafico perché questo è il risultato dei dati acquisiti in tutte le semi-orbite comprese nell'intervallo di tempo interessato dal flare, intervallate quindi dal passaggio ai poli in cui non c'è acquisizione di dati. Guardando il pannello di correlazione tra i dati di conteggio e quelli del valore di latitudine è lampante la correlazione tra i picchi relativi al flare e i valori di picco della latitudine (risultano perfettamente coincidenti), mentre i picchi della SAA si trovano tutti alle latitudini caratteristiche della SAA. (Fig. 4.38).

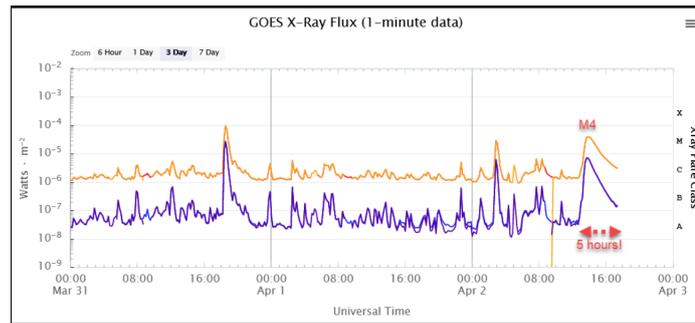


Figura 4.36: Solar flare di classe M4 registrato il 2 aprile 2022. Dati [spaceweather.com](https://spaceweather.com)

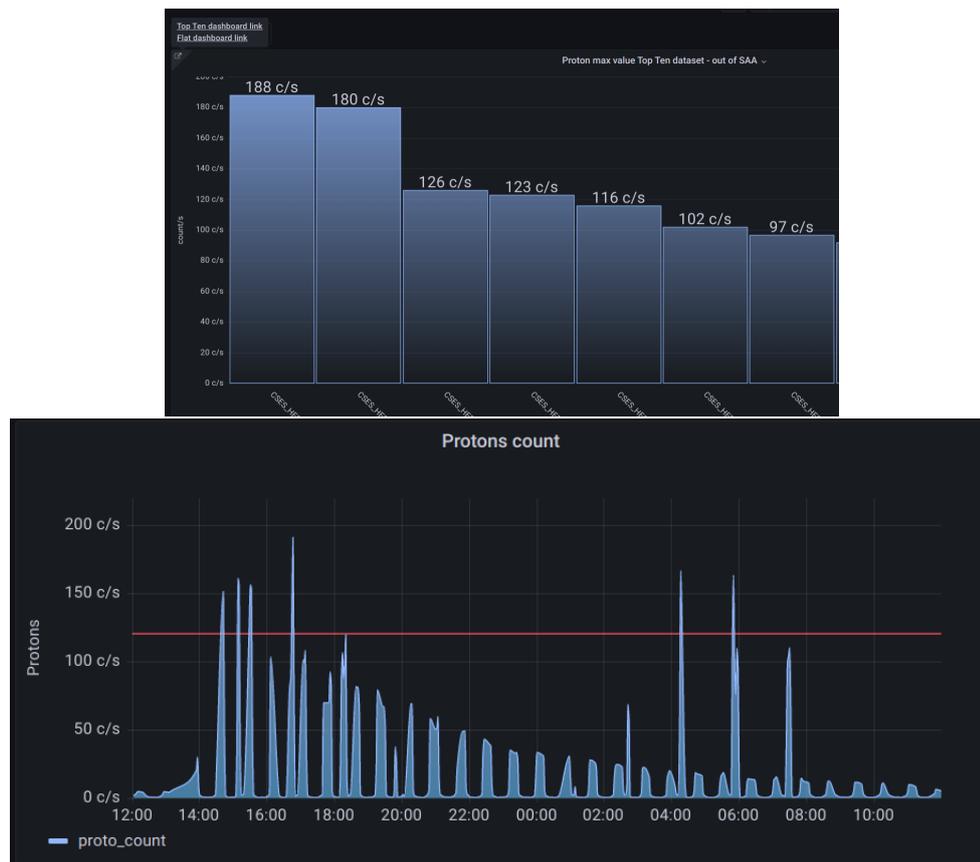


Figura 4.37: Pannello Top Ten per il conteggio di protoni fuori dalla SAA (sopra). Dal link in alto a dx si può accedere alla Flat dashboard (sotto).

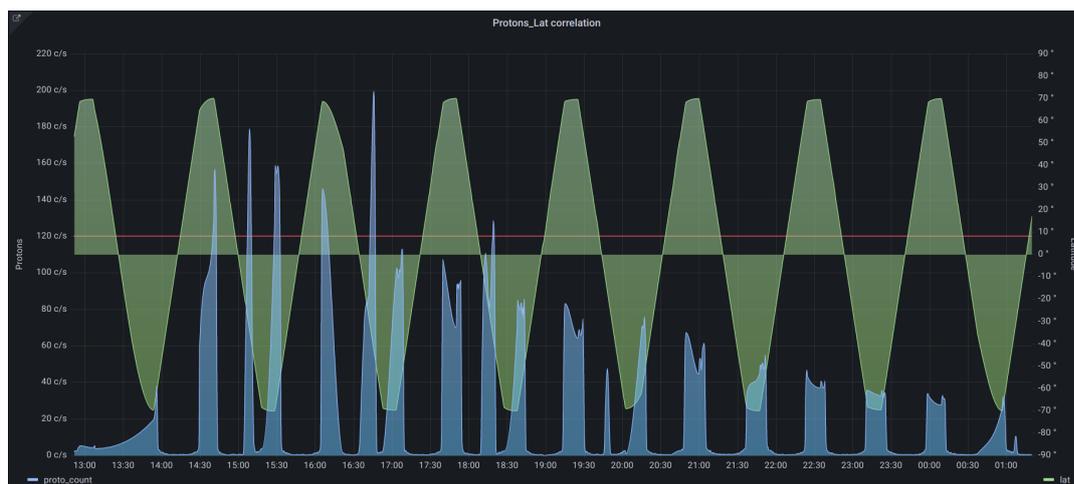


Figura 4.38: Correlazione tra i picchi del flare e quelli del valore di latitudine.

Anche se un sistema di monitoraggio pensato soprattutto per lo studio di fenomeni può essere in generale visto fuori da uno schema a «ciclo continuo», come accennato nell'introduzione (Sez. 3.1), spesso invece si trova il modo per sfruttarlo anche in questo paradigma. Ad esempio potremmo sfruttare questo sistema per capire la qualità dei dati ed eventualmente migliorarla. I dati di livello L3 sono infatti il risultato di un processo di elaborazione software che si occupa anche di separare elettroni e protoni in base all'analisi delle misure rilevate da HEPD. Confrontando quindi i dati di eventi noti anche meno evidenti con quelli acquisiti da fonti esterne, come in questo esempio, si può verificare la qualità di questa separazione ed eventualmente intervenire sul software per migliorarla. Per eventi di questa portata guardando Fig. 4.39 si riscontra comunque un ottimo livello di separazione tra elettroni e protoni.

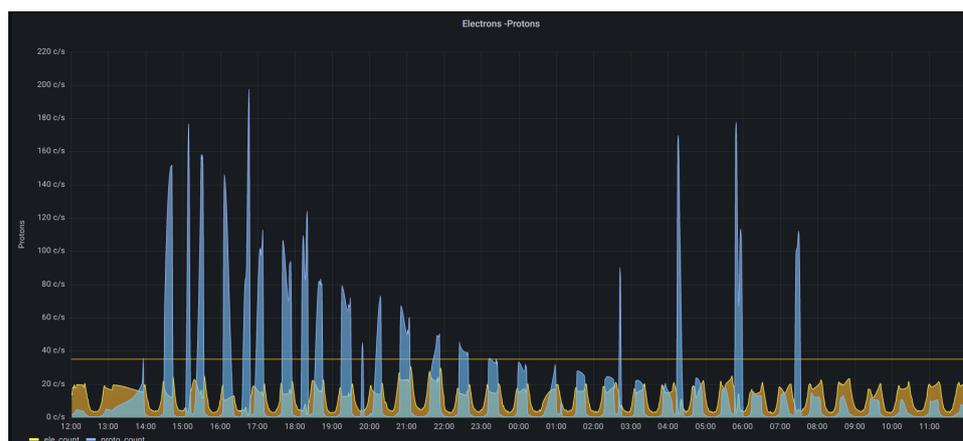


Figura 4.39: Sovrapposizione dei grafici elettroni-protoni per l'evento del 2 aprile 2022.

Ancora più imponente è il flare di lunga durata e di classe X1 sprigionato il 28 ottobre 2021 nella regione 2887, in quel momento allineata in direzione della terra, che in questo caso ha causato anche un'eruzione di massa coronale che ha investito la Terra il 31 ottobre (Fig 4.40).

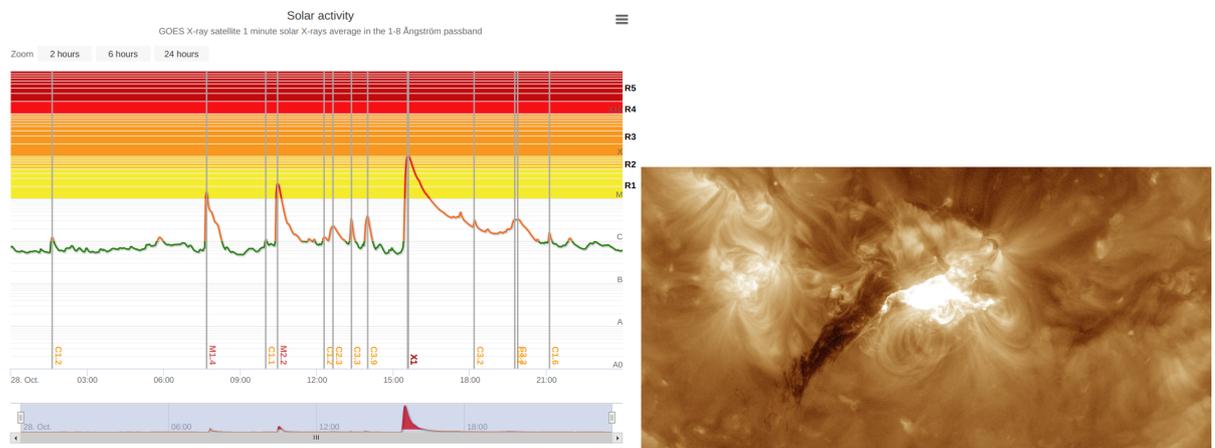


Figura 4.40: Plot e immagine del flare di classe X1 del 28/10/2022. Dati SpaceWeatherLive.com

Anche in questo caso, inserendo nel sistema l'intervallo di tempo tra il 28 ottobre e il 1 novembre, dalla TopTen dashboard si può notare subito un elevato valore per i conteggi dei protoni per tutti dataset del periodo selezionato (Fig. 4.41). Passando alla visualizzazione della Flat Dashboard nel pannello Protoni-Elettroni possiamo subito notare l'importanza e la durata dell'evento e ancora l'ottima separazione tra protoni ed elettroni del livello L3 dei dati (Fig. 4.42).

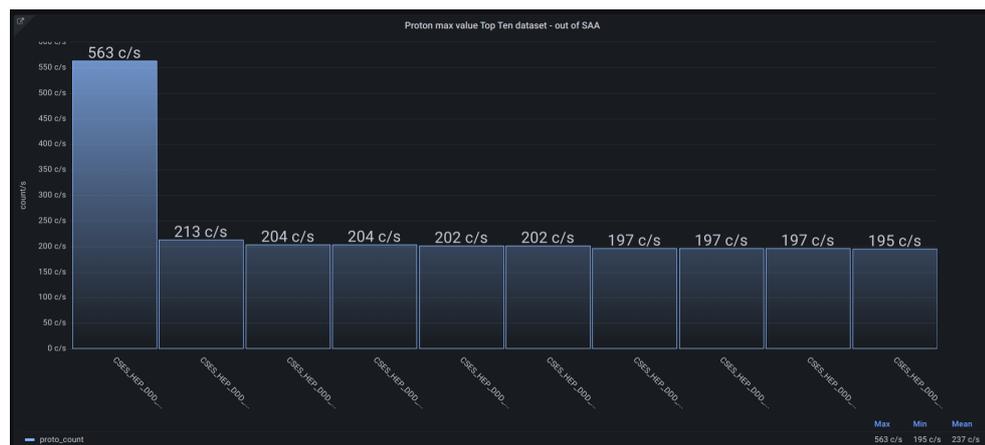


Figura 4.41: Top ten panel del conteggio di protoni nel periodo relativo al flare di classe X1.

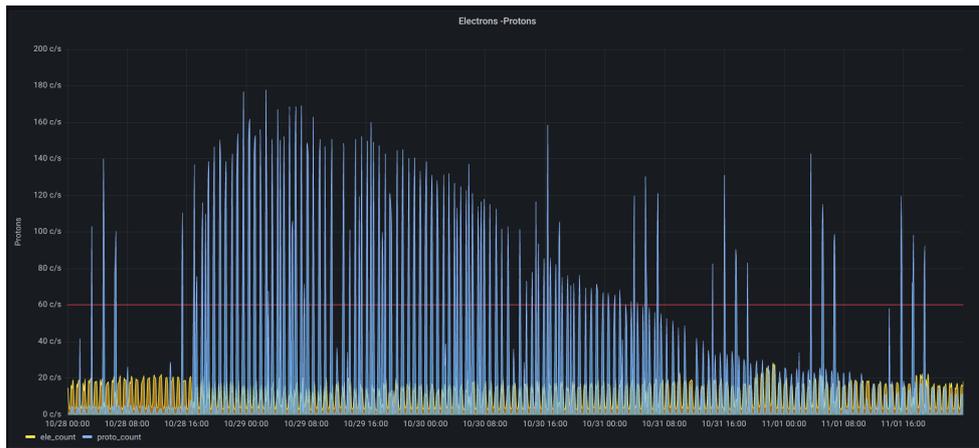


Figura 4.42: Visualizzazione della correlazione protoni-elettroni nel periodo relativo al flare di classe X1.

Considerando un periodo più ampio e tagliando la componente di conteggi in zona SAA otteniamo il grafico in figura 4.43 si può notare come l'attività si durata per diversi giorni con una coda fino a oltre il 3 novembre 2021.

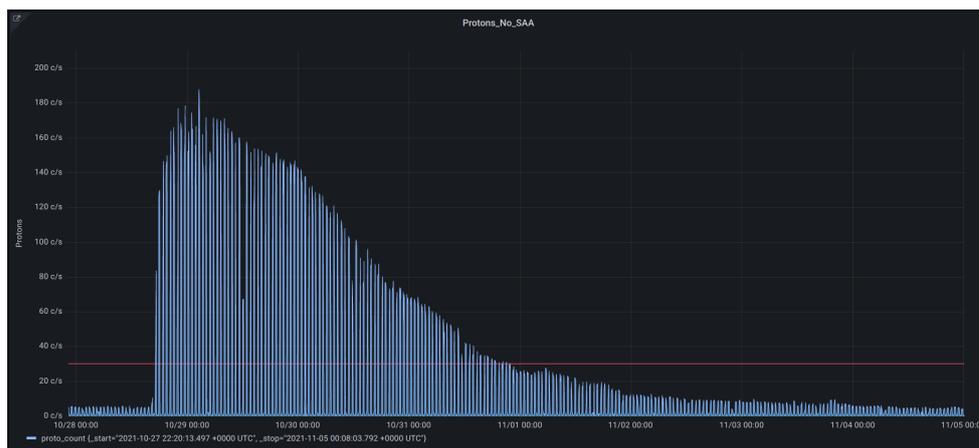


Figura 4.43: Grafico del rate di conteggio dei protoni fuori dalla zona dell'Anomalia del Sud Atlantico.



## 5 Roadmap e casi d'uso

Questo sistema di monitoraggio è tutt'altro che definitivo. Soprattutto considerando che il monitoraggio è un processo e un sistema come questo, con le sue caratteristiche di flessibilità, possibilità di configurazione e dinamicità, deve rientrare esso stesso in un ciclo *Plan, Do, Check, Act* per ottenere delle performance sempre maggiori ed una sempre più vasta gamma di casi d'uso.

### 5.1 Integrazioni e miglioramenti a breve termine

Già nell'immediato futuro è previsto l'arricchimento degli attuali data set di HEPD attraverso l'integrazione nativa con un database MySQL che verrà popolato con tutti i meta dati relativi. Conto poi di inserire anche i dati provenienti dal rilevatore HEPP che saranno sempre in formato HDF5. Sarà quindi possibile farlo con una modifica dell'agent che adatterò alla nuova struttura dati. HEPP infatti è costituito da nove sensori separati e formatterò i punti da iniettare nel database facendo un uso più completo degli array multidimensionali messi a disposizione dalla libreria python NumPy.

Un altro obiettivo del prossimo futuro è quello di inserire nel sistema i dati di livello L2. Questo formato, a differenza di L3, contiene informazioni relative ad eventi catturati anche con altre configurazioni di trigger [2.7](#), ad esempio eventi di particelle in un range di energia più basso. Questo potrebbe aiutare nella rivelazione di tempeste geomagnetiche.

Dal punto di vista sistemistico il sistema è già predisposto per essere scalato verticalmente e questo ci dà già la possibilità di gestire oltre un milione di serie dati uniche e avere performance più che adeguate in termini di scrittura e query. In ogni caso l'architettura basata su docker-compose permette di replicare l'applicazione in modo pressoché immediato e si potrebbe pensare ad una gestione snella basata su un app per tenant (ovvero per organizzazione Grafana) nel caso le collaborazioni e gli utilizzatori dovessero aumentare, anche senza ricorrere ad uno scaling orizzontale (clustering).

Per l'interfaccia è prevista una rivisitazione continua con correzioni frequenti e apporti di nuove funzionalità in base al feedback degli utilizzatori. Già da questo momento sto lavorando ad un più vasto e ottimizzato utilizzo dei link e alla loro parametrizzazione attraverso le variabili in modo da rendere le visualizzazioni sempre più immediate nel passaggio da una all'altra e al pannello dedicato alla ricerca globale per tag sullo storico dei dataset. Le possibilità sono davvero moltissime e il sistema viene costruito quotidianamente e dinamicamente durante l'utilizzo.

Un'altra integrazione possibile sulla quale mi sto preparando a lavorare è quella degli Open Data messi a disposizione dalla comunità scientifica. Questo permetterebbe di avere un confronto visivo immediato con molte serie di dati già disponibili online. Molto utile in questo ambito è la possibilità di integrare i dati relativi ad eventi sismici da affiancare alle visualizzazioni che creerò per il monitoraggio in ambito precursori sismici con i dati di HEPP.

## 5.2 Collaborazioni

Il sistema implementato con questo lavoro nasce nel contesto della missione CSES-Limadou, ciò non toglie che sia un servizio già pronto e aperto all'utilizzo da parte di altri esperimenti e gruppi di lavoro che avessero serie storiche da analizzare per la loro ricerca. Anzi, è un sistema fortemente e strutturalmente orientato ad essere uno strumento collaborativo. Soprattutto in ambito Space Weather, dove l'analisi di time series diventa fondamentale nella previsione e nello studio degli eventi e dei fenomeni ad essi associati, è importante avere a disposizione strumenti che permettano in modo semplice e flessibile di integrare dati provenienti da molte fonti differenti. I plugin e le librerie presenti nativamente sia in InfluxDB che in Grafana, come ho già spiegato nel capitolo precedente, permettono l'inserimento e l'integrazione di dati da qualsiasi tipo di sorgente senza necessità di spostarli o replicarli su altri volumi o altre sedi geografiche. L'efficacia del linguaggio funzionale Flux e le infinite possibilità di personalizzazione offerte da Grafana saranno sfruttate ampiamente per il resto del lavoro.

Per il prossimo futuro è già prevista la collaborazione e l'utilizzo di questa piattaforma da parte del gruppo di Fisica Solare e Spaziale di Tor Vergata sempre in ambito Space Weather.

Abbiamo già in preparazione dei pannelli specifici per i gruppi coinvolti in CSES-Limadou del TIFPA di Trento nell'ambito dello studio dei precursori sismici non ap-

pena avrò inserito e configurato i dati del rivelatore HEPP.

### 5.2.1 CSES-02

Siamo in attesa inoltre del lancio del secondo satellite della costellazione prevista dalla missione CSES-Limadou che dovrebbe avvenire entro dicembre del 2022 ed avere un ciclo di vita di 6 anni. Lo scopo della missione è legato allo studio dei fenomeni elettromagnetici collegati allo Space Weather, alla fisica della ionosfera terrestre e ai meccanismi di accoppiamento tra magnetosfera-ionosfera-litosfera. La componente italiana sarà coordinata dall'INFN e vede ancora la partecipazione delle sezioni e laboratori di Bologna, dei Laboratori Nazionali di Frascati (LNF), Napoli, Roma Tor Vergata e TIFPA (Trento) in collaborazione con l'INAF-IAPS di Roma, l'Università degli Studi di Roma Tor Vergata, l'Università degli Studi di Trento, l'Università degli Studi di Napoli Federico II e Uninettuno.

Il satellite CSES-02 è realizzato sulla stessa piattaforma di CSES (*CAST2000*), avrà uno sfasamento di  $180^\circ$  rispetto a CSES-01 e ospiterà a bordo un sistema integrato composto da 9 strumenti scientifici tra i quali quelli per cui è coinvolta la componente italiana che si è occupata:

- della realizzazione, il test e la calibrazione di un rivelatore di campo elettrico denominato Electric Field Detector (EFD2) – in quota INAF-IAPS;
- della realizzazione, il test e la calibrazione del rivelatore di particelle High Energy Particle Detector (HEPD2) – in quota INFN;
- di consentire a personale cinese (scienziati ed ingegneri) di seguire lo sviluppo dei vari modelli previsti per il rivelatore EFD, realizzati a guida INAF-IAPS
- di fornire l'accesso alle infrastrutture di test per la calibrazione dei rivelatori di plasma Langmuir probes e Plasma Analyzer – INAF-IAPS;

In particolare agganceremo al sistema di monitoraggio in prima battuta i dati del nuovo rivelatore HEPD-02 che porterà molti miglioramenti rispetto alla prima versione. non ultimo il fatto di avere un'area di lavoro che va dai  $-90^\circ$  ai  $+90^\circ$  di latitudine. Infatti con CSES-02 i payload non verranno messi in stand-by durante il passaggio ai poli. Tutto questo, insieme allo sfasamento dei due satelliti ci permetterà di coprire molto meglio anche il monitoraggio dei transienti solari.

## 5.3 Sistema o servizio? - SaaM-MaaS

Tutto ciò che ho descritto finora mette in luce la natura di questo sistema pensato in realtà come un servizio. L'ho pensato e progettato infatti per poter essere gestito a pieno anche dall'utilizzatore finale che, semplicemente collegandosi via web sul suo portale, avrà accesso completo (nell'ambito del suo progetto e/o del suo ruolo) alla creazione e modifica di dashboard e pannelli e alla scrittura delle query.

Con licenza di creazione degli acronimi ho pensato di chiamare questo sistema SaaM (System of alerting and Monitoring) che però rovesciato diventa MaaS che nell'ambito dei paradigmi cloud e del monitoraggio significa Metric as a Service.<sup>1</sup> Questo descrive benissimo la natura di questo servizio: un *web service* con architettura *cloud* destinato al monitoraggio di *time series*, integrato, raggiungibile ed utilizzabile da qualsiasi parte del mondo in modo autenticato.

---

<sup>1</sup>Più spesso nei paradigmi cloud la notazione «MaaS» è riferita a Metal as a Service, ovvero alla possibilità di offrire sistemi bare metal on demand, ma in questo ambito non è inusuale che sia riferito alle «metriche».

# Conclusioni

Il lavoro descritto in questa tesi riguarda la progettazione e l'implementazione di un sistema di *alerting* e *monitoring* dedicato all'analisi di serie temporali dei dati di livello L3 acquisiti dal rivelatore di particelle HEPD (*High Energy Particle Detector*) a bordo del satellite Cinese CSES nell'ambito della missione Italo-Cinese CSES\_Limadou. Questi dati al momento sono sottoutilizzati. Avere molti dati poco utilizzati significa avere poche informazioni quindi poca «conoscenza».

Ho scelto di progettare il sistema basandomi su uno stack di tre componenti ognuno dei quali svolge specifici task. Un agent scritto da me in Python per l'interfaccia verso i dati si occupa dell'estrazione di questi dal loro formato originale e della formattazione dei «punti» *Flux* da inserire in una piattaforma di gestione di time series (*InfluxDB*). Quest'ultimo è il secondo componente, che configurato ad hoc consente una completa gestione di serie temporali di dati in modo efficiente ed efficace, mettendo a disposizione un completo e potente linguaggio funzionale di analisi e query. Il terzo componente (*Grafana*) è specificamente pensato e realizzato per la visualizzazione dei dati consentendo, in perfetta integrazione con *InfluxDB*, la configurazione di *Panels* e *Dashboard* dinamiche oltre ad una completa centralizzazione dell'interfaccia di *alerting*. La scelta di questa architettura permette da un lato di avere componenti separati e quindi maggiore flessibilità e possibilità di adattamento a differenti esigenze senza implicare lo stravolgimento dell'intero sistema, dall'altra di sfruttare a pieno le possibilità e le funzioni messe a disposizione dai tre componenti ognuno dei quali si esprime al meglio nell'ambito degli specifici task per cui ho deciso di utilizzarlo.

Il servizio che questo sistema offre è di fatto un *web service*. Una volta collegate, da parte dell'amministratore, le sorgenti dati, l'utilizzatore finale ha la possibilità di collegarsi in modo autenticato via web da qualsiasi parte del mondo ed interagire col sistema nell'ambito dei suoi progetti con specifici permessi e ruoli sfruttando le sue funzionalità completamente via web. La possibilità di condividere semplicemente dashboard e pannelli all'interno di un progetto o anche, in alcuni casi, pubblicamente, così come la possibilità di integrare qualsiasi *data source*, la gestione interna di link,

variabili, annotazioni e sistemi di comunicazione degli alert, ne fanno inoltre un servizio pensato per essere uno strumento fortemente collaborativo. Tutto questo mette in luce la natura «cloud» di questo sistema e proprio secondo i paradigmi cloud lo possiamo vedere con un servizio di MasS (*Metric as a Service*)[46].

In conclusione il mio lavoro fornisce una piattaforma integrata di gestione e visualizzazione dei dati che permette di individuare rapidamente i dataset più interessanti e di essere avvisati su eventuali anomalie. Permette inoltre di accedere allo storico completo dei dati in modo semplice e veloce condividendo con altri ogni tipo di visualizzazione. La progettazione, lo sviluppo e l'implementazione del servizio sono completi e può essere consolidato e messo in produzione nel giro di poco tempo. Funzionalità aggiuntive e nuove integrazioni di sorgenti dati, pannelli e dashboard sono in sviluppo in un processo di continuo miglioramento che coinvolge fortemente gli utilizzatori finali.

# Codice

Listato 5.1: *Python agent*

```
1 #!/usr/bin/env python
2 #####
3 # PYTHON MODULES IMPORT
4 #####
5 from posixpath import basename, dirname, relpath
6 #from turtle import shape
7 #import influxdb
8 #import numpy as np
9 import logging
10 import h5py
11 import time
12 import os
13 import sys
14 #import more_itertools
15 from datetime import datetime
16 from urllib3 import Retry
17 #import influxdb_client
18 #from influxdb import InfluxDBClient
19 from influxdb_client.client.write_api import SYNCHRONOUS,
       ASYNCHRONOUS
20 from influxdb_client import InfluxDBClient, WriteOptions
21 from influxdb_client.client.exceptions import InfluxDBError
22 #####
23
24 # LOGGING CONFIGURATION
25 logging.basicConfig(filename='influx.log', level=logging.DEBUG)
26
27 # THIS FUNCTION READS HEPD H5 FILES AND FILENAME
28 def read_hepd_h5_info(file):
```

```

29
30     with h5py.File(file, 'r') as f:
31         #print("filename: %s" % file)
32         if not f.keys():
33             logging.debug('EMPTY FILE!!!: '+file)
34             return
35         rel_path = basename(file)
36         keys = rel_path.split('_')
37         orbitid = int(keys[3])
38         print(orbitid)
39         startid = keys[4]
40         endid = keys[6]
41
42         ##### CREATE INFLUXDB CONNECTION #####
43         token = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
44         org = "hepd"
45         bucket = "hepd_data"
46         retries = Retry(connect=5, read=2, redirect=5)
47         client = InfluxDBClient(url="http://hepd-mon.rmlab.infn.it
48             :8086", token=token, org=org, retries=retries,
49             enable_gzip=True)
50
51         # INSTANTIATE THE WriteAPI AND QueryAPI
52         write_api = client.write_api(write_options=SYNCHRONOUS)
53         #query_api = client.query_api()
54
55         #####
56         # GETTING KEYS OF HD5 FILE
57         #for key in f.keys():
58         #    print(key)
59         #####
60
61         #####
62         # HEPD DATASETS LIST
63         #alt_data=f['Altitude']
64         # b_data=f['B']
65         #gmlonlat_data=f['GMLonLat']

```

```

64     eleccounts_data=f['HEPD_ele_counts']
65     # eleenepitch_data=f['HEPD_ele_energy_pitch']
66     # eleenetable_data=f['HEPD_ele_energy_table']
67     # elepitch_data=f['HEPD_ele_pitch_table']
68     procounnts_data=f['HEPD_pro_counts']
69     # proenepitch_data=f['HEPD_pro_energy_pitch']
70     # proenetable_data=f['HEPD_pro_energy_table']
71     # propitch_data=f['HEPD_pro_pitch_table']
72     l_data=f['L_parameter']
73     lonlat_data=f['LonLat']
74     #attr_data=f['ProductAttributes']
75     utctime_data=f['UTCTime']
76     #####
77
78     ##### CREATE LIST OF POINTS #####
79     records = (utctime_data.shape[0])
80     print(" CREATING AND WRITING POINTS OF: "+rel_path)
81     points = []
82     for i in range(records):
83         s = str(utctime_data[i])
84         t = datetime.strptime(s, '%Y%m%d%H%M%S')
85         ft = t.isoformat('T') + "Z"
86         point_body = {
87             "measurement": "hepd_counts",
88             "tags": {
89                 "Filename": rel_path,
90                 "Orbit": orbitid
91             },
92             "time": ft,
93             "fields": {
94                 "ele_count": eleccounts_data[i],
95                 "proto_count": procounnts_data[i],
96                 "L_shell": l_data[i],
97                 "lon": lonlat_data[i][0],
98                 "lat": lonlat_data[i][1]
99             }
100         }

```

```

101         points.append(point_body)
102
103         ##### WRITE LIST OF POINTS TO INFLUXDB #####
104         try:
105             write_api.write(bucket=bucket,org=org,record=points)
106         except InfluxDBError as e:
107             if e.response.status == 401:
108                 raise Exception(f"Insufficient write permissions
109                               to 'bucket'") from e
110             if e.response.status == 404:
111                 raise Exception(f"Not found: probably org or
112                               bucket") from e
113             if e.response.status == 413:
114                 raise Exception(f"Request entity too large") from
115                               e
116             raise
117
118         print()
119         print("Wait to finishing ingesting...")
120         print()
121
122         ##### MAIN #####
123
124         def _main():
125             #rootDir = walkdir
126             rootDir = sys.argv[1]
127             #print(rootDir)
128
129             # MAIN DIRECTORY CYCLE
130             start_time=time.time()
131             for dirName, subdirList, fileList in os.walk(rootDir):
132                 #print("Found directory: %s" % dirName)
133                 fileList.sort()
134                 dict = {}
135                 for fname in fileList:
136                     ext = os.path.splitext(fname)[-1].lower()
137                     noext = os.path.splitext(fname)[0]

```

```

135         tok = noext.split('_')
136         fullfilename = os.path.join(dirName, fname)
137         #if fname.endswith(".h5"):
138         if ext == ".h5" and tok[2] == 'DDD':
139             key = tok[3]
140             if key not in dict:
141                 #if not (dict.get(key)):
142                 dict[key] = fullfilename
143             else:
144                 diffsize = (os.path.getsize(dict[key]) - os.
145                             path.getsize(fullfilename))
146                 if (diffsize < 0):
147                     dict[key] = fullfilename
148         for i in dict:
149             read_hepd_h5_info(dict[i])
150
151     print('finished cycle, it took %d seconds' % (time.time() -
152           start_time))
153
154 if __name__ == '__main__':
155     _main()

```

Listato 5.2: *docker-compose.yml*

```

1 version: '3.7'
2 services:
3   influxdb:
4     container_name: influx
5     image: influxdb:2.1.1
6     ports:
7       - '8086:8086'
8     volumes:
9       - influxdb-storage:/var/lib/influxdb2
10    restart: always
11    privileged: true
12
13 grafana:
14   container_name: grafana
15   image: grafana/grafana:8.4.2

```

```
16     ports:
17         - '3000:3000'
18     volumes:
19         - ./grafana-custom.ini:/etc/grafana/grafana.ini
20         - grafana-storage:/var/lib/grafana
21     links:
22         - influxdb:influxdb
23     depends_on:
24         - influxdb
25     restart: always
26     privileged: true
27
28 volumes:
29     influxdb-storage:
30     grafana-storage:
```

# Bibliografia

- [1] «<https://www.influxdata.com/products/influxdb-overview/>» 4
- [2] «<https://grafana.com/>» 4
- [3] P. Picozza, R. Battiston, G. Ambrosi, S. Bartocci, L. Basara, W. J. Burger, D. Campana, L. Carfora, M. Casolino, G. Castellini, P. Cipollone, L. Conti, A. Contin, C. De Donato, C. De Santis, F. M. Follega, C. Guandalini, M. Ionica, R. Iuppa, G. Laurenti, I. Lazzizzera, M. Lolli, C. Manea, L. Marcelli, M. Martucci, G. Masciantonio, M. Mergé, G. Osteria, L. Pacini, F. Palma, F. Palmonari, B. Panico, A. Parmentier, L. Patrizii, F. Perfetto, M. Piersanti, M. Pozzato, M. Puel, I. Rashevskaya, E. Ricci, M. Ricci, S. Ricciarini, V. Scotti, A. Sotgiu, R. Sparvoli, B. Spataro, V. Vitale, P. Zuccon, and S. Zoffoli. Scientific Goals and In-orbit Performance of the High-energy Particle Detector on Board the CSES. *Astrophys. J. Suppl.*, 243(1):16, July 2019. 15
- [4] M. E. Aleshina, S. A. Voronov, A. M. Gal'Per, S. V. Koldashov, and L. V. Maslennikov. Correlation between earthquake epicenters and regions of high-energy particle precipitations from the radiation belt. *Cosmic Research*, 30:65–68, July 1992. 21
- [5] , C. F., 1956. Magnitude and Energy of Earthquakes. *Annali di Geofisica*, 9: 1–15
- [6] V. Sgrigna. ESPERIA Experiment: Phase A Report Submitted to the Italian Space Agency (ASI). AGU Fall Meeting Abstracts, December 2001. 9, 23
- [7] G. Calderoni, B. De Simoni, R. Di Giovambattista e C. Salvaterra - Istituto Nazionale di Geofisica «Variazioni anomale del potenziale spontaneo: monitoraggio e analisi preliminare in due siti dell'Italia Centrale» 19
- [8] A. Sotgiu «Development of the control and data handling software for the High-Energy Particle Detector and preliminary data analysis»

- [9] A. Balogh, L. J. Lanzerotti, and S. T. Suess. «The Heliosphere through the Solar Activity Cycle». Springer-Praxis Books and SpringerScience+Business Media, 2008. [30](#)
- [10] A. B. Struminsky and I. V. Zimovetz. Observations of the December 6, 2006 solar flare: Electron acceleration and plasma heating. *Astronomy Letters*, 36:430–437, June 2010. [30](#)
- [11] O. Adriani, G. C. Barbarino, G. A. Bazilevskaya, R. Bellotti, M. Boezio, E. A. Bogomolov, L. Bonechi, M. Bongi, V. Bonvicini, S. Borisov, S. Bottai, A. Bruno, F. Cafagna, D. Campana, R. Carbone, P. Carlson, M. Casolino, G. Castellini, L. Consiglio, M. P. De Pascale, C. De Santis, N. De Simone, V. Di Felice, V. Formato, A. M. Galper, L. Grishantseva, W. Gillard, G. Jerse, A. V. Karelin, S. V. Koldashov, S. Y. Krutkov, A. N. Kvashnin, A. Leonov, V. Malakhov, L. Marcelli, A. G. Mayorov, W. Menn, V. V. Mikhailov, E. Mocchiutti, A. Monaco, N. Mori, N. Nikonov, G. Osteria, F. Palma, P. Papini, M. Pearce, P. Picozza, C. Pizzolotto, M. Ricci, S. B. Ricciarini, R. Sarkar, L. Rossetto, M. Simon, R. Sparvoli, P. Spillantini, Y. I. Stozhkov, A. Vacchi, E. Vannuccini, G. Vasilyev, S. A. Voronov, J. Wu, Y. T. Yurkin, G. Zampa, N. Zampa, and V. G. Zverev. Observations of the 2006 December 13 and 14 Solar Particle Events in the 80 MeV  $n-1$  -3 GeV  $n-1$  Range from Space with the PAMELA Detector. *ApJ*, 742:102, December 2011. [30](#)
- [12] D. Lario, A. Aran, R. Gomez-Herrero, N. Dresing, B. Heber, G. C. Ho, R. B. Decker, and E. C. Roelof. Longitudinal and Radial Dependence of Solar Energetic Particle Peak Intensities: STEREO, ACE, SOHO, GOES, and MESSENGER Observations. *ApJ*, 767:41, April 2013. [31](#)
- [13] S. Della Torre and AMS-02 Collaboration. Results on Solar Physics from AMS-02. ArXiv e-prints, December 2016. [31](#)
- [14] M. Storini, K. Kudela, E. G. Cordaro, and S. Massetti. Ground-level enhancements during solar cycle 23: results from SVIRCO, LOMNICKY STIT and LARC neutron monitors. *Advances in Space Research*, 35:416–420, 2005. [31](#)
- [15] O. Adriani, G. C. Barbarino, G. A. Bazilevskaya, R. Bellotti, and M. Boezio. Ten years of pameLA in space. *La Rivista del Nuovo Cimento*, 40:473521, Sep 2017. [31](#)
- [16] O. Adriani, G. C. Barbarino, G. A. Bazilevskaya, R. Bellotti, M. Boezio, E. A. Bogomolov, M. Bongi, V. Bonvicini, S. Bottai, U. Bravar, A. Bruno, F. Cafagna, D.

- Campana, R. Carbone, P. Carlson, M. Casolino, G. Castellini, E. R. Christian, C. De Donato, G. A. de Nolfo, C. De Santis, N. De Simone, V. Di Felice, V. Formato, A. M. Galper, A. V. Karelin, S. V. Koldashov, S. Koldobskiy, S. Y. Krutkov, A. N. Kvashnin, M. Lee, A. Leonov, V. Malakhov, L. Marcelli, M. Martucci, A. G. Mayorov, W. Menn, M. Merge, V. V. Mikhailov, E. Mocchiutti, A. Monaco, N. Mori, R. Munini, G. Osteria, F. Palma, B. Panico, P. Papini, M. Pearce, P. Picozza, M. Ricci, S. B. Ricciarini, J. M. Ryan, R. Sarkar, V. Scotti, M. Simon, R. Sparvoli, P. Spillantini, S. Stochaj, Y. I. Stozhkov, N. Thakur, A. Vacchi, E. Vannuccini, G. I. Vasilyev, S. A. Voronov, Y. T. Yurkin, G. Zampa, and N. Zampa. Pamela Measurements of Magnetospheric Effects On High Energy Solar Particles. *ApJ Letters*, 801:L3, March 2015. [31](#)
- [17] T. Mathews and D. Venkatesan. Unique series of increases in cosmic-ray intensity due to solar flares. *Nature*, 345:600–602, June 1990. [31](#)
- [18] S. E. Forbush. Cosmic-Ray Intensity Variations during Two Solar Cycles. *JGR*, 63:651–669, December 1958. [31](#)
- [19] O. Adriani, G. C. Barbarino, G. A. Bazilevskaya, R. Bellotti, M. Boezio, E. A. Bogomolov, M. Bonghi, V. Bonvicini, S. Bottai, A. Bruno, F. Cafagna, D. Campana, P. Carlson, M. Casolino, G. Castellini, C. De Donato, G. A. de Nolfo, C. De Santis, N. De Simone, V. Di Felice, A. M. Galper, A. V. Karelin, S. V. Koldashov, S. Koldobskiy, S. Y. Krutkov, A. N. Kvashnin, A. Leonov, V. Malakhov, L. Marcelli, M. Martucci, A. G. Mayorov, W. Menn, M. Merge, V. V. Mikhailov, E. Mocchiutti, A. Monaco, N. Mori, R. Munini, G. Osteria, F. Palma, B. Panico, P. Papini, M. Pearce, P. Picozza, M. Ricci, S. B. Ricciarini, R. Sarkar, V. Scotti, M. Simon, R. Sparvoli, P. Spillantini, Y. I. Stozhkov, A. Vacchi, E. Vannuccini, G. I. Vasilyev, S. A. Voronov, Y. T. Yurkin, G. Zampa, and N. Zampa. PAMELA's measurements of geomagnetic cutoff variations during the 14 December 2006 storm. *Space Weather*, 14:210–220, March 2016. [32](#)
- [20] S. E. FORBUSH: *Phys. Rev.*, 54, 975 (1938). [31](#)
- [21] M. S. Potgieter, E. E. Vos, M. Boezio, N. De Simone, V. Di Felice, and V. Formato. Modulation of Galactic Protons in the Heliosphere During the Unusual Solar Minimum of 2006 to 2009. *Solar Physics*, 289:391–406, January 2014. [31](#)

- [22] E. N. Parker. The passage of energetic charged particles through interplanetary space. *Plan. Sp. Sci.*, 13:9–49, January 1965. [31](#)
- [23] M. S. Potgieter. The charge-sign dependent effect in the solar modulation of cosmic rays. *Advances in Space Research*, 53:1415–1425, May 2014. [31](#)
- [24] J. A. van Allen and A. V. Gangnes. On the Azimuthal Asymmetry of Cosmic-Ray Intensity above the Atmosphere at the Geomagnetic Equator. *Physical Review*, 79:51–53, July 1950. [32](#)
- [25] S. B. Treiman. The Cosmic-Ray Albedo. *Physical Review*, 91:957–959, August 1953. [32](#)
- [26] J. Moritz. Energetic protons at low equatorial altitudes. *Zeitschrift für Geophysik*, 38:701–717, 1972. [32](#)
- [27] J. Alcaraz, D. Alvisi, B. Alpat, G. Ambrosi, H. Anderhub, L. Ao, A. Arefiev, P. Az-zarello, E. Babucci, L. Baldini, M. Basile, D. Barancourt, F. Barao, G. Barbier, G. Barreira, R. Battiston, R. Becker, U. Becker, L. Bellagamba, P. Ben´e, J. Berdugo, ´ P. Berges, B. Bertucci, A. Biland, S. Bizzaglia, S. Blasko, G. Boella, M. Boschini, M. Bourquin, G. Bruni, M. Buenerd, J. D. Burger, W. J. Burger, X. D. Cai, R. Cavalletti, C. Camps, P. Cannarsa, M. Capell, D. Casadei, J. Casaus, G. Castellini, Y. H. Chang, H. F. Chen, H. S. Chen, Z. G. Chen, N. A. Chernoplekov, A. Chiarini, T. H. Chiueh, Y. L. Chuang, F. Cindolo, V. Commichau, A. Contin, A. Cotta-Ramusino, P. Crespo, M. Cristinziani, J. P. da Cunha, T. S. Dai, J. D. Deus, N. Dinu, L. Djambazov, I. D’Antone, Z. R. Dong, P. Emonet, J. Engelberg, F. J. Epp-ling, T. Eronen, G. Esposito, P. Extermann, J. Favier, C. C. Feng, E. Fiandrini, F. Finelli, P. H. Fisher, R. Flaminio, G. Fluegge, N. Fouque, Y. Galaktionov, M. Ger-vasi, P. Giusti, D. Grandi, W. Q. Gu, K. Hangarter, A. Hasan, V. Hermel, H. Hofer, M. A. Huang, W. Hungerford, M. Ionica, R. Ionica, M. Jongmanns, K. Karlamaa, W. Karpinski, G. Kenney, J. Kenny, W. Kim, A. Klimentov, R. Kossakowski, V. Koutsenko, G. Laborie, T. Laitinen, G. Lamanna, G. Laurenti, A. Lebedev, S. C. Lee, G. Levi, P. Levtchenko, C. L. Liu, H. T. Liu, M. Lolli, I. Lopes, G. Lu, Y. S. Lu, K. Lubelsmeyer, D. Luckey, W. Luster mann, C. Ma´na, A. Margotti, F. Massera, ´ F. Mayet, R. R. McNeil, B. Meillon, M. Menichelli, F. Mezzanotte, R. Mezzen- ga, A. Mihul, G. Molinari, A. Mourao, A. Mujunen, F. Palmonari, G. Pancaldi, A. Papi, I. H. Park, M. Pauluzzi, F. Pauss, E. Perrin, A. Pesci, A. Pevsner, R. Pilastrini, M. Pimenta, V. Plyaskin, V. Pojidaev, H. Postema, V. Postolache, E. Prati, N.

- Produit, P. G. Rancoita, D. Rapin, F. Raupach, S. Recupero, D. Ren, Z. Ren, M. Ribordy, J. P. Richeux, E. Riihonen, J. Ritakari, U. Roeser, C. Roissin, R. Sagdeev, D. Santos, G. Sartorelli, A. Schultz von Dratzig, G. Schwering, E. S. Seo, V. Shoutko, E. Shoumilov, R. Siedling, D. Son, T. Song, M. Steuer, G. S. Sun, H. Suter, X. W. Tang, S. C. C. Ting, S. M. Ting, M. Tornikoski, G. Torrromeo, J. Torsti, J. Trumper, J. Ulbricht, S. Urpo, I. Usoskin, E. Valtonen, J. Vandenhirtz, F. Velcea, E. Velikhov, B. Verlaat, I. Vetlitsky, F. Vezzu, J. P. Vialle, G. Viertel, D. Vite, H. Von Gunten, S. Waldmeier Wicki, W. Wallraff, B. C. Wang, J. Z. Wang, Y. H. Wang, K. Wiik, C. Williams, S. X. Wu, P. C. Xia, J. L. Yan, L. G. Yan, C. G. Yang, M. Yang, S. W. Ye, P. Yeh, Z. Z. Xu, H. Y. Zhang, Z. P. Zhang, D. X. Zhao, G. Y. Zhu, W. Z. Zhu, H. L. Zhuang, and A. Zichichi. Protons in near earth orbit. *Physics Letters B*, 472:215–226, January 2000. [32](#)
- [28] O. Adriani, G. C. Barbarino, G. A. Bazilevskaya, R. Bellotti, M. Boezio, E. A. Bogomolov, M. Bongi, V. Bonvicini, S. Bottai, A. Bruno, F. Cafagna, D. Campana, P. Carlson, M. Casolino, G. Castellini, C. de Donato, C. De Santis, N. De Simone, V. Di Felice, V. Formato, A. M. Galper, A. V. Karelin, S. V. Koldashov, S. Koldobskiy, S. Y. Krutkov, A. N. Kvashnin, A. Leonov, V. Malakhov, L. Marcelli, M. Martucci, A. G. Mayorov, W. Menn, M. Merge, V. V. Mikhailov, E. Mocchiutti, A. Monaco, N. Mori, R. Munini, G. Osteria, F. Palma, B. Panico, P. Papini, M. Pearce, P. Picozza, M. Ricci, S. B. Ricciarini, R. Sarkar, V. Scotti, M. Simon, R. Sparvoli, P. Spillantini, Y. I. Stozhkov, A. Vacchi, E. Vannuccini, G. I. Vasilyev, S. A. Voronov, Y. T. Yurkin, G. Zampa, and N. Zampa. Reentrant albedo proton fluxes measured by the PAMELA experiment. *Journal of Geophysical Research (Space Physics)*, 120:3728–3738, May 2015. [32](#)
- [29] T. Zharaspayev, S. Aleksandrin, A. M. Galper, and S. Koldashov. High-energy electron bursts in the inner Earth magnetosphere caused by precipitation from radiation belt. *ArXiv e-prints*, January 2017. [9](#), [33](#), [34](#)
- [30] V. Sgrigna, L. Carota, L. Conti, M. Corsi, A. M. Galper, S. V. Koldashov, A. M. Murashov, P. Picozza, R. Scrimaglio, and L. Stagni. Correlations between earthquakes and anomalous particle bursts from SAMPEX/PET satellite observations. *Journal of Atmospheric and Solar-Terrestrial Physics*, 67:1448–1462, October 2005. [33](#)

- [31] S.A. Voronov. Maria-2 charged particle magnetic spectrometer. *Pribory i Tekhnika Eksperimenta*, 1991. 33
- [32] V. V. Akimov, V. M. Balebanov, A. S. Belousov, I. D. Blokhintsev, G. V. Veselova, M. B. Dobrijan, L. F. Kalinkin, S. V. Kovalenko, V. D. Kozlov, N. G. Leikov, N. K. Mordvov, Y. I. Nagornih, V. E. Nesterov, O. F. Prilutsky, V. L. Prohin, V. G. Rodin, S. R. Tabaldiev, V. N. Chuprov, V. I. Fuks, I. A. Gerasimov, V. S. Ovtchinnikov, V. P. Poluektov, A. V. Serov, V. Y. Tugaenko, L. V. Kurnosova, M. A. Rusakovich, N. P. Topchiev, M. I. Fradkin, I. F. Bugakov, G. M. Gorodinsky, E. I. Chuikin, S. A. Voronov, A. M. Galper, V. A. Grigoriev, M. V. Guzenko, V. G. Kirillov-Ugriumov, S. V. Koldashov, M. G. Korotkov, B. I. Luchkov, A. A. Moiseev, Yu. V. Ozerov, A. V. Popov, V. A. Rud'ko, M. F. Runtsso, B. Yu. Chesnokov, B. Agrinier, A. Bouere, M. Gros, J. P. Leray, A. Leconte, P. Masse, B. Mougine, P. Keirle, J. Cretolle, J. Paul, A. Raviart, B. Parlier, M. Poiviller, C. Hugot, F. Soroka, G. Serra, A. R. Bazer-Bachi, C. Doulade, J. Ducros, G. Vedrenne, F. Cotin, Y. M. Lavigne, P. Mandrou, E. Orsal, M. Avignon, J. Durand, J. Joli, F. Gardon, J. Mouli, M. Nobileau, and D. Fournier. The gamma-ray telescope gamma-1. *Space Science Reviews*, 49(1):111–124, Jan 1989. 33
- [33] D. N. Baker, G. M. Mason, O. Figueroa, G. Colon, J. G. Watzin, and R. M. Aleman. An overview of the Solar, Anomalous, and Magnetospheric Particle Explorer (SAMPEX) mission. *IEEE Transactions on Geoscience and Remote Sensing*, 31:531–541, May 1993 33
- [34] G Hall. Semiconductor particle tracking detectors. *Reports on Progress in Physics*, 57(5):481, 1994 42
- [35] Tsurutani, B.T., Gonzalez, W.D. & Kamide, Y. Magnetic storms. *Surveys in Geophysics* 18, 363–383 (1997). 35
- [36] «<https://lasco-www.nrl.navy.mil/>» 38
- [37] Basu, Su, et al. "Specification and forecasting of scintillations in communication/navigation links: current status and future plans." *Journal of atmospheric and solar-terrestrial physics* 64.16 (2002): 1745-1754. 35
- [38] «Fondazione bruno kessler. <https://www.fbk.eu/it/>» 42
- [39] «STAT 510 Applied Time Series Analysis - Penn State - Eberly College of Science»

- [40] «<https://www.hdfgroup.org/solutions/hdf5/>» 67
- [41] «<https://www.h5py.org/>» 68
- [42] «<https://www.docker.com/>» 64
- [43] «<https://wiki.infn.it/cn/ccr/aai/howto/godiva/home>» 65
- [44] «<https://wiki.infn.it/cn/ccr/aai/home>» 65
- [45] «<https://wiki.infn.it/cn/ccr/aai/howto/authz/home>» 65
- [46] «<https://www.influxdata.com/solutions/metrics-as-a-service-maas/>»

116