



**Politecnico  
di Torino**

**Politecnico di Torino**

Corso di Laurea

A.a. 2025/2026

Sessione di laurea Marzo/Aprile 2026

**Progettazione e sviluppo di  
un'interfaccia immersiva per la  
visualizzazione di dati astronomici in  
Astra Data Navigator**

Relatore:

Prof. Andrea SANNA

Dott. Eugenio TOPA

Candidato:

Stefano ATRIA



## Abstract

Questo lavoro presenta l'integrazione di un'interfaccia utente immersiva nel software di simulazione astronomica Astra Data Navigator, estendendone le funzionalità per creare una potente applicazione di realtà virtuale (VR) su piattaforma PC-VR. Il progetto ha richiesto un significativo aggiornamento tecnologico dell'infrastruttura esistente, con la migrazione a Unity 6 e l'adozione di standard moderni come OpenXR, per garantire la portabilità e l'adozione del nuovo Input System di Unity per una gestione flessibile e astratta dei controlli. Il cuore del lavoro si è concentrato sulla creazione di un'esperienza utente (UX) che fosse al tempo stesso stabile, funzionale e confortevole, affrontando le sfide intrinseche dell'interazione in VR. Per garantire la stabilità visiva, è stato ingegnerizzato un sistema di "camera lock" robusto, basato su coordinate sferiche e quaternioni, che elimina il "jitter" della telecamera durante il puntamento di piccoli oggetti in rapido movimento orbitale. Parallelamente, per mitigare la cinetosi durante la navigazione libera, è stata introdotta una piattaforma virtuale statica come ancoraggio visivo per l'utente. L'interfaccia spazializzata, basata su un sistema di pannelli rotanti accessibili da un menù centrale, è stata progettata per consentire un'interazione fluida con i dati minimizzando i movimenti della testa. Le funzionalità chiave della versione desktop, come il sistema di ricerca e il puntamento a distanza, sono state trasposte in modo nativo per la VR, creando un'esperienza d'uso coerente. Il prototipo finale valida con successo la trasposizione di un complesso strumento di analisi scientifica in un ambiente immersivo. Il risultato non solo offre un'interfaccia più intuitiva, ma migliora drasticamente la percezione della scala e delle relazioni spaziali all'interno di vasti cataloghi stellari, dimostrando il valore aggiunto della realtà virtuale per l'esplorazione dei dati astronomici.



# Ringraziamenti

Desidero innanzitutto esprimere la mia sincera gratitudine al professor Andrea Sanna, relatore di questa tesi, per la disponibilità e l'attenzione dimostrate durante tutto il percorso. La rapidità con cui ha sempre risposto ai miei dubbi, insieme ai consigli e alle indicazioni che mi ha fornito, sono stati fondamentali per orientare correttamente il lavoro svolto. A questo si aggiunge il valore formativo del suo insegnamento: il corso da lui tenuto è stato uno di quelli che ho seguito con maggiore interesse e passione, e ha contribuito in modo significativo alla mia crescita accademica e personale.

Un sentito ringraziamento va anche a Eugenio Topa, che mi ha seguito come referente aziendale durante lo sviluppo del progetto. I suoi suggerimenti e le sue osservazioni sono stati determinanti per impostare correttamente il lavoro e affrontare il progetto con il giusto approccio. Grazie anche a lui ho avuto l'opportunità di entrare in contatto con un ambiente di lavoro estremamente piacevole e stimolante. Desidero quindi ringraziare anche tutti i colleghi in ufficio e il personale di Altec, che con la loro gentilezza, accoglienza e disponibilità hanno reso le giornate in azienda particolarmente piacevoli. La loro curiosità verso il progetto, e la disponibilità a testare l'applicazione e compilare il questionario di valutazione, hanno contribuito concretamente alla realizzazione di questo lavoro.

Un ringraziamento speciale va alla mia famiglia, che mi ha sostenuto lungo tutto il percorso universitario. Grazie a loro ho avuto la possibilità di trasferirmi a Torino e frequentare il Politecnico, vivendo un'esperienza che mi ha fatto crescere profondamente. Nei momenti di dubbio e di incertezza sono stati loro a spingermi ad andare avanti con determinazione. In questi anni siamo cresciuti insieme, e oggi riesco a vedere i miei genitori anche come amici. Col tempo ho imparato a comprendere meglio quanto impegno e quante scelte difficili abbiano affrontato per permettermi di arrivare fin qui, lasciandomi allo stesso tempo la libertà di fare le mie. Un grazie va anche a mia sorella e ai miei cugini, con cui ho condiviso momenti importanti della mia crescita. Nel tempo abbiamo imparato a volerci bene e a starci vicini, costruendo un legame che va oltre quello familiare. Per me sono tra le persone che mi conoscono meglio in assoluto, i confidenti con cui posso essere davvero me stesso e con cui ho sempre potuto condividere pensieri, dubbi e

momenti importanti della mia vita.

Ringrazio poi i miei amici che hanno reso Torino una seconda casa: Carla, Mari, Gabriele, Peppe, Andrea, Alessio, Michele, Saverio, Salvo e tutti gli altri del campus e dintorni. Con loro ho trovato una seconda famiglia, fatta di persone che mi hanno sempre fatto sentire apprezzato e accolto. Nei momenti difficili sono stati un sostegno prezioso, mentre nei momenti più leggeri sono stati la compagnia perfetta per ridere, divertirsi o semplicemente stare insieme. Con loro ho imparato anche a muovermi in una città nuova e ad affrontare insieme le sfide della vita universitaria.

Un pensiero va anche agli amici di Palermo, Vittoria, Michele e Giancarlo. Nonostante la distanza e i periodi in cui non riuscivamo a sentirci spesso, ogni volta siamo sempre riusciti a trovare il modo di vederci e ritrovarci, come se il tempo non fosse mai passato. I nostri incontri sono sempre stati occasioni per raccontarci come stiamo cambiando, come stiamo crescendo e come la vita ci stia portando su strade diverse ma ancora intrecciate.

Ringrazio anche tutti gli amici che hanno accompagnato questi anni torinesi, cioè Altea, James, Pablo, Mary, Ramona, Simone, Salah e gli altri, con cui ho condiviso momenti spensierati, risate interminabili e ricordi che porterò sempre con me. La loro presenza ha alleggerito anche i periodi più intensi e faticosi di questo percorso.

Un ringraziamento speciale va a Jordan, l'amorcito mio, che con il suo affetto dolcissimo e la sua personalità vulcanica mi ha dato la spinta per affrontare gli ultimi sforzi prima della laurea, sempre con energia e con il giusto ritmo latino. Grazie anche alla sua famiglia, che fin dal primo momento mi ha fatto sentire accolto come a casa.

Desidero inoltre dedicare una breve ma sincera menzione al dottor Caci, il mio psicoterapeuta, che mi ha accompagnato proprio nei momenti più delicati di questa fase di crescita. Con il suo punto di vista lucido e spesso molto più efficace del mio, mi ha aiutato a guardare le cose con maggiore consapevolezza e ad affrontare le difficoltà con uno sguardo più maturo, anche se dentro di me rimane sempre una parte di bambino.

Infine, un piccolo ringraziamento lo dedico anche a me stesso, per essere arrivato fino alla fine di questo percorso seguendo le mie passioni e ciò che riesce ancora a farmi brillare gli occhi. Questo corso di studi mi ha messo di fronte a difficoltà e complessità che non avevo immaginato, e non sono mancati momenti di esitazione e di dubbio. Proprio per questo il traguardo raggiunto oggi rappresenta non solo la conclusione di un percorso accademico, ma anche uno dei momenti di crescita più importanti della mia vita.



# Indice

<b>Elenco delle figure</b>	IX
<b>1 Introduzione</b>	1
1.1 Contesto . . . . .	1
1.2 Motivazione . . . . .	2
1.3 Obiettivi . . . . .	2
1.4 Struttura della tesi . . . . .	3
<b>2 Stato dell'arte</b>	6
2.1 Classificazione delle applicazioni per la visualizzazione astronomica	6
2.2 Software scientifici basati su dataset reali . . . . .	7
2.2.1 Cosmographia . . . . .	7
2.2.2 CosmoScout VR . . . . .	8
2.2.3 Astra Data Navigator . . . . .	9
2.3 Software divulgativi ed esplorativi . . . . .	10
2.3.1 SpaceEngine . . . . .	10
2.3.2 Celestia . . . . .	11
2.4 Esperienze immersive e paradigmi di interazione spaziale . . . . .	12
2.4.1 Universe Sandbox . . . . .	12
2.4.2 Elite Dangerous . . . . .	13
2.4.3 No Man's Sky . . . . .	14
2.5 Limiti attuali nella visualizzazione astronomica immersiva . . . . .	15
2.5.1 Precisione scientifica e prestazioni . . . . .	15
2.5.2 Gestione della scala e stabilità numerica . . . . .	15
2.5.3 Paradigmi di locomozione . . . . .	16
2.5.4 Interfaccia diegetica e limiti ergonomici . . . . .	16
2.5.5 Gestione del target e controllo dell'orientamento . . . . .	16
2.5.6 Sintesi . . . . .	17
2.6 Posizionamento di Astra Data Navigator . . . . .	17

<b>3</b>	<b>Tecnologie e infrastruttura di sviluppo</b>	<b>18</b>
3.1	Piattaforme hardware per la realtà virtuale . . . . .	18
3.1.1	Meta Quest 2 e Meta Quest 3 . . . . .	18
3.1.2	Varjo e vincolo verso OpenXR . . . . .	19
3.2	Motore di sviluppo . . . . .	20
3.2.1	Unity 6.0 . . . . .	21
3.2.2	TextMeshPro e leggibilità della UI in VR . . . . .	21
3.2.3	Input System . . . . .	22
3.3	Standard e framework XR . . . . .	22
3.3.1	OpenXR . . . . .	23
3.3.2	XR Interaction Toolkit . . . . .	23
3.3.3	XR Hands . . . . .	24
3.4	Librerie e sistemi di supporto . . . . .	25
3.4.1	SPICE Toolkit . . . . .	25
3.4.2	Space Graphics Toolkit . . . . .	25
3.5	Sintesi . . . . .	26
<b>4</b>	<b>Implementazione dell'interfaccia immersiva</b>	<b>27</b>
4.1	Architettura generale del sistema immersivo . . . . .	28
4.1.1	Unicità della sorgente di trasformazione: AdnCameraControls come “camera authority” . . . . .	28
4.1.2	Target unico e propagazione event-driven . . . . .	29
4.1.3	Producer/Consumer tra LateUpdate e Update . . . . .	29
4.1.4	Stabilizzazione in VR: posizione target “stabile” e calcolo geometrico dedicato . . . . .	29
4.1.5	Implicazioni sulla locomozione: movimento libero vs orbitale	30
4.1.6	Sintesi . . . . .	30
4.2	Configurazione dell'ambiente XR e gestione dell'input . . . . .	30
4.2.1	Adozione di OpenXR . . . . .	31
4.2.2	Architettura dell'Input System . . . . .	31
4.2.3	Action Map “Player” . . . . .	32
4.2.4	Separazione tra input e logica . . . . .	33
4.2.5	Considerazioni progettuali . . . . .	33
4.3	Sistema di locomozione e gestione della scala . . . . .	34
4.3.1	Obiettivi progettuali della locomozione . . . . .	34
4.3.2	Movimento libero in sei gradi di libertà (6DoF controllato) .	34
4.3.3	Gestione della velocità e adattamento alla scala . . . . .	35
4.3.4	Smorzamento del movimento e stabilità percettiva . . . . .	36
4.3.5	Interazione con eventi di rivoluzione e rotazione . . . . .	36
4.3.6	Considerazioni finali . . . . .	36
4.4	Sistema di puntamento e selezione del target . . . . .	37

4.4.1	Puntamento in modalità VR . . . . .	37
4.4.2	Puntamento in modalità desktop . . . . .	38
4.4.3	Interazione con LookAt e Warp . . . . .	38
4.4.4	Coerenza architetturale . . . . .	38
4.5	Sistema di stabilizzazione della camera . . . . .	39
4.5.1	Definizione del problema percettivo . . . . .	39
4.5.2	Floating origin e posizione target stabile in XR . . . . .	39
4.5.3	Separazione temporale: produzione e consumo della richiesta . . . . .	40
4.5.4	Calcolo della rotazione con quaternioni . . . . .	40
4.5.5	Accumulo e damping . . . . .	41
4.5.6	Locomozione orbitale come strategia di stabilità . . . . .	41
4.5.7	Limiti residui . . . . .	41
4.5.8	Considerazioni finali . . . . .	42
4.6	Interfaccia utente spazializzata e sistema di menu orbitale . . . . .	42
4.6.1	Architettura generale del sistema di menu . . . . .	42
4.6.2	Rotazione orbitale dei pannelli . . . . .	43
4.6.3	Navigazione delle opzioni interne . . . . .	43
4.6.4	Descrizione dei pannelli interagibili . . . . .	44
4.6.5	Separazione tra livello informativo e livello simulativo . . . . .	45
4.6.6	Estendibilità del sistema . . . . .	45
<b>5</b>	<b>Valutazione e test sperimentali</b> . . . . .	<b>46</b>
5.1	Metodologia dei test con utenti . . . . .	46
5.1.1	Obiettivi della valutazione . . . . .	46
5.1.2	Contesto di prova . . . . .	47
5.1.3	Profilo del campione . . . . .	47
5.1.4	Procedura sperimentale . . . . .	48
5.2	Analisi quantitativa dei questionari . . . . .	49
5.2.1	Profilo del campione . . . . .	49
5.2.2	Intuitività dell'interfaccia . . . . .	49
5.2.3	Navigazione e controllo dei movimenti . . . . .	50
5.2.4	Comfort e tollerabilità dell'esperienza VR . . . . .	51
5.2.5	Valutazione complessiva e percezione della VR come strumento divulgativo . . . . .	52
5.3	Analisi qualitativa e interpretazione critica . . . . .	52
5.3.1	Usabilità e curva di apprendimento . . . . .	53
5.3.2	Comfort e stabilità dell'esperienza immersiva . . . . .	53
5.3.3	Coinvolgimento e valore esperienziale . . . . .	54
5.4	Test sulle prestazioni e stabilità del sistema . . . . .	54
5.4.1	Prestazioni durante la simulazione e la navigazione . . . . .	54
5.4.2	Stress test dell'interfaccia e del sistema temporale . . . . .	56

5.4.3	Test di risposta agli input . . . . .	58
5.5	Sintesi finale della valutazione . . . . .	60
<b>6</b>	<b>Conclusioni, risultati e sviluppi futuri</b>	<b>62</b>
6.1	Valutazione dei risultati dell'esperienza immersiva . . . . .	62
6.1.1	Comfort e sostenibilità dell'esperienza . . . . .	62
6.1.2	Intuitività dell'interazione e qualità dell'esperienza immersiva	63
6.2	Limitazioni tecniche del sistema . . . . .	64
6.2.1	Visualizzazione e interazione con oggetti di piccole dimensioni	64
6.2.2	Stabilità del sistema di orbitamento della camera . . . . .	64
6.3	Sviluppi futuri . . . . .	65
6.3.1	Estensione dei dataset e delle funzionalità di esplorazione . .	65
6.3.2	Miglioramento della rappresentazione grafica . . . . .	66
6.3.3	Espansione dei fenomeni astronomici rappresentati . . . . .	66
6.3.4	Integrazione con tecnologie di realtà aumentata . . . . .	67
6.4	Considerazioni finali . . . . .	68
	<b>Bibliografia</b>	<b>70</b>

# Elenco delle figure

1.1	<i>Modello di Gaia al Salone Aeronautico di Parigi 2013.</i>	5
1.2	<i>Mappa del cielo in luce visibile e alcune scale colorimetriche, ricavate dall'attività di Gaia.</i>	5
2.1	<i>Cosmographia: Diverse schermate di visualizzazione dell'orientamento dei sistemi di riferimento di alcuni satelliti artificiali e tracciamento delle traiettorie.</i>	7
2.2	<i>Cosmoscout: Grafica informativa per l'apprendimento di informazioni e curiosità sulla Terra.</i>	8
2.3	<i>Astra Data Navigator: Visualizzazione con tracciamento dell'orbita del satellite artificiale TGO, con pannello informativo.</i>	9
2.4	<i>Space Engine: Selezione di oggetti appartenenti al Sistema Solare per la visualizzazione delle relative informazioni.</i>	10
2.5	<i>Celestia: Vista dell'interfaccia utente del software nella versione per dispositivi mobile.</i>	11
2.6	<i>Universe Sandbox: L'utente utilizza un laser per distruggere parte dell'anello di detriti attorno a un pianeta generato proceduralmente.</i>	12
2.7	<i>Elite Dangerous VR: Interfaccia diegetica distribuita radialmente attorno al giocatore, che richiede una torsione laterale del capo per l'accesso completo ai pannelli informativi.</i>	13
2.8	<i>No Man's Sky VR: Scena di combattimento spaziale con interfaccia HUD sovrapposta, ambientata nei pressi di un pianeta con anelli.</i>	14
3.1	<i>Meta Quest 2/3 e Varjo XR-4: Visori di realtà virtuale utilizzati come dispositivi di visualizzazione immersiva di riferimento durante lo sviluppo e i test dell'interfaccia VR del software.</i>	19
3.2	<i>Unity 6.0: Interfaccia di sviluppo dell'applicazione.</i>	20
3.3	<i>XR Interaction Toolkit: Scena di esempio per testare gli strumenti per lo sviluppo di ambienti immersivi offerti dal toolkit.</i>	23
3.4	<i>XR Hands: Interfaccia per il rilevamento in tempo reale delle "gesture".</i>	24

3.5	<i>Space Graphics Toolkit: Varietà di pianeti e oggetti generabili tramite l'utilizzo del Toolkit. . . . .</i>	26
4.1	<i>Diagramma architetturale del sistema di input e navigazione in realtà virtuale di Astra Data Navigator, che mostra il flusso delle informazioni tra dispositivi XR, moduli di gestione della camera, sistema di input e componenti di interazione come il VR pointer. . . . .</i>	32
4.2	<i>Vista della piattaforma col menu aperto e orientamento della piattaforma rispetto agli assi cartesiani di movimento (rosso: X, verde: Y, blu: Z). . . . .</i>	35
4.3	<i>Vista dell'interfaccia utente isolata dal contesto immersivo. . . . .</i>	43
5.1	<i>Distribuzione di età, genere e frequenza di utilizzo del PC e della VR fra il campione. . . . .</i>	48
5.2	<i>Valutazioni da parte del campione sull'intuitività dell'interfaccia utente e sul senso di controllo sui movimenti e di orientamento nello spazio. . . . .</i>	50
5.3	<i>Grafici a torta sui feedback ricevuti riguardo eventuali sensazioni di nausea o fastidio provate durante l'esperienza. . . . .</i>	51
5.4	<i>Valutazioni dell'esperienza con Astra Data Navigator VR e dell'efficacia della VR per la divulgazione in ambito astronomico. . . . .</i>	52

# Capitolo 1

## Introduzione

### 1.1 Contesto

Negli ultimi decenni l'astronomia osservativa ha prodotto una quantità crescente di dati grazie a missioni spaziali dedicate alla misura precisa delle proprietà delle stelle. Cataloghi astrometrici come quelli derivati dalle missioni Hipparcos e Gaia (Fig. 1.1) contengono informazioni dettagliate sulla posizione, distanza e moto di un numero estremamente elevato di oggetti celesti, consentendo la ricostruzione tridimensionale della struttura della Via Lattea.

La disponibilità di tali dataset ha reso sempre più rilevante il ruolo della visualizzazione scientifica, che permette di rappresentare e interpretare grandi quantità di dati complessi attraverso strumenti interattivi. Nel caso dei cataloghi stellari, la visualizzazione tridimensionale rappresenta un elemento particolarmente importante, poiché consente di comprendere la distribuzione spaziale degli oggetti astronomici e le relazioni tra le diverse strutture galattiche.

Tradizionalmente, l'analisi di tali dataset avviene tramite strumenti bidimensionali basati su monitor, mouse e tastiera, attraverso la consultazione di cataloghi astronomici, grafici, immagini osservative e rappresentazioni derivate dai dati (Fig. 1.2). Sebbene tali strumenti risultino efficaci per l'analisi numerica e per la manipolazione di dataset complessi, essi presentano alcune limitazioni quando si tratta di rappresentare e comprendere strutture tridimensionali su larga scala.

Negli ultimi anni la realtà virtuale (VR) è stata progressivamente adottata anche nel campo della visualizzazione scientifica. La possibilità di rappresentare i dati all'interno di un ambiente immersivo consente all'utente di esplorare direttamente lo spazio tridimensionale, migliorando la percezione delle relazioni spaziali tra gli oggetti e facilitando l'interpretazione di dataset complessi.

In questo contesto si colloca Astra Data Navigator, un software sviluppato da ALTEC per la simulazione e la visualizzazione interattiva di corpi del sistema solare

e di cataloghi stellari. L'applicazione consente di navigare nello spazio simulato, selezionare oggetti celesti e consultare informazioni relative alle loro proprietà astronomiche attraverso strumenti di interazione grafica.

## 1.2 Motivazione

Nonostante l'efficacia degli strumenti di visualizzazione tridimensionale su desktop, l'interazione con grandi dataset astronomici tramite interfacce tradizionali presenta alcune limitazioni. In particolare, la navigazione nello spazio tridimensionale e la selezione di oggetti celesti risultano meno intuitive quando l'utente osserva la scena attraverso una proiezione bidimensionale su schermo.

La realtà virtuale offre la possibilità di affrontare queste limitazioni introducendo un ambiente immersivo in cui l'utente può esplorare direttamente la scena tridimensionale e interagire con gli oggetti presenti nello spazio simulato. Tuttavia, il passaggio da un'interfaccia desktop a un sistema immersivo non rappresenta un semplice adattamento tecnologico, ma richiede una riprogettazione dei paradigmi di interazione.

Le modalità di navigazione, selezione e gestione dell'interfaccia devono essere ridefinite per garantire stabilità visiva, chiarezza dell'interazione e comfort durante l'utilizzo del sistema. In particolare, la visualizzazione di oggetti astronomici molto distanti o di piccole dimensioni introduce problematiche specifiche legate alla precisione della selezione, alla gestione della telecamera e alla leggibilità delle informazioni visualizzate.

Queste considerazioni hanno motivato lo sviluppo di una modalità di interazione immersiva per Astra Data Navigator, con l'obiettivo di esplorare le potenzialità della realtà virtuale come strumento per la visualizzazione e l'esplorazione di dataset astronomici tridimensionali.

## 1.3 Obiettivi

L'obiettivo principale di questa tesi è la progettazione e l'implementazione di un'interfaccia immersiva in realtà virtuale per il software Astra Data Navigator, finalizzata all'esplorazione interattiva di cataloghi stellari e oggetti astronomici.

Il lavoro si è concentrato sull'integrazione delle tecnologie di realtà virtuale all'interno dell'architettura esistente dell'applicazione, utilizzando il motore di sviluppo Unity e lo standard OpenXR per la gestione dei dispositivi XR e dei sistemi di input.

Per raggiungere questo obiettivo sono state affrontate diverse attività progettuali e di sviluppo, tra cui:

- integrazione del supporto alla realtà virtuale nell'infrastruttura software esistente;
- progettazione di un sistema di navigazione tridimensionale che consenta all'utente di muoversi nello spazio simulato mantenendo stabilità visiva e comfort;
- sviluppo di un sistema di selezione e puntamento degli oggetti astronomici basato su tecniche di raycasting;
- progettazione di un'interfaccia utente spazializzata composta da pannelli interattivi e strumenti di controllo dei parametri di visualizzazione;
- adattamento alla realtà virtuale di funzionalità già presenti nella versione desktop dell'applicazione, tra cui i sistemi di ricerca, puntamento automatico e navigazione verso oggetti selezionati.

Il contributo principale di questo lavoro consiste nella progettazione di un sistema di interazione immersiva per l'esplorazione di dati astronomici, che ridefinisce le modalità di navigazione e interazione con i dataset stellari all'interno di un ambiente tridimensionale immersivo.

Il prototipo sviluppato dimostra la fattibilità dell'integrazione di interfacce immersive all'interno di strumenti di visualizzazione astronomica, evidenziando il potenziale della realtà virtuale per migliorare la comprensione delle relazioni spaziali tra gli oggetti celesti e facilitare l'esplorazione di dataset scientifici complessi.

## 1.4 Struttura della tesi

Il **Capitolo 2** presenta lo stato dell'arte relativo ai software per la visualizzazione di cataloghi stellari e per l'esplorazione di ambienti astronomici digitali. Vengono analizzate diverse categorie di applicazioni, comprendenti strumenti scientifici basati su dataset reali, software divulgativi orientati alla navigazione astronomica e applicazioni immersive o ludiche che offrono spunti rilevanti per lo studio dei paradigmi di interazione spaziale.

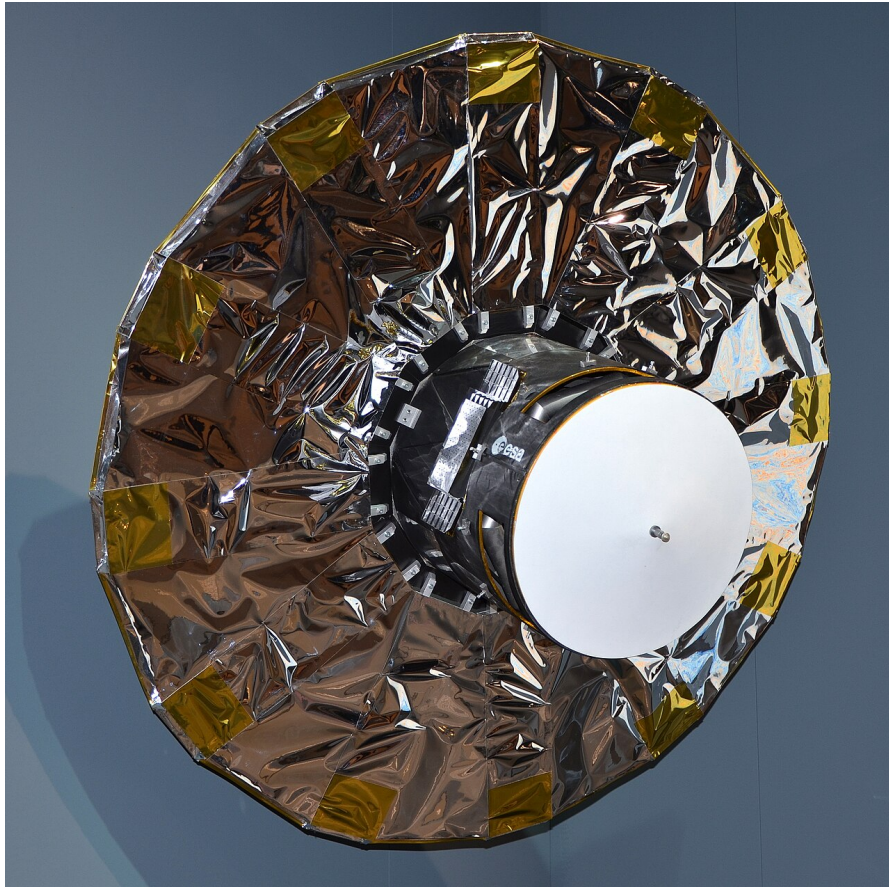
Il **Capitolo 3** descrive le principali tecnologie utilizzate durante lo sviluppo dell'applicazione. In particolare vengono presentate le piattaforme hardware di realtà virtuale impiegate nei test e nello sviluppo, nonché gli strumenti software utilizzati per l'implementazione del sistema, tra cui il motore grafico Unity, lo standard OpenXR e librerie di supporto come Space Graphics Toolkit.

Il **Capitolo 4** illustra lo stato attuale dell'applicazione e l'implementazione dell'interfaccia immersiva. Vengono descritti l'integrazione di OpenXR e del nuovo Input System di Unity e le principali scelte architettoniche adottate per adattare

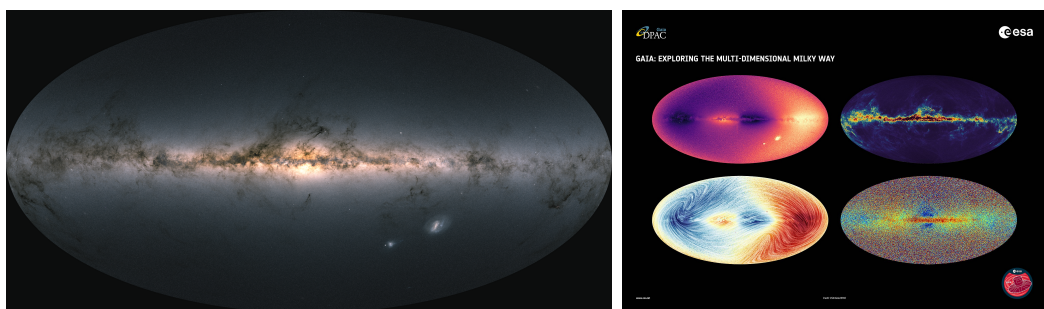
l'applicazione a un contesto immersivo. Particolare attenzione è dedicata agli adattamenti necessari per trasporre in realtà virtuale il paradigma di esplorazione originariamente progettato per la versione desktop di Astra Data Navigator.

Il **Capitolo 5** presenta i risultati della fase di valutazione sperimentale. In questo capitolo vengono descritti la metodologia dei test con gli utenti, l'analisi dei questionari di valutazione dell'esperienza e le considerazioni emerse dall'osservazione dell'utilizzo del sistema. Vengono inoltre riportati i risultati dei test di prestazione e stabilità dell'applicazione.

Infine, il **Capitolo 6** riassume i risultati del lavoro svolto e discute i principali limiti emersi durante lo sviluppo, proponendo possibili sviluppi futuri del sistema. Il capitolo include inoltre alcune considerazioni sull'efficacia della realtà virtuale come strumento per la visualizzazione e la divulgazione di dati astronomici.



**Figura 1.1:** *Modello di Gaia al Salone Aeronautico di Parigi 2013.*



**Figura 1.2:** *Mappa del cielo in luce visibile e alcune scale colorimetriche, ricavate dall'attività di Gaia.*

# Capitolo 2

## Stato dell'arte

La visualizzazione di ambienti astronomici digitali si colloca all'intersezione tra accuratezza scientifica, capacità computazionale e progettazione dell'interazione. Nel tempo sono state sviluppate applicazioni con finalità differenti, dall'analisi rigorosa dei dati alla divulgazione, fino alle esperienze immersive ludiche.

Il presente capitolo analizza tali soluzioni al fine di inquadrarne approcci, limiti e paradigmi di interazione, individuando il contesto teorico e applicativo in cui si inserisce il lavoro di tesi.

### 2.1 Classificazione delle applicazioni per la visualizzazione astronomica

Le applicazioni per la rappresentazione e l'esplorazione di ambienti astronomici possono essere suddivise in categorie distinte in base a obiettivi, livello di accuratezza scientifica e paradigma di interazione.

Una prima categoria comprende gli strumenti scientifici basati su dataset reali, orientati alla precisione delle effemeridi e alla coerenza con modelli fisici validati. In questi sistemi l'accuratezza dei dati costituisce il requisito principale, mentre l'esperienza immersiva e la resa grafica assumono un ruolo secondario. *Astra Data Navigator* rientra in questa categoria, in quanto fondato su cataloghi astronomici reali e progettato per garantire coerenza geometrica e correttezza nella rappresentazione dei corpi celesti.

Una seconda categoria include i software divulgativi ed esplorativi, che privilegiano l'accessibilità e l'impatto visivo. Pur potendo integrare dati osservativi, tali applicazioni non sono progettate per contesti di analisi mission-critical, ma per la navigazione libera e la comunicazione scientifica.

Infine, si possono considerare le esperienze immersive a carattere ludico, che, pur non essendo finalizzate alla visualizzazione dell'universo reale, rappresentano

casi rilevanti per l'analisi dei paradigmi di interazione spaziale in realtà virtuale. In tali sistemi la componente esperienziale e diegetica assume un ruolo centrale, offrendo spunti utili per la progettazione dell'interazione immersiva.

Questa classificazione consente di analizzare le soluzioni esistenti secondo criteri comparabili, evidenziandone approcci progettuali, punti di forza e limitazioni rispetto agli obiettivi del presente lavoro.

## 2.2 Software scientifici basati su dataset reali

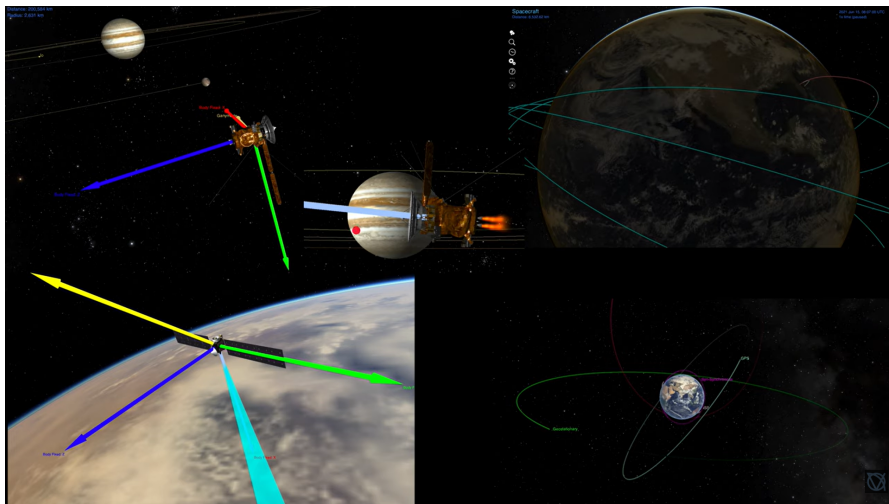
Gli strumenti scientifici per la visualizzazione astronomica si caratterizzano per l'utilizzo di dataset osservativi ufficiali, modelli matematici validati e sistemi di riferimento coerenti con le missioni spaziali. In tali applicazioni l'obiettivo primario è la correttezza fisica della simulazione, piuttosto che l'esperienza immersiva.

### 2.2.1 Cosmographia

Cosmographia è un'applicazione tridimensionale orientata alla visualizzazione di missioni spaziali e corpi del sistema solare a partire da dataset SPICE [1].

Il sistema consente di:

- caricare kernel relativi a pianeti, satelliti e spacecraft;
- visualizzare traiettorie e orientamenti (Fig. 2.1);
- esplorare scenari missione in modo interattivo.



**Figura 2.1:** Cosmographia: *Diverse schermate di visualizzazione dell'orientamento dei sistemi di riferimento di alcuni satelliti artificiali e tracciamento delle traiettorie.*

L'approccio è fortemente data-driven: la scena è costruita a partire da configurazioni esplicite e dataset ufficiali, garantendo coerenza con le effemeridi reali. Tale caratteristica rende lo strumento particolarmente adatto a contesti di analisi e supporto scientifico.

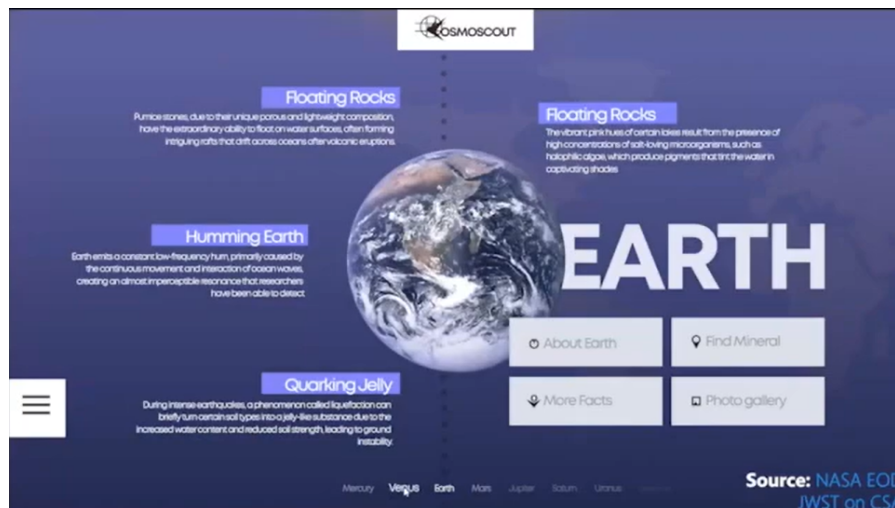
Dal punto di vista dell'interazione, tuttavia, Cosmographia mantiene un paradigma prevalentemente desktop. Il supporto alla realtà virtuale non costituisce l'asse progettuale principale, e l'esperienza immersiva risulta subordinata alla precisione del dato.

### 2.2.2 CosmoScout VR

CosmoScout VR (Fig. 2.2) rappresenta un'evoluzione verso l'integrazione tra accuratezza scientifica e ambiente immersivo. Il sistema è progettato per supportare dataset planetari di grandi dimensioni e ambienti di realtà virtuale fin dalla sua concezione [2].

Tra le sue caratteristiche principali si evidenziano:

- architettura modulare basata su plugin;
- integrazione di dataset topografici ad alta risoluzione;
- supporto a traiettorie basate su SPICE;
- strumenti di misura e analisi del terreno.



**Figura 2.2:** Cosmoscout: Grafica informativa per l'apprendimento di informazioni e curiosità sulla Terra.

L'approccio modulare consente un'elevata estendibilità, rendendo il sistema adatto a contesti di ricerca e visualizzazione avanzata. Tuttavia, la complessità dell'infrastruttura e l'orientamento verso ambienti ad alte prestazioni possono limitarne l'accessibilità in scenari più leggeri o orientati alla semplicità d'uso.

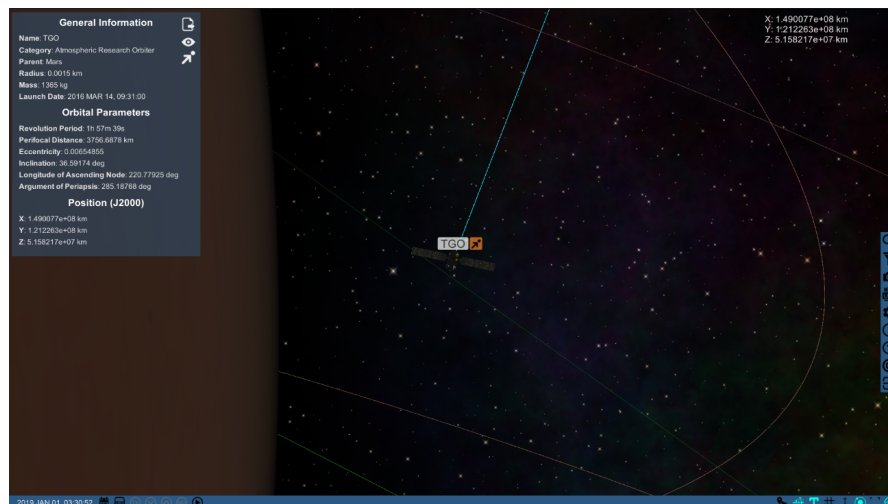
Nel complesso, i software scientifici analizzati privilegiano l'accuratezza e la coerenza fisica, ma presentano limiti in termini di ergonomia dell'interazione e immediatezza dell'esperienza immersiva.

### 2.2.3 Astra Data Navigator

Astra Data Navigator è un'applicazione tridimensionale per la visualizzazione di dati astronomici basata su dataset osservativi reali. Il sistema integra cataloghi stellari quali Hipparcos e Gaia, contenenti informazioni astrometriche su un numero estremamente elevato di stelle, e consente la rappresentazione spaziale dei corpi celesti nel sistema solare mediante modelli geometrici coerenti con dati scientifici disponibili.

La posizione dei corpi può essere determinata tramite modelli fisici e sistemi di effemeridi, basati sul toolkit SPICE, descritto più approfonditamente nel Capitolo 3, permettendo la ricostruzione della configurazione spaziale in funzione del tempo simulato. L'architettura del software è orientata alla coerenza geometrica e alla gestione di scale astronomiche estese, includendo meccanismi di navigazione libera, selezione di oggetti e funzionalità di warp verso target specifici.

Dal punto di vista dell'interazione, la versione originaria dell'applicazione è progettata per un contesto desktop (Fig. 2.3), con controlli basati su mouse e



**Figura 2.3:** Astra Data Navigator: *Visualizzazione con tracciamento dell'orbita del satellite artificiale TGO, con pannello informativo.*

tastiera e interfaccia a pannelli bidimensionali sovrapposti alla scena tridimensionale. Pur garantendo solidità nella gestione del dato e nella rappresentazione spaziale, l'applicazione non nasce come sistema immersivo nativo.

In questo senso, Astra Data Navigator si configura come uno strumento scientificamente fondato e strutturalmente solido, ma non originariamente concepito per un paradigma di interazione VR. Tale caratteristica costituisce il presupposto per l'estensione progettuale sviluppata nel presente lavoro.

## 2.3 Software divulgativi ed esplorativi

Accanto ai software strettamente scientifici, esiste una categoria di applicazioni orientate principalmente all'esplorazione e alla divulgazione astronomica. In questi sistemi l'obiettivo prioritario non è l'analisi quantitativa dei dati, bensì la fruizione visiva dell'universo e la navigazione libera tra corpi celesti su scale spaziali estremamente ampie. La precisione fisica può essere mantenuta entro limiti ragionevoli, ma spesso viene affiancata o sostituita da componenti procedurali, artistiche o semplificate.

### 2.3.1 SpaceEngine

SpaceEngine è un simulatore spaziale che consente l'esplorazione continua dell'universo osservabile combinando dati astronomici reali con generazione procedurale. Le stelle e gli oggetti noti sono modellati sulla base di cataloghi scientifici (Fig. 2.4), mentre regioni non osservate vengono generate algoritmicamente secondo modelli astrofisici plausibili.



**Figura 2.4:** Space Engine: *Selezione di oggetti appartenenti al Sistema Solare per la visualizzazione delle relative informazioni.*

L'utente può muoversi liberamente tra scale differenti, da superfici planetarie a galassie, senza soluzione di continuità, modificando la velocità temporale e spaziale. È presente anche un supporto VR che permette la navigazione immersiva, prevalentemente orientata alla percezione spettacolare dell'ambiente cosmico [3].

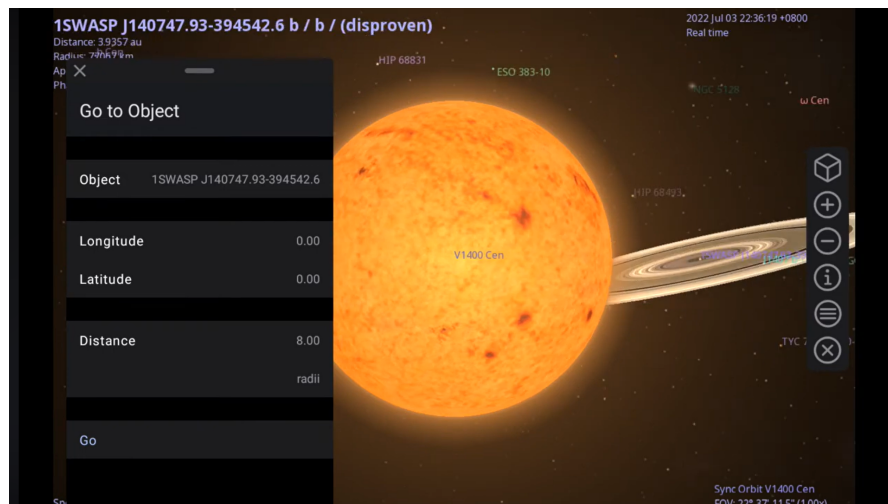
Dal punto di vista dell'interazione, il paradigma è quello della libertà totale di movimento, con controlli che simulano inerzia e dinamica di volo. Tuttavia, l'interfaccia rimane principalmente orientata all'esplorazione visiva e non fornisce strumenti analitici strutturati. La componente immersiva privilegia l'effetto di presenza rispetto alla gestione rigorosa del dato.

SpaceEngine rappresenta quindi un esempio di equilibrio tra divulgazione scientifica e spettacolarizzazione, ma non si configura come strumento di analisi astronomica in senso stretto.

### 2.3.2 Celestia

Celestia è un planetario digitale multiplatforma che consente l'esplorazione tridimensionale del sistema solare e di cataloghi stellari estesi. A differenza di SpaceEngine, l'accento è posto maggiormente sulla consultazione di oggetti astronomici reali e sulla possibilità di navigare tra essi con continuità.

Il sistema permette zoom esponenziali e movimenti su scale molto ampie, offrendo un'esperienza di esplorazione relativamente fedele ai dati disponibili. È inoltre estendibile tramite script e moduli aggiuntivi, favorendo un utilizzo anche in ambito educativo. [4]



**Figura 2.5:** Celestia: Vista dell'interfaccia utente del software nella versione per dispositivi mobile.

L'interazione, tuttavia, è prevalentemente desktop-oriented, basata su mouse e tastiera, e l'eventuale supporto a configurazioni immersive non rappresenta il fulcro dell'architettura del software. L'interfaccia resta tradizionale, con menu e controlli bidimensionali (Fig. 2.5).

Celestia costituisce un esempio significativo di software divulgativo basato su dati astronomici reali, ma non integra in maniera nativa paradigmi di interazione VR immersiva strutturata.

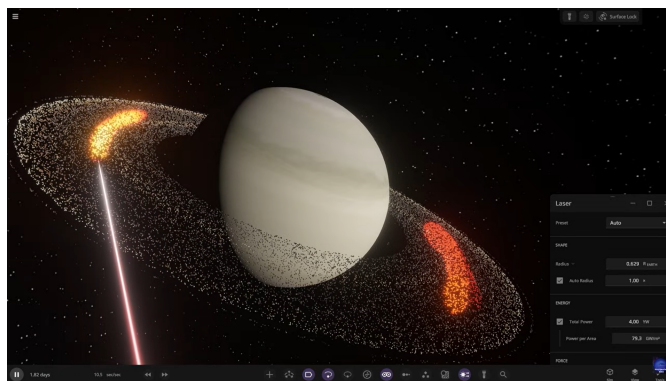
## 2.4 Esperienze immersive e paradigmi di interazione spaziale

Accanto ai software scientifici e divulgativi esiste una categoria di applicazioni che, pur non essendo finalizzate alla rappresentazione rigorosa dell'universo reale, costituiscono casi studio rilevanti per l'analisi dei paradigmi di interazione spaziale immersiva. In questi sistemi l'obiettivo primario è l'esperienza, la presenza e l'interazione corporea, più che l'accuratezza dei dati astronomici.

L'analisi di tali applicazioni risulta significativa poiché evidenzia i vantaggi e i limiti dell'interfaccia diegetica in ambienti spaziali virtuali.

### 2.4.1 Universe Sandbox

Universe Sandbox è una simulazione interattiva che permette di manipolare sistemi planetari (Fig. 2.6), stelle e galassie modificandone parametri fisici come massa, velocità, distanza e forza gravitazionale. A differenza dei software precedentemente analizzati, il focus non è l'esplorazione narrativa o la consultazione di cataloghi, bensì la sperimentazione dinamica di fenomeni astrofisici. [5]



**Figura 2.6:** Universe Sandbox: *L'utente utilizza un laser per distruggere parte dell'anello di detriti attorno a un pianeta generato proceduralmente.*

Il supporto alla realtà virtuale consente all'utente di "afferrare" corpi celesti, spostarli nello spazio e alterarne le proprietà tramite controller VR. L'interazione assume una dimensione fisica diretta: i pianeti diventano oggetti manipolabili.

Questo paradigma enfatizza l'immediatezza e l'intuizione, ma introduce alcune criticità:

- l'interfaccia è fortemente centrata sull'azione manuale;
- la precisione del controllo può risultare limitata;
- la manipolazione diretta non è sempre compatibile con contesti di consultazione strutturata.

Universe Sandbox dimostra come la VR possa rendere tangibile la simulazione gravitazionale, ma evidenzia anche la difficoltà di integrare interazioni "fisiche" con strumenti analitici complessi.

## 2.4.2 Elite Dangerous

Elite Dangerous è un simulatore spaziale ambientato in una galassia su scala 1:1 generata proceduralmente. Pur non essendo uno strumento astronomico in senso scientifico, rappresenta uno dei casi più maturi di integrazione VR in un ambiente spaziale.

In modalità realtà virtuale l'utente si trova all'interno della cabina di una navicella spaziale, con un'interfaccia prevalentemente diegetica: i pannelli informativi sono integrati nel cockpit e si attivano tramite lo sguardo o il movimento della testa. [6]

Questo approccio aumenta fortemente il senso di presenza, ma presenta alcune implicazioni:

- l'interazione è mediata da un contesto narrativo (la cabina);
- i menu sono vincolati alla geometria dell'ambiente (Fig. 2.7);



**Figura 2.7:** Elite Dangerous VR: *Interfaccia diegetica distribuita radialmente attorno al giocatore, che richiede una torsione laterale del capo per l'accesso completo ai pannelli informativi.*

- la navigazione non è libera nello spazio, ma subordinata al modello di volo.

Dal punto di vista ergonomico, la VR è ben integrata, ma il paradigma di controllo è legato alla simulazione di un veicolo, non all'osservazione diretta dell'universo. Ciò lo rende poco trasferibile a un contesto di esplorazione astronomica svincolata da un avatar o da un mezzo.

### 2.4.3 No Man's Sky

No Man's Sky rappresenta un esempio significativo di trasformazione del paradigma di controllo tra modalità desktop e modalità VR. In versione tradizionale molte azioni sono eseguite tramite comandi diretti su tastiera o controller. In modalità VR, invece, tali azioni vengono tradotte in gesti corporei e interazioni spazializzate. [7]

Ad esempio, l'estrazione di strumenti avviene tramite un gesto sopra la spalla; l'interazione con interfacce avviene tramite manipolazione diretta di pannelli olografici (Fig. 2.8); diverse operazioni richiedono movimenti fisici continui.

Questo approccio massimizza l'immersione e la coerenza diegetica, ma introduce potenziali criticità:

- aumento dell'affaticamento fisico nelle operazioni ripetitive;
- riduzione dell'efficienza in attività frequenti;
- maggiore complessità cognitiva nell'accesso a funzioni non immediatamente visibili;
- ergonomia non sempre ottimale per sessioni prolungate.



**Figura 2.8:** No Man's Sky VR: *Scena di combattimento spaziale con interfaccia HUD sovrapposta, ambientata nei pressi di un pianeta con anelli.*

Nel contesto della visualizzazione astronomica, tali limiti diventano particolarmente rilevanti: l'interazione analitica con grandi quantità di dati richiede rapidità, precisione e stabilità percettiva, caratteristiche non sempre compatibili con un'interfaccia completamente diegetica.

## **2.5 Limiti attuali nella visualizzazione astronomica immersiva**

Dall'analisi delle categorie precedenti emergono alcune criticità ricorrenti nei software di visualizzazione astronomica, soprattutto quando si tenta di integrare supporto alla realtà virtuale. Tali limiti non riguardano esclusivamente la componente tecnologica, ma investono in modo trasversale aspetti di precisione, ergonomia, gestione della scala e progettazione dell'interazione.

### **2.5.1 Precisione scientifica e prestazioni**

Nei software scientifici basati su dataset reali, la necessità di mantenere accuratezza geometrica e temporale comporta spesso compromessi prestazionali. Il calcolo continuo di effemeridi e trasformazioni spaziali può risultare oneroso in contesti real-time, soprattutto in VR, dove la stabilità del frame rate è requisito essenziale per evitare discomfort visivo.

Al contrario, le applicazioni divulgative o ludiche privilegiano fluidità e resa grafica, accettando modelli semplificati o generazione procedurale non rigorosamente aderente ai dati osservativi. Il risultato è una polarizzazione tra accuratezza e fruibilità.

La coesistenza di entrambe le esigenze rimane una problematica aperta.

### **2.5.2 Gestione della scala e stabilità numerica**

La rappresentazione dell'universo implica ordini di grandezza estremamente eterogenei: dal livello planetario a quello galattico. Questo comporta difficoltà nella gestione della precisione numerica e nella stabilità delle trasformazioni spaziali.

In ambienti VR, tali criticità si amplificano: micro-oscillazioni, jitter o imprecisioni percettive risultano immediatamente evidenti all'utente, compromettendo il senso di presenza. La gestione della floating origin e delle trasformazioni relative diventa quindi elemento centrale.

Molti software non progettati originariamente per la VR mostrano limiti strutturali quando trasposti in un contesto immersivo.

### 2.5.3 Paradigmi di locomozione

Nei software scientifici la navigazione è generalmente libera e diretta, con movimenti scalabili e warp istantanei. Nei software ludici, invece, la locomozione è spesso mediata da veicoli, avatar o meccaniche narrative.

In VR, la locomozione rappresenta uno dei principali fattori di rischio per motion sickness. Movimenti continui su larga scala, accelerazioni improvvise o rotazioni incontrollate possono generare disagio.

La progettazione di un sistema di movimento che consenta esplorazione astronomica su scale estreme mantenendo comfort percettivo costituisce una delle sfide principali non completamente risolte nei sistemi analizzati.

### 2.5.4 Interfaccia diegetica e limiti ergonomici

Le applicazioni ludiche immersive adottano frequentemente interfacce diegetiche, in cui strumenti e menu fanno parte del mondo virtuale. Sebbene questo approccio aumenti la coerenza narrativa e il senso di presenza, presenta alcune limitazioni:

- aumento dell'affaticamento fisico nelle operazioni ripetitive;
- minore efficienza nell'accesso a funzioni frequenti;
- complessità nella gestione di informazioni strutturate;
- difficoltà di integrazione con dataset estesi.

In un contesto di visualizzazione astronomica, dove l'interazione analitica e la consultazione di dati rivestono un ruolo centrale, un'interfaccia completamente diegetica può risultare poco funzionale.

### 2.5.5 Gestione del target e controllo dell'orientamento

Un ulteriore aspetto critico riguarda la selezione e il mantenimento del target di osservazione. Nei software desktop la funzione di "punta e clicca" è semplice e immediata; in VR, invece, il sistema di selezione deve essere ripensato per evitare instabilità visiva o conflitti tra input manuali e rotazioni automatiche.

Il mantenimento dell'orientamento verso un oggetto in movimento introduce problematiche geometriche e temporali che raramente vengono affrontate in modo sistematico nelle applicazioni esistenti.

### **2.5.6 Sintesi**

L'analisi condotta evidenzia una frammentazione tra:

- applicazioni scientifiche accurate ma poco immersive,
- applicazioni divulgative visivamente efficaci ma non orientate all'analisi,
- esperienze ludiche immersive ma ergonomicamente discutibili per uso analitico.

La progettazione di un ambiente che integri precisione scientifica, stabilità percettiva e interazione ergonomica rimane quindi un problema aperto. Su questo spazio di intersezione si colloca il lavoro sviluppato nella presente tesi, come verrà approfondito nel paragrafo seguente.

## **2.6 Posizionamento di Astra Data Navigator**

Astra Data Navigator si colloca in una posizione intermedia tra applicazione scientifica e ambiente esplorativo basato su dati reali. L'utilizzo di cataloghi astronomici e la possibilità di integrare sistemi di effemeridi garantiscono coerenza con modelli osservativi, mentre le funzionalità di navigazione libera, ricerca e warp ne favoriscono la fruizione esplorativa.

Tuttavia, nella sua configurazione originaria desktop, l'applicazione non integra un paradigma di interazione immersiva progettato specificamente per la realtà virtuale. Le logiche di controllo, selezione e orientamento risultano pensate per input bidimensionali tradizionali e non per un contesto tridimensionale embodied.

Si individua pertanto uno spazio progettuale non ancora esplorato: l'estensione di un sistema di visualizzazione astronomica basato su dati reali verso un ambiente VR coerente, stabile e ergonomicamente sostenibile. Su tale ambito si concentra il contributo della presente tesi.

## Capitolo 3

# Tecnologie e infrastruttura di sviluppo

La realizzazione dell'estensione in realtà virtuale di Astra Data Navigator si fonda su un insieme integrato di piattaforme hardware, standard di interoperabilità e strumenti software. In questo capitolo vengono presentate le tecnologie effettivamente impiegate durante lo sviluppo, con particolare attenzione agli elementi che hanno influenzato le scelte architettoniche, le modalità di interazione e i vincoli prestazionali del sistema.

### 3.1 Piattaforme hardware per la realtà virtuale

La scelta delle piattaforme hardware ha influenzato in maniera significativa l'architettura e le scelte progettuali adottate nello sviluppo della versione VR di Astra Data Navigator. Non si tratta soltanto di dispositivi di visualizzazione, ma di sistemi complessi che determinano modalità di tracking, gestione degli input, vincoli prestazionali e compatibilità con gli standard XR.

Nel presente lavoro sono state considerate principalmente tre famiglie di dispositivi: la linea Meta Quest (in particolare Quest 2 e Quest 3), i visori Varjo (Fig. 3.1) e, in maniera più indiretta, le piattaforme compatibili con lo standard OpenXR.

#### 3.1.1 Meta Quest 2 e Meta Quest 3

Meta Quest 2 e Meta Quest 3 hanno costituito le principali piattaforme di sviluppo e test dell'estensione VR.

Entrambi i dispositivi appartengono alla categoria dei visori stand-alone con tracking inside-out, dotati di controller con analogici, trigger e pulsanti principali. Questo ha imposto fin dalle prime fasi una progettazione dell'interazione basata su

input continui (analogici) e discreti (pressioni), evitando soluzioni che richiedessero periferiche esterne o sistemi di tracciamento aggiuntivi.

Il passaggio da Quest 2 a Quest 3 non ha comportato modifiche sostanziali dal punto di vista delle funzionalità sfruttate all'interno del progetto. Caratteristiche quali il passthrough a colori o le funzionalità di mixed reality non sono state integrate nel flusso applicativo, poiché l'obiettivo rimane una simulazione immersiva completamente virtuale. [8]

La differenza più rilevante si è manifestata invece sul piano prestazionale. Con l'avanzare dello sviluppo e l'incremento della complessità della scena (numero di oggetti, gestione dei pannelli UI, sistemi di locking e calcoli dinamici), la maggiore potenza computazionale di Quest 3 ha consentito una gestione più stabile del framerate e una riduzione delle latenze percepite. Questo aspetto è risultato particolarmente importante in un contesto di navigazione astronomica, dove movimenti ad ampia scala e rotazioni continue possono amplificare eventuali instabilità percettive.

In altre parole, l'evoluzione hardware non ha modificato il paradigma di interazione, ma ha migliorato la robustezza esecutiva del sistema in condizioni operative più complesse.

### 3.1.2 Varjo e vincolo verso OpenXR

L'utilizzo di dispositivi Varjo, orientati a contesti professionali ad alta fedeltà visiva, ha introdotto un vincolo tecnologico rilevante: l'adozione obbligata dello standard OpenXR come layer di astrazione.

Questo ha escluso soluzioni proprietarie legate a singoli vendor e ha imposto un'architettura compatibile con uno standard aperto e multiplatforma. Tale



**Figura 3.1:** Meta Quest 2/3 e Varjo XR-4: *Visori di realtà virtuale utilizzati come dispositivi di visualizzazione immersiva di riferimento durante lo sviluppo e i test dell'interfaccia VR del software.*

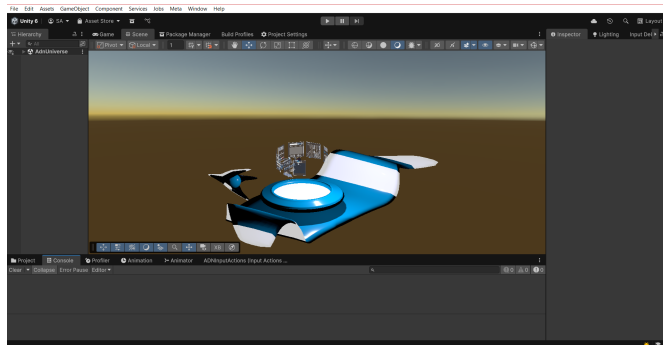
scelta si è rivelata coerente con l'obiettivo di mantenere il sistema il più possibile indipendente dall'hardware specifico, favorendo la portabilità tra dispositivi differenti.

L'integrazione con OpenXR ha inoltre garantito coerenza nella gestione del tracking, degli input dei controller e dell'eventuale hand tracking, pur senza vincolare l'applicazione a un singolo ecosistema.

## 3.2 Motore di sviluppo

L'estensione VR di Astra Data Navigator è stata sviluppata mediante il motore grafico Unity, ambiente multiplatforma ampiamente utilizzato per applicazioni tridimensionali interattive e sistemi di realtà virtuale. [9] Unity adotta un'architettura di tipo Entity-Component, in cui ogni entità della scena è rappresentata da un GameObject, al quale vengono associati componenti funzionali e script che ne definiscono il comportamento e le proprietà (Fig. 3.2). Questo paradigma consente di separare in modo chiaro la logica simulativa, la gestione dell'input, il controllo delle trasformazioni spaziali e il rendering, facilitando l'integrazione di un layer VR senza alterare radicalmente la struttura desktop preesistente.

L'utilizzo del linguaggio C# per lo scripting permette una gestione modulare delle responsabilità, rendendo possibile intervenire selettivamente sui comportamenti in funzione della modalità di esecuzione (desktop o XR). L'ecosistema XR integrato nel motore fornisce inoltre strumenti maturi per la gestione del tracking, dell'interazione con controller e della UI tridimensionale, aspetti fondamentali in un contesto immersivo.



**Figura 3.2:** Unity 6.0: *Interfaccia di sviluppo dell'applicazione.*

### 3.2.1 Unity 6.0

Lo sviluppo è stato condotto utilizzando Unity 6.0, versione che consolida e stabilizza l'integrazione con lo standard OpenXR e con i pacchetti XR ufficiali. Rispetto alle versioni precedenti, Unity 6.0 offre maggiore robustezza nell'esecuzione di applicazioni immersive, migliore compatibilità multiplatforma e un'integrazione più matura del nuovo Input System.

Nel contesto di Astra Data Navigator, tali miglioramenti hanno facilitato in particolare:

- l'astrazione dell'hardware VR attraverso OpenXR,
- la coesistenza tra modalità desktop e modalità immersiva nello stesso progetto,
- la stabilità del rendering in presenza di scene su larga scala.

Il modello del ciclo di esecuzione (Update, LateUpdate, FixedUpdate) rimane invariato rispetto alle versioni precedenti; tuttavia, la maggiore stabilità del runtime ha consentito di implementare in modo affidabile una separazione tra fase di raccolta delle richieste di orientamento e fase di applicazione delle trasformazioni, elemento centrale nell'architettura VR sviluppata.

### 3.2.2 TextMeshPro e leggibilità della UI in VR

Per la gestione dei testi e degli elementi tipografici dell'interfaccia è stato utilizzato TextMeshPro (TMP), sistema di rendering del testo basato su Signed Distance Field (SDF). [10]

In ambiente VR, la leggibilità della UI costituisce un aspetto critico: testi troppo piccoli, poco definiti o soggetti a aliasing possono generare affaticamento visivo e ridurre la fruibilità dei pannelli informativi. L'approccio SDF di TextMeshPro consente:

- elevata nitidezza anche a diverse distanze di osservazione;
- scalabilità senza perdita significativa di qualità;
- migliore gestione del contrasto e dei contorni del testo;
- controllo fine su spaziatura, interlinea e gerarchia tipografica.

Tali caratteristiche risultano particolarmente rilevanti in un sistema a pannelli spazializzati, nei quali l'utente può modificare dinamicamente distanza e angolo di visualizzazione.

### 3.2.3 Input System

Un elemento centrale dell'architettura VR è rappresentato dal Unity Input System, adottato in sostituzione del sistema di input legacy per garantire maggiore flessibilità e astrazione rispetto all'hardware. Il nuovo sistema separa le azioni logiche dell'applicazione dai dispositivi fisici che le generano: operazioni come movimento, rotazione, selezione o warp non sono più direttamente legate a un tasto o a un controller specifico, ma vengono definite come entità astratte a cui possono essere associati diversi binding. [11]

Questa impostazione consente di disaccoppiare la logica applicativa dall'hardware sottostante, rendendo possibile:

- il riutilizzo delle stesse azioni in modalità desktop e VR;
- la gestione differenziata di tastiera, mouse e controller XR;
- la compatibilità multiplatforma attraverso OpenXR.

Nel caso di Astra Data Navigator, le azioni legate alla modalità immersiva sono state centralizzate in un livello intermedio, rappresentato da `AdnInputManager`, che funge da punto di raccordo tra il sistema di input e i componenti responsabili del controllo della camera e dell'interazione con la scena. L'input non agisce quindi direttamente sulle trasformazioni, ma viene prima raccolto e interpretato, evitando dipendenze dirette tra dispositivi e logica simulativa.

L'organizzazione in Action Map dedicate alla modalità VR ha permesso di strutturare in modo coerente i comandi dei controller, distinguendo chiaramente tra input di movimento, rotazione, selezione e interazione con l'interfaccia. Alcune funzionalità contestuali, come il movimento verticale o la rotazione attorno all'asse forward attivabili tramite combinazioni con gripper o secondary trigger, sono state implementate come binding separati, mantenendo la struttura dell'input modulare ed estendibile.

L'Input System costituisce quindi un elemento chiave nella costruzione del layer VR, in quanto permette di mantenere la coerenza architetturale dell'applicazione pur introducendo un paradigma di interazione tridimensionale complesso.

## 3.3 Standard e framework XR

L'integrazione della realtà virtuale all'interno di Astra Data Navigator non si è limitata all'aggiunta di componenti hardware, ma ha richiesto l'adozione di uno standard di astrazione e di framework dedicati alla gestione dell'interazione tridimensionale. In questo contesto, un ruolo centrale è stato assunto dallo standard OpenXR e dai pacchetti ufficiali di Unity dedicati all'interazione immersiva.



Le scene di esempio fornite dal pacchetto hanno costituito un riferimento iniziale per la configurazione dei controller e delle interazioni con la UI. Tuttavia, l'architettura è stata adattata per integrarsi con il sistema di input centralizzato e con la logica di selezione del target già presente nell'applicazione desktop.

Il toolkit non impone un paradigma di interazione specifico, ma offre strumenti modulari che possono essere estesi o modificati. Questa caratteristica ha reso possibile l'integrazione del layer VR senza introdurre ridondanze nei sistemi di controllo già esistenti.

### 3.3.3 XR Hands

Per supportare il tracciamento delle mani è stato utilizzato il pacchetto XR Hands, anch'esso integrato nell'ecosistema OpenXR (Fig. 3.4). Questo consente di accedere ai dati di tracking delle mani quando il dispositivo lo supporta, mantenendo la compatibilità con controller tradizionali. [14]

Nel contesto della presente estensione, il tracking delle mani rappresenta un elemento di compatibilità e apertura futura più che un paradigma di interazione principale. La progettazione dell'interfaccia non si basa su gesture complesse o interazioni interamente diegetiche, privilegiando invece controlli stabili e ripetibili tramite analogici e pulsanti.

Questa scelta non è soltanto tecnica ma anche legata a considerazioni di accessibilità: l'utilizzo prioritario dei controller consente di mantenere compatibilità con dispositivi assistivi e soluzioni hardware alternative, rendendo l'interazione potenzialmente fruibile anche da utenti con limitazioni motorie o assenza parziale degli arti superiori. L'adozione del solo hand tracking come paradigma primario

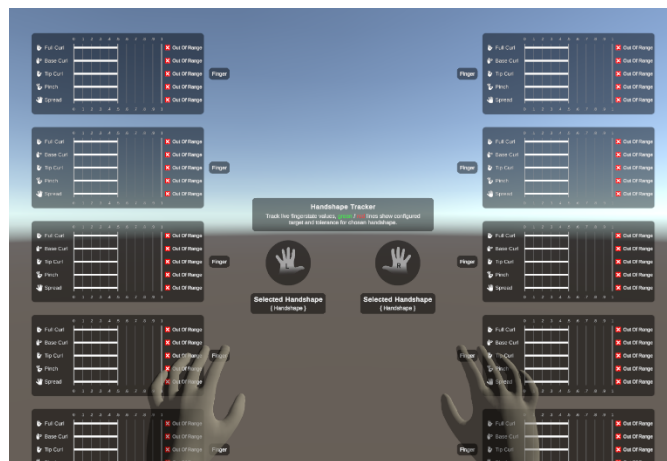


Figura 3.4: XR Hands: *Interfaccia per il rilevamento in tempo reale delle "gesture".*

avrebbe infatti introdotto una dipendenza più forte da specifiche capacità fisiche dell'utente.

## 3.4 Librerie e sistemi di supporto

Oltre al motore grafico e agli standard XR, lo sviluppo dell'estensione VR di Astra Data Navigator si basa su una serie di librerie e strumenti di supporto che costituiscono l'infrastruttura tecnica della simulazione. Tali componenti non riguardano direttamente l'interazione immersiva, ma rendono possibile la gestione della precisione numerica, della scala astronomica e dell'integrazione con dataset scientifici reali.

In particolare, due sistemi risultano centrali: il toolkit SPICE per il calcolo delle effemeridi e Space Graphics Toolkit per la gestione della scena in ambienti a scala cosmologica.

### 3.4.1 SPICE Toolkit

SPICE (Spacecraft, Planet, Instrument, C-matrix and Events) è il sistema sviluppato dal NAIF della NASA per la gestione di dati geometrici e temporali relativi a missioni spaziali e corpi celesti. [15]

All'interno di Astra Data Navigator, SPICE viene utilizzato per ottenere le posizioni dei corpi celesti in funzione del tempo simulato, garantendo coerenza con dataset scientifici reali. L'integrazione avviene tramite librerie native e interfacce che consentono di interrogare i kernel caricati, restituendo coordinate in sistemi di riferimento definiti.

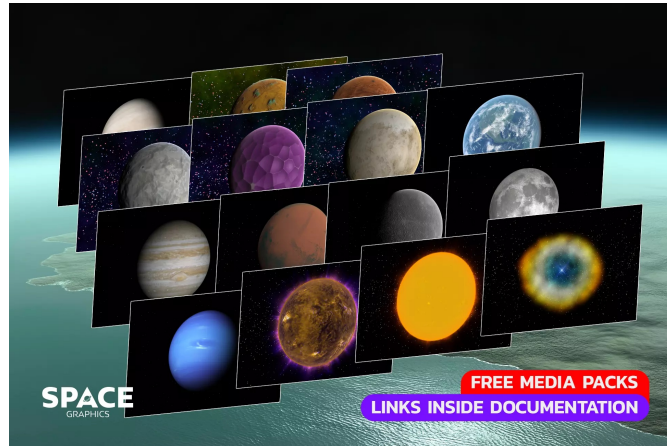
Dal punto di vista architetturale, SPICE non è legato alla componente VR, ma rappresenta il fondamento scientifico della simulazione. L'estensione immersiva sviluppata nella presente tesi opera quindi su dati la cui origine è esterna al motore grafico, preservando la separazione tra livello simulativo e livello di visualizzazione.

### 3.4.2 Space Graphics Toolkit

Space Graphics Toolkit (SGT) è una libreria per Unity progettata per la rappresentazione di ambienti spaziali su larga scala (Fig 3.5). La sua funzione principale all'interno del progetto è la gestione della floating origin e delle coordinate ad alta precisione tramite strutture come `SgtPosition` e sistemi di snapping. [16]

La rappresentazione dell'universo comporta infatti l'utilizzo di distanze astronomiche incompatibili con la precisione dei float standard di Unity. SGT consente di mantenere stabilità numerica attraverso un sistema di riposizionamento dinamico dell'origine della scena, evitando jitter e artefatti visivi durante la navigazione.

Nel contesto della VR, questa caratteristica assume un'importanza ancora maggiore: eventuali instabilità nella rappresentazione spaziale vengono amplificate dalla visione stereoscopica e possono compromettere il comfort dell'utente. L'utilizzo di SGT costituisce quindi un elemento abilitante per la realizzazione di un ambiente immersivo coerente su scala astronomica.



**Figura 3.5:** Space Graphics Toolkit: *Varietà di pianeti e oggetti generabili tramite l'utilizzo del Toolkit.*

### 3.5 Sintesi del capitolo

Il presente capitolo ha delineato le tecnologie e gli strumenti che costituiscono l'infrastruttura dell'estensione VR di Astra Data Navigator. Motore grafico, standard XR, piattaforme hardware e librerie scientifiche operano su livelli distinti ma complementari. Su tale base tecnologica si innesta l'architettura applicativa descritta nel capitolo successivo, in cui vengono analizzate le scelte progettuali e le soluzioni implementate.

## Capitolo 4

# Implementazione dell'interfaccia immersiva

L'estensione di Astra Data Navigator alla modalità immersiva non ha rappresentato una semplice operazione di porting tecnologico, bensì una riorganizzazione strutturale dell'interazione, del controllo della camera e dell'interfaccia utente.

L'obiettivo non è stato replicare in realtà virtuale il comportamento desktop, ma integrare un sistema di navigazione e selezione coerente con le specificità percettive e tecniche dell'ambiente XR, mantenendo al contempo la compatibilità con l'architettura preesistente.

Il capitolo descrive le scelte progettuali e implementative adottate per introdurre il layer VR all'interno dell'applicazione, analizzando:

- l'integrazione con il sistema di controllo della camera;
- la gestione centralizzata del target;
- l'adattamento dell'input tramite OpenXR e Input System;
- il sistema di puntamento e selezione;
- la navigazione immersiva;
- la progettazione dell'interfaccia utente spazializzata.

Particolare attenzione viene dedicata alla stabilità percettiva e alla coerenza tra spazio simulato e spazio interattivo, aspetti critici in un contesto che combina coordinate astronomiche ad alta precisione e rendering immersivo in tempo reale.

## 4.1 Architettura generale del sistema immersivo

L'introduzione della modalità immersiva in Astra Data Navigator è stata affrontata come un'integrazione controllata nell'architettura esistente, e non come un modulo completamente esterno. La scelta deriva dal fatto che il sistema desktop possiede già un comportamento consolidato per navigazione, rotazione e aggiornamento della camera; replicare tali funzioni in un layer separato avrebbe introdotto ridondanze e, soprattutto, aggiornamenti concorrenti su posizione e rotazione provenienti da più sorgenti (input XR, camera lock, warp, funzioni di puntamento automatico). In un contesto VR, tale concorrenza produce facilmente instabilità percettiva (jitter) e comportamenti non deterministici.

Per evitare questi effetti, l'architettura è stata rifattorizzata secondo due principi:

- unicità della sorgente di trasformazione della camera, concentrata in un singolo componente;
- target unico condiviso, propagato tramite eventi e consumato coerentemente dai moduli.

### 4.1.1 Unicità della sorgente di trasformazione: AdnCameraControls come “camera authority”

Il controllo effettivo delle trasformazioni della camera è accentrato nel componente AdnCameraControls. Questo componente agisce come camera authority: è l'unico punto in cui vengono applicate, a ogni frame, le trasformazioni di:

- rotazione manuale (desktop o XR),
- movimento libero,
- movimento orbitale durante lock,
- damping/smoothing di rotazione e traslazione,
- delta di rivoluzione/rotazione indotte dal target (quando rilevanti).

Invece di permettere a più moduli di applicare direttamente `Transform.LookAt()` o modificare localmente posizione/rotazione, tutte le operazioni di puntamento vengono ricondotte a un'unica pipeline di aggiornamento che risiede in `Update()`. Questo riduce drasticamente la possibilità che due sistemi scrivano sul transform nello stesso frame, e garantisce coerenza tra modalità desktop e modalità VR.

### 4.1.2 Target unico e propagazione event-driven

L'applicazione mantiene un target unico per impostazione architetturale precedente: la selezione dell'oggetto attivo viene gestita centralmente (tramite `AdnInputManager`, che notifica ai listener). Ne consegue che i diversi sistemi che dipendono dal target (lock, warp, pannelli informativi, selezione) non competono su quale oggetto sia "corrente", ma consumano tutti lo stesso riferimento.

Questo evita un'intera classe di problemi tipici della VR: mismatch tra oggetto selezionato, oggetto evidenziato e oggetto effettivamente seguito dalla camera.

### 4.1.3 Producer/Consumer tra `LateUpdate` e `Update`: `RequestLookAt` invece di `Transform.LookAt`

Un aspetto fondamentale dell'architettura è la separazione tra generazione della richiesta di puntamento e applicazione della trasformazione.

- In `AdnCameraLock.LateUpdate()`, quando il sistema è in stato di lock, viene generata una richiesta di orientamento verso il target tramite `AdnCamera.Instance.Controls.RequestLookAt(Target.Position, LockRotationFactor)`.

Questa scelta colloca la produzione della richiesta a fine frame, quando le trasformazioni dei corpi (e l'eventuale accumulo di delta) sono già state aggiornate.

- In `AdnCameraControls.Update()`, la richiesta viene consumata: se `lookAtRequested` è true, la rotazione "del frame" viene calcolata e accumulata nella pipeline comune (`remainingLookDelta`), invece di applicare direttamente un `LookAt` locale.

Questa architettura è intenzionalmente asimmetrica: `LateUpdate` non applica trasformazioni, ma prepara lo stato; `Update` applica trasformazioni in modo centralizzato. In questo modo il puntamento non è una "azione immediata" distribuita nei moduli, ma un input alla pipeline della camera.

### 4.1.4 Stabilizzazione in VR: posizione target "stabile" e calcolo geometrico dedicato

In modalità desktop, la posizione world-space del target può essere ricavata in modo stabile tramite `FloatingCamera.CalculatePosition(...)`. In modalità XR, invece, l'uso di un XR Origin e di una camera fisica/offset introduce condizioni in cui un calcolo naïve può amplificare jitter e incoerenze (parallasse tra camera reale e pivot di simulazione).

Per questo motivo è stata introdotta una funzione dedicata per ottenere una posizione target stabile per frame (`GetStableTargetWorldPosition`), che ricostruisce un pivot coerente rispetto all'offset tra XR Origin e floating camera. La

stabilizzazione non è quindi demandata a “più smoothing”, ma affrontata come problema geometrico: prima si definisce correttamente il riferimento, poi si applica la rotazione.

#### 4.1.5 Implicazioni sulla locomozione: movimento libero vs orbitale

Quando non è presente una richiesta di look-at, la locomozione è libera e aggiornata in coordinate relative all'orientamento corrente della camera. Quando è attivo il lock (cioè quando la richiesta di look-at è presente), la locomozione cambia paradigma e diventa orbitale: la camera non trasla liberamente nello spazio, ma mantiene un offset rispetto al target (distanza orbitale + rotazione orbitale), riducendo disorientamento e mantenendo il target come riferimento percettivo.

Questo comportamento è implementato all'interno dello stesso componente di controllo, evitando che “movimento” e “puntamento” vivano in moduli differenti e si contraddicano.

#### 4.1.6 Sintesi

L'architettura risultante non coincide con un disaccoppiamento totale “a plugin”, ma con una coabitazione controllata: la modalità VR viene integrata preservando il comportamento desktop, mentre la camera è gestita da una pipeline unica che:

- riceve input manuale,
- riceve richieste di puntamento,
- applica smoothing,
- applica delta accumulati,
- evita scritture concorrenti sul transform.

Questa scelta rende difendibile la tesi: la stabilità non è affidata a “trucchi”, ma a un modello coerente di responsabilità e flusso dati.

## 4.2 Configurazione dell'ambiente XR e gestione dell'input

L'integrazione della modalità immersiva è stata realizzata adottando lo standard OpenXR all'interno dell'ambiente Unity, con il supporto dei pacchetti XR Interaction Toolkit e XR Hands. La gestione dell'input è stata interamente demandata al

nuovo Input System di Unity, evitando l'accesso diretto ai dispositivi XR a basso livello.

Questa scelta architetturale garantisce indipendenza dall'hardware, maggiore manutenibilità e coerenza tra modalità desktop e VR.

### **4.2.1 Adozione di OpenXR**

OpenXR è stato scelto come standard di interfaccia tra applicazione e dispositivi XR per garantire compatibilità multiplatforma e ridurre il lock-in verso SDK proprietari.

L'adozione di OpenXR consente:

- supporto a visori stand-alone e PC-VR;
- interoperabilità con controller e hand tracking;
- astrazione delle caratteristiche hardware specifiche.

Le componenti di interazione sono state costruite a partire dalle scene di esempio ufficiali fornite dai pacchetti XR Interaction Toolkit e XR Hands, che includono:

- gestione dei controller XR;
- hand tracking;
- sistemi di interazione ray-based e direct interaction;
- elementi di UI tridimensionale (bottoni, toggle, input field, tastiera virtuale).

Tali esempi non sono stati utilizzati come soluzione finale, ma come base strutturale per integrare le funzionalità immersive all'interno dell'architettura di Astra Data Navigator.

### **4.2.2 Architettura dell'Input System**

Tutti gli input relativi alla modalità VR vengono gestiti tramite il nuovo Input System di Unity. Nessuna parte del sistema legge direttamente i valori dai dispositivi XR tramite InputDevice o API specifiche del controller.

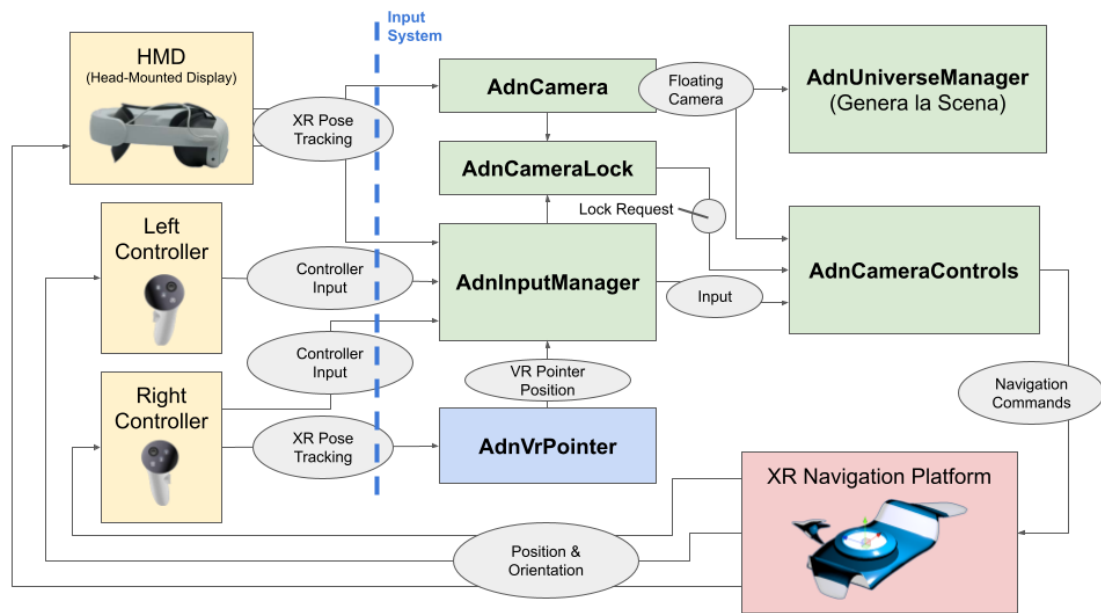
Le azioni vengono definite in Action Map e successivamente interpretate dal componente AdnInputManager, che funge da intermediario tra input fisico e logica applicativa (Fig. 4.1).

Questo approccio garantisce:

- separazione tra dispositivo e comportamento;

- riutilizzabilità della logica su hardware differenti;
- testabilità delle azioni senza dipendenza diretta dall'hardware.

L'unico input letto direttamente tramite la classe `Keyboard` è il comando di uscita (Esc) in modalità VR su PC, utilizzato esclusivamente come fallback per chiudere l'applicazione.



**Figura 4.1:** Diagramma architetturale del sistema di input e navigazione in realtà virtuale di Astra Data Navigator, che mostra il flusso delle informazioni tra dispositivi XR, moduli di gestione della camera, sistema di input e componenti di interazione come il VR pointer.

### 4.2.3 Action Map “Player”

È stata definita una Action Map dedicata, denominata “Player”, che raccoglie tutte le azioni necessarie alla navigazione e all'interazione in ambiente immersivo.

Le azioni sono organizzate per categorie funzionali:

Le azioni sono state associate ai controlli standard dei dispositivi XR (assi analogici, pressione degli assi, trigger e pulsanti principali) tramite binding dell'Input System, mantenendo separata la logica applicativa dall'hardware specifico.

La struttura della Action Map consente di mantenere coerenza con la modalità desktop: le azioni di movimento e rotazione esistono in entrambe le modalità, ma sono alimentate da binding differenti (tastiera/mouse vs controller XR).

<b>Navigazione spaziale:</b>	<b>Rotazione:</b>	<b>Interazione:</b>
Move (XZ)	Rotate (Yaw/Pitch)	Select
Move (Y)	Rotate (Roll/Pitch)	Cancel
Move (XY)	Rotate (Roll)	ShowPointer
		ShowTarget
<b>Controllo della velocità:</b>		WarpTo
SpeedControl		LookAt

**Tabella 4.1:** *Lista delle azioni associate agli input dentro la ActionMap "Player".*

In modalità VR, il movimento verticale (asse Y) e la rotazione attorno all'asse di vista (roll) sono gestiti tramite azioni dedicate nell'Action Map "Player". Tali azioni sono associate a controlli specifici dei controller XR (ad esempio secondary trigger o gripper), separandole dagli assi analogici principali utilizzati per traslazione sul piano e rotazione yaw/pitch. Questa separazione riduce il rischio di attivazioni involontarie e mantiene la navigazione più controllabile.

#### 4.2.4 Separazione tra input e logica

`AdnInputManager` interpreta le azioni attive e fornisce valori normalizzati (ad esempio `PitchInput`, `YawInput`, `ForwardInput`, `SpeedDelta`) ai sistemi di controllo della camera.

La camera non legge mai direttamente il dispositivo, ma consuma valori già filtrati e soggetti a:

- soglie minime (threshold);
- scaling per velocità;
- disabilitazione temporanea durante lock o warp.

Questo livello intermedio consente di:

- bloccare selettivamente l'input manuale durante il lock;
- modificare la sensibilità senza alterare i binding;
- mantenere coerenza tra desktop e VR.

#### 4.2.5 Considerazioni progettuali

La scelta di centralizzare l'input nel nuovo Input System e in `AdnInputManager` non è solo tecnica, ma architetturale:

- evita dipendenze dirette dall'hardware XR;
- permette estensione futura verso nuovi dispositivi;
- mantiene la camera indipendente dal tipo di controller.

In questo modo la modalità immersiva non introduce una nuova logica di controllo, ma riutilizza e adatta quella esistente, garantendo uniformità comportamentale tra ambienti di esecuzione differenti.

## 4.3 Sistema di locomozione e gestione della scala

La locomozione in un ambiente astronomico presenta criticità peculiari rispetto a scenari VR tradizionali: l'utente deve potersi spostare su distanze estremamente variabili (da prossimità di un corpo celeste fino a scale interplanetarie e oltre), mantenendo al contempo controllo, comfort e stabilità visiva. In Astra Data Navigator l'obiettivo non è simulare un moto fisico realistico, ma fornire una navigazione funzionale all'esplorazione e alla comprensione spaziale dei dati. Ne deriva una scelta esplicita: movimento parametrico controllato, con velocità regolabile e con soglie di input atte a prevenire micro-movimenti involontari.

### 4.3.1 Obiettivi progettuali della locomozione

Il sistema di locomozione è progettato per soddisfare tre requisiti principali:

- **Scalabilità:** la stessa interazione deve funzionare a scala locale (vicino a pianeti/feature) e a scala ampia (tra oggetti distanti);
- **Robustezza:** ridurre oscillazioni e input accidentali, particolarmente in VR;
- **Coerenza tra modalità:** mantenere un comportamento concettualmente analogo tra desktop e VR, pur con binding differenti.

La locomozione è quindi implementata come calcolo di un delta di movimento per frame, derivato dagli input e scalato in base a una velocità corrente. Tale delta viene poi applicato tramite la camera "floating" per mantenere precisione su grandi distanze.

### 4.3.2 Movimento libero in sei gradi di libertà (6DoF controllato)

In assenza di vincoli di puntamento (assenza di richiesta di look-at), la locomozione avviene in modalità free movement, dove l'utente può:

- traslare lungo gli assi locali della camera, o della piattaforma in VR, cioè avanti/indietro, destra/sinistra, alto/basso (Fig. 4.2);
- ruotare la vista tramite yaw/pitch/roll (in VR tramite input analogici, in desktop tramite mouse/roll input).

Il delta di traslazione viene calcolato a partire dall'input normalizzato  $[-1,1]$ , scalato per la velocità corrente e per il tempo di frame, quindi convertito nello spazio della camera applicando la rotazione corrente. Questo approccio rende il movimento coerente con l'orientamento dell'utente e mantiene un controllo immediato e prevedibile.

Per limitare instabilità dovute a noise del controller, in VR vengono applicate soglie (threshold) agli input: valori sotto una soglia minima vengono azzerati, evitando drift e micro-spostamenti.

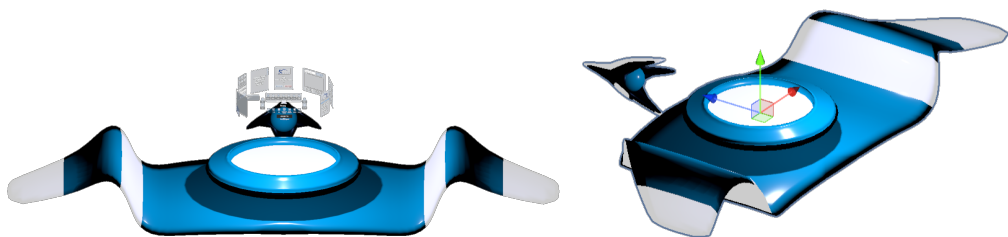
### 4.3.3 Gestione della velocità e adattamento alla scala

La velocità di movimento rappresenta il principale parametro di controllo della scala percepita. Un valore unico non sarebbe sufficiente: troppo basso renderebbe impossibile attraversare distanze astronomiche, troppo alto renderebbe ingestibile la navigazione di prossimità.

Per questo motivo il sistema prevede:

- velocità variabile, modificabile dall'utente durante l'esplorazione;
- limiti min/max coerenti con lo scaling della simulazione;
- un meccanismo di aggiornamento della velocità che non interferisce con l'applicazione del delta di movimento (evitando salti di posizione dovuti a modifiche di speed e movement nello stesso frame).

In aggiunta, eventi specifici della simulazione possono suggerire un aggiornamento della velocità (es. dopo un warp verso un target, la velocità viene impostata su un



**Figura 4.2:** Vista della piattaforma col menu aperto e orientamento della piattaforma rispetto agli assi cartesiani di movimento (rosso: X, verde: Y, blu: Z).

valore più adatto alla prossimità del nuovo oggetto), in modo da ridurre il rischio di overshooting.

#### 4.3.4 Smorzamento del movimento e stabilità percettiva

In VR, la locomozione continua può generare discomfort se il moto è troppo “scattoso” o se la camera risponde con accelerazioni brusche. Per mitigare questo comportamento, il sistema introduce uno smorzamento (damping) che:

- accumula il delta di movimento in una variabile residua;
- interpola progressivamente verso zero in base a un parametro di damping;
- applica a ogni frame solo la parte “consumata” del delta.

Questo schema stabilizza il moto e rende più continuo l’aggiornamento della posizione, senza introdurre ritardi eccessivi o perdita di controllo.

#### 4.3.5 Interazione con eventi di rivoluzione e rotazione

La simulazione astronomica può aggiornare la posizione/orientamento di entità (revolution/rotation). Per evitare jitter dovuto a eventi asincroni richiamati più volte tra due frame, i delta generati dagli eventi vengono accumulati e consumati nella pipeline di movimento della camera. In tal modo il movimento risultante resta deterministico e applicato una sola volta per frame, coerentemente con l’architettura centralizzata descritta nel paragrafo 4.1.

#### 4.3.6 Considerazioni finali

Il sistema di locomozione implementato non mira a riprodurre una dinamica fisica realistica, ma a garantire un compromesso efficace tra esplorabilità e comfort in un contesto a scala variabile. Le scelte principali - movimento parametrico, soglie sugli input, velocità regolabile e damping - sono tutte orientate a:

- mantenere la navigazione controllabile;
- evitare drift e instabilità percettiva;
- supportare transizioni rapide tra scala “locale” e scala “astronomica”.

La modalità di movimento vincolata al target (orbitale) viene trattata separatamente nel contesto del camera lock, in quanto introduce un paradigma differente legato alla stabilizzazione del puntamento (par. 4.5).

## 4.4 Sistema di puntamento e selezione del target

Il sistema di puntamento costituisce il meccanismo principale attraverso cui l'utente identifica e seleziona i corpi celesti all'interno della simulazione. La progettazione di tale sistema ha dovuto conciliare tre esigenze:

- precisione nella selezione di oggetti potenzialmente molto piccoli;
- coerenza tra modalità desktop e modalità VR;
- integrazione con i pannelli informativi e le funzioni di navigazione (look-at e warp).

Il sistema implementato supporta differenti modalità di selezione, mantenendo tuttavia un modello a target unico globale condiviso dall'intera applicazione.

### 4.4.1 Puntamento in modalità VR

In modalità VR il puntamento avviene tramite un sistema ray-based associato al controller destro. L'attivazione del puntatore avviene mediante la pressione del tasto primario del controller.

Quando il puntatore è attivo:

1. viene proiettato un raggio nello spazio;
2. il sistema identifica il corpo celeste più vicino alla direzione del raycast;
3. viene visualizzato un pannello informativo (info box) associato all'oggetto candidato.

Per evitare selezioni accidentali dovute a micro-movimenti del controller, l'assegnazione effettiva del target non avviene immediatamente. L'oggetto deve rimanere il più vicino al puntatore per un intervallo di tempo minimo (cooldown), impostato a 0.4 secondi. Solo al termine di questo intervallo viene eseguita la funzione di selezione (`SelectObject()`), che aggiorna il target globale dell'applicazione.

Questo meccanismo introduce una forma di stabilizzazione temporale della selezione, riducendo errori involontari e migliorando il controllo in ambiente immersivo.

In alternativa, l'utente può selezionare un corpo celeste attraverso il pannello di ricerca (Search Panel), inserendone il nome o l'identificativo (ad esempio nel catalogo Hipparcos). La selezione dall'elenco aggiorna direttamente il target globale, senza necessità di puntamento spaziale.

#### 4.4.2 Puntamento in modalità desktop

In modalità desktop, la selezione avviene tramite mouse con paradigma “punta e clicca”. L'utente orienta la camera verso l'oggetto desiderato e seleziona direttamente tramite click.

Non essendo presente instabilità dovuta al tracking fisico del controller, non è necessario un cooldown temporale esplicito come in VR.

#### 4.4.3 Interazione con LookAt e Warp

Una volta selezionato un target - indipendentemente dal metodo - l'utente può attivare funzioni di navigazione avanzata:

- **LookAt**: orientamento automatico della camera verso il target;
- **WarpTo**: spostamento rapido verso il target.

Queste funzioni possono essere attivate:

- dal pannello informativo (se aperto);
- dal pannello di ricerca;
- direttamente tramite binding dedicati in modalità VR.

In particolare, in VR:

- la pressione dell'analogico sinistro attiva l'azione associata a **LookAt**;
- la pressione dell'analogico destro attiva l'azione associata a **WarpTo**.

Tali azioni sono definite nella Action Map “Player” e integrate nel sistema di controllo della camera, mantenendo la separazione tra input fisico e logica applicativa.

#### 4.4.4 Coerenza architetturale

Nonostante la molteplicità delle modalità di selezione, il sistema mantiene una proprietà fondamentale: esiste un solo target attivo in ogni istante.

Questo garantisce:

- coerenza tra pannelli informativi e camera;
- assenza di conflitti tra metodi di selezione;
- semplicità di integrazione con il sistema di lock e stabilizzazione.

Il sistema di puntamento non è quindi un modulo isolato, ma un'interfaccia coerente con l'architettura centralizzata della camera descritta nei paragrafi precedenti.

## 4.5 Sistema di stabilizzazione della camera

La stabilizzazione della camera rappresenta uno dei principali problemi tecnici affrontati nello sviluppo della modalità immersiva di Astra Data Navigator. In ambienti astronomici, caratterizzati da scale spaziali estreme e oggetti di dimensioni angularmente ridotte, piccole variazioni numeriche o geometriche possono tradursi in oscillazioni percettive significative, particolarmente evidenti in realtà virtuale.

Il problema non è architetturale - poiché la trasformazione è centralizzata - ma geometrico e numerico.

### 4.5.1 Definizione del problema percettivo

Quando la camera è vincolata a un target puntiforme (stella o pianeta lontano), la direzione di puntamento è data dal vettore:

$$\mathbf{d} = \mathbf{p}_{\text{target}} - \mathbf{p}_{\text{camera}} \quad (4.1)$$

La rotazione desiderata è quindi:

$$R_{\text{target}} = \text{LookRotation}(\mathbf{d}) \quad (4.2)$$

In condizioni ideali, se il target è statico e la camera non si muove, tale rotazione dovrebbe rimanere costante. Tuttavia, in pratica si verificano:

- micro-variazioni della posizione target (revolution updates);
- micro-variazioni del pivot della camera (floating origin);
- offset dinamico tra XR Origin e floating camera;
- accumulo di delta tra frame.

Poiché l'oggetto osservato può occupare pochi pixel sullo schermo, anche variazioni minime di  $\mathbf{d}$  producono variazioni angolari percepibili.

### 4.5.2 Floating origin e posizione target stabile in XR

In modalità desktop, la posizione world-space del target può essere ricavata tramite:

$$\mathbf{p}_{\text{world}} = \text{FloatingCamera.CalculatePosition}(\text{target}) \quad (4.3)$$

In modalità XR, tuttavia, la presenza di un XR Origin separato dalla floating camera introduce una differenza tra:

- posizione fisica della camera reale;

- pivot logico della simulazione.

Un calcolo naïve della posizione world-space amplifica jitter e disallineamenti. Per questo motivo è stata introdotta una funzione dedicata (`GetStableTargetWorldPosition`) che:

1. calcola il pivot coerente rispetto alla floating camera;
2. considera l'offset tra XR Origin e camera reale;
3. ricostruisce una posizione target stabile nel sistema di riferimento corrente.

La stabilizzazione è quindi affrontata prima a livello geometrico, riducendo l'errore alla sorgente.

### 4.5.3 Separazione temporale: produzione e consumo della richiesta

La richiesta di puntamento viene generata in `LateUpdate()` dal sistema di lock, quando le trasformazioni del target sono già aggiornate per il frame corrente.

L'applicazione della rotazione avviene invece in `Update()` all'interno di `AdnCameraControls`.

Questo modello producer/consumer:

- evita scritture concorrenti sul `Transform`;
- garantisce che la rotazione venga applicata una sola volta per frame;
- impedisce che aggiornamenti multipli tra due frame producano jitter.

### 4.5.4 Calcolo della rotazione con quaternioni

La rotazione verso il target viene calcolata tramite:

$$R_{\text{target}} = \text{Quaternion.LookRotation}(\mathbf{p}_{\text{target}} - \mathbf{p}_{\text{camera}}) \quad (4.4)$$

Se il fattore di interpolazione è inferiore a 1, viene applicato smoothing:

$$R_{\text{final}} = \text{Slerp}(R_{\text{current}}, R_{\text{target}}, \alpha) \quad (4.5)$$

dove:

$$\alpha = \frac{1}{\text{LookAtSlerpFactor}} \quad (4.6)$$

L'uso di quaternioni evita problemi di gimbal lock e garantisce interpolazione uniforme nello spazio delle rotazioni.

### 4.5.5 Accumulo e damping

La rotazione calcolata non viene applicata direttamente, ma accumulata in `remainingLookDelta`, successivamente viene applicato un damping:

$$R_{\text{new}} = \text{Slerp}(\text{remainingLookDelta}, I, f) \quad (4.7)$$

dove  $I$  è il quaternione identità e  $f$  è funzione del parametro `lookDamping`. Questo approccio evita variazioni brusche e consente una transizione progressiva verso la nuova direzione.

### 4.5.6 Locomozione orbitale come strategia di stabilità

Quando il lock è attivo, la locomozione non è più libera ma orbitale. La posizione della camera viene definita come:

$$\mathbf{p}_{\text{camera}} = \mathbf{p}_{\text{target}} + R_{\text{orbital}} * (0, 0, \mathbf{d})$$

dove:

- $R_{\text{orbital}}$  è un quaternione accumulato tramite yaw/pitch;
- $\mathbf{d}$  è la distanza orbitale dal target.

Questo modello:

- mantiene il target come riferimento visivo centrale;
- riduce disorientamento;
- evita accumulo di errore direzionale dovuto a traslazioni libere.

### 4.5.7 Limiti residui

Nonostante le strategie adottate, persistono condizioni critiche:

- target con raggio molto piccolo (come piccoli asteroidi o satelliti artificiali);
- distanza elevata e soglia percettiva inferiore a un pixel;
- aggiornamenti rapidi di rivoluzione durante tempo accelerato;
- variazioni di floating origin ravvicinate.

In tali casi, è stata introdotta una misura mitigativa: sospensione del tempo in XR quando si è molto vicini a oggetti di piccole dimensioni, al fine di ridurre motion sickness.

### 4.5.8 Considerazioni finali

La stabilizzazione non è stata ottenuta tramite un singolo meccanismo, ma mediante una combinazione di:

- correzione geometrica della posizione target in XR;
- centralizzazione dell'applicazione delle trasformazioni;
- uso di quaternioni per la rotazione;
- interpolazione slerp controllata;
- locomozione orbitale vincolata al target.

Il problema, inizialmente percepito come instabilità visiva, si è rivelato essere un'interazione complessa tra precisione numerica, scala spaziale e percezione umana in ambiente immersivo.

## 4.6 Interfaccia utente spazializzata e sistema di menu orbitale

L'interfaccia utente in modalità immersiva non rappresenta una semplice trasposizione della UI desktop in ambiente tridimensionale, ma una riorganizzazione completa della logica di interazione. L'obiettivo principale è stato garantire leggibilità, accessibilità e libertà visiva in uno spazio astronomico potenzialmente illimitato.

Tutti gli elementi dell'interfaccia sono implementati in world space e rigidamente ancorati alla XR Origin, più precisamente al Camera Offset. In questo modo la UI si muove solidalmente con il sistema di riferimento della realtà virtuale, rimanendo indipendente dalle trasformazioni del floating origin astronomico.

### 4.6.1 Architettura generale del sistema di menu

Il sistema di menu VR è organizzato come un insieme di pannelli disposti radialmente attorno all'asse Y della XR Origin.

L'apertura e chiusura dei pannelli avviene tramite toggle nel menu principale, oppure tramite un piccolo tasto "Close" posto sul bordo di ciascun pannello. Tuttavia, l'elemento distintivo dell'implementazione non è la semplice attivazione dei Canvas, bensì la gestione orbitale della loro disposizione.

## 4.6.2 Rotazione orbitale dei pannelli

I pannelli sono disposti lungo una circonferenza virtuale centrata sulla XR Origin (Fig. 4.3).

Tramite l'analogico sinistro del controller, l'utente può ruotare l'intero sistema attorno all'asse Y della XR Origin, facendo scorrere i pannelli lateralmente.

Il sistema garantisce che:

- un solo pannello sia sempre frontale;
- anche un pannello non attivo possa essere portato in posizione frontale;
- la vista frontale possa essere completamente liberata.

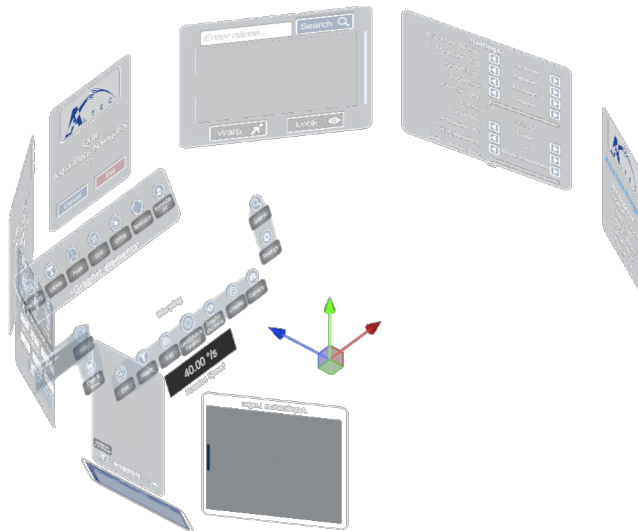
Quest'ultimo aspetto è cruciale: durante la navigazione astronomica, specialmente in fase di warp o osservazione ravvicinata, è necessario evitare che la UI ostruisca il campo visivo centrale. La possibilità di ruotare i pannelli consente di mantenere accessibili le funzionalità senza compromettere l'esperienza immersiva.

## 4.6.3 Navigazione delle opzioni interne

La selezione delle opzioni all'interno del pannello attivo non avviene esclusivamente tramite ray interactor.

L'analogico destro consente di navigare tra le opzioni disponibili nel pannello corrente. Le opzioni sono rappresentate da istanze della classe `AdnVrMenuOption`, raccolte nel vettore `options` della classe `AdnVrMenu`.

Caratteristiche principali del sistema:



**Figura 4.3:** Vista dell'interfaccia utente isolata dal contesto immersivo.

- le opzioni vengono inserite manualmente nel vettore dal programmatore;
- ogni pannello espone solo le opzioni rilevanti;
- il sistema di navigazione è indipendente dalla tipologia del widget (button, toggle, input field).

Questa impostazione evita automatismi poco controllabili e consente una progettazione consapevole della gerarchia interattiva.

#### 4.6.4 Descrizione dei pannelli interagibili

I pannelli interagibili disponibili nella modalità VR sono i seguenti:

- **Info Panel:** Mostra le informazioni relative al target selezionato e consente l'esecuzione diretta delle funzioni di `LookAt` e `Warp`. Il contenuto viene aggiornato dinamicamente in base al target globale attivo.
- **Graphic Elements Panel:** Permette l'attivazione e la disattivazione di elementi grafici della scena (labels, assi, griglia, orbite, backdrop, ecc.), influenzando la rappresentazione visiva senza modificare la logica simulativa.
- **Settings Panel:** Raccoglie impostazioni relative a sistema di coordinate, unità di misura, epoca, IPD multiplier, parametri audio e opzioni di visualizzazione.
- **Search Panel:** Consente la ricerca di oggetti tramite nome o identificativo (ad esempio nel catalogo Hipparcos), e di attivare le funzioni di `LookAt` e `WarpTo`. La selezione aggiorna direttamente il target globale.
- **"Set Date & Time" Panel:** Permette la configurazione della data e del tempo simulato, influenzando l'aggiornamento delle effemeridi.
- **Log Panel:** Visualizza messaggi di sistema e informazioni diagnostiche utili durante l'esecuzione.
- **Filters Panel:** Consente il filtraggio delle categorie di oggetti visibili (stelle, pianeti, satelliti, spacecraft, asteroidi).
- **Credits Panel:** Mostra informazioni relative allo sviluppo dell'applicazione.
- **Quit Panel:** Gestisce l'uscita dall'applicazione.

### **4.6.5 Separazione tra livello informativo e livello simulativo**

La scelta di ancorare i pannelli alla XR Origin e non allo spazio astronomico introduce una netta separazione tra:

- spazio simulato (coordinate astronomiche, floating origin);
- spazio interattivo (sistema VR centrato sull'utente).

Questo evita che la UI venga influenzata da cambi di scala o spostamenti astronomici e garantisce stabilità percettiva costante.

### **4.6.6 Estendibilità del sistema**

L'architettura adottata consente l'aggiunta di nuovi pannelli con intervento minimo sul sistema esistente.

Poiché:

- la logica di rotazione orbitale è centralizzata;
- la navigazione tramite `AdnVrMenuOption` è generica;
- i binding XR sono già implementati;

un nuovo pannello può essere integrato inserendo le relative opzioni nel vettore del menu, senza modificare il comportamento globale del sistema.

Questo rende il sistema modulare e scalabile, facilitando future estensioni funzionali.

# Capitolo 5

## Valutazione e test sperimentali

Il presente capitolo descrive le attività di valutazione condotte sull'estensione VR di Astra Data Navigator, con l'obiettivo di analizzarne l'usabilità, la qualità dell'interazione e la stabilità operativa. La verifica è stata articolata in una fase di test con utenti reali, accompagnata da questionari strutturati, e in una successiva analisi delle prestazioni tecniche del sistema in condizioni controllate.

### 5.1 Metodologia dei test con utenti

La valutazione della versione VR di Astra Data Navigator è stata condotta attraverso una combinazione di osservazione diretta e raccolta di feedback strutturato tramite questionario.

L'obiettivo non è stato esclusivamente quello di misurare parametri tecnici, ma di analizzare l'esperienza complessiva dell'utente in termini di intuitività, controllo, comfort e percezione dell'efficacia della realtà virtuale come strumento di esplorazione astronomica.

L'attività sperimentale si è concentrata su sessioni individuali di utilizzo in ambiente controllato, con successiva compilazione di un modulo di valutazione.

#### 5.1.1 Obiettivi della valutazione

La fase di test con utenti ha avuto i seguenti obiettivi principali:

- valutare l'intuitività dell'interfaccia e dei comandi di navigazione;
- analizzare il grado di controllo percepito durante movimento, rotazione e orbiting;

- verificare l'eventuale insorgenza di motion sickness o disorientamento;
- misurare la percezione dell'efficacia della VR per la divulgazione astronomica;
- raccogliere suggerimenti qualitativi per miglioramenti futuri.

L'attenzione è stata posta in particolare sul ciclo di interazione principale (puntamento, selezione, warp, camera lock, esplorazione libera), che costituisce il nucleo dell'esperienza immersiva.

### 5.1.2 Contesto di prova

Le prove sono state svolte utilizzando un visore Meta Quest collegato a PC tramite modalità Quest Link.

L'applicazione veniva avviata dal computer dall'operatore, mentre l'utente indossava il visore già configurato. Ogni sessione aveva una durata media compresa tra 15 e 20 minuti.

Prima dell'utilizzo, all'utente venivano spiegati:

- il contesto dell'applicazione;
- le principali funzionalità sviluppate nella tesi;
- i controlli di base per movimento, rotazione, apertura menu e selezione.

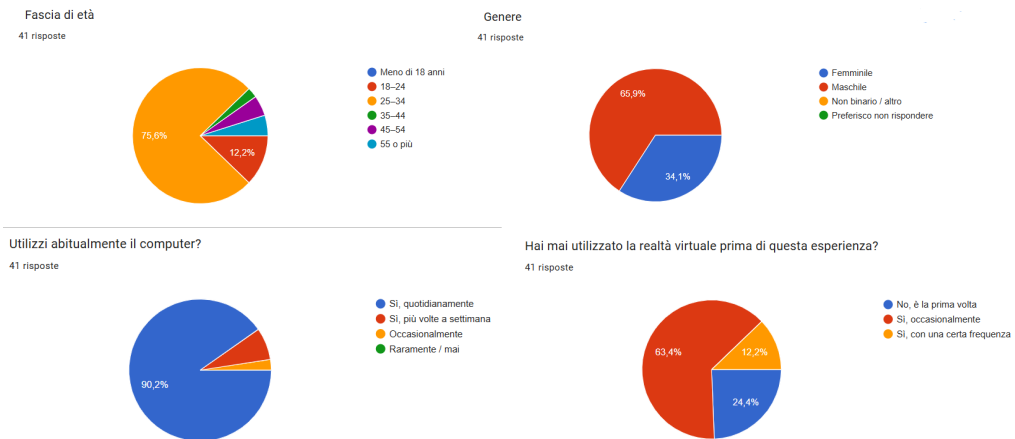
Durante la prova l'utente veniva inizialmente lasciato libero di esplorare, osservandone il comportamento spontaneo, per poi essere guidato gradualmente nell'utilizzo delle principali funzionalità (puntamento, ricerca, warp, orbiting, regolazione del tempo).

### 5.1.3 Profilo del campione

Il campione è composto da circa quaranta partecipanti, appartenenti prevalentemente alla fascia di età 25–34 anni, con una minoranza di utenti più giovani (18–24) e over 45.

La maggior parte dei partecipanti utilizza abitualmente il computer; una parte significativa aveva già avuto esperienze precedenti con la realtà virtuale, mentre per alcuni si trattava del primo utilizzo di un visore VR.

Una porzione iniziale del campione proviene da ambiente lavorativo tecnico (ALTEC), mentre i restanti partecipanti appartengono a un contesto eterogeneo, prevalentemente lavoratori tra i 25 e i 30 anni (Fig. 5.1).



**Figura 5.1:** Distribuzione di età, genere e frequenza di utilizzo del PC e della VR fra il campione.

### 5.1.4 Procedura sperimentale

La procedura seguita durante ogni sessione può essere riassunta come segue:

1. Introduzione all'applicazione e spiegazione preliminare dei comandi.
2. Indossamento del visore e regolazione da parte dell'operatore.
3. Avvio della simulazione.
4. Fase di familiarizzazione con movimento e rotazione.
5. Esplorazione libera iniziale.
6. Introduzione guidata al puntamento o alla ricerca di un pianeta (spesso Saturno, scelto per l'impatto visivo).
7. Attivazione delle orbite e regolazione della velocità del tempo.
8. Prova del camera lock e successivo sgancio.
9. Esplorazione di ulteriori stelle tramite puntamento.
10. Eventuale navigazione manuale senza sistemi assistiti.
11. Presentazione del buco nero Sagittarius A\* come elemento finale di forte impatto.
12. Conclusione libera della sessione.

Non si sono verificati casi di interruzione anticipata volontaria; al contrario, la maggior parte degli utenti ha mostrato interesse a prolungare l'esperienza.

## 5.2 Analisi quantitativa dei questionari

L'analisi dei questionari post-esperienza ha l'obiettivo di valutare l'intuitività dell'interfaccia, il grado di controllo percepito, il comfort fisico e l'efficacia complessiva della modalità VR come strumento di esplorazione astronomica. I dati raccolti provengono da circa quaranta partecipanti, con una prevalenza nella fascia 25–34 anni e con livelli eterogenei di familiarità sia con la realtà virtuale sia con simulatori astronomici.

L'analisi viene articolata in sottosezioni tematiche.

### 5.2.1 Profilo del campione

Il campione è composto prevalentemente da utenti che utilizzano abitualmente il computer in modo quotidiano, elemento che riduce la variabile legata alla scarsa alfabetizzazione digitale.

La familiarità con la realtà virtuale risulta eterogenea: una parte dei partecipanti era alla prima esperienza VR, mentre altri avevano già utilizzato visori in modo occasionale o frequente.

La maggior parte non aveva mai utilizzato Astra Data Navigator in precedenza, né altri simulatori astronomici in VR, condizione che rende le valutazioni particolarmente significative in termini di prima impressione e curva di apprendimento iniziale.

Va inoltre considerato che il campione non è stato selezionato con criterio statistico randomizzato, ma costituisce un gruppo misto composto da utenti provenienti da contesto professionale e rete personale dell'autore.

### 5.2.2 Intuitività dell'interfaccia

Le valutazioni relative all'intuitività generale dell'interfaccia risultano mediamente elevate (prevalentemente valori 4 e 5 su scala 1–5).

Anche la facilità di orientamento nello spazio e la comprensione di ciò che veniva visualizzato hanno ricevuto punteggi medi alti, indicando che la rappresentazione spaziale e il sistema di riferimento risultano generalmente chiari anche per utenti non esperti.

Dalle risposte aperte emergono tuttavia alcune criticità puntuali:

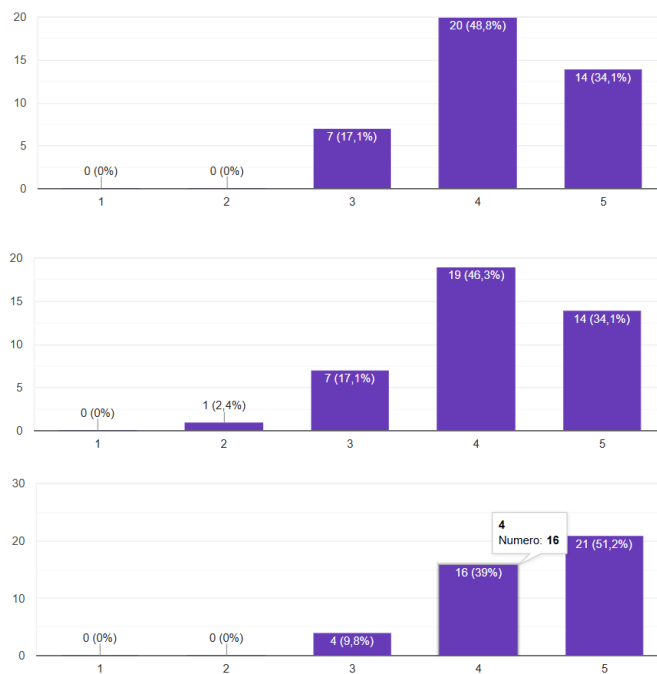
- richiesta di maggiore evidenza visiva per alcuni elementi (tag, indicatori);
- suggerimenti relativi alla chiarezza dei controlli nei menu;
- proposta di feedback aggiuntivi (vibrazione o segnale audio) alla pressione dei comandi.

Questi elementi indicano che l'architettura generale dell'interfaccia è percepita come comprensibile, ma può essere migliorata in termini di feedback e visibilità dello stato del sistema.

### 5.2.3 Navigazione e controllo dei movimenti

Il sistema di navigazione (movimento, rotazione, spostamenti rapidi) ha ricevuto valutazioni mediamente positive, con una leggera variabilità maggiore rispetto ai punteggi sull'interfaccia.

Il senso di controllo dei movimenti è stato generalmente percepito come buono o molto buono (Fig. 5.2), segnale che il mapping degli analogici e la gestione della camera risultano coerenti per la maggior parte degli utenti.



**Figura 5.2:** Valutazioni da parte del campione sull'intuitività dell'interfaccia utente e sul senso di controllo sui movimenti e di orientamento nello spazio.

Tuttavia, emergono osservazioni ricorrenti:

- richiesta di inversione degli analogici (movimento su sinistro, rotazione su destro), in linea con lo standard videoludico;
- percezione di lag durante alcune rotazioni;

- difficoltà nel comprendere intuitivamente la velocità di spostamento nello spazio;
- suggerimenti per rendere il sistema di orbiting più fluido o più libero nello spazio tridimensionale.

Un dato significativo è che il ciclo puntamento-warp è risultato particolarmente apprezzato e frequentemente utilizzato spontaneamente dagli utenti, suggerendo che questo paradigma di navigazione rappresenti uno dei punti di forza dell'esperienza.

### 5.2.4 Comfort e tollerabilità dell'esperienza VR

La maggior parte dei partecipanti ha dichiarato di non aver provato alcun fastidio fisico durante l'esperienza (Fig. 5.3).

Nei casi in cui è stato segnalato un lieve o moderato disagio, questo si è manifestato principalmente:

- durante le rotazioni della visuale;
- durante il movimento nello spazio;
- in rari casi durante l'aggancio a oggetti.

È rilevante osservare che diversi utenti hanno specificato che il lieve senso di disorientamento è tipico delle esperienze VR in generale e non attribuibile specificamente all'applicazione.

Non si sono verificati casi di interruzione anticipata della prova per nausea o malessere, e la permanenza media (15-20 minuti) è risultata stabile o addirittura prolungata per interesse spontaneo.



**Figura 5.3:** Grafici a torta sui feedback ricevuti riguardo eventuali sensazioni di nausea o fastidio provate durante l'esperienza.

### 5.2.5 Valutazione complessiva e percezione della VR come strumento divulgativo

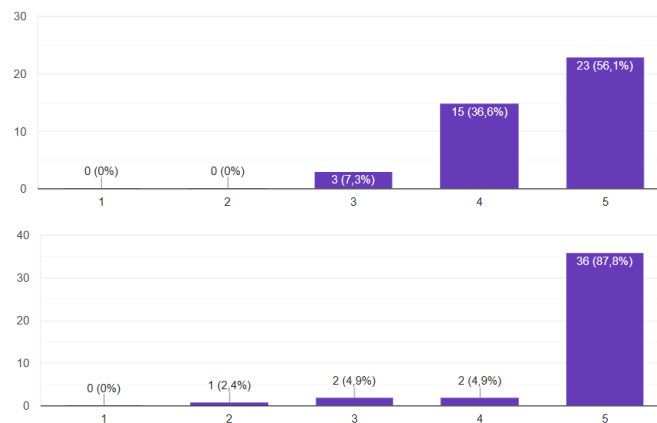
La valutazione complessiva dell'esperienza risulta molto elevata, con una forte concentrazione di punteggi massimi (Fig. 5.4).

Anche la percezione della VR come strumento efficace per la divulgazione astronomica ha ricevuto valutazioni quasi unanimemente positive.

Dal punto di vista qualitativo, gli utenti hanno mostrato particolare interesse per:

- l'osservazione ravvicinata dei pianeti (in particolare Saturno);
- la possibilità di esplorare stelle differenti;
- l'esperienza legata al buco nero Sagittarius A\*;
- la libertà di movimento nello spazio tridimensionale.

Gli elementi meno esplorati spontaneamente risultano invece i filtri grafici, gli assi di riferimento e alcune opzioni avanzate, suggerendo che l'attrattiva primaria dell'esperienza sia legata alla navigazione immersiva piuttosto che alla manipolazione di parametri grafici.



**Figura 5.4:** Valutazioni dell'esperienza con Astra Data Navigator VR e dell'efficacia della VR per la divulgazione in ambito astronomico.

## 5.3 Analisi qualitativa e interpretazione critica

I risultati raccolti attraverso i questionari e l'osservazione diretta delle sessioni di prova consentono di formulare alcune considerazioni qualitative sull'efficacia

dell'estensione VR di Astra Data Navigator, evidenziandone punti di forza e margini di miglioramento.

### 5.3.1 Usabilità e curva di apprendimento

Le prove hanno evidenziato una buona accessibilità iniziale del sistema, a condizione che venga fornita una breve introduzione preliminare ai controlli. Una volta compreso il funzionamento dei joystick, del menu principale e del sistema di puntamento, la maggior parte degli utenti è risultata autonoma nell'esplorazione.

La distinzione tra movimento libero e orbiting è stata generalmente compresa senza difficoltà, così come il meccanismo di camera lock. Le incertezze più frequenti hanno riguardato:

- la regolazione della velocità di movimento (tendenza a mantenere premuto il pulsante anziché utilizzare piccoli incrementi);
- l'apertura del menu in assenza di istruzioni iniziali;
- l'interazione con la tastiera virtuale tramite near/far interactor.

Tali criticità non hanno tuttavia compromesso l'esperienza complessiva, risultando superabili con una breve spiegazione iniziale. Questo suggerisce che il sistema possiede una curva di apprendimento contenuta, pur non essendo ancora completamente autoesplicativo.

### 5.3.2 Comfort e stabilità dell'esperienza immersiva

Dal punto di vista del comfort, i dati mostrano un livello di tollerabilità generalmente elevato. I casi di lieve o moderato fastidio sono risultati limitati e prevalentemente associati alle rotazioni della visuale o a movimenti prolungati nello spazio, dinamiche tipiche della realtà virtuale e non specifiche dell'applicazione.

Non sono stati osservati fenomeni significativi di micro-jitter o instabilità dei controlli. Episodi sporadici di calo di framerate sono stati rilevati soprattutto nelle prime esecuzioni dopo l'avvio del sistema, suggerendo una possibile correlazione con il caricamento iniziale delle risorse piuttosto che con limiti strutturali dell'architettura.

Un aspetto emerso riguarda la gestione della velocità di movimento e della scala temporale. In particolare:

- la regolazione lineare dello slider temporale rende poco percepibili le differenze ai valori più elevati;
- la gestione manuale della velocità nello spazio può risultare complessa quando si tenta di raggiungere oggetti molto distanti tramite movimento libero.

Queste osservazioni indicano possibili direzioni di miglioramento (ad esempio una scala logaritmica per il tempo o l'introduzione di sistemi di velocità adattiva), senza tuttavia compromettere la stabilità generale del sistema.

### 5.3.3 Coinvolgimento e valore esperienziale

L'elemento che ha generato maggiore impatto positivo è stato il ciclo costituito da puntamento, warp e osservazione immersiva. Il momento dell'arrivo in prossimità di un corpo celeste, in particolare Saturno o Sagittarius A\*, ha suscitato reazioni di sorpresa e forte coinvolgimento.

In modo inatteso, anche la visualizzazione delle orbite ha contribuito significativamente alla percezione di "pienezza" della scena, compensando la naturale rarefazione spaziale dovuta alle enormi distanze astronomiche.

L'interesse suscitato dalla possibilità di esplorare liberamente, selezionare stelle rare (come quelle bianche) e osservare il moto orbitale suggerisce che l'estensione VR riesca a trasformare un dataset astronomico in un'esperienza spaziale percepita come concreta e immersiva.

Pur riconoscendo che la presenza dell'operatore possa aver facilitato l'esperienza iniziale, l'osservazione diretta dei comportamenti e la permanenza media volontaria di circa 15–20 minuti indicano un coinvolgimento autentico e sostenuto nel tempo.

## 5.4 Test sulle prestazioni e stabilità del sistema

### 5.4.1 Prestazioni durante la simulazione e la navigazione

Per completare la valutazione delle prestazioni dell'applicazione sono stati eseguiti ulteriori test in tre condizioni operative differenti: simulazione a velocità normale, simulazione alla velocità massima e navigazione tramite warp. L'obiettivo di queste prove era verificare il comportamento del sistema durante l'esecuzione della simulazione astronomica e durante le operazioni di navigazione nello spazio virtuale, che rappresentano due delle situazioni più frequenti durante l'utilizzo dell'applicazione.

Come nei test precedenti, l'analisi è stata condotta utilizzando il Unity Profiler, osservando principalmente il tempo di generazione dei frame, l'utilizzo della CPU e della GPU e l'eventuale presenza di picchi o instabilità durante l'esecuzione.

#### Simulazione a velocità di movimento iniziale

Nel primo scenario la simulazione è stata eseguita alla velocità standard prevista dall'applicazione. In queste condizioni il tempo medio di generazione di un frame si mantiene attorno a 11–12 ms, valore che corrisponde a un framerate di circa

85–90 FPS. Il tempo di attività della CPU risulta pari a circa 23,781 ms nel frame analizzato, mentre il tempo di elaborazione della GPU si mantiene su valori significativamente inferiori, intorno a 4,905 ms.

L'analisi del rendering mostra un carico grafico relativamente contenuto, con circa 2,72 milioni di triangoli e poco più di 200 batch di rendering per frame. Anche l'utilizzo della memoria rimane stabile durante l'intero test, con un consumo totale intorno a 4,59 GB, senza evidenze di crescita progressiva o allocazioni anomale da parte del garbage collector.

Nel complesso, la simulazione a velocità normale presenta un comportamento fluido e stabile, senza variazioni significative del frame time né fenomeni di instabilità percepibili durante l'esecuzione.

### **Simulazione alla velocità di movimento massima**

Nel secondo test la velocità della simulazione è stata impostata al valore massimo disponibile. Questa configurazione comporta un aggiornamento più rapido dei parametri temporali del sistema e, di conseguenza, un maggior numero di operazioni di calcolo legate all'aggiornamento delle orbite e degli stati dei corpi celesti.

Nonostante l'aumento del carico computazionale, il sistema mantiene prestazioni molto simili a quelle osservate nel caso precedente. Il frame time medio rimane infatti nell'intervallo 10–11 ms, con un framerate stimato ancora compreso tra 85 e 90 FPS. Il tempo di attività della CPU nel frame analizzato è pari a circa 22,736 ms, mentre la GPU impiega circa 5,114 ms.

Dal punto di vista grafico il numero di elementi renderizzati rimane comparabile al caso precedente, con circa 2,56 milioni di triangoli e circa 300 batch per frame. Anche in questo scenario l'utilizzo della memoria rimane stabile, con un consumo totale intorno a 4,39 GB e senza allocazioni significative da parte del garbage collector.

I risultati indicano quindi che l'aumento della velocità della simulazione non introduce variazioni sostanziali nelle prestazioni complessive dell'applicazione, suggerendo che il sistema di aggiornamento temporale è implementato in modo efficiente.

### **Navigazione tramite warp**

L'ultimo test ha riguardato la navigazione tramite warp, ovvero il teletrasporto rapido della camera in un punto diverso dello spazio virtuale. Questa operazione comporta un aggiornamento immediato della scena e della posizione della camera, con conseguente ricalcolo degli oggetti visibili e dei parametri di rendering.

Durante l'attivazione del warp si osserva un picco temporaneo nel tempo di attività della CPU, che nel frame analizzato raggiunge circa 38,851 ms, superando momentaneamente il valore di riferimento di 33,333 ms corrispondente a 30 FPS.

Questo comportamento è dovuto principalmente al lavoro eseguito nel Main Thread, necessario per aggiornare la scena e ricalcolare gli elementi visibili.

Tuttavia, il picco risulta limitato a un singolo frame o a un numero molto ridotto di frame consecutivi. Subito dopo l'operazione di warp, il sistema torna rapidamente a valori di frame time comparabili a quelli osservati negli altri test, mantenendosi nuovamente intorno a 11–12 ms.

Dal punto di vista del rendering la scena raggiunge circa 5,76 milioni di triangoli e circa 528 batch, valori più elevati rispetto agli altri scenari a causa della diversa porzione di scena visibile dopo il teletrasporto. Nonostante ciò, l'utilizzo della memoria rimane stabile, con un consumo totale intorno a 3,71 GB e allocazioni minime del garbage collector (circa 9 KB per frame).

Nel complesso, i risultati indicano che il sistema di navigazione tramite warp introduce un picco computazionale breve e circoscritto, ma non comporta degradazioni persistenti delle prestazioni né instabilità durante l'utilizzo dell'applicazione.

### 5.4.2 Stress test dell'interfaccia e del sistema temporale

Al fine di valutare la stabilità dell'applicazione in condizioni di utilizzo intensivo delle componenti interattive, sono stati condotti tre stress test distinti. L'obiettivo non era simulare un uso realistico, bensì forzare ripetutamente le funzionalità più critiche dell'interfaccia e del controllo temporale per osservare eventuali degradazioni delle prestazioni.

Le metriche considerate sono state principalmente il frame time medio (espresso in millisecondi) e la sua variabilità nel tempo, da cui è possibile stimare il framerate effettivo ( $\text{FPS} \approx 1000 / \text{frame time}$ ).

#### Stress UI - Apertura/chiusura e interazioni multiple

Il primo test ha riguardato l'interfaccia grafica nel suo complesso (attivazione pannelli, toggle, pulsanti), simulando interazioni rapide e ripetute.

Durante il test:

- Il frame time medio si è mantenuto intorno ai 10-11 ms;
- Il framerate risultante è quindi rimasto stabile intorno ai 90 FPS;
- Non sono stati osservati picchi prolungati superiori ai 13-14 ms.

Questo comportamento indica che l'interfaccia, pur coinvolgendo numerosi elementi UI e aggiornamenti grafici, non introduce carichi significativi sulla CPU né sulla pipeline di rendering.

Dal punto di vista qualitativo, il sistema ha mantenuto fluidità costante e non sono stati rilevati fenomeni di micro-stuttering visibile all'utente.

### **Stress Time - Play/Pause ripetuto**

Il secondo test ha riguardato l'attivazione e disattivazione ripetuta della simulazione (play/pause), operazione che comporta:

- abilitazione/disabilitazione dell'aggiornamento fisico
- aggiornamento dello stato globale della simulazione
- sincronizzazione dei sistemi dipendenti dal tempo

Anche in questo caso:

- Il frame time medio è rimasto nell'intervallo 10-12 ms;
- Il framerate si è mantenuto stabilmente intorno ai 90 FPS;
- Sono stati osservati sporadici picchi brevi fino a circa 14-15 ms, non persistenti.

È importante sottolineare che tali picchi si verificano in corrispondenza del cambio di stato della simulazione e non durante l'esecuzione continua. Non si è verificato alcun accumulo progressivo di carico, segno che il sistema non introduce overhead cumulativo.

### **Stress Time - Variazione continua della Simulation Speed**

Il terzo test ha riguardato la modifica continua e rapida della velocità di scorrimento del tempo tramite slider.

Questa operazione è potenzialmente più critica, poiché comporta:

- aggiornamento della scala temporale globale
- ricalcolo continuo delle velocità orbitali
- propagazione dei nuovi parametri ai sistemi dipendenti dal tempo

Durante il test:

- Il frame time medio si è attestato tra 11 e 12 ms;
- Il framerate è rimasto intorno agli 85-90 FPS;
- Non sono stati osservati spike superiori ai 15 ms in modo persistente.

Si evidenzia un leggero incremento medio rispetto al test UI puro, coerente con il maggiore numero di operazioni matematiche eseguite. Tuttavia, la variazione è contenuta e non compromette la fluidità.

## Considerazioni complessive sui tre test

I risultati dei test mostrano un comportamento complessivamente stabile e coerente. Il tempo medio di generazione di un frame si mantiene tra 10 e 12 ms, corrispondente a un framerate di circa 85-90 FPS, senza evidenze di degradazione progressiva delle prestazioni o comportamenti anomali anche sotto stress ripetuto.

Questi dati suggeriscono che la gestione della simulazione temporale è ben separata dal livello dell'interfaccia e che gli aggiornamenti non introducono operazioni particolarmente onerose o accumulo di stato nel tempo. Nel complesso, i test indicano quindi una buona robustezza strutturale dell'applicazione in ambiente PC+VR, anche in condizioni di utilizzo intenzionalmente intensive.

### 5.4.3 Test di risposta agli input

#### Metodo di misurazione

Per valutare la reattività del sistema di interazione sono stati eseguiti test specifici finalizzati a misurare il tempo intercorrente tra la ricezione di un input da parte dell'utente e l'attivazione dell'azione corrispondente all'interno dell'applicazione.

La misurazione è stata implementata tramite un sistema di logging dedicato integrato tra il gestore degli input dell'applicazione e le funzioni che eseguono le azioni associate. Per ogni evento vengono registrati due timestamp: il momento in cui l'input viene rilevato e il momento in cui la funzione che implementa l'azione viene effettivamente eseguita. Il tempo di risposta è quindi calcolato come differenza tra questi due valori.

Per evitare interferenze con la scala temporale della simulazione, i timestamp sono stati registrati utilizzando la funzione `Time.realtimeSinceStartup`, che misura il tempo reale trascorso dall'avvio dell'applicazione indipendentemente dal valore di `Time.timeScale`.

I test sono stati eseguiti durante sessioni di utilizzo dell'applicazione che includevano movimento libero, rotazione della camera, selezione di oggetti e utilizzo della funzione di warp. Complessivamente sono stati registrati oltre 66.000 eventi di input, permettendo di ottenere una base di dati sufficientemente ampia per l'analisi.

#### Risultati

L'analisi dei log mostra che gli input più frequenti sono quelli associati alla navigazione nello spazio virtuale. In particolare sono stati registrati oltre 60.000 eventi di movimento e circa 5.700 eventi di rotazione, a conferma del fatto che la navigazione rappresenta la principale modalità di interazione con l'ambiente.

Gli input continui presentano durate medie relativamente elevate, ad esempio circa 78 s per il movimento e 73 s per la rotazione. Questo comportamento è atteso, poiché tali input rimangono attivi per l'intera durata dell'interazione dell'utente e la durata registrata corrisponde quindi al tempo complessivo dell'azione piuttosto che alla latenza del sistema.

Gli input discreti mostrano invece tempi significativamente più ridotti. In particolare il comando di warp presenta una latenza media di circa 0,012 s (circa 12 ms) e un valore massimo inferiore a 0,04 s, indicando che l'azione viene eseguita quasi immediatamente dopo la ricezione dell'input. Anche il comando LookAt mostra tempi contenuti, con una media di circa 0,10 s.

Oltre alla latenza dell'input di attivazione del warp è stata misurata anche la durata della transizione di teletrasporto nello spazio virtuale. I risultati indicano una durata media di circa 0,68 s, con un valore massimo inferiore a 0,82 s, suggerendo che lo spostamento avviene in tempi brevi e prevedibili senza introdurre interruzioni percepibili nell'esperienza di navigazione.

## Discussione

Nel complesso i risultati indicano che il sistema di gestione degli input mantiene un comportamento stabile anche durante sessioni di utilizzo prolungate e con un numero elevato di eventi registrati. Gli input continui vengono gestiti senza perdita di eventi durante la navigazione nello spazio virtuale, mentre gli input discreti presentano tempi di risposta molto ridotti.

In particolare la latenza osservata per l'attivazione del warp risulta ampiamente inferiore alla durata di un frame nelle tipiche configurazioni VR, suggerendo che l'azione viene percepita dall'utente come praticamente immediata. Questo

Tipo	Input	Eventi	Tempo medio (s)	Tempo massimo (s)
Continuo	Move	60541	78,39	269,69
Continuo	Rotate	5776	73,28	287,77
Continuo	PointerPosition	164	137,10	285,41
Discreto	Pick	388	103,51	205,23
Discreto	WarpTo	47	<b>0,012</b>	<b>0,038</b>
Discreto	LookAt	3	0,102	0,119
Transizione	WarpDuration	47	0,68	0,82

**Tabella 5.1:** Tempi registrati tra input e risposta del sistema durante i test di interazione. Gli input continui rappresentano la durata dell'interazione dell'utente, mentre per gli input discreti il valore corrisponde alla latenza tra input e attivazione dell'azione.

comportamento contribuisce a mantenere un'interazione fluida e reattiva durante l'esplorazione dell'ambiente immersivo.

## 5.5 Sintesi finale della valutazione

Le attività di valutazione presentate in questo capitolo hanno permesso di analizzare l'estensione VR di Astra Data Navigator sotto tre prospettive complementari: l'esperienza utente, la qualità dell'interazione e la stabilità tecnica del sistema. L'integrazione di test con utenti reali, analisi qualitative e verifiche prestazionali consente quindi di formulare una valutazione complessiva dell'efficacia della soluzione sviluppata.

Dal punto di vista dell'esperienza utente, i risultati dei questionari indicano un livello generalmente elevato di soddisfazione. L'interfaccia è risultata comprensibile anche per utenti privi di esperienza precedente con l'applicazione, mentre il sistema di navigazione nello spazio tridimensionale è stato percepito come efficace nella maggior parte dei casi. In particolare, il ciclo di interazione basato su puntamento, warp e osservazione ravvicinata dei corpi celesti è emerso come uno degli elementi più apprezzati dell'esperienza. Le osservazioni raccolte evidenziano tuttavia alcune possibili direzioni di miglioramento, soprattutto in termini di chiarezza dei controlli, feedback delle azioni e gestione della velocità di movimento nello spazio.

Per quanto riguarda il comfort dell'esperienza immersiva, la maggioranza dei partecipanti non ha riportato disagi significativi durante le sessioni di prova. I rari casi di lieve disorientamento sono risultati principalmente associati alle rotazioni della visuale o ai movimenti nello spazio, fenomeni comunemente osservati nelle applicazioni di realtà virtuale. La durata media delle sessioni, compresa tra 15 e 20 minuti e spesso prolungata spontaneamente dagli utenti, suggerisce comunque un buon livello di tollerabilità complessiva.

Le analisi prestazionali condotte tramite Unity Profiler confermano la stabilità tecnica dell'applicazione. Nei diversi scenari di simulazione e navigazione il tempo medio di generazione di un frame si mantiene generalmente nell'intervallo tra 10 e 12 ms, corrispondente a un framerate di circa 85–90 FPS. Anche in condizioni di stress dell'interfaccia e del sistema temporale non sono state osservate degradazioni progressive delle prestazioni né comportamenti anomali. Le operazioni più onerose, come la navigazione tramite warp, generano soltanto picchi computazionali temporanei limitati a pochi frame, senza effetti persistenti sulla fluidità dell'esperienza.

Ulteriori verifiche hanno riguardato la reattività del sistema di input. L'analisi dei log mostra che gli input discreti presentano tempi di risposta molto ridotti: ad esempio il comando di warp viene eseguito mediamente dopo circa 0,012 s dalla ricezione dell'input. Questo valore è inferiore alla durata di un frame nelle normali

condizioni di utilizzo VR, indicando che l'azione risulta percepita dall'utente come praticamente immediata. Anche durante sessioni con oltre 66.000 eventi registrati non sono emerse perdite di eventi o instabilità nel sistema di gestione degli input.

Nel complesso, i risultati suggeriscono che l'estensione VR di Astra Data Navigator riesce a combinare efficacemente la visualizzazione scientifica dei dati astronomici con un'esperienza immersiva accessibile e tecnicamente stabile. Pur in presenza di alcuni margini di miglioramento — in particolare nella progettazione dell'interfaccia e nella gestione di alcuni parametri di navigazione — l'architettura sviluppata dimostra una buona robustezza sia dal punto di vista dell'usabilità sia dal punto di vista delle prestazioni. Questi elementi indicano che la realtà virtuale può rappresentare un valido strumento per l'esplorazione e la divulgazione dei dati astronomici, offrendo modalità di interazione che rendono percepibili strutture e fenomeni difficilmente comprensibili attraverso rappresentazioni bidimensionali tradizionali.

## Capitolo 6

# Conclusioni, risultati e sviluppi futuri

Questo capitolo presenta una sintesi dei risultati ottenuti nello sviluppo dell'interfaccia immersiva per Astra Data Navigator e ne discute le principali implicazioni. Dopo una valutazione dell'esperienza d'uso e delle soluzioni di interazione adottate, vengono analizzate le limitazioni tecniche emerse durante il progetto e le possibili direzioni di evoluzione del sistema. Il capitolo si conclude con alcune considerazioni sul ruolo della realtà virtuale nella visualizzazione e nella divulgazione dei dati astronomici.

### 6.1 Valutazione dei risultati dell'esperienza immersiva

Lo sviluppo dell'interfaccia immersiva per Astra Data Navigator ha permesso di verificare concretamente l'efficacia delle soluzioni progettuali adottate per l'interazione e la navigazione all'interno dell'ambiente virtuale. Le prove di utilizzo condotte durante le fasi di test hanno evidenziato risultati complessivamente positivi sia in termini di comfort dell'esperienza sia in relazione alla facilità di apprendimento delle modalità di interazione.

#### 6.1.1 Comfort e sostenibilità dell'esperienza

Uno degli aspetti più rilevanti nella progettazione di applicazioni di realtà virtuale riguarda il comfort dell'utente durante l'utilizzo del sistema. Ambienti immersivi caratterizzati da movimenti nello spazio tridimensionale possono infatti generare

fenomeni di disagio, come nausea o affaticamento visivo, soprattutto in presenza di locomozione virtuale o variazioni rapide del punto di vista.

Nel caso dell'applicazione sviluppata, le sessioni di prova hanno mostrato come l'esperienza risulti generalmente confortevole e sostenibile anche per periodi di utilizzo prolungati. I partecipanti hanno riportato soltanto sporadici episodi di lieve fastidio o disorientamento, generalmente limitati alle prime fasi di utilizzo del sistema. Nel complesso, la progettazione dei meccanismi di navigazione e delle modalità di movimento si è dimostrata efficace nel mantenere un buon equilibrio tra libertà di esplorazione e stabilità percettiva.

Questo risultato suggerisce che le scelte progettuali adottate per la gestione della locomozione e della visualizzazione dell'ambiente contribuiscano a ridurre l'insorgenza dei principali fenomeni di discomfort associati all'utilizzo della realtà virtuale, rendendo l'applicazione adatta anche a sessioni di esplorazione relativamente lunghe.

### **6.1.2 Intuitività dell'interazione e qualità dell'esperienza immersiva**

Un secondo aspetto emerso dalle prove di utilizzo riguarda l'intuitività dei metodi di interazione e dell'interfaccia utente sviluppata. Le modalità di controllo e i sistemi di interazione implementati si sono dimostrati generalmente facili da apprendere anche per utenti che non avevano precedentemente utilizzato l'applicazione.

L'introduzione di una breve guida iniziale consente infatti di presentare rapidamente le principali funzionalità del sistema, permettendo agli utenti di familiarizzare con i meccanismi di navigazione e con gli strumenti disponibili. Questa fase di introduzione risulta particolarmente utile in contesti dimostrativi o durante sessioni di prova di breve durata. Una volta comprese le funzioni principali, tuttavia, l'applicazione si presta anche a un utilizzo più esplorativo, nel quale l'utente può interagire liberamente con l'ambiente virtuale e scoprire progressivamente le diverse funzionalità disponibili.

Durante le sessioni di prova è emerso inoltre come l'interazione diretta con gli oggetti e la navigazione nello spazio tridimensionale contribuiscano a generare un forte senso di immersione nell'ambiente simulato. Questo livello di coinvolgimento tende a spostare l'attenzione degli utenti dagli aspetti tecnici dell'interfaccia verso i contenuti rappresentati, favorendo una maggiore concentrazione sulle informazioni astronomiche osservate durante l'esperienza.

Nel complesso, i risultati ottenuti indicano che le soluzioni adottate per l'interazione e per la progettazione dell'interfaccia utente permettono di costruire un'esperienza immersiva accessibile e facilmente esplorabile, nella quale l'utente può concentrarsi principalmente sull'osservazione e sulla comprensione dei fenomeni astronomici rappresentati.

## 6.2 Limitazioni tecniche del sistema

Nonostante i risultati positivi ottenuti in termini di usabilità e capacità di rappresentazione delle relazioni spaziali tra oggetti astronomici, il sistema sviluppato presenta alcune limitazioni tecniche che costituiscono possibili ambiti di miglioramento per sviluppi futuri. Tali limitazioni riguardano principalmente due aspetti: la rappresentazione e l'interazione con oggetti di dimensioni molto ridotte e la gestione dell'orientamento della camera durante le operazioni di orbitamento attorno ai corpi celesti.

### 6.2.1 Visualizzazione e interazione con oggetti di piccole dimensioni

Una prima criticità riguarda la rappresentazione e l'individuazione di oggetti astronomici di dimensioni molto ridotte, come satelliti artificiali o piccoli asteroidi. In un ambiente immersivo che deve rappresentare scale spaziali estremamente ampie, tali oggetti possono risultare difficili da individuare o da selezionare quando la loro dimensione apparente diventa molto ridotta rispetto alla scena complessiva.

Questo problema è in parte intrinseco alla visualizzazione di dataset astronomici, nei quali oggetti caratterizzati da dimensioni e distanze estremamente diverse devono essere rappresentati all'interno dello stesso spazio tridimensionale. Per mantenere prestazioni adeguate in ambiente VR è inoltre necessario limitare il livello di dettaglio grafico e il numero di elementi simultaneamente visibili, al fine di garantire un frame rate stabile e prevenire fenomeni di discomfort visivo.

Di conseguenza, la rappresentazione di alcune categorie di oggetti minori risulta attualmente semplificata o limitata in termini di visibilità. Sebbene questo compromesso consenta di preservare la fluidità dell'esperienza immersiva, rappresenta una limitazione nella rappresentazione completa di alcune componenti del sistema solare, come ad esempio la popolazione di piccoli corpi presenti nella fascia degli asteroidi o nelle orbite artificiali terrestri.

### 6.2.2 Stabilità del sistema di orbitamento della camera

Una seconda limitazione riguarda la gestione del sistema di orbitamento della camera attorno ai corpi celesti selezionati. In determinate configurazioni di orientamento possono verificarsi situazioni riconducibili al fenomeno noto come gimbal lock, che si manifesta quando specifiche combinazioni di rotazioni portano alla perdita di un grado di libertà nella rappresentazione dell'orientamento della camera.

Nonostante l'implementazione utilizzi rotazioni basate su quaternioni per ridurre la probabilità di tali condizioni, alcune configurazioni limite possono comunque emergere durante l'osservazione di oggetti caratterizzati da traiettorie orbitali

complesse o durante rapide variazioni dell'orientamento della visuale. Questo comportamento diventa particolarmente evidente quando l'utente tenta di orbitare attorno a oggetti molto piccoli o difficili da tracciare, come satelliti artificiali o asteroidi di dimensioni ridotte.

Per mitigare temporaneamente questo problema è stata introdotta la possibilità di disattivare manualmente il sistema di camera lock, consentendo all'utente di riprendere il controllo diretto dell'orientamento della visuale. Questa soluzione permette di evitare situazioni di blocco della camera durante l'esplorazione, ma rappresenta principalmente una misura di mitigazione piuttosto che una soluzione definitiva.

L'individuazione sistematica delle configurazioni che possono portare a tali condizioni e lo sviluppo di strategie più robuste per la gestione dell'orientamento della camera costituiscono quindi un aspetto che richiede ulteriori approfondimenti. Questo risulta particolarmente rilevante nel contesto di un ambiente immersivo destinato alla visualizzazione di oggetti astronomici su scale molto diverse, dove la stabilità del sistema di osservazione rappresenta un requisito fondamentale per garantire un'esperienza di esplorazione efficace.

## **6.3 Sviluppi futuri**

L'applicazione sviluppata in questo lavoro rappresenta un primo passo verso la realizzazione di un ambiente immersivo dedicato alla visualizzazione e all'esplorazione di dati astronomici. Tuttavia, le potenzialità di evoluzione del sistema sono numerose e riguardano sia l'estensione delle funzionalità disponibili sia il miglioramento della qualità della simulazione e dell'esperienza immersiva complessiva. Gli sviluppi futuri possono essere individuati principalmente in quattro direzioni: l'espansione dei dataset astronomici integrati, il miglioramento della rappresentazione grafica della simulazione, l'introduzione di nuovi fenomeni cosmici esplorabili e l'integrazione con tecnologie emergenti come la realtà aumentata.

### **6.3.1 Estensione dei dataset e delle funzionalità di esplorazione**

Uno degli sviluppi più rilevanti riguarda l'integrazione di cataloghi astronomici più completi rispetto a quelli attualmente utilizzati. La struttura del software consente infatti di estendere relativamente facilmente il numero e la varietà di oggetti celesti visualizzati, qualora siano disponibili dataset adeguati. L'integrazione di cataloghi stellari più estesi permetterebbe di rappresentare regioni della Via Lattea con una densità maggiore di stelle e con una maggiore varietà di informazioni associate agli oggetti osservati.

Parallelamente, potrebbe essere integrata anche in modalità immersiva la funzionalità di atterraggio sui corpi celesti, già presente nella versione desktop di Astra Data Navigator. Questa funzione consentirebbe all'utente di passare da una visione macroscopica del sistema solare a un'esplorazione ravvicinata delle superfici planetarie, rendendo possibile una navigazione multi-scala che va dalle strutture cosmiche di grandi dimensioni fino all'osservazione locale delle superfici dei pianeti o dei satelliti naturali. L'integrazione di questa funzionalità in ambiente VR rappresenterebbe un passo importante verso un'esperienza di esplorazione astronomica ancora più completa.

### **6.3.2 Miglioramento della rappresentazione grafica**

Un'altra importante area di sviluppo riguarda il miglioramento del realismo visivo della simulazione. Sebbene l'implementazione attuale privilegi la stabilità delle prestazioni e la leggibilità dei dati, esistono diverse possibilità per incrementare la qualità grafica della rappresentazione degli oggetti astronomici.

Tra gli interventi più significativi rientra il miglioramento della rappresentazione degli anelli dei pianeti gassosi, che potrebbero essere resi tramite tecniche di rendering più avanzate capaci di simulare con maggiore accuratezza la distribuzione delle particelle e gli effetti di illuminazione. Analogamente, la fascia principale degli asteroidi potrebbe essere rappresentata con un numero maggiore di oggetti visibili e distinguibili, ad esempio attraverso l'utilizzo di sistemi di particelle ottimizzati o tecniche di instancing che permettano di gestire grandi quantità di elementi mantenendo prestazioni adeguate.

Questi miglioramenti contribuirebbero a rendere l'ambiente simulato più credibile dal punto di vista visivo e ad aumentare il senso di immersione percepito dagli utenti, mantenendo al tempo stesso il ruolo principale dell'applicazione come strumento di visualizzazione e comprensione dei dati astronomici.

### **6.3.3 Espansione dei fenomeni astronomici rappresentati**

Un ulteriore sviluppo possibile consiste nell'introduzione di nuove tipologie di fenomeni astronomici esplorabili all'interno della simulazione. Attualmente l'applicazione si concentra principalmente sulla rappresentazione dei corpi celesti del sistema solare e delle stelle presenti nei cataloghi integrati, ma l'ambiente potrebbe essere ampliato includendo una maggiore varietà di strutture cosmiche.

Tra gli elementi che potrebbero essere integrati in futuro vi sono, ad esempio, nebulose, comete e buchi neri, rappresentati attraverso modelli visivi e simulazioni semplificate che ne evidenzino le principali caratteristiche fisiche e morfologiche. L'introduzione di questi fenomeni permetterebbe di ampliare notevolmente il potenziale educativo e divulgativo dell'applicazione, offrendo agli utenti la possibilità

di osservare e comprendere strutture cosmiche che non appartengono direttamente al sistema solare ma che rivestono un ruolo fondamentale nello studio dell'universo.

#### **6.3.4 Integrazione con tecnologie di realtà aumentata**

Infine, una possibile evoluzione del sistema riguarda l'integrazione con tecnologie di realtà aumentata. I visori di nuova generazione consentono infatti di combinare elementi virtuali con la percezione dell'ambiente reale attraverso sistemi di passthrough, aprendo nuove possibilità per la fruizione di contenuti scientifici.

L'applicazione potrebbe evolvere verso modalità ibride in cui la simulazione astronomica viene parzialmente integrata con lo spazio fisico circostante. Ad esempio, potrebbero essere introdotte "finestre" virtuali attraverso cui osservare porzioni della simulazione mantenendo la percezione dell'ambiente reale, oppure modalità in cui oggetti astronomici vengono visualizzati direttamente nello spazio fisico davanti all'utente.

Questo tipo di integrazione potrebbe risultare particolarmente interessante in contesti educativi, museali o espositivi, dove la combinazione tra ambiente reale e simulazione digitale permetterebbe di creare esperienze divulgative altamente coinvolgenti e accessibili anche a utenti non esperti.

## 6.4 Considerazioni finali

Il lavoro presentato in questa tesi ha avuto come obiettivo principale lo sviluppo di un'interfaccia immersiva per il software Astra Data Navigator, con l'intento di esplorare le potenzialità della realtà virtuale nella visualizzazione e nell'esplorazione di dati astronomici. Più che un semplice adattamento tecnologico di un'interfaccia esistente, il progetto ha rappresentato un tentativo di verificare come un ambiente immersivo possa contribuire alla comprensione delle relazioni spaziali che caratterizzano l'universo.

Uno degli aspetti più significativi emersi durante lo sviluppo e l'utilizzo del sistema riguarda infatti la capacità della realtà virtuale di rendere percepibili le scale astronomiche. Le distanze tra pianeti, satelliti e stelle sono talmente vaste da risultare spesso difficili da rappresentare e comprendere attraverso strumenti tradizionali basati su schermi bidimensionali. L'immersione all'interno di un ambiente tridimensionale navigabile permette invece all'utente di osservare direttamente le relazioni spaziali tra gli oggetti, rendendo più tangibili le proporzioni e le distanze che caratterizzano il sistema solare e le strutture stellari circostanti.

In questo contesto, l'esperienza immersiva non si limita a migliorare l'interazione con il sistema, ma contribuisce a trasformare la modalità con cui le informazioni astronomiche vengono percepite. Muoversi nello spazio simulato, osservare da vicino pianeti e stelle e comprendere la loro posizione relativa all'interno della scena permette di costruire una rappresentazione mentale più chiara e intuitiva delle strutture cosmiche osservate. La realtà virtuale si configura quindi non soltanto come un mezzo di visualizzazione più coinvolgente, ma come uno strumento capace di facilitare la comprensione di fenomeni che, per loro natura, risultano spesso difficili da immaginare.

Parallelamente, il livello di immersione offerto dalla realtà virtuale contribuisce a stimolare la curiosità e l'interesse verso l'astronomia anche in utenti che non possiedono una formazione specifica in questo ambito. La possibilità di esplorare direttamente un ambiente che rappresenta il sistema solare e lo spazio circostante rende i concetti astronomici meno astratti e più accessibili, favorendo un approccio esplorativo alla conoscenza scientifica. In questo senso, strumenti come Astra Data Navigator possono assumere un ruolo importante anche nel campo della divulgazione scientifica, offrendo modalità di apprendimento basate sull'esperienza diretta piuttosto che sulla semplice osservazione passiva di contenuti informativi.

Guardando alle prospettive future, il potenziale di un sistema come Astra Data Navigator risiede anche nella sua capacità di evolvere insieme all'avanzamento delle conoscenze scientifiche. La possibilità di integrare nuovi dataset, nuovi fenomeni astronomici e nuove tipologie di oggetti celesti consente infatti di immaginare l'applicazione non soltanto come un simulatore, ma come una piattaforma dinamica

per l'esplorazione dell'universo. In un contesto in cui le missioni spaziali e le osservazioni astronomiche continuano ad ampliare rapidamente il nostro patrimonio di conoscenze, un ambiente immersivo aggiornabile potrebbe diventare uno strumento capace di rendere accessibile al pubblico l'evoluzione stessa della ricerca scientifica.

In questa prospettiva, Astra Data Navigator potrebbe progressivamente trasformarsi in un archivio interattivo della conoscenza astronomica contemporanea, nel quale i dati provenienti dalla ricerca scientifica e dalle missioni di esplorazione spaziale vengono resi accessibili attraverso un ambiente tridimensionale esplorabile. Un sistema di questo tipo permetterebbe non solo agli appassionati di seguire e comprendere le più recenti scoperte nel campo dell'astronomia, ma anche a un pubblico più ampio di avvicinarsi alla comprensione dell'universo attraverso un'esperienza immersiva e intuitiva.

La realtà virtuale, in questo contesto, non rappresenta soltanto una tecnologia emergente, ma uno strumento capace di trasformare il modo in cui l'essere umano può osservare, comprendere ed esplorare il cosmo.

# Bibliografia

- [1] B. V. Semenov. «WebGeocalc and Cosmographia: Modern tools to access OPS SPICE data». In: *Proc. 2018 SpaceOps Conf.* Art. no. 2366. 2018 (cit. a p. 7).
- [2] S. Schneegans, M. Zeumer, J. Gilg e A. Gerndt. «CosmoScout VR: A Modular 3D Solar System Based on SPICE». In: *2022 IEEE Aerospace Conf.* Big Sky, MT, USA, 2022, pp. 1–13 (cit. a p. 8).
- [3] U. A. Almeida et al. «SpaceEngine como ferramenta didática para o ensino de astronomia». In: *Revista Latino-Americana de Educação em Astronomia* 35 (2023), pp. 57–76 (cit. a p. 11).
- [4] F. Cuartero, J. Ruiz e J. J. Pardo. «Educational Innovation in Astronomy Teaching». In: *6th WSEAS Int. Conf. on Education and Educational Technology.* Venice, Italy, 2007 (cit. a p. 11).
- [5] S. R. Kusumaningrum, Khamdun e L. A. Riswari. «Exploring the Effectiveness of PjBL in Enhancing Students’ Understanding of Astronomy Concepts Using Universe Sandbox». In: *Mimbar Sekolah Dasar* 12.2 (2025), pp. 307–325 (cit. a p. 12).
- [6] A. T. Hayes e M. Fowler. «User Experience and Intent to Adopt VR across Levels of Immersion: A Case Study of the Flight Simulation Game Elite Dangerous». In: *IS&T Int. Symp. on Electronic Imaging.* 2025 (cit. a p. 13).
- [7] M. Foxman, A. P. Leith, D. Beyea, B. Klebig, V. H. H. Chen e R. Ratan. «Virtual Reality Genres: Comparing Preferences in Immersive Experiences and Games». In: *CHI PLAY ’20 Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play.* Nov. 2020, pp. 244–248 (cit. a p. 14).
- [8] L. Daricello et al. «Discover the secrets of the highest-energy Universe: The CTAO Virtual Tour». In: *Memorie della Società Astronomica Italiana* 96 (2025), p. 70 (cit. a p. 19).

- [9] A. Eidikis e M. Svensson. «Exploring Astronomy in VR». Tesi di laurea mag. Gothenburg, Sweden: Chalmers Univ. Technol. e Univ. of Gothenburg, 2022 (cit. a p. 20).
- [10] Unity Technologies. *TextMeshPro Documentation*. Accessed: 2026-02-18. 2024. URL: <https://docs.unity3d.com/Packages/com.unity.textmeshpro@latest> (cit. a p. 21).
- [11] D. E. Rzig, N. Iqbal, I. Attisano, X. Qin e F. Hassan. «Virtual Reality (VR) Automated Testing in the Wild: A Case Study on Unity-Based VR Applications». In: *Proc. ISSSTA '23*. Seattle, WA, USA, 2023, pp. 1–13 (cit. a p. 22).
- [12] Khronos Group. *OpenXR Specification*. Accessed: 2026-02-18. 2024. URL: <https://registry.khronos.org/OpenXR/specs/1.0/html/xrspec.html> (cit. a p. 23).
- [13] Unity Technologies. *XR Interaction Toolkit Documentation*. Accessed: 2026-02-18. 2024. URL: <https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@latest> (cit. a p. 23).
- [14] Unity Technologies. *XR Hands Documentation*. Accessed: 2026-02-18. 2024. URL: <https://docs.unity3d.com/Packages/com.unity.xr.hands@latest> (cit. a p. 24).
- [15] C. H. Acton. «NASA's SPICE system models the solar system». In: *Dynamics and Astrometry of Natural and Artificial Celestial Bodies*. A cura di J. M. Wyrzyszcak, J. H. Lieske e R. A. Feldman. Dordrecht: Kluwer Academic Publishers, 1997, p. 257 (cit. a p. 25).
- [16] Space Graphics Toolkit. *Space Graphics Toolkit Documentation*. Accessed: 2026-02-18. 2024. URL: <https://carloswilkes.com/Documentation/SpaceGraphicsToolkit> (cit. a p. 25).