



UNIVERSITÀ DI PISA

Progetto Basi di Dati

Simone Gallo, Riccardo Ciucci

Indice

1	Introduzione	6
2	Progettazione Concettuale	7
2.1	Dizionario dell'Entità	7
2.2	Dizionario delle Relazioni	11
2.3	Regole di Vincolo	15
2.4	Regole di Derivazione	15
2.5	Descrizione ER	16
2.5.1	Critico e Utente	16
2.5.2	Vincita e Premio	17
2.5.3	Edizione	17
2.5.4	File	17
2.5.5	FormatoCodifica	17
2.5.6	Qualità di VideoFile	18
2.5.7	Qualità di AudioFile	18
2.5.8	IP Range	18
2.5.9	Server	20
2.5.10	Abbonamento	20
2.5.11	Popolarità di Artista	21
2.5.12	Posizione di un Paese	21
3	Ristrutturazione	22
3.1	Eliminazione delle Generalizzazioni	22
3.1.1	Artista	22
3.1.2	Recensore	22
3.1.3	FormatoCodifica	23
3.2	Eliminazione degli Attributi Multivalore	24
3.2.1	LinguaDoppiaggio	24
3.2.2	LinguaSottotitoli	24
3.3	Analisi Ridondanze	24
3.3.1	Ridondanze presenti	24
3.3.2	Introduzione nuove Ridondanze	24
3.4	Partizionamento e Accorpamento ER	25
3.4.1	Qualità di VideoFile	25
3.4.2	Qualità di AudioFile	25
3.4.3	Posizione	25
3.4.4	Carta di Credito	26

4	Tavole dei Volumi	27
4.1	Area Contenuti	27
4.2	Area Formato	29
4.3	Area Clienti	30
4.4	Area streaming	33
5	Operazioni sui Dati	33
5.1	Analisi delle Ridondanze	33
5.1.1	CaricoAttuale	33
5.1.2	DistanzaPrecalcolata	36
5.1.3	MediaRecensioni NumeroRecensioni	38
5.1.4	Visualizzazioni Giornaliere	40
5.2	Analisi delle Operazioni	44
5.2.1	Vincite di un Film	44
5.2.2	Generi di un Film	46
5.2.3	Massima Risoluzione di un Film	47
5.2.4	Numero di Film non disponibili in un Abbonamento	49
5.2.5	Film disponibili in Lingua specifica	51
5.2.6	Film più visti recentemente	53
5.2.7	Cambio Abbonamento	55
5.2.8	Film con Recensioni migliori	57
6	Progettazione Logica	59
6.1	Ridenominazione Nomi	59
6.2	Traduzione in schema Logico	59
6.3	Vincoli	63
6.3.1	Vincoli di Integrità Referenziale	63
6.3.2	Vincoli di Tupla	64
6.3.3	Vincoli di Dominio	65
6.3.4	Vincoli Inter-Relazionali	65
6.4	Analisi delle Dipendenze Funzionali e Normalizzazione	66
7	Raccomandazione Utente	69
7.1	Rating	69
7.1.1	Rating del Film	69
7.1.2	Rating per Utente	70
7.1.3	Sistema di Raccomandazione	70
8	Area Analytics	72
8.1	Classifiche	72
8.2	Custom Analytics	74
8.3	Bilanciamento carico	76

9	Area Streaming	77
9.1	Individuazione Server	77
9.2	Ribilanciamento del carico	80
9.3	Caching previsionale	81
10	Implementazione Fisica	83
10.1	Note su Vincoli	83
10.2	Indicizzazioni e Motori	83
10.3	Popolamento	83
10.4	Benchmark e Testing	84
10.4.1	Ip2Paese	86
10.4.2	RatingFilm	87
10.4.3	RatingUtente	88
10.5	Trigger e Event	89
10.5.1	AnnoEdizioneValido	89
10.5.2	DataCriticaValido	89
10.5.3	InserimentoFile	89
10.5.4	ModificaFile	89
10.5.5	ModificaErogazione	89
10.5.6	AggiungiErogazione	90
10.5.7	RimuoviErogazione	90
10.5.8	IpRangeControlloAggiornamento	90
10.5.9	InserimentoRecensione	90
10.5.10	ModificaRecensione	90
10.5.11	CancellazioneRecensione	90
10.5.12	GestioneVisualizzazione	91
10.5.13	GestioneConnessione	91
10.5.14	GestioneErogazione	91
10.6	API	91
10.6.1	AggiungiErogazioneServer	91
10.6.2	BilanciamentoDelCarico	91
10.6.3	CachingPrevisionale	92
10.6.4	CalcolaDistanzaPaese	92
10.6.5	CalcolaDistanzaServer	92
10.6.6	CambioAbbonamento	92
10.6.7	FileMiglioreQualita	92
10.6.8	FilmDisponibiliInLinguaSpecifica	93
10.6.9	FilmEsclusiAbbonamento	93
10.6.10	FilmPiuVistiRecentemente	93
10.6.11	GeneriDiUnFilm	93
10.6.12	RimuoviErogazioneServer	93
10.6.13	VinciteDiUnFilm	93

10.6.14 Ip2Paese	94
10.6.15 Ip2PaeseStorico	94
10.6.16 IpAppartieneRangeInData	94
10.6.17 IpRangeCollidono	94
10.6.18 IpRangeValidoInData	94
10.6.19 IpRangePossoInserire	95
10.6.20 IpRangeInserisciFidato	95
10.6.21 IpRangeInserisciAdessoFidato	95
10.6.22 IpRangeProvaInserire	95
10.6.23 IpRangeProvaInserireAdesso	95
10.6.24 VisualizzazioniGiornaliereBuild	96
10.6.25 VisualizzazioniGiornaliereFullBuild	96
10.6.26 MathMap	96
10.6.27 StrListContains	96
10.6.28 CalcolaDelta	96
10.6.29 MigliorServer	97
10.6.30 TrovaMigliorServer	97
10.6.31 RibilanciamentoCarico	97
10.6.32 RatingUtente	98

11 Codice sorgente e Bibliografia	99
--	-----------

1 Introduzione

Si desidera progettare e realizzare una base di dati che permetta sia la memorizzazione di dati ed informazioni relativi a FilmSphere, un servizio internazionale di streaming di contenuti video online, che l'utilizzo di operazioni e funzioni di data analytics che permettono di migliorare l'esperienza dell'utente.

Il gestore del sistema consente l'inserimento di Film e le informazioni ad essi correlate, la memorizzazione dei vari formati utilizzati insieme alle relative versioni di questi ultimi ed, infine, l'immagazzinamento di dati che riguardano gli utenti compresi abbonamento, fatture, con eventuali pagamenti effettuati, e raccomandazione di contenuti attraverso algoritmi di valutazione di contenuti.

Oltretutto il suddetto sistema sovrintende e gestisce anche la CDN, ossia la rete di distribuzione del servizio costituita da server dislocati nel globo, grazie all'implementazione di funzionalità lato server di analisi di dati, come il caching.

Ad ogni componente dello schema ER che riguarda una delle diverse Aree della base di dati viene univocamente associato un colore:

- Area Contenuti
- Area Formati
- Area Clienti
- Area Streaming

2 Progettazione Concettuale

2.1 Dizionario dell'Entità

Entità	Attributi	Identificatore	Descrizione
Film*	Id, Descrizione, Titolo	ID	Film offerto dal servizio di streaming online
Genere	Nome	Nome	Genere dei Film offerti da FilmSpere
Attore*	Nome, Cognome, Popolarità	Nome, Cognome	Specializzazione di Artista, attore recitante in un Film
Regista*	Nome, Cognome, Popolarità	Nome, Cognome	Specializzazione di Artista, regista dirigente un Film
Artista*	Nome, Cognome, Popolarità	Nome, Cognome	Generalizzazione di Attore e Regista
Premio*	Macrotipo, Microtipo	Macrotipo, Microtipo	Premio attribuito a Film, Attori e/o Registi
Vincita*	Data	Data, Macrotipo(FK), Microtipo(FK)	Vincita di un Premio da parte di Film, Attori e/o Registi

Entità	Attributi	Identificatore	Descrizione
Recensore*	Codice, Nome, Cognome	Codice	Generalizzazione di Utente e Critico
Critico*	Codice, Nome, Cognome	Codice	Critico Cinematografico
Casa-Produzione	Nome	Nome	Casa di Produzione Cinematografica
Paese*	Codice, Nome, Posizione	Codice	Paese dal quale provengono dei Film
File*	Id, Dimensione, LingueDoppiaggio, LingueSottotitoli, BitRate, FormatoContenitore	Id	File in cui è stato salvato un Formato di un Film
Edizione	Id, Anno, Rapporto d'Aspetto, Lunghezza	Id	Edizione di un Film (e.g. "Godfather" fu rilasciato una seconda volta nel 1977 con alcune differenze)
Formato-Codifica*	Famiglia, Versione, MaxBitrate, Lossy	Famiglia, Versione	Generalizzazione di Audio e Video, Formato di Codifica e Decodifica
Audio*	Famiglia, Versione, MaxBitrate, Lossy	Famiglia, Versione	Specializzazione di Formato Codifica

Entità	Attributi	Identificatore	Descrizione
Video*	Famiglia, Versione, Anno, Bitrate, Lossy	Famiglia, Versione	Specializzazione di Formato Codifica
Server	Id, Lunghezza Banda, MTU, MaxConessioni, Carico Attuale, Posizione	Id	Server appartenente alla rete distribuita
IP Range	Inizio, Fine, DataInizio, DataFine	Inizio, Fine, DataInizio	Range di IP appartenenti ad una Zona Geografia
Utente	Codice, Nome, Cognome, Email, Password	Codice	Utente registrato al servizio di streaming
Connessione	IP, Inizio, Fine, Hardware	IP, Inizio, Codice(FK)	Connessione al servizio da parte di un Utente
Fattura	ID	ID	Fattura emessa da FilmSphere
Pagamento*	Data, Carta di Credito	ID(FK)	Pagamento effettuato da un Utente per una Fattura
Visualizzazione	Timestamp	Timestamp, ID(FK), Codice(FK)	Visualizzazione di un File da parte di un Utente

Entità	Attributi	Identificatore	Descrizione
Abbonamento	Tipo, Tariffa, Durata, Definizione, Offline, GB Mensili, Max Ore	Tipo	Abbonamento fornito dal servizio di streaming

Le entità indicate con "*" sono generalizzazione, specializzazioni o altre entità che sono state modificate durante la ristrutturazione.

In particolare durante la ristrutturazione sono state eliminate le seguenti entità: **Attore, Regista, Recensore, Audio, Video**.

Sono state modificate le seguenti: **File, Pagamento, Film**.

Mentre sono state aggiunte le seguenti: **Lingua, Carta di Credito, Visualizzazioni Giornaliere**.

Entità	Attributi	Identificatore	Descrizione
Film	ID, Titolo, Descrizione, MediaRecensioni, NumeroRecensioni	ID	Film offerto dal servizio di streaming online
File	ID, Dimensione, BitRate, FormatoContenitore	ID	File in cui è stato salvato un Film
Pagamento	Data	ID(FK)	Pagamento effettuato da un Utente per una Fattura
Lingua	Nome	Nome	Lingua in cui possono essere disponibili sia audio che sottotitoli
Carta di Credito	PAN, Data di Scadenza, CVV	PAN	Carta di Credito utilizzata per un Pagamento
Visualizzazioni Giornaliere	NumeroVisualizzazioni, Data	ID(FK), Codice(FK), Data	Visualizzazioni Giornaliere per Film e per Paese

2.2 Dizionario delle Relazioni

Relazione	Descrizione	Componenti	Attributi
Vincita-Artista	Vincita, da parte di un Artista, di un premio	Vincita (0, 1), Artista (0, N)	-
Vincita-Premio	Vincita di un particolare Premio	Vincita (1, 1), Premio (1, N)	-
Vincita-Film	Vincita, da parte di un Film, di un premio	Vincita (1, 1), Film (0, N)	-
Recitazione*	Recitazione da parte di un Attore in un Film	Film (1, N), Attore (1, N)	-
Regia*	Direzione da parte di un Regista di un Film	Regista (1, N), Film (1, 1)	-
Genere-Film	Appartenenza di un Film ad uno o più Generi	Genere (1, N), Film (1, N)	-
Critica	Valutazione, da parte di un Critico, di un Film	Critico (1, N), Film (0, N)	Testo, Data, Voto
Recensione	Valutazione, da parte di un Utente, di un Film	Utente (0, N), Film (0, N)	Voto
Produzione	Produzione, da parte di una Casa di Produzione, di un Film	CasaPro- duzione (1, N), Film (1, 1)	Anno

Relazione	Descrizione	Componenti	Attributi
Appartenenza	Residenza di una Casa di Produzione in un Paese	CasaProduzione (1, 1), Paese (0, N)	-
Restrizione	Impossibilità di vedere un Edizione in un Paese	Edizione (0, N), Paese (0, N)	-
Edizione-File	Corrispondenza di uno o più File ad un'Edizione di un film	File (1, 1), Edizione (1, N)	-
VideoFile*	Presenza di un Formato Video in un File	Video (1, N), File (1, 1)	Qualità
AudioFile*	Presenza di un Formato Audio in un File	Audio (1, N), File (1, 1)	Qualità
P.o.P.	Presenza di un File in un Server	File (1, N), Server (1, N)	-
Geolocal	Corrispondenza di un IP Range ad un Paese	Paese (1, N), IP Range (1, 1)	-
Erogazione	Erogazione, da parte di un Server, di un file per la Visualizzazione di un utente	Paese (1, N), IP Range (1, 1)	Inizio-Erogazione

Relazione	Descrizione	Componenti	Attributi
Connes- sioneUtente	Corrispondenza tra un Utente e la sua Connessione	Utente (0, N), Conessione (1, 1)	-
Emissione	Emissione di una Fattura per un Utente	Utente (0, N), Fattura (1, 1)	Data
Pagamen- toFattura	Corrispondenza tra una Fattura ed il suo, eventuale, Pagamento	Fattura (0, 1), Pagamento (1, 1)	-
Visualizza- zioneCon- nessione	Corrispondenza tra la Visualizzazione e la Connessione di un utente	Connessione (0, N), Visualizzazione (1, 1)	-
Abbona- mento Utente	Corrispondenza tra un Utente ed il suo ultimo Abbonamento	Utente (1, 1), Abbonamento (0, N)	DataInizio
Visualiz- zazioneEdi- zione	Corrispondenza tra la Visualizzazione e l'Edizione visionata	Visualizzazione (1, 1), Edizione (0, N)	-
Esclusione	Esclusione di un Genere per un Abbonamento	Abbonamento (0, N), Genere (0, N)	-

Le relazioni indicate con "*" sono state modificate durante la ristrutturazione.

Durante la ristrutturazione sono state modificate le seguenti relazioni: **VideoFile**, **AudioFile**, **Regia**, **Recitazione**.

Mentre sono state aggiunte le seguenti: **Doppiaggio**, **Sottotitolaggio** **DistanzaPrecalcolata**, **Pagamento Carta**, **Film Vis Giorni**, **Paese Vis Giorni**.

Relazione	Descrizione	Componenti	Attributi
VideoFile	Presenza di un Formato Video in un File	FormatoCodifica (0,N), File (1,1)	FPS, Risoluzione
AudioFile	Presenza di un Formato Audio in un File	FormatoCodifica (0, N), File (1, 1)	Frequenza, BitDepth
Recitazione	Recitazione da parte di un Artista in un Film	Film (1, N), Artista (0, N)	-
Regia	Direzione da parte di un Artista di un Film	Artista (1, N), Film (1, 1)	-
Doppiaggio	Doppiaggio in una Lingua disponibile in un File	Lingua (0, N), File (1, N)	-
Sottotitolaggio	Sottotitolaggio in una Lingua disponibile in un File	Lingua (0, N), File (1, N)	-
DistanzaPrecalcolata	Distanza Precalcolata tra un Paese e un Server	Paese (1, N), Server (1, N)	Valore Distanza

Relazione	Descrizione	Componenti	Attributi
Pagamento-Carta	Corrispondenza tra un Pagamento e la Carta utilizzata	Pagamento (1, 1), Carta di Credito (1, N)	-
Film Vis Giorni	Corrispondenza tra un Film e le Visualizzazioni Giornaliere di un Paese	Film (1, N), Visualizzazioni Giornaliere (1, 1)	-
Paese Vis Giorni	Corrispondenza tra un Paese e le Visualizzazioni Giornaliere di un Film	Paese (1, N), Visualizzazioni Giornaliere (1, 1)	-

2.3 Regole di Vincolo

Le regole indicate con "*" sono state aggiunte durante la ristrutturazione.

Area	Vincolo	Entità
Contenuti*	Un Artista deve aver recitato o diretto almeno un Film	Film, Artista, Vincita
Contenuti*	Una Lingua deve comparire in almeno un Doppiaggio o in almeno un Sottotitolaggio	Lingua, Formato

2.4 Regole di Derivazione

Le regole indicate con "*" sono state aggiunte durante la ristrutturazione.

Area	Vincolo	Entità
Contenuti	Il rating critico di un Film è uguale alla media dei voti delle Critiche di quel film	Film, Critico
Contenuti	Il rating popolare di un Film è uguale alla media dei voti delle Recensioni di quel film ed è memorizzato come attributo ridondante nell'entità Film	Film, Utente
Streaming*	ValoreDistanza di DistanzaPrecalcolata è uguale alla distanza in km tra Posizione di Paese e Posizione di Server	Server, Paese
Streaming	CaricoAttuale di Server è una ridondanza che indica il numero di Erogazioni attive che impiegano tale Server. Quando una Erogazione viene avviata (inserita nel DB) il contatore aumenta di 1 e quando termina (viene cancellata) diminuisce di 1	Server, Visualizzazione
Streaming	CaricoAttuale di Server deve essere sempre minore di MaxConnessioni, se così non è il Server è in sovraccarico	Server

2.5 Descrizione ER

In questo paragrafo sono trattati nel dettaglio alcuni attributi, entità o relazioni il cui significato o funzionamento potrebbe non risultare chiaro ed evidente guardando il diagramma Entità-Relazioni.

2.5.1 Critico e Utente

Ciascuna delle due entità ha la possibilità di esprimere un giudizio su un Film tramite un Voto. Un **Critico** però non è una persona qualunque: è un delegato di un giornale o un esperto di cinema e della sua **Critica** si memorizza anche un Testo: un commento testuale riguardante il **Film** criticato. Gli utenti che guardano un Film potranno leggere queste critiche prima di guardarlo e decidere quindi se iniziarne la visione o meno.

Entrambe le entità condividono inoltre tre attributi, motivo per cui è stata effettuata la generalizzazione (esclusiva), ma poiché le medie dei voti dei critici e quelle dei voti degli utenti saranno poi distinte, assieme al fatto che solo

un Critico può allegare commenti è stato deciso, in fase di Ristrutturazione, di separare le entità.

2.5.2 Vincita e Premio

Per **Macrotipo** e **Microtipo** si intendono, rispettivamente, la tipologia e la categoria del premio, ad esempio nel caso di "Oscar al miglior film" il **Macrotipo** sarebbe "Oscar", mentre il **Microtipo** "miglior film".

Per **Vincita** si intende un premio relativo ad uno specifico anno (e.g. Oscar al miglior film 2021)

2.5.3 Edizione

Uno stesso **Film** può essere ridistribuito più volte dalla casa di produzione e avere, per ogni nuova distribuzione piccoli o grandi cambiamenti, come l'aggiunta o la rimozione di scene. Un esempio in questo caso può essere *Star Wars: Una Nuova Speranza* di George Lucas che, sebbene sia stato prodotto nel 1977 ha avuto decine di riedizioni (di cui l'ultima nel 2020); tra esse sono state effettuate piccole modifiche a dettagli o anche grandi a personaggi.

Il Rapporto d'Aspetto è un altro attributo che può variare: uno stesso **Film** può averne vari, un esempio è il recente *Oppenheimer* di Christopher Nolan, che è stato rilasciato sia in rapporto d'aspetto IMAX che in 16:9.

Questo fa sì che la durata dello stesso **Film** possa cambiare tra le varie **Edizioni**: entità che ha lo scopo di raccogliere queste informazioni nella Base di Dati.

2.5.4 File

Per **File** si intende un file in cui è stato salvato il contenuto di un edizione di un film. Esistono file, per una stessa **Edizione**, con risoluzione, dimensione e FPS differenti per ottemperare a varie necessità, ad esempio il limite di risoluzione per alcuni abbonamenti.

Con **FormatoContenitore** intendiamo l'estensione del file (e.g. .AVI .mp4).

2.5.5 FormatoCodifica

Per **FormatoCodifica** si intende lo standard utilizzato per codificare dati audio e/o video (e.g. MPEG-4 Part 10 è lo standard più utilizzato dai file MP4)

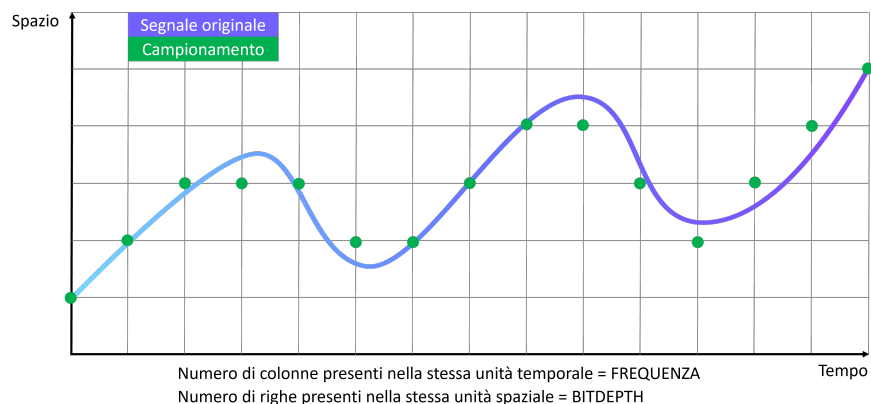
2.5.6 Qualità di VideoFile

La Qualità di un **File** Video viene misurata mediante due valori: Risoluzione e FPS (Frame Per second). La prima è la risoluzione (qualità d'immagine) dei fotogrammi, ovvero il numero di pixel che ciascun fotogramma possiede; il secondo è il numero di fotogrammi che verranno visualizzati ogni secondo. Maggiori sono entrambi questi valori maggiore è la Qualità Video ma anche la dimensione del **File** e il suo BitRate.

2.5.7 Qualità di AudioFile

La Qualità di un File Audio viene misurata mediante due valori: Frequenza e BitDepth. La prima è la frequenza con cui il segnale audio è stato campionato (più alta è più la Qualità è maggiore); la seconda è la precisione con cui ogni punto è individuato (maggiore è il BitDepth maggiore è la Qualità del segnale).

Bisogna segnalare che, tuttavia, **File** con maggiore Qualità Audio hanno anche Dimensione maggiore e maggiore BitRate di fruizione.



2.5.8 IP Range

Vi è la necessità di sapere da dove si connettono gli **Utenti** per avere una migliore fruizione dei contenuti ed anche applicare eventuali restrizioni dovute a leggi.

Se fosse l'utente stesso a comunicare il **Paese** (nazione) dove si trova ci

sarebbe il problema di richieste al sistema con valori falsi: un **Utente** potrebbe dalle impostazioni del proprio dispositivo modificare sia la posizione riportata dal GPS (che diventa quindi inutile alla base di dati) che il **Paese** dove il dispositivo pensa di trovarsi.

L'utente non può però nascondere il suo indirizzo IP (senza usare una VPN) durante la connessione e da esso si può tramite questa entità risalire al **Paese** dove si trova.

L'indirizzo IP di un dispositivo viene assegnato dal proprio ISP (Internet Service Provider, esempi sono TIM e Vodafone) al momento in cui ci si aggancia al ripetitore di quest'ultimo.

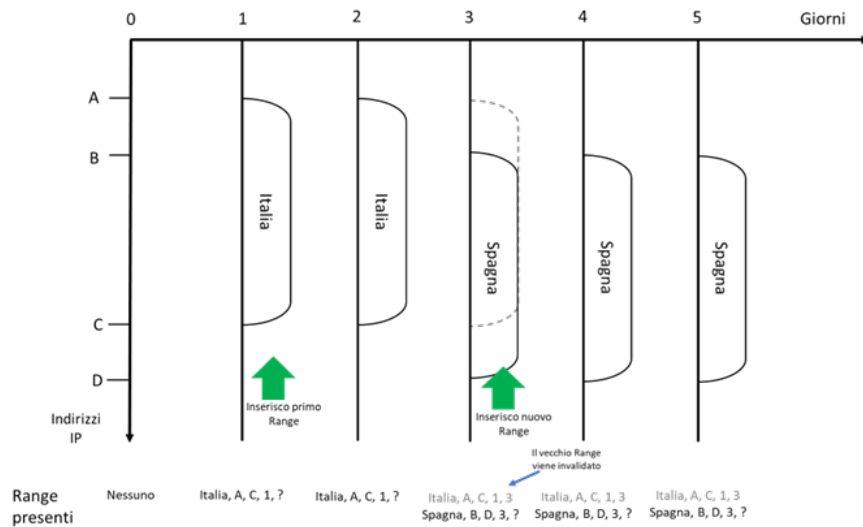
Gli indirizzi IP che un ISP può assegnare vengono comprati (e posseduti) in blocchi continui, individuabili quindi da un IP di inizio e da uno di Fine. La DataInizio di un range è la data in cui il range è stato acquistato ed è diventato quindi valido.

Ciascun ISP può avere sede in uno e un solo **Paese**, quindi un blocco di IP permette di risalire al **Paese** proprietario.

Un blocco (o parte) di indirizzi può però essere rivenduto ad altri ISP e quindi cambiare **Paese** di riferimento.

Dei Vincoli di Integrità faranno in modo che nella Base di Dati non vi siano range che collidono entrambi validi in uno stesso istante temporale e faranno ciò assegnando una data di fine validità (DataFine) ai range quando ne viene inserito uno nuovo che collide con uno già presente e impostando come scaduti range ineriti con data di inizio validità inferiore ad altri con cui collidono già presenti

Due diversi range collidono se e solo se esiste un indirizzo IP che appartiene a entrambi i range.



2.5.9 Server

Il **CaricoAttuale** del server viene misurato in numero di connessioni attualmente aperte, mentre con **MaxConnessioni** ci riferiamo al massimo numero di connessioni che possono essere aperte in un certo istante.

Il server è considerato in sovraccarico se il numero di connessioni attualmente aperte, ossia **CaricoAttuale**, è superiore ad una certa percentuale del numero massimo di connessioni (e.g. se CaricoAttuale supera l'80% di MaxConnessioni il Server è in sovraccarico).

Maximum Transmission Unit (**MTU**), in italiano Unità massima di trasmissione, si indica la dimensione massima, misurata in byte, di un pacchetto dati che può essere inviato.

2.5.10 Abbonamento

La **Durata** dell'**Abbonamento** è il numero di giorni dopo il quale l'abbonamento scade, insieme all'attributo **DataInizio** della relazione **AbbonamentoUtente** è possibile determinare se l'attuale abbonamento è in vigore o scaduto.

Non viene salvato lo storico di tutti gli abbonamenti di un **Utente** ma solamente l'abbonamento più recente dell'utente, sia esso scaduto o in vigore.

Con **Definizione** si intende la definizione massima consentita da quel tipo di

abbonamento. **Offline** è un booleano che indica se l'abbonamento permette all'utente di scaricare contenuti in locale.

Con **MaxOre** ci si riferisce al numero massimo di ore mensili di cui un utente può usufruire per la fruizione di contenuti. **GB Mensili** sono il massimo numero di GB utilizzabili da un utente ogni mese.

2.5.11 Popolarità di Artista

La **Popolarità di Artista** è un valore numerico decimale che va da 0 a 10 e funge, per l'appunto, da metrica della fama, notorietà e celebrità di un artista.

Essa verrà, inoltre, presa in considerazione per fornire valutazioni riguardo i film in cui gli artisti prendono parte sia come attori che come registi.

2.5.12 Posizione di un Paese

La **Posizione** di un **Paese** è una coppia Latitudine, Longitudine (posizione sulla superficie terrestre) che identifica il "*centro di popolazione*" di quest'ultimo.

Tale valore può essere calcolato come la media ponderata in base al numero di abitanti delle posizioni geografiche dei centri di un **Paese**.

È necessario sfruttare questo valore fittizio e, di per sé, poco rappresentativo per individuare i **Server** migliori (più vicini) ad un **Utente** che si trova in un determinato **Paese**.

3 Ristrutturazione

3.1 Eliminazione delle Generalizzazioni

3.1.1 Artista

Artista è una generalizzazione totale sovrapposta specializzata in **Attore** e **Regista** per distinguere i diversi ruoli che una persona può assumere all'interno di un film.

Entrambe le specializzazioni di **Artista** non sono dotate di attributi propri ma possiedono unicamente gli attributi relativi ad **Artista**.

Questa generalizzazione è stata ristrutturata decidendo di conservare l'entità **Artista** la quale mantiene la relazione **VincitaArtista** ed, inoltre, guadagna le due relazioni **Recitazione** e **Regia** che erano precedentemente collegate, rispettivamente, a **Attore** e **Regista** cambiandone, tuttavia, le cardinalità.

Infatti, le cardinalità di **Regia** sono **(1, N)** rispetto a **Film**, come nell'ER non ristrutturato, e **(0, N)** verso **Artista**, diversamente da prima in quanto bisogna contemplare la presenza di un regista che non reciti in nessun film. Mentre le cardinalità di **Recitazione** sono **(1, 1)** rispetto a **Film**, come nell'ER non ristrutturato, e **(0, N)** verso **Artista**, diversamente da prima in quanto bisogna contemplare la presenza di un attore che non diregga nessun film.

La nostra scelta di ristrutturazione deriva, innanzitutto, da una gestione sicuramente più efficiente del caso sporadico in cui esista un artista che ricopra sia un ruolo di **Attore** che un ruolo di **Regista** ed, oltretutto, dalla consapevolezza che non vi sia uno spreco di memoria, che si sarebbe verificato qualora avessimo ristrutturato conservando sia **Attore** che **Regista**, dovuto alla natura sovrapposta della generalizzazione.

3.1.2 Recensore

Recensore è una generalizzazione totale ed esclusiva specializzata in **Critico** e **Utente** per distinguere le diverse valutazioni, ossia critica e recensione, che vengono espresse nei confronti di un film.

La specializzazione **Critico** non è dotata di attributi propri, mentre la specializzazione **Utente** possiede due attributi propri, ossia email e password. Questa generalizzazione è stata ristrutturata decidendo di conservare le due entità **Critico** ed **Utente** eliminando l'entità **Recensore**.

La nostra scelta di ristrutturazione deriva, innanzitutto, da una gestione indubbiamente più efficiente della memoria giacché, ristrutturando solamente con l'entità **Recensore**, ci sarebbero valori NULL, sebbene sporadici, per email e password nel caso dei critici ed, oltretutto, dalla chiara separazione sia a livello concettuale e logico ma anche pratico e funzionale tra un **Critico**, che fornisce unicamente critiche, ed un utente il quale, oltre a recensire, ha un evidente ruolo nell'**Area Clienti**, ovvero tutto ciò che riguarda abbonamenti, visualizzazioni e pagamenti.

3.1.3 FormatoCodifica

FormatoCodifica è una generalizzazione totale e sovrapposta specializzata in **Audio** e **Video** per distinguere i formati utilizzati codificazione e decodificazione di dati uditivi e/o visivi.

Entrambe le specializzazioni di **FormatoCodifica** non sono dotate di attributi propri ma possiedono unicamente gli attributi relativi a **FormatoCodifica**.

Questa generalizzazione è stata tradotta decidendo di mantenere solamente l'entità **FormatoCodifica** la quale preserva le due relazioni con **File**, ossia **AudioFile** e **VideoFile**, modificandone, però, le cardinalità.

Infatti, le cardinalità di **VideoFile** rimangono **(1,1)** rispetto a **File** ma mutano in **(0, N)** rispetto a **FormatoCodifica** poiché bisogna considerare l'esistenza di un formato unicamente audio. Invece, le cardinalità di **AudioFile** rimangono **(1,1)** rispetto a **File** ma mutano in **(0, N)** rispetto a **FormatoCodifica** poiché bisogna considerare l'esistenza di un formato unicamente video.

La nostra decisione di ristrutturazione è dovuta sia alla semplicità di introdurre una sola entità con una relazione piuttosto che due entità con due relazioni, senza nessuno spreco di memoria a cagione dell'assenza di attributi annessi alle singole specializzazioni, che alla gestione più efficiente del caso in cui un **FormatoCodifica** sia sia **Audio** che **Video**, essendo la generalizzazione sovrapposta, la quale non comporterebbe alcuno spreco di memoria.

3.2 Eliminazione degli Attributi Multivalore

3.2.1 LinguaDoppiaggio

Si considera che ogni **File** debba contenere al suo interno audio in una o più lingue. L'attributo multivalore è, pertanto, stato gestito introducendo sia un'entità **Lingua**, usata altresì per le lingue in cui sono disponibili i sottotitoli, che una relazione **Doppiaggio** che intercorre tra **Lingua**, con cardinalità **(0, N)**, in quanto una lingua potrebbe essere disponibile nei sottotitoli di qualche film ma mai disponibile come audio in nessun film, e **File**, con cardinalità **(1, N)** in quanto ogni file deve avere l'audio disponibile in almeno una lingua (si ignora la presenza del cinema muto in quanto ormai desueto).

3.2.2 LinguaSottotitoli

Si considera che ogni **File** possa contenere al suo interno **Sottotitoli** in varie lingue. L'attributo multivalore è stato trasformato in una relazione **Sottotitolaggio** che intercorre tra **Lingua** (creata al punto precedente), con cardinalità **(0, N)**, in quanto una lingua può essere presente come audio in un film e mai nei sottotitoli di alcun film, e **File**, con cardinalità **(0, N)**, in quanto un film non deve necessariamente avere sottotitoli.

3.3 Analisi Ridondanze

3.3.1 Ridondanze presenti

Nell'ER iniziale è presente una sola ridondanza: *CaricoAttuale* di **Server**. Essa è stata inserita in modo da facilitare (velocizzare) l'individuazione dei server sovraccarichi e, se ve ne sono, operare sulle connessioni attive su quei server per bilanciare la distribuzione degli utenti. Per garantire una fluidità del servizio questa operazione è ripetuta con alta frequenza e deve quindi completare velocemente in modo da non sovraccaricare lei stessa il DB. Essa è calcolata contando tutte le **Erogazioni** attive (presenti nel DB) sul server

3.3.2 Introduzione nuove Ridondanze

È stata inserita una ulteriore ridondanza sotto la forma della relazione **DistanzaPrecalcolata** tra **Server** e **Paese**, che conterrà all'interno la di-

stanza (espressa in chilometri) tra ogni **Server** e ogni **Paese**, in modo da velocizzare l'esecuzione della query che restituisce il server più adatto per la fruizione di un contenuto da parte dell'utente evitando il ricalcolo della distanza tra ogni **Server** ed il **Paese** dell'utente.

Sono stati aggiunte all'entità **Film** due attributi ridondanti: *MediaRecensioni* e *NumeroRecensioni*.

Essi sono necessari per evitare di calcolare la media delle recensioni di ogni **Film**, operazione molto comune perché avviene ogni volta che un **Utente** apre la pagina relativa ad un **Film**.

NumeroRecensioni permette di modificare la *MediaRecensioni* alla aggiunta / modifica / rimozione di una **Recensione** senza ricalcolare il valore dall'intero insieme dei dati.

È stata aggiunta la ridondanza *NumeroVisualizzazioni* in **VisualizzazioniGiornaliere**: essa ha lo scopo di evitare la conta delle Visualizzazioni avvenute in un determinato **Paese** (che comportarebbe moltissimi accessi a **IPRange**). Oltretutto il raggruppamento per **Film** permette di individuare facilmente i **Film** più visti in un determinato arco temporale o in una determinata area geografica, facilitando molte operazioni comuni.

3.4 Partizionamento e Accorpamento ER

3.4.1 Qualità di VideoFile

L'attributo composto **Qualità** non è altro che un raggruppamento di specifiche tecniche nell'ambito video di un **File** multimediale; è quindi scomposto in due attributi *Risoluzione* e *FPS*.

3.4.2 Qualità di AudioFile

L'attributo composto **Qualità** non è altro che un raggruppamento di specifiche tecniche nell'ambito audio di un **File** multimediale; è quindi scomposto in due attributi *Frequenza* e *BitDepth*.

3.4.3 Posizione

L'attributo composto **Posizione** descrive una coppia Latitudine e Longitudine in modo da individuare un punto sulla superficie terrestre.

Esso, a differenza degli altri attributi composti, non necessita di essere scomposto per facilitare l'Implementazione Fisica, in quanto il DBMS in cui la Base di Dati sarà implementata (Oracle MySQL) supporta attributi di tipo *POINT*, che possono essere utilizzati per memorizzare coppie di valori numerici assieme.

Oltretutto, memorizzare una coppia *Latitudine* e *Longitudine* in questo modo facilita successivamente il calcolo di distanza tra due **Posizioni**, sfruttando la funzione nativa di MySQL *ST_DISTANCE_SPHERE*.

3.4.4 Carta di Credito

L'attributo composto **Carta di Credito**, dell'entità **Pagamento** contiene le informazioni sulla carta impegnata nel pagamento di una **Fattura**.

Essa è stata trasformata in un'entità, poiché si prevede che un **Utente** utilizzi più volte la stessa carta; pertanto, se tutti i dati della carta fossero attributi di **Pagamento** non vi sarebbero una gestione efficiente della memoria a causa di valori ripetuti.

È stata quindi creata l'entità **Carta di Credito** con i seguenti attributi: *PAN*, *Scadenza*, *CVV*. La chiave dell'entità è *PAN*, in quanto ogni carta può essere individuata unicamente dal suo numero.

È stata quindi aggiunta anche una relazione **PagamentoCarta** tra le entità **Pagamento**, con cardinalità **(1, 1)**, e **Carta di Credito**, con cardinalità **(1, N)**, in quanto una carta può essere usata in più pagamenti ed in un pagamento può essere usata una sola carta.

4 Tavole dei Volumi

Le entità e le relazioni con segante con "*" hanno volume assegnato per ipotesi.

4.1 Area Contenuti

Concetto	Tipo	Volume	Note
Film*	E	1 000	Assunzione iniziale
Produzione	R	1 000	Cardinalità (1, 1) con Film
Casa di Produzione*	E	20	Assunzione iniziale
Appar-tenenza	R	20	Cardinalità (1, 1) con Casa di Produzione
Paese	E	195+1	Nel mondo ci sono 195 stati riconosciuti, il 196-esimo rappresenta lo stato <i>Mondo</i> , utilizzato come ancora di salvezza per dare una posizione agli utenti quando non si riesce a trovare lo stato in cui si trovano
Lingua	E	141	Nel mondo ci sono 141 lingue
Doppiaggio	R	18 000	Consideriamo che ogni File sia doppiato, in media, in 4 Lingue

Concetto	Tipo	Volume	Note
Sotto-titolaggio	R	13 500	Consideriamo che ogni File sia sottotitolato, in media, in 3 Lingue
Edizione Film	R	1 500	Cardinalità (1, 1) con Edizione
Artista*	E	4 000	Assunzione Iniziale
Regia	R	1 000	Cardinalità (1, 1) con Film
Recitazione	R	20 000	Consideriamo che, in media, 20 attori recitino in un Film
Genere*	E	27	Assunzione iniziale
GenereFilm	R	2 000	Consideriamo che, in media, un Film appartenga a due Generi
Critico*	E	25	Assunzione iniziale
Critica	R	400	Si stima che un Critico faccia in media poco più di 15 critiche a Film diversi
Recensione	R	25 000	Si stima che in media un Film abbia 25 recensioni da Utenti diversi
Premio*	E	50	Assunzione iniziale

Concetto	Tipo	Volume	Note
Vincita	E	1 500	Si stima che ogni Premio sia stato vinto una volta l'anno negli ultimi 30 anni
VincitaPremio	R	1 500	Cardinalità (1, 1) con Vincita
VincitaFilm	R	1 500	Cardinalità (1, 1) con Vincita
VincitaArtista	R	1 200	Si stima che la maggior parte (80%) delle vincite riguardi in maniera specifica un Artista (Attore o Regista)

4.2 Area Formato

Concetto	Tipo	Volume	Note
Edizione	E	1 500	Si stima che, in media, la metà dei Film abbia due edizioni
File	E	4 500	Si stima che ogni Edizione abbia in media 3 File associati, in modo da permettere la fruizione su dispositivi diversi (che possono supportare solo alcuni formati) e anche velocità di connessione deboli
Edizione File	R	4 500	Cardinalità (1, 1) con File

Concetto	Tipo	Volume	Note
Formato Codifica*	E	100	Si stima che esistano circa 100 Formati di codifica tra video e audio
VideoFile	R	4 500	Cardinalità (1, 1) con File
AudioFile	R	4 500	Cardinalità (1, 1) con File
Restrizione	R	1 000	Si stima che ogni Paese bandisca, in media, 5 Edizioni

4.3 Area Clienti

Concetto	Tipo	Volume	Note
Utente*	E	1 000 000	Assunzione iniziale
Abbon-Utente	R	1 000 000	Cardinalità (1, 1) con Utente
Abbonamento*	E	5	Assunzione iniziale
Esclusione	R	15	Si stima che ogni Abbonamento escluda, in media, circa 3 Generi
Fattura	E	12 000 000	Fattura contiene le fatture emesse in un anno. Per ogni utente viene emessa una fattura ogni mese, quindi 12 in un anno.

Concetto	Tipo	Volume	Note
Emissione	R	12 000 000	Cardinalità (1, 1) con Fattura
Pagamento	E	11 000 000	Si stima che per ogni utente le 11 ultime fatture siano state pagate mentre l'ultima no. Quindi 11 su 12, ovvero il 92% delle emissioni.
Pagamento Fattura	R	11 000 000	Cardinalità (1, 1) con Pagamento
Carta di credito	E	1 500 000	Si stima che la metà degli Utenti abbia usato, in media, due Carte di Credito
Pagamento Carta	R	11 000 000	Cardinalità (1, 1) con Pagamento
Connessione	E	30 000 000	Si stima che un Utente si connetta 1 volta al giorno. Il sistema conserva le connessioni degli ultimi 30 giorni
Conn-Utente	R	30 000 000	Cardinalità (1, 1) con Connessione
Visualizzazione	E	15 000 000	Si stima che un Utente guardi, in media, un' Edizione ogni due giorni. Il sistema tiene traccia dei contenuti guardati negli ultimi 30 giorni
Visualizzazione Connessione	R	15 000 000	Cardinalità (1, 1) con Visualizzazione

Concetto	Tipo	Volume	Note
Visualizzazione Edizione	R	15 000 000	Cardinalità (1, 1) con Visualizzazione
Visualizzazioni Giornaliere	E	30 000	1000 Film per 30 giorni di distanza nel passato
Film Vis Giorni	R	30 000	Cardinalità (1, 1) con Visualizzazioni Giornaliere
Paese Vis Giorni	R	30 000	Cardinalità (1, 1) con Visualizzazioni Giornaliere

4.4 Area streaming

Concetto	Tipo	Volume	Note
Server*	E	20	Circa il 10% dei Paesi
Distanza Precalcolata	R	3920	Esattamente ogni combinazione possibile tra Server e Paese
P.o.P.	R	9 000	Ogni file è presente, in media, su 2 server
Erogazione	R	40 000	Corrisponde al numero di utenti mediamente connessi in ogni istante. Stimando il numero medio di ore giornaliere di attività a 1 si deduce che il 4% degli utenti è mediamente connesso in ogni istante
Ip-Range*	E	130 000	Assunzione iniziale basata su valori reali
GeoLocal	R	130 000	Cardinalità (1, 1) con Ip-Range

5 Operazioni sui Dati

5.1 Analisi delle Ridondanze

5.1.1 CaricoAttuale

CaricoAttuale, attributo di **Server** contiene, in ogni momento, il numero di connessioni in corso di un server CDN. Tale valore potrebbe essere calcolato

contando le **Erogazioni** che puntano a quel **Server**. Per misurare il carico di un singolo server, senza la ridondanza, sarebbe necessario quindi analizzare tutte le Erogazioni e contare quelle che soddisfano il criterio.

Tale operazione avviene però quando un **Utente** è alla ricerca del **Server** migliore che gli può offrire il contenuto che cerca, rendendo necessario analizzare tutti il CaricoAttuale di tutti i Server.

Questa operazione è molto comune e deve concludersi in fretta in modo da non far aspettare l'Utente e garantire una migliore esperienza.

Il tipo di aggiornamento della ridondanza è l'**Immediate Refresh**: essa viene aggiornata ad ogni inserimento/modifica/cancellazione nella relazione **Erogazione**.

Di seguito è riportata la frequenza giornaliera delle attività che coinvolgono tale ridondanza:

Azione	FrequenzaGiornaliera	Nota
Lista Server	500 000	Si considera che ogni utente guardi un file ogni due giorni

Input: chiave del **File** che si intende guardare

Output: Lista di Server disponibili, ciascuno con il suo CaricoAttuale

I Volumi interessati sono:

Nome	Tipo	Volume
P.o.P	R	9 000
Erogazione	R	40 000
Server	E	20

Gli accessi per Visualizzazione sarebbero in media, senza ridondanza:

Concetto	Costrutto	Accessi	Tipo	Note
P.o.P	R	9 000	R	Conoscendo la chiave di File si può passare direttamente a leggere i P.o.P. per individuare i Server che possiedono il File che si cerca. La lettura di tutte le occorrenze della relazione restituirà, in media, 2 chiavi di Server
Erogazione	R	40 000	R	Vengono lette tutte le occorrenze per individuare quelle che soddisfano il criterio. La chiave si Server è stata ottenuta da P.o.P.
Totale:		49 000		Soluzione non ottimale

Mentre, con la ridondanza sono in media:

Concetto	Costrutto	Accessi	Tipo	Note
P.o.P	R	9 000	R	Conoscendo la chiave di File si può passare direttamente a leggere i P.o.P. per individuare i Server che possiedono il File che si cerca. La lettura di tutte le occorrenze della relazione restituirà, in media, 2 chiavi di Server
Server	E	2	R	La chiave di Server è già stata individuata e accedendo all'entità si ha già il CaricoAttuale: parametro della scelta
Totale:		9 002		Soluzione ottimale

È importante far notare che, in questo caso, ogni volta che un'**Erogazione** viene aggiunta/rimossa sarà necessario accedere a **Server** per alterarne il CaricoAttuale. Questo costringe a sommare ai valori giornalieri ottenuti il costo ottenuto dalla periodica modifica di CaricoAttuale.

Azione	FrequenzaGiornaliera	Nota
Aggiornamento <i>CaricoAttuale</i>	750 000	Si considera che ogni Visualizzazione sfrutti, in media, 1,5 Server diversi nel corso della sua durata

I Volumi coinvolti sono:

Nome	Tipo	Volume
Server	E	20

La cui tabella degli Accessi è:

Concetto	Costrutto	Accessi	Tipo	Note
Server	E	1	W	La chiave di Server è già nota alla relazione Erogazione , permettendo di accedere al Server interessato e impostare il <i>CaricoAttuale</i> a +1 o -1 il suo precedente valore se l' Erogazione è stata aggiunta o rimossa.
Totale:		1		

Gli accessi giornalieri delle due tipologie sono quindi:

Metodologia	Accessi al giorno		
	Lettura	Scrittura	Totale
Senza ridondanza	24,5 X 10 ⁹	0	24,5 X 10 ⁹
Con ridondanza	4,5 X 10 ⁹	1 500 000	4,5 X 10 ⁹

5.1.2 DistanzaPrecalcolata

DistanzaPrecalcolata è una relazione tra **Paese** e **Server** che ha, come attributo, il valore della distanza tra un server ed un paese. Tale valore potrebbe essere calcolato a partire dalle posizioni sia di paese che di server. Qualora si stia ricercando il server ottimale per un utente ubicato in un certo paese occorre conoscere, oltre al carico del server, anche la distanza tra di esso ed il paese. Invece di ricalcolare tutte le distanze ogni volta è più efficiente andarle a leggere da **DistanzaPrecalcolata**.

DistanzaPrecalcolata non viene aggiornata giornalmente in quanto la posizione dei server e dei paesi è fissa.

Di seguito è riportata la frequenza giornaliera delle attività che coinvolgono tale ridondanza:

Azione	FrequenzaGiornaliera	Nota
Distanza Server	1 000 000	Si considera che ogni Utente guardi un file ogni due giorni, e che per farlo usi circa due Server , dei quali si deve calcolare la distanza dalla posizione dell' Utente . Per ogni richiesta si stima che ci siano due Server (di cui si conosce già la chiave) che ospitano il contenuto richiesto

Input: chiave del **Paese** dove si trova l'utente e dei Server (2 in media) che ospitano il Film richiesto

Output: Lista di ID di tutti i Server, ciascuno con la sua distanza dal paese dell'utente

I Volumi interessati sono:

Nome	Tipo	Volume
Distanza Precalcolata	R	3 920

Gli accessi per Distanza Server sarebbero in media, senza ridondanza:

Concetto	Costrutto	Accessi	Tipo	Note
Paese	E	1	R	Conoscendo la chiave di Paese vi si può accedere puntualmente
Server	E	2	R	Viene letta la posizione dei Server richiesti
Totale:		3		Soluzione non ottimale

Mentre, con la ridondanza sono in media:

Concetto	Costrutto	Accessi	Tipo	Note
Distanza-Precalcolata	R	2	R	Conoscendo sia la chiave di Paese che di Server si può leggere la relazione in maniera puntuale.
Totale:		2		Soluzione ottimale

Questa soluzione è ottimale anche perché i valori di *Posizione* sia di un **Paese** che di un **Server** sono praticamente immutabili: è un evento rarissimo che una nazione nasca/scompaia o cambi notevolmente i suoi confini, come un **Server** non può cambiare posizione mentre sta erogando contenuti. Le distanze tra le *Posizioni* possono quindi essere calcolate una volta sola all'inserimento di queste entità e praticamente mai cambiate. Oltretutto, calcolare la distanza date le due posizioni possiede un costo computazionale non indifferente che, tuttavia, non può essere stimato in accessi.

Gli accessi giornalieri delle due tipologie sono quindi:

Metodologia	Accessi al giorno		
	Lettura	Scrittura	Totale
Senza ridondanza	3×10^6	0	3×10^6
Con ridondanza	2×10^6	0	2×10^6

5.1.3 MediaRecensioni NumeroRecensioni

MediaRecensioni e **NumeroRecensioni** sono due attributi di **Film** che tengono conto, rispettivamente, della media delle recensioni del film e del numero di recensioni che esso ha ricevuto.

Il valore della media ha un'utilità pratica e immediata mentre il numero di recensione assolve alla funzione di alleggerimento del mantenimento della ridondanza.

Il valore della media delle recensione potrebbe essere calcolando andando a considerare tutte le recensioni del film e, poi, calcolandone la media. Qualore, però, si stia ricercando il film con valutazione media maggiore, la quale è un'operazione piuttosto comune, si andrebbe a ricalcolare molteplici volte le medie di tutti i film.

Invece di ricalcolarle ogni volta ex nihilo è più efficiente salvarle per poi rileggerle.

Il tipo di aggiornamento della ridondanza è l'**Immediate Refresh**: essa viene aggiornata ad ogni inserimento/modifica/cancellazione nella relazione **Recensione**.

Di seguito è riportata la frequenza giornaliera delle attività che coinvolgono tale ridondanza:

Azione	FrequenzaGiornaliera	Nota
Film Migliori	30 000 000	Si considera che ogniqualvolta un utente si connetta gli venga mostrato il film con recensioni migliori come raccomandazione automatica
Inserimento Recensione	70	Si stima che le 25 000 recensioni siano state accumulate nel corso di un anno

Input: Nulla

Output: Titoli e media recensioni dei 20 Film con media più alta

I Volumi interessati sono:

Nome	Tipo	Volume
Film	E	1 000
Recensione	R	25 000

Gli accessi per Visualizzazione sarebbero in media, senza ridondanza:

Concetto	Costrutto	Accessi	Tipo	Note
Recensione	R	25 000	R	Devono essere lette tutte le occorrenze, raggruppate in base alla chiave di Film e ordinate in base alla media di <i>Voto</i>
Film	E	20	R	Noti gli <i>ID</i> dei Film di cui si deve riportare il <i>Titolo</i> si accede direttamente a essi dalla chiave
Totale:		25 020		Soluzione non ottimale

Mentre, con la ridondanza sono in media:

Concetto	Costrutto	Accessi	Tipo	Note
Film	E	1 000	R	Si leggono tutti i Film e si trovano le migliori 20 medie
Totale:		1 000		Soluzione ottimale

Gli accessi relativi all'aggiornamento della ridondanza sono invece:

Concetto	Costrutto	Accessi	Tipo	Note
Film	E	1	W	Conoscendo la chiave di Film e memorizzando sia <i>NumeroRecensioni</i> che <i>MediaRecensioni</i> si possono calcolare i nuovi valori senza effettuare altre letture
Totale:		1		

Gli accessi giornalieri delle due tipologie sono quindi:

Metodologia	Accessi al giorno		
	Lettura	Scrittura	Totale
Senza ridondanza	25 X 10 ⁹	0	25 X 10 ⁹
Con ridondanza	1 X 10 ⁹	70	1 X 10 ⁹

5.1.4 Visualizzazioni Giornaliere

Visualizzazioni Giornaliere è un'entità che tiene il conto, giorno per giorno, del numero di visualizzazioni totali avvenute, raggruppate sia per il **Film** visualizzato che per il **Paese** dove la visualizzazione è avvenuta.

Il **Paese** viene calcolato dall'*IP* e da *InizioConnessione*, mentre il **Film** viene ottenuto accedendo all'Edizione visualizzata.

Il valore dell'attributo **NumeroVisualizzazioni** è particolarmente utile poiché funzionale all'alleggerimento del carico, infatti previene l'accesso, in alcune situazioni, a tutte le occorrenze dell'entità **Visualizzazione**.

Tutte le occorrenze di quest'entità potrebbero essere calcolate semplicemente attraverso raggruppamenti sull'entità **Visualizzazione**. Tuttavia, qualora si volesse ottenere una classifica dei **Film** sulla base della visualizzazioni tutti gli accessi su **Visualizzazione** porterebbero ad un eccessivo appesantimento del carico.

È possibile, con la stessa facilità, stilare la classifica dei **Film** più visti in un determinato **Paese**.

Il tipo di aggiornamento previsto per tale ridondanza è il **Deferred Refresh** con cadenza giornaliera: ogni giorno, solo una volta al giorno, si accede a **Visualizzazione** per inserire i dati aggregati del giorno precedente; non è quindi possibile avere i dati relativi al giorno in corso.

Di seguito è riportata la frequenza giornaliera delle attività che coinvolgono tale ridondanza:

Azione	FrequenzaGiornaliera	Nota
Classifiche Film	100 000	Si considera che a un utente vengano mostrati i film più visti nell'ultimo mese il 10% delle volte che si connette
Ricalcolo Classifiche	1	Ogni giorno, vengono contate le Visualizzazioni di ogni Film avvenute durante la giornata precedente

Input: Nulla

Output: *Titoli e Numero Visualizzazioni* dei **Film** con almeno una visualizzazione

Porzione della Tabella dei Volumi:

Concetto	Tipo	Volume
Film	E	1 000
Edizione	E	1 500
Visualizzazione	E	15 000 000
FilmzEdizione	E	1 500
Visualizzazione-Edizione	R	15 000 000
Visualizzazioni Giornaliere	E	30 000
Film Vis-Giorno	R	30 000

Gli accessi sarebbero in media, senza ridondanza:

Concetto	Costrutto	Accessi	Tipo	Note
Visualizzazione	E	0	R	Essendo tutte le Visualizzazioni memorizzate già riguardanti gli ultimi 31 giorni si può leggere direttamente Visualizzazione-Edizione
Visualizzazione-Edizione	R	15 000 000	R	Si legge ogni occorrenza e si estraе la chiave di Edizione
Edizione-Film	R	1 500	R	Si sommano, raggruppando per <i>ID</i> di Film , le visualizzazioni di ogni Edizione
Film	E	1 000	R	Noti gli <i>ID</i> dei Film se ne estraggono i <i>Titoli</i>
Totale:		15 000 000		
Totale Giornalier:		1 500 000 000 000		Soluzione non ottimale

Mentre, con la ridondanza sono in media:

Concetto	Costrutto	Accessi	Tipo	Note
Visualizzazioni-Giornaliere	E	30 000	R	Si leggono tutte le visualizzazioni giornaliere
Film Vis-Giorni	E	30 000	R	Di tutte le Visualizzazioni Giornaliere selezionate al passaggio precedente si legge l' <i>ID</i> del relativo Film e, in base a esso, si raggruppa trovando il numero complessivo di <i>Visualizzazioni</i> per Film
Film	E	1 000	R	Noti gli <i>ID</i> dei Film se ne estraggono i <i>Titoli</i>
Totale:		61 000		
Totale Giornalier:		6 100 000 000		Soluzione ottimale

Gli accessi relativi all'aggiornamento della ridondanza sono invece :

Concetto	Costrutto	Accessi	Tipo	Note
Visualizzazione	E	0	R	Essendo <i>Timestamp</i> chiave di Visualizzazione lo si può trovare direttamente in le Visualizzazione-Edizione
Visualizzazione-Edizione	R	15 000 000	R	Si leggono tutte le VisualizzazioniEdizione avvenute nelle ultime 24h. Si stima che le corrispondenze siano 1/30 del totale. Si estraggono le chiavi delle edizioni interessate.
Edizione-Film	R	1 500	R	Si contano, raggruppando per <i>ID</i> di Film , le visualizzazioni di ogni Edizione
Film Vis-Giorno	R	1 000	W	Per ogni Film creo una nuova relazione che servirà per creare un'associazione con Visualizzazioni Giornaliere
Visualizzazioni Giornaliere	E	1 000	W	Per ogni Film scrivo il <i>Numero Visualizzazioni</i> e la <i>Data</i> (data del giorno precedente)
Totale:		15 005 500		

Gli accessi giornalieri delle due tipologie sono quindi:

Metodologia	Accessi al giorno		
	Lettura	Scrittura	Totale
Senza ridondanza	$1,5 \times 10^{12}$	0	$1,5 \times 10^{12}$
Con ridondanza	$6,1 \times 10^9$	4 000	$6,1 \times 10^9$

5.2 Analisi delle Operazioni

Le stime sulle tabelle degli accessi vengono fatte supponendo di non sapere assolutamente nulla sulla struttura dati utilizzati per implementare sia le Relazioni che le Entità, assumiamo solamente che essendo a conoscenza, per interezza, dell'identificativo di un costrutto si possa accedere ad esso in maniera puntuale.

Pertanto, non prenderemo in considerazione, in quanto logicamente sbagliato non trattandosi né di progettazione logica né di progettazione fisica, l'eventualità di ottimizzazioni di accessi quali la presenza di indici oppure Foreign Key.

Le operazioni di scrittura nel conteggio finale verranno raddoppiate in quanto evidentemente più pesanti di una lettura.

5.2.1 Vincite di un Film

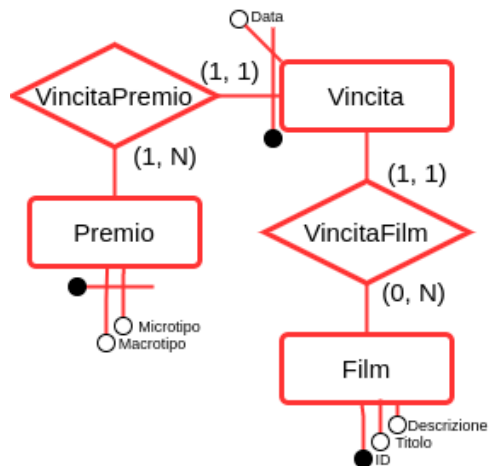
Descrizione: Non appena viene selezionato il film da visionare la piattaforma carica una lista contenenti sia i premi vinti dal film, comprendenti anche le date, e il numero di premi totali vinti.

Input: ID di Film

Output: Lista di Macrotipo, Microtipo e Data, Numero di Premi

Frequenza Giornaliera: 500 000 Poiché un **Utente**, in media, vede un **Film** ogni due giorni e i premi vengono mostrati nella pagina dell'app relativa al **Film**

Porzione di Diagramma Interessato:



Porzione della Tavola dei Volumi:

Concetto	Tipo	Volume
Premio	E	50
Film	E	1 000
Vincita	E	1 500
VincitaFilm	R	1 500
VincitaPremio	R	1 500

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
VincitaFilm	R	1 500	R	Conoscendo solo la chiave di Film si deve leggere necessariamente tutto VincitaFilm per trovare la chiave di Vincita contenente tutte le informazioni richieste.
Totale:		1 500		
Totale Giornaliero:		750 000 000		

5.2.2 Generi di un Film

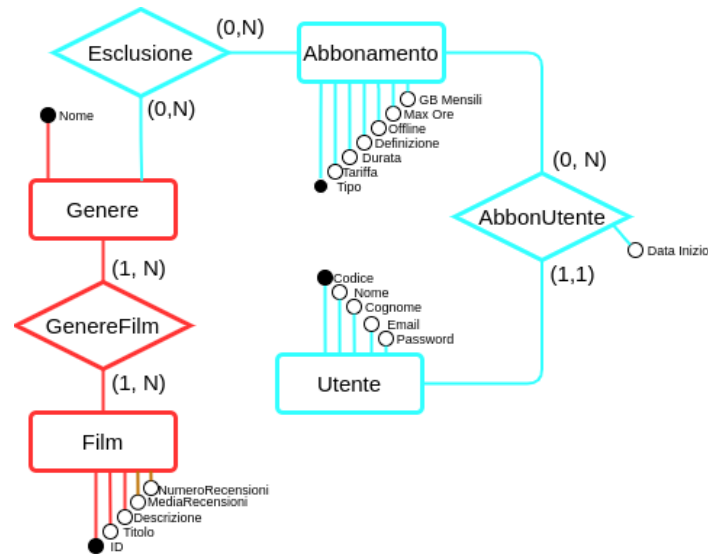
Descrizione: Non appena viene selezionato un **Film** viene caricata una lista contenente i **Generi** a cui appartiene il **Film**. Oltretutto, controllando il tipo di abbonamento dell'**Utente** si verifica che egli sia abilitato o meno alla fruizione del contenuto.

Input: ID, Codice

Output: Lista Generi, Abilitazione alla fruizione

Frequenza Giornaliera: 500 000 un **Utente**, in media, vede un **Film** ogni due giorni

Porzione di Diagramma Interessato:



Porzione della Tavola dei Volumi:

Concetto	Tipo	Volume
Genere	E	27
GenereFilm	R	2000
Film	E	1000
Utente	E	1 000 000
Abbonamento Utente	R	1 000 000
Abbonamento	E	5
Esclusione	R	15

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
GenereFilm	R	2 000	R	Conoscendo solo la chiave di Film si devono leggere tutte le occorrenze di GenereFilm . Si stima che i generi ottenuti siano due poiché un film ha in media due generi.
Abbonamento Utente	R	1 000 000	R	Conoscendo solo la chiave di Utente la relazione AbbonamentoUtente va letta per intero per determinare il tipo di abbonamento
Esclusione	R	15	R	Conoscendo solo la chiave di Abbonamento si devono leggere tutte le occorrenze di Esclusione .
Totale:		1 002 015		
Totale Giornaliero:		501 007 500 000		

5.2.3 Massima Risoluzione di un Film

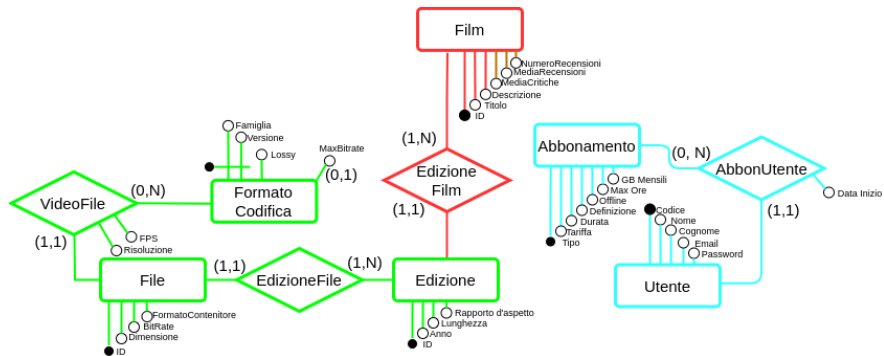
Descrizione: dato un **Film** restituire, tra tutte le **Edizioni**, il/i **File** con la *Risoluzione* maggiore permessa all'utente.

Input: ID di Film, ID di Utente

Output: ID di File con associata Risoluzione

Frequenza Giornaliera: 50 000 Perché si stima che ogni **Utente** guardi un **Film** ogni due giorni e che una volta su 10 cerchi il **File** con la massima *Risoluzione* possibile

Porzione di Diagramma interessato:



Porzione della Tabella dei Volumi:

Concetto	Tipo	Volume
EdizioneFilm	R	1 500
EdizioneFile	R	4 500
VideoFile	R	4 500
Utente	E	1 000 000
Abbonamento	E	5
Abbonamento- Utente	R	1 000 000

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Abbonamento- Utente	R	1 000 000	R	Avendo solo la chiave di Utente la relazione Abbonamento va letta per interezza.
Abbonamento	E	1	R	Occorre leggere la massima risoluzione consentita dall'abbonamento
Edizione- Film	R	1 500	R	Conoscendo solo la chiave di Film si accede a tutta la relazione per trovare l'ID dell' Edizione associata.

File-Edizione	E	4 500	R	Conoscendo solo la chiave di Edizione si accede a tutta la relazione per trovare l'ID dell' File associato.
VideoFile	R	4 500	R	Noti gli ID dei File interessati si estrae la <i>Risoluzione</i> trovandone la massima
Totale:		1 010 501		
Totale Giornaliero:		50 525 050 000		

5.2.4 Numero di Film non disponibili in un Abbonamento

Descrizione: Quando l'utente sceglie quale abbonamento acquistare può informarsi sul numero di **Film** esclusi da uno specifico **Abbonamento**

Input: Tipo

Output: Numero di Film Esclusi

Frequenza Giornaliera: 80 000 Poiché gli abbonamenti hanno durata mensile si stima che ogni fine mese il 50% degli utenti decida di cambiare abbonamento e, dunque, controlli il numero di **Film** esclusi per tutti i 5 abbonamenti

Porzione di Diagramma Interessato:

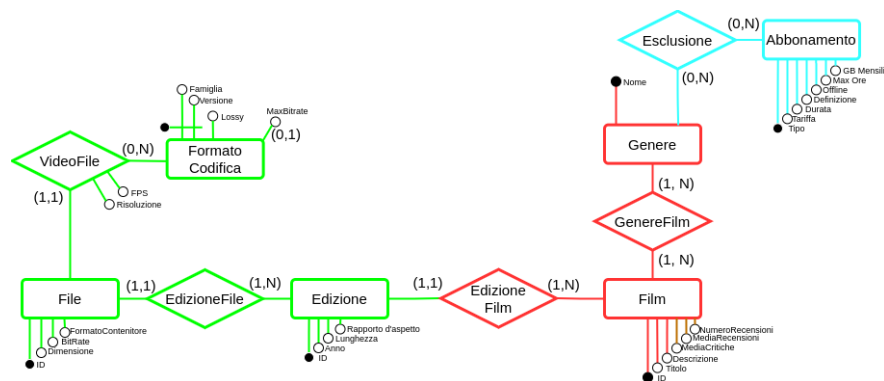


Tavola dei Volumi:

Concetto	Tipo	Volume
Abbonamento	E	5
Esclusione	R	15
Genere	E	27
GenereFilm	R	2 000
Film	E	1 000
EdizioneFilm	R	1 500
Edizione	E	1 500
EdizioneFile	R	4 500
File	E	4 500
VideoFile	R	4 500
FormatoCodifica	E	100

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Esclusione	R	15	R	Conoscendo la chiave di Abbonamento si accede a tutta Esclusione . Si assume che ogni abbonamento esclude, in media, 3 generi.
GenereFilm	R	2 000	R	Conoscendo solo la chiave di Genere si accede direttamente a tutta GenereFilm .
Abbonamento	E	1	R	Si legge la massima risoluzione disponibile
VideoFile	R	4 500	R	Si legge la risoluzione di tutti i File .
Edizione-File	R	4 500	R	Si legge la minima risoluzione per ogni Edizione .
Film-Edizione	R	1 500	R	Si legge la minima risoluzione per ogni Film .
Totale:		12 516		
Totale Giornaliero:		1 001 280 000		

5.2.5 Film disponibili in Lingua specifica

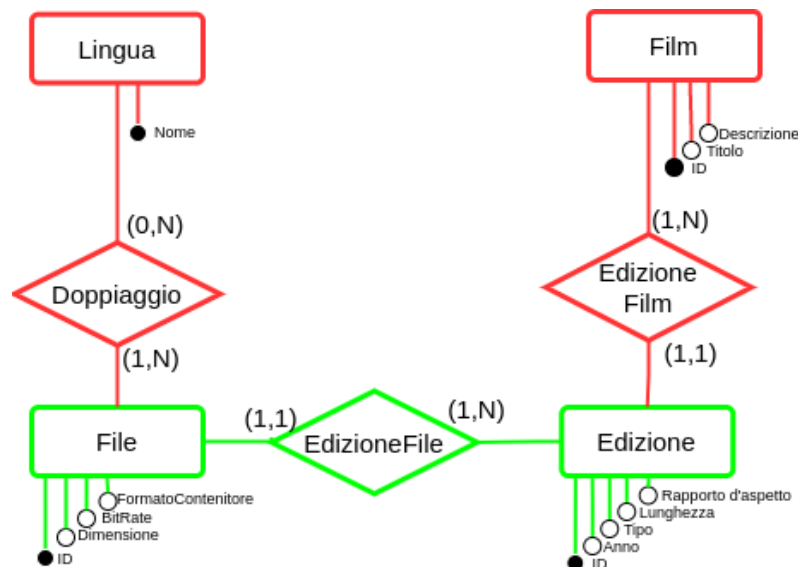
Descrizione: data una **Lingua** restituire, tutti i titoli di **Film** che hanno almeno un **Doppiaggio**, presente nel sistema, nella lingua richiesta.

Input: ID di Lingua

Output: Lista di ID e Titoli di Film che soddisfano il criterio

Frequenza Giornaliera: 100 000 Perché si stima che ogni Utente guardi un Film ogni due giorni e che un quinto delle volte ne scelga uno in base ad una Lingua specifica. La ricerca dei Film disponibili in una lingua avviene quindi, per ogni Utente, una volta ogni 10 giorni.

Porzione di Diagramma interessato:



Porzione della Tabella dei Volumi:

Concetto	Tipo	Volume
Lingua	E	141
Doppiaggio	R	18 000
EdizioneFile	R	4 500
EdizioneFilm	R	1 500
Film	E	1 000
File	E	4 500

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Doppiaggio	R	18 000	R	Conoscendo solo la chiave di Lingua va letta tutta la relazione. Si stimano circa 130 file trovati (18 000 / 141)
Edizione-File	R	4 500	R	Conoscendo solo la chiave di File va letta tutta la relazione. Si stimano circa 43 edizioni trovate (ci sono 3 file per ogni edizione)
Edizione-Film	R	1 500	R	Conoscendo solo la chiave di Edizione va letta tutta la relazione. Si stimano circa 29 film (un film ogni due ha due edizioni)
Film	E	29	R	Noti gli ID dei Film cercati se ne estraggono i <i>Titoli</i>
Totale:		24 029		
Totale Giornaliero:		2 402 500 000		

5.2.6 Film più visti recentemente

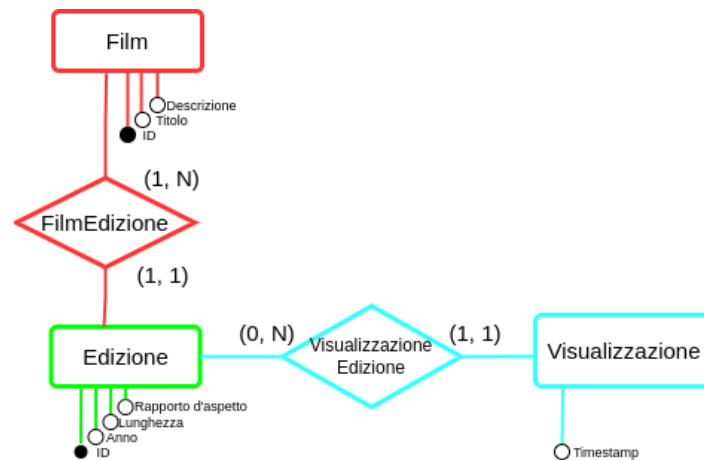
Descrizione: Un utente potrebbe voler conoscere gli n film più visti nell'ultimo mese nella piattaforma di streaming.

Input: Numero di Film

Output: Titoli di Film più visti nell'ultimo mese

Frequenza Giornaliera: 100 000 Si stima che il 10% delle volte che un utente si connette vada a controllare la classifica dei **Film** più visti di recente

Porzione di Diagramma Interessato:



Porzione della Tabella dei Volumi:

Concetto	Tipo	Volume
Film	E	1 000
EdizioneFilm	R	1 000
Edizione	E	1 500
Visualizzazione-Edizione	R	15 000 000
Visualizzazione	E	15 000 000
Visualizzazioni Giornaliere	E	30 000
FilmVis-Giorni	R	30 000

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
FilmVis-Giorni	R	30 000	R	Si leggono tutte le visualizzazioni, raggruppando per Film e individuando gli n migliori.
Film	E	n	R	Si i titoli dei Film precedentemente individuati. Si considera n come insignificante rispetto a 30 000.
Totale:		30 000		
Totale Giornaliero:		3 000 000 000		Soluzione Ottimale

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Visualizzazione Edizione	R	15 000 000	R	Si leggono tutte le visualizzazioni delle edizioni, e si raggruppando per Edizione .
Edizione Film	R	1 500	R	Si leggono tutte le coppie Edizione-Film e si estraggono i Film dalle Edizioni trovate al precedente punto. Si raggruppa per Film
Film	E	n	R	Si leggono i titoli degli n Film (si ha già la loro chiave) con più visualizzazioni. Si considera n come insignificante rispetto a 15 000 000
Totale:		15 001 500		
Totale Giornaliero:		$1,5 \times 10^{12}$		Soluzione Non Ottimale

5.2.7 Cambio Abbonamento

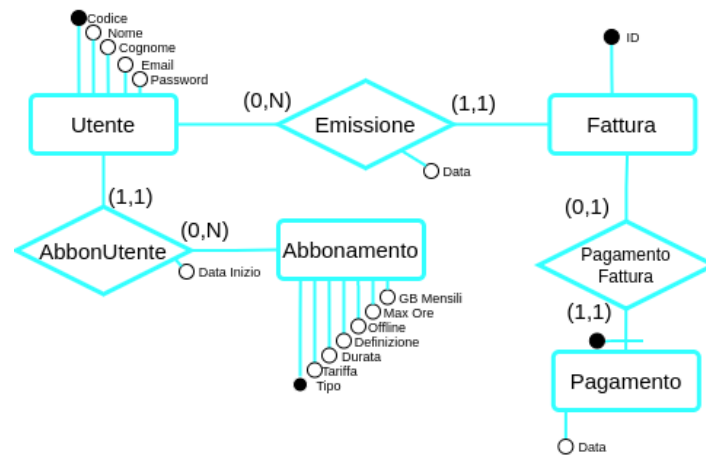
Descrizione: Un **Utente** può decidere di cambiare piano di abbonamento, ovviamente prima di fare ciò occorre controllare che sia in pari coi pagamenti.

Input: Codice, Tipo

Output:

Frequenza Giornaliera: 1 667 Poiché gli abbonamenti hanno durata mensile si stima che ogni fine mese il 5% degli utenti decida di cambiare abbonamento

Porzione di Diagramma Interessata:



Porzione della Tabella dei Volumi:

Concetto	Tipo	Volume
Utente	E	1 000 000
Abbonamento	E	5
Fattura	E	12 000 000
Pagamento	E	11 000 000
Abbonamento- Utente	R	1 000 000
Emissione	R	12 000 000
Pagamento- Fattura	R	11 000 000

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Abbon- Utente	R	1 000 000	R	Si determina l' <i>ID</i> dell'attuale Abbonamento accedendo a tutto Abbonamento Utente . Se l' <i>ID</i> ottenuto è il medesimo al quale si vuole passare (caso improbabile) si termina l'operazione
Emissione	R	12 000 000	R	Dall' <i>ID</i> dell' Utente si leggono le Emissioni relative a egli. Si ottengono quindi gli <i>ID</i> delle Fatture . Si ricorda che solo i dati degli ultimi 12 mesi sono memorizzati nella Base di Dati
Pagamento	E	11 000 000	R	Avendo Pagamento stessa chiave di Fattura , ne si controlla l'esistenza (tentativo di lettura)
Abbon- Utente	R	1	W	Si modifica la relazione sostituendo il vecchio Abbonamento con quello nuovo
Totale:		24 000 002		
Totale Giornaliero:		40 008 003 334		

5.2.8 Film con Recensioni migliori

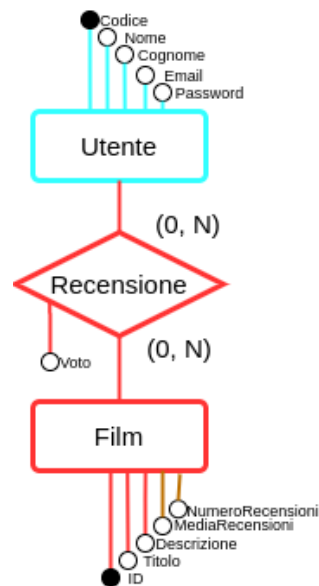
Descrizione: Restituire i migliori 20 **Film** con la più alta media di *Voto* nelle **Recensioni** associate.

Input: Nulla

Output: Lista di *ID* e *Titoli* di **Film**.

Frequenza Giornaliera: 500 000 Si considera che ogni volta che un utente si connette gli vengano mostrato i **Film** con **Recensioni** migliori come raccomandazione automatica

Porzione di Diagramma interessato:



Porzione della Tabella dei Volumi:

Concetto	Tipo	Volume
Film	E	1 000
Recensione	R	25 000

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Film	E	1 000	R	La ridondanza <i>MediaRecensioni</i> permettere di individuare il miglior Film senza dover calcolare le recensioni medie
Totale:		1 000		
Totale Giornaliero:		500 000 000		Soluzione Ottimale

Tavola degli Accessi:

Concetto	Costrutto	Accessi	Tipo	Note
Recensione	R	25 000	R	Bisogna leggere tutte le Recensioni per calcolare la media per film
Film	E	1 000	R	Occorre leggere i Titoli dei Film
Totale:		26 000		
Totale Giornaliero:		98 000 000 000		Soluzione Non Ottimale

6 Progettazione Logica

6.1 Ridenominazione Nomi

6.2 Traduzione in schema Logico

Concetti ER	Traduzione	Note
Film, Produzione, Regia	Film(<u>ID</u> , Titolo, Descrizione, Anno, CasaProduzione, NomeRegista, CognomeRegista, MediaRecensioni, NumeroRecensioni)	
Artista	Artista(<u>Nome</u> , <u>Cognome</u> , Popolarità)	
Premio, VincitaPremio, Vincita, VincitaArtista, VincitaFilm	VincitaPremio(Macrotipo, Microtipo, <u>Data</u> , Film, NomeArtista, CognomeArtista)	1
Recitazione	Recitazione(<u>Film</u> , <u>NomeAttore</u> , <u>CognomeAttore</u>)	
Critico	Critico(<u>Codice</u> , Nome, Cognome)	
Critica	Critica(<u>Critico</u> , <u>Film</u> , Testo, Data, Voto)	
Recensione	Recensione(<u>Utente</u> , <u>Film</u> , Voto)	

Concetti ER	Traduzione	Note
Genere	Genere(<u>Nome</u>)	
GenereFilm	GenereFilm(<u>Genere</u> , <u>Film</u>)	
CasaProduzione, Appartenenza	CasaProduzione(<u>Nome</u> , Paese)	
Lingua	Lingua(<u>Nome</u>)	
Doppiaggio	Doppiaggio(<u>File</u> , <u>Lingua</u>)	
Sottotitolaggio	Sottotitolaggio(<u>File</u> , <u>Lingua</u>)	
Paese	Paese(<u>Codice</u> , Nome, Posizione)	
File, VideoFile, AudioFile, EdizioneFile	File(<u>ID</u> , Dimensione, BitRate, FormatoContenitore, FamigliaAudio, VersioneAudio, FamigliaVideo, VersioneVideo, Risoluzione, FPS, BitDepth, Frequenza, Edizione)	
FormatoCodifica	FormatoCodifica(Famiglia, <u>Versione</u> , Lossy, MaxBitrate)	
Edizione, EdizioneFilm	Edizione(<u>ID</u> , Film, Anno, Tipo, Lunghezza, Rapporto D'Aspetto)	

Concetti ER	Traduzione	Note
Restrizione	Restrizione(<u>Edizione</u> , <u>Paese</u>)	
IP Range, Geolocal	IPRange(<u>Inizio</u> , <u>Fine</u> , <u>DataInizio</u> , DataFine, Paese)	
Server	Server(<u>ID</u> , LunghezzaBanda, MTU, MaxConnessioni, CaricoAttuale, Posizione)	
P.o.P.	P.o.P.(<u>File</u> , <u>Server</u>)	
Distanza- Precalcolata	DistanzaPrecalcolata(<u>Paese</u> , <u>Server</u> , ValoreDistanza)	
Visualizzazione, VisualizzazioneE- dizione, Visualizzazione- Connession	Visualizzazione(Timestamp, <u>Edizione</u> , <u>Utente</u> , <u>IP</u> , <u>InizioConnessione</u>)	
Erogazione	Erogazione(Timestamp, <u>Edizione</u> , <u>Utente</u> , <u>IP</u> , <u>InizioConnessione</u> , InizioErogazione, Server)	2
Connessione, ConnessioneU- tente	Connessione(<u>Utente</u> , <u>IP</u> , <u>Inizio</u> , Fine, Hardware)	
Utente, Abbona- mentoUtente	Utente(<u>Codice</u> , Nome, Cognome, Email, Password, Abbonamento, DataInizioAbbonamento)	

Concetti ER	Traduzione	Note
Abbonamento	Abbonamento(Tipo, Tariffa, Durata, Definizione, Offline, MaxOre, GBMensili)	
Esclusione	Esclusione(<u>Abbonamento</u> , <u>Genere</u>)	
Fattura, Emissione, Pagamento, PagamentoCarta	Fattura(<u>ID</u> , Utente, DataEmissione, DataPagamento, CartaDiCredito)	3
Carta di Credito	CartaDiCredito(<u>PAN</u> , Scadenza, CVV)	
Film Vis Giorni, Paese Vis Giorni, Visualizzazioni Giornaliere	VisualizzazioniGiornaliere(<u>Film</u> , <u>Data</u> , <u>Paese</u> , NumeroVisualizzazioni)	

Note:

1. Stimiamo che la metà dei premi siano vinti da un artista, dunque è più conveniente avere due valori nulli invece di ripetere metà delle volte i 3 attributi chiave in una tabella che avrebbe tradotto la relazione VincitaArtista
2. Le Erogazioni conterranno i dati solo delle visualizzazioni attive. Se **Erogazione** fosse stato tradotto come un attributo *ServerAttuale* in **Visualizzazione**, al termine di essa sarebbe dovuto essere impostato come nullo. Visualizzazione avrebbe avuto quindi la maggior parte delle occorrenze con un campo nullo.
3. Pagamento e PagamentoCarta confluiscono dentro Fattura in quanto entrambe le entità condividono la chiave e si stima che al più 1/12 delle fatture possa non essere pagata. Si ha quindi anche un risparmio

di memoria in quanto non serve memorizzare più volte la chiave di Fattura

6.3 Vincoli

6.3.1 Vincoli di Integrità Referenziale

1. *CasaProduzione* di **Film** in *Nome* di **CasaProduzione**
2. *NomeRegista, CognomeRegista* di **Film** in *Nome, Cognome* di **Artista**
3. *Film* di **VincitaPremio** in *ID* di **Film**
4. *NomeArtista, CognomeArtista* di **VincitaPremio** in *Nome, Cognome* di **Artista**
5. *Film* di **Recitazione** in *ID* di **Film**
6. *NomeAttore, CognomeAttore* di **Recitazione** in *Nome, Cognome* di **Artista**
7. *Critico* di **Critica** in *Codice* di **Critico**
8. *Film* di **Critica** in *ID* di **Film**
9. *Utente* di **Recensione** in *Codice* di **Utente**
10. *Film* di **Recensione** in *ID* di **Film**
11. *Genere* di **GenereFilm** in *Nome* di **Genere**
12. *Film* di **GenereFilm** in *ID* di **Film**
13. *Paese* di **CasaProduzione** in *Codice* di **Paese**
14. *File* di **Doppiaggio** in *ID* di **File**
15. *Lingua* di **Doppiaggio** in *Nome* di **Lingua**
16. *File* di **Sottotitolaggio** in *ID* di **File**
17. *Lingua* di **Sottotitolaggio** in *Nome* di **Lingua**
18. *FamigliaAudio, VersioneAudio* di **File** in *Famiglia, Versione* di **FormatoCodifica**
19. *FamigliaVideo, VersioneVideo* di **File** in *Famiglia, Versione* di **FormatoCodifica**
20. *Edizione* di **File** in *ID* di **Edizione**
21. *Film* di **Edizione** in *ID* di **Film**
22. *Edizione* di **Restrizione** in *ID* di **Edizione**
23. *Paese* di **Restrizione** in *Codice* di **Paese**
24. *Paese* di **IP Range** in *Codice* di **Paese**
25. *File* di **P.o.P** in *ID* di **File**
26. *Server* di **P.o.P** in *ID* di **Server**

27. *Paese* di **DistanzaPrecalcolata** in *Codice* di **Paese**
28. *Server* di **DistanzaPrecalcolata** in *Server* di **Paese**
29. *Edizione* di **Erogazione** in *ID* di **Edizione**
30. *Timestamp, Utente, IP, Edizione, InizioConnessione* di **Erogazione** in *Timestamp, Utente, IP, Edizione, InizioConnessione* di **Visualizzazione**
31. *Server* di **Erogazione** in *ID* di **Server**
32. *Edizione* di **Visualizzazione** in *ID* di **Edizione**
33. *Utente, IP, InizioConnessione* di **Visualizzazione** in *Utente, IP, Inizio* di **Connessione**
34. *Utente* di **Connessione** in *Codice* di **Utente**
35. *Abbonamento* di **Utente** in *Tipo* di **Abbonamento**
36. *Abbonamento* di **Esclusione** in *Tipo* di **Abbonamento**
37. *Genere* di **Esclusione** in *Nome* di **Genere**
38. *Utente* di **Fattura** in *Codice* di **Utente**
39. *CartaDiCredito* di **Fattura** in *PAN* di **Carta di Credito**
40. *Film* di **Visualizzazioni Giornaliere** in *ID* di **Film**
41. *Paese* di **Visualizzazioni Giornaliere** in *Codice* di **Paese**

6.3.2 Vincoli di Tupla

Ogni attributo che fa parte di una chiave ha il vincolo *NOT NULL*.

I seguenti attributi hanno la possibilità di essere *NULL*:

1. *DataFine* di **IPRange**, quando un range è in corso di validità
2. *MaxBitRate* di **FormatoCodifica**, quando la metodologia in questione non pone massimi al *BitRate* di un **File**

I seguenti attributi, invece, non potranno essere *NULL* (anche se un'altra interpretazione potrebbe dargli questa possibilità) in quanto:

1. *MaxOre* di **Abbonamento**, quando non vi è un limite sulle ore massime di streaming al giorno lo si imposta a 100
2. *GBMensili* di **Abbonamento**, quando non vi è un limite sulla mole massima di streaming al mese lo si imposta a 10 000

Questa scelta è stata effettuata nell'ottica di semplificare i controlli di tali valori durante lo streaming.

Sono individuati, oltretutto, i seguenti vincoli intra-relazionali:

1. L'istante di *Fine* di una **Connessione** deve essere non minore di quello di *Inizio*
2. Se impostata, la *DataFine* di un **IPRange** deve essere non minore di *DataInizio*
3. Un **IPRange** deve avere IP di *Fine* non minore di quello di *Inizio*

6.3.3 Vincoli di Dominio

Sono individuati i seguenti vincoli sul dominio di alcuni attributi.

1. Una **Recensione** deve avere un *Voto* compreso tra 0.0 e 5.0
2. Una **Critica** deve avere un *Voto* compreso tra 0.0 e 5.0
3. In ogni momento, una **Critica** deve avere *Data* non maggiore della data corrente
4. In ogni momento il *Timestamp* di una **Visualizzazione** deve essere non maggiore di *CURRENT_TIMESTAMP*
5. L'istante di *Fine* di una **Connessione** deve essere non minore di quello di *Inizio*
6. Sia l'istante di *Inizio* che quello di *Fine* di una **Connessione** devono essere, in ogni momento, non maggiori di *CURRENT_TIMESTAMP*
7. La *DataInizio* di **IPRange** deve essere non maggiore della data corrente

6.3.4 Vincoli Inter-Relazionali

Sono individuati i seguenti vincoli Inter-Relazionali.

Il loro scopo è mantenere coerenza tra i vari dati memorizzati in relazioni diverse nella stessa Base di Dati.

1. Una **Critica** deve avere l'anno della *Data* non minore dell'*Anno* di produzione del **Film** criticato
2. Un'**Edizione** deve avere *Anno* di pubblicazione non minore dell'*Anno* di produzione del **Film**
3. Un **File**, se ha associati **FormatoAudio** e **Videoentrambi** con un *MaxBitRate* deve avere l'attributo *BitRate* non maggiore della somma degli altri due
4. L'anno del *Timestamp* di una **Visualizzazione** deve essere non minore dell'*Anno* di pubblicazione dell'**Edizione** visualizzata
5. Il *Timestamp* di una **Visualizzazione** deve essere compreso tra gli istanti di *Inizio* e di *Fine* della **Connessione** associata

6. L'istante di *Fine* di una **Connessione** deve essere non minore di quello di *Inizio*
7. Una **CartadiCredito** utilizzata in un **Pagamento** non deve essere scaduta nella **DataPagamento**
8. Il timestamp *InizioErogazione* di **Erogazione** non può essere maggiore di *CURRENT_TIMESTAMP* e non può essere minore del *TimeStamp* di inizio della Visualizzazione associata
9. In uno stesso istante non possono esistere due o più **IPRange** validi che collidono. Due range collidono quando esiste un indirizzo IP che appartiene ad entrambi i range

6.4 Analisi delle Dipendenze Funzionali e Normalizzazione

A questo punto si procede con l'analisi delle dipendenza funzionali delle relazioni ottenute dalla traduzione. Il risultato ottenuto, come di seguito, è in forma BCNF. Per ogni tabella in elenco, si analizzano le dipendenza funzionali in essa presenti e si inserisce un ✓ ove la BCNF viene soddisfatta, altrimenti si opera una decomposizione.

- **Film**(ID, Titolo, Descrizione, Anno, CasaProduzione, NomeRegista, CognomeRegista, MediaRecensioni, NumeroRecensioni) ✓
ID → Titolo, Descrizione, Anno, CasaProduzione, NomeRegista, CognomeRegista, MediaRecensioni, NumeroRecensioni
- **Artista**(Nome, Cognome, Popolarità) ✓
Nome, Cognome → Popolarità
- **VincitaPremio**(Macrotipo, Microtipo, Data, Film, NomeArtista, CognomeArtista) ✓
Macrotipo, Microtipo, Data → Film, NomeArtista, CognomeArtista
- **Recitazione**(Film, NomeAttore, CognomeAttore) ✓
- **Critico**(Codice, Nome, Cognome) ✓
Codice → Nome, Cognome
- **Critica**(Critico, Film, Testo, Data, Voto) ✓
Critico, Film → Testo, Data, Voto

- **Recensione**(Utente, Film, Voto) ✓
Utente, Film → Voto
- **Genere**(Nome) ✓
- **GenereFilm**(Genere, Film) ✓
- **CasaProduzione**(Nome, Paese) ✓
Nome → Paese
- **Lingua**(Nome) ✓
- **Doppiaggio**(File, Lingua) ✓
- **Sottotitolaggio**(File, Lingua) ✓
- **Paese**(Codice, Nome, Posizione) ✓
Codice → Nome, Posizione
Nome → Codice
- **File**(ID, Dimensione, BitRate, FormatoContenitore, FamigliaAudio, VersioneAudio, FamigliaVideo, VersioneVideo, Risoluzione, FPS, BitDepth, Frequenza, Edizione) ✓
ID → Dimensione, BitRate, FormatoContenitore, FamigliaAudio, VersioneAudio, FamigliaVideo, VersioneVideo, Risoluzione, FPS, BitDepth, Frequenza, Edizione
- **FormatoCodifica**(Famiglia, Versione, Lossy, MaxBitrate) ✓
Famiglia, Versione → Lossy, MaxBitrate
- **Edizione**(ID, Film, Anno, Tipo, Lunghezza, Rapporto D'Aspetto) ✓
ID → Film, Anno, Tipo, Lunghezza, Rapporto D'Aspetto
- **Restrizione**(Edizione, Paese) ✓
- **IPRange**(Inizio, Fine, DataInizio, DataFine, Paese) ✓
Inizio, Fine, DataInizio → DataFine, Paese
- **Server**(ID, Risoluzione, LunghezzaBanda, MTU, MaxConnessioni, CaricoAttuale, Posizione) ✓
ID → LunghezzaBanda, MTU, MaxConnessioni, CaricoAttuale, Posizione
- **P.o.P.**(File, Server) ✓
- **DistanzaPrecalcolata**(Paese, Server, ValoreDistanza) ✓
Paese, Server → ValoreDistanza

- **Erogazione**(Timestamp, Edizione, Utente, IP, InizioConnessione, InizioErogazione, Server) ✓
Timestamp, Edizione, Utente, IP, InizioConnessione → InizioErogazione, Server
- **Visualizzazione**(Timestamp, Edizione, Utente, IP, InizioConnessione) ✓
- **Connessione**(Utente, IP, Inizio, Fine, Hardware) ✓
Utente, IP, Inizio → Fine, Hardware
- **Utente**(Codice, Nome, Cognome, Email, Password, Abbonamento, DataInizioAbbonamento) ✓
Codice → Nome, Cognome, Email, Password, Abbonamento, DataInizioAbbonamento
- **Abbonamento**(Tipo, Tariffa, Durata, Definizione, Offline, MaxOre, GBMensili) ✓
Tipo → Tariffa, Durata, Definizione, Offline, MaxOre, GBMensili
- **Esclusione**(Abbonamento, Genere) ✓
- **Fattura**(ID, Utente, DataEmissione, DataPagamento, CartaDiCredito) ✓
ID → Utente, DataEmissione, DataPagamento, CartaDiCredito
- **CartaDiCredito**(PAN, Scadenza, CVV) ✓
PAN → Scadenza, CVV
- **Visualizzazioni Giornaliere**(Film, Data, Paese, NumeroVisualizzazioni) ✓
Film, Data, Paese → NumeroVisualizzazioni

Per quanto riguarda la tabella *Connessione* si potrebbe pensare che esista la dipendenza funzionale $IP, Inizio \rightarrow Fine, Hardware$, tuttavia qualora qualcuno si collegasse dallo stesso PC, aprendo due tab in due browser diversi nello stesso istante, avrei stesso IP e Inizio ma due User-Agent diversi, dunque due Hardware diversi.

7 Raccomandazione Utente

7.1 Rating

Il rating di un film è una valutazione espressa in numero di stelle, con la possibilità di averne mezze, che parte da un minimo di zero ed arriva ad un massimo di 5.

Ogni film possiede due diverse tipologie di rating, ossia il rating del film ed il rating personalizzato, la prima è una metrica oggettiva che è, dunque, fissa per ogni film, la seconda, invece, varia sulla base del singolo utente in quanto dipendente dalla "history" di quest'ultimo.

Precedentemente si era già parlato di una forma di valutazione, espressa come media delle varie recensioni. Tuttavia, essa differisce dal rating non solo in merito al sistema di valutazione utilizzato, stelle invece che numerico, ma anche e soprattutto per le idee dietro di esse. Mentre la media valutazione è una misura molto superficiale, che corrisponde solamente al pensiero dei vari utenti, il rating si propone come una valutazione più sofisticata, complessa e sicuramente completa, infatti essa prende in considerazione una moltitudine di parametri, quali le recensioni di critici, le popolarità sia del regista che degli attori, il numero di premi vinti ed, eventualmente, le preferenze dell'utente.

7.1.1 Rating del Film

Il rating del film consente, come detto precedentemente, di dare una valutazione a tutto tondo e soprattutto oggettiva di un film.

Siano R_U la media delle recensioni degli utenti, R_C la media delle recensioni dei critici, RM_U la media delle recensioni degli utenti più alta tra tutti i film aventi almeno un genere in comune con il film, P_A la media delle popolarità degli attori, P_R la media delle popolarità dei registi ed, infine, PV i premi vinti dal film.

L'algoritmo è il seguente:

$$V = \frac{[0.5 * (R_U + R_C) + 0.1 * (P_A + P_R) + 0.1 * PV + (R_U/RM_U)]}{2}$$

L'idea dietro di esso è assegnare un valore che va da 0 a 10 ai seguenti

parametri: media recensioni e valutazioni, media popolarità attori e regista, numero di premi vinti e media recensioni rispetto al massimo per il genere. Assegnati i valori occorre fare una media pesata coi seguenti pesi: 0.5, 0.2, 0.2 e 0.1, prendere la parte intera del risultato e dividerla per due (in questo modo otteniamo le mezze stelle per un giudizio che arriva al massimo a 5 stelle).

7.1.2 Rating per Utente

Il rating per utente consente, come detto precedentemente, di dare una valutazione soggettiva di un film, basandosi sulle preferenze dell'utente.

Siano G_1, G_2 i 2 generi preferiti dell'utente, A_1, A_2, A_3 i 3 attori preferiti dell'utente, L_1, L_2 le 2 lingue, in cui l'audio è disponibile, preferite dell'utente, R il regista preferito ed, infine, RA il rapporto d'aspetto preferito.

Questi valori valgono 1 se sono presenti nel film altrimenti 0 (e.g. Se G_1 corrisponde a *Fantascienza*, in ogni film di *Fantascienza* G_1 vale 1)

Per preferiti si intende i più visti nell'ultimo mese.

L'algoritmo è il seguente:

$$V = \frac{\lfloor 2 * G_1 + G_2 + 1.5 * A_1 + A_2 + 0.5 * A_3 + L_1 + L_2 + R + RA \rfloor}{2}$$

L'idea dietro di esso è assegnare un bonus per ogni preferenza dell'utente che viene soddisfatta dal film in questione, ovviamente ad ogni preferenza viene attribuito uno specifico peso in base alla rilevanza. La somma di tutti questi pesi equivale a 10, ossia il massimo valore ottenibile che corrisponde ad una valutazione di 5 stelle.

Sommati tutti i bonus occorre prendere la parte intera per poi dividerla per due (in questo modo otteniamo le mezze stelle per un giudizio che arriva al massimo a 5 stelle).

7.1.3 Sistema di Raccomandazione

Descrizione: Il sistema di raccomandazione di contenuti è in grado di, dato un utente e la sua "history" di visualizzazioni, consigliare e proporre film potenzialmente graditi da quest'ultimo.

Una metrica perfetta per misurare quanto uno specifico film possa essere potenzialmente apprezzato da un particolare utente è il suddetto meccanismo di **Rating Utente** in quanto riesce a coniugare alla perfezione i gusti soggettivi con l'esigenza di raccomandazione.

Input: Codice dell'Utente e Numero di film richiesti

Output: Codici dei Film Raccomandati

```
DROP PROCEDURE IF EXISTS 'RaccomandazioneContenuti';
DELIMITER //
CREATE PROCEDURE 'RaccomandazioneContenuti' (
    IN codice_utente VARCHAR(100),
    IN numero_film INT
)
BEGIN

    WITH
        FilmRatingUtente AS (
            SELECT
                ID,
                RatingUtente(ID, codice_utente) AS Rating
            FROM Film
        )
    SELECT ID
    FROM FilmRatingUtente
    ORDER BY Rating DESC, ID
    LIMIT numero_film;

END
//
DELIMITER ;
```

Commento Codice: Sfruttando la funzione RatingUtente è possibile creare una tabella $T(Film, RatingUtente)$, sulla quale poi verrà operato un ordinamento decrescente in base al RatingUtente, usando ORDER BY, per poi andare a selezionare solo i primi N film graditi dall'utente, mediante LIMIT N.

8 Area Analytics

8.1 Classifiche

Descrizione: Questa funzionalità si occupa di stilare delle classifiche di **Film** e di **Edizioni** più gettonate sulla base sia del **Paese** che dell'**Abbonamento**. L'idea dietro questa funzionalità è di poter comporre una classifica degli N **Film** oppure **Edizioni** più visti in uno specifico **Paese** solo da **Utenti** con uno specifico **Abbonamento**.

Input: N , Codice del Paese, Tipo di Abbonamento, P (Variabile che determina cosa classificare)

Output: Classifica

```
CREATE PROCEDURE IF NOT EXISTS 'Classifica'(  
    N INT,  
    codice_paese CHAR(2),  
    tipo_abbonamento VARCHAR(50),  
    P INT -- 1 -> Film 2 -> Edizioni  
)  
BEGIN  
  
    IF p = 1 THEN  
  
        WITH 'FilmVisualizzazioni' AS (  
            SELECT  
                E.'Film',  
                COUNT(*) AS "Visualizzazioni"  
            FROM 'Visualizzazione' V  
            INNER JOIN 'Utente' U ON V.'Utente' = U.'Codice'  
            INNER JOIN 'Edizione' E ON E.'ID' = V.'Edizione'  
            LEFT OUTER JOIN 'IPRange' R ON  
                (V.'IP' BETWEEN R.'Inizio' AND R.'Fine') AND  
                (V.'InizioConnessione' BETWEEN R.'DataInizio' AND  
                    IFNULL(R.'DataFine', CURRENT_TIMESTAMP))  
            WHERE U.'Abbonamento' = tipo_abbonamento AND IFNULL  
                (R.'Paese', '??') = codice_paese  
            GROUP BY E.'Film'  
        )  
        SELECT 'Film'
```



```

FROM 'FilmVisualizzazioni'
ORDER BY 'Visualizzazioni' DESC
LIMIT N;

ELSEIF p = 2 THEN

    WITH 'EdizioneVisualizzazioni' AS (
        SELECT
            V.'Edizione',
            COUNT(*) AS "Visualizzazioni"
        FROM 'Visualizzazione' V
        INNER JOIN 'Utente' U ON V.'Utente' = U.'Codice'
        LEFT OUTER JOIN 'IPRange' R ON
            (V.'IP' BETWEEN R.'Inizio' AND R.'Fine') AND
            (V.'InizioConnessione' BETWEEN R.'DataInizio' AND
                IFNULL(R.'DataFine', CURRENT_TIMESTAMP))
        WHERE U.Abbonamento = tipo_abbonamento AND IFNULL
            (R.'Paese', '??') = codice_paese
        GROUP BY V.'Edizione'
    )
    SELECT 'Edizione'
    FROM 'EdizioneVisualizzazioni'
    ORDER BY 'Visualizzazioni' DESC
    LIMIT N;

ELSE

    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'Parametro P non Valido';

END IF;

END

```

Commento Codice: La stored procedure prende in input i parametri sopra descritti ed, in base al valore di *P* decide come agire, ovvero se classificare i **Film** oppure le **Edizioni**.

Successivamente, crea una **CTE** *VisualizzazioniFilm* contenente tutti i Film con il relativo numero di visualizzazioni che soddisfano sia i criteri riguardanti la collocazione geografica che quelli riguardanti il tipo di abbonamento. Per controllare il primo occorre fare un JOIN con la tabella **IPRange** selezionando il range valido.

Invece, per controllare il secondo occorre fare un *JOIN* tra la tabella **Visualizzazione** e quella **Utente** per determinare se il tipo di abbonamento

sia quello desiderato.

Si conclude ordinando in base a Visualizzazioni e restituendo gli N valori più grandi.

Qualora si volessero invece le **Edizioni** non occorre un *JOIN* ma si procede direttamente col raggruppamento, l'ordinamento e la restituzione dei migliori N valori.

8.2 Custom Analytics

La Custom Analytics ideata è quella di *Migliore Attore e Migliore Regista Inaspettato* della settimana.

Ogni settimana, considerando artisti aventi una popolarità minore di una certa soglia, viene stilata una classifica degli artisti e registi più gettonati.

Questa funzionalità serve a dare rilievo e lustro ad artisti debuttanti, emergenti o semplicemente meno noti al grande pubblico. Siamo convinti che questa opzione susciterebbe indubbiamente grande interesse tra gli utenti di *FilmSphere* in quanto sarebbe un modo per aprire i propri orizzonti ad artisti minori e, pertanto, ampliare la propria cultura cinematografica anche, magari, sviluppando un giudizio critico verso attori o registi debuttanti.

Innanzitutto, occorre definire la soglia massima di popolarità tra gli artisti considerati, che decidiamo di fissare a 2.5 in quanto siamo particolarmente interessati ad artisti poco rilevanti.

Per quanto concerne il punteggio da attribuire ad un attore o regista vengono presi in considerazione tutti i **Film** in cui quest'ultimo ha recitato oppure ha diretto e si somma al suo punteggio la valutazione media del **Film**, a cui vanno aggiunti eventuali 5 punti qualora il **Film** in questione risultasse vincitore di un qualsivoglia tipo di premio. Inoltre, per ogni premio eventualmente vinto da egli (artista) vengono attribuiti 100 punti.

Siano v_i la valutazione media dell' i -esimo **Film** in cui ha partecipato l'**Artista**, p_i una variabile che assume come valore 1 se l' i -esimo film ha vinto almeno un premio e 0 altrimenti, n il numero di premi vinti dall'**Artista**, allora il suo punteggio P è uguale a:

$$P = \sum_{i=1}^n (v_i + 5 * p_i) + 100 * n$$

Implementazione:

```
CREATE FUNCTION 'ValutazioneRegista'(  
    Nome VARCHAR(50),  
    Cognome VARCHAR(50)  
)  
RETURNS FLOAT DETERMINISTIC  
BEGIN  
  
    DECLARE sum_v FLOAT;  
    DECLARE sum_p FLOAT;  
    DECLARE n INT;  
  
    SET sum_v := (  
        SELECT  
            SUM(MediaRecensioni)  
        FROM Film  
        WHERE NomeRegista = Nome AND CognomeRegista = Cognome  
    );  
  
    SET sum_p := (  
        SELECT  
            COUNT(DISTINCT VP.Film)  
        FROM Film F  
        INNER JOIN VincitaPremio VP  
            ON VP.Film = F.ID  
        WHERE F.NomeRegista = Nome AND F.CognomeRegista = Cognome  
    );  
  
    SET n := (  
        SELECT  
            COUNT(*)  
        FROM VincitaPremio  
        WHERE NomeArtista = Nome AND CognomeArtista = CognomeArtista  
    );  
  
    RETURN sum_v + sum_p * 5 + n * 100.0;  
END
```

Questa è la funzione che permette, nel caso di un regista, di calcolare il suo punteggio. Per ottenere la somma della media delle recensioni dei film in cui ha recitato si accede direttamente a film e la si legge, per il numero di film vincitori di premi in cui ha visitato, invece, si accede a film, imponendo che il regista del film sia il regista "target", e si fa un JOIN con **VincitaPremio** per selezionare solamente i film vincitori di premi. Infine, per ottenere il numero di premi vinti dal regista si accede diretta-

mente a **VincitaPremio**.

```
DROP PROCEDURE IF EXISTS 'MiglioreRegista';
DELIMITER //
CREATE PROCEDURE IF NOT EXISTS 'MiglioreRegista'()
BEGIN

    WITH
        RegistaValutazione AS (
            SELECT
                Nome, Cognome,
                ValutazioneRegista(Nome, Cognome) AS Valutazione
            FROM Artista
            WHERE Popolarita <= 2.5
        )
    SELECT
        Nome, Cognome
    FROM RegistaValutazione
    WHERE Valutazione = (
        SELECT MAX(Valutazione)
        FROM RegistaValutazione
    );

END
//
DELIMITER ;
```

Questa è la procedura che ritorna il migliore, o i migliori in caso di ex aequo, regista. Innanzitutto, si procede creando un CTE contenente tutti i registi con una popolarità non sopra il 2.5, assieme alla loro valutazione ottenuta con la funzione precedente.

A questo punto, si selezionano solamente i registi avente Valutazione massima.

8.3 Bilanciamento carico

Descrizione: Per prevenire sovraccarichi sui **Server**, si cerca di prevedere le **Edizioni** che saranno più richieste nel futuro prossimo in un determinato Paese, in modo da copiare/spostare i **File** di tali **Edizioni** sui **Server** più

vicini a quel **Paese**.

Input: N, M numeri naturali

Output: Lista di spostamenti da effettuare

La procedura funziona al seguente modo:

1. Per ogni **Paese** si individuano le **Visualizzazioni** avvenute in quel **Paese** dall'*IP*
2. Si raggruppa in base al **Paese** e all'**Edizione** visualizzata contando le visualizzazioni per ogni **Edizione** in ogni **Paese**
3. Per ogni **Paese** si selezionano le N **Edizioni** più visualizzate
4. Per ogni **Paese** si individuano gli M **Server** più vicini
5. Ogni coppia Edizione-Server, ignorando il **Paese** che non serve più, viene aggiunta, con attenzione a non aggiungere duplicati, alla lista da restituire
6. Per ogni **Edizione** si individuano tutti i **File** e si trasforma la lista in una File-Server; si scartano le coppie dove esiste già un **P.o.P.** tra quel **File** e quel **Server**

Allo stesso modo si può utilizzare lo stesso criterio per individuare i **File** che non è necessario tenere su un **Server**, seguendo la stessa procedura ma scegliendo, invece delle più visualizzate, le **Edizioni** meno visualizzate e, invece di prendere le coppie dove non vi è un **P.o.P.**, si prendono solo le coppie dove esiste già il **P.o.P.**

9 Area Streaming

9.1 Individuazione Server

Descrizione: Quando un **Utente** richiede di guardare un'**Edizione**, l'applicazione invia la richiesta al DBMS che deve restituire il miglior Server al quale connettersi.

Input: *Codice Utente*, *ID* dell'**Edizione**, *IP* dell'Utente, *MaxBitRate*, *MaxRis*, *ListaEncodingAudio*, *ListaEncodingVideo*

Output: *ID* del **Server** migliore assieme all'*ID* del **File** migliore o nulla in caso di non disponibilità

Dei parametri passati alla procedura, *Codice*, *ID*, *IP* servono ad individuare l'**Utente**, il suo **Abbonamento** e le relative permissions, il **Paese** dove si trova, mentre *MaxBitRate* e *MaxRis* sono parametri determinati dalle caratteristiche del dispositivo e dalla potenza di segnale attuale della rete al quale si è connessi. Tali parametri sono calcolati dall'applicazione lato client e servono per scegliere il **File** dell'**Edizione** più adatto.

ListaEncodingAudio e *ListaEncodingVideo* sono stringhe contenenti le famiglie di **FormatoCodifica** che il dispositivo client supporta per mostrare il contenuto. La selezione si assicura che il **File** rispetti queste caratteristiche.

La procedura seguita per l'individuazione del Server migliore è la seguente:

1. Vengono lette l'entità **Utente** e il relativo *Abbonamento*, il **Film** associato all'**Edizione** e i **Generi** del **Film** e, se esistono, le **Esclusioni** tra l'**Abbonamento** e il **Genere**. Se l'interrogazione restituisce dei valori allora la procedura termina: non si è autorizzati a guardare il contenuto.
2. Dall'*IP* si risale al *Codice* di **Paese** tramite gli **IPRange** e si controlla se vi siano **Restrizioni** tra il **Paese** e l'**Edizione**. Se vi sono restrizioni la procedura termina: non si è autorizzati (per legge) a guardare il contenuto.
3. Si individuano tutti i **File** dell'**Edizione** che hanno codifiche supportate dal client
4. Per ogni **File** trovato, si calcolano i seguenti parametri:

$$\Delta Rate = \begin{cases} MaxBitRate - File.BitRate & \text{se } File.BitRate < MaxBitRate \\ -2(MaxBitRate - File.BitRate) & \text{altrimenti} \end{cases}$$

$$\Delta Ris = \begin{cases} MaxRis - File.Risoluzione & \text{se } File.Risoluzione < MaxRis \\ -2(MaxRis - File.Risoluzione) & \text{altrimenti} \end{cases}$$

Tali valori serviranno poi nella scelta del **File** migliore, più alti sono questi valori, minori saranno le probabilità che il **File** sia scelto

5. Per ogni **File** vengono individuati i **P.o.P.** e da essi i **Server** che li contengono. Dal **Paese** e i **Server**, passando per **DistanzaPrecalcolata** si ottiene la distanza, in km, tra la capitale del **Paese** dell'**Utente** e ogni **Server**. Di ogni **Server** si estrae il *CaricoPercentuale* col seguente modo:

$$CaricoPercentuale = CaricoAttuale / MaxConnessioni$$

6. Si effettuano le seguenti operazioni matematiche per riscalarli i valori trovati:

$$\Delta Rate' = map(\Delta Rate, 0, MAX_RATE, 0, w_{Rate})$$

$$\Delta Ris' = map(\Delta Ris, 0, MAX_RIS, 0, w_{Ris})$$

$$\Delta Pos' = map(ValoreDistanza, 0, MAX_DIST, 0, w_{Pos})$$

$$CaricoPercentuale' = CaricoPercentuale * w_{Carico}$$

Dove *map* è una funzione che riscalda un numero appartenente ad un intervallo portandolo in un nuovo intervallo.

Essa è calcolata al seguente modo:

$$map(x, IN_{Min}, IN_{Max}, OUT_{Min}, OUT_{Max}) = OUT_{Min} + \frac{x - IN_{Min}}{IN_{Max} - IN_{Min}} \cdot (OUT_{Max} - OUT_{Min})$$

I valori *MAX_RATE*, *MAX_RIS* e *MAX_DIST* rappresentano invece il massimo valore che le relative grandezze possono raggiungere.

I valori *w_{Rate}*, *w_{Ris}*, *w_{Pos}* e *w_{Carico}* sono i pesi che permetteranno di confrontare i tali parametri tra di loro, potendo cambiare facilmente l'importanza di uno rispetto all'altro.

Lo score Finale di ogni combinazione File-Server è assegnato come

$$Score = \Delta Rate' + \Delta Ris' + \Delta Pos' + CaricoPercentuale'$$

7. La combinazione con lo *Score* minore sarà restituita all'**Utente**.

Un esempio di valori dei pesi può essere:

- $w_{Rate} = 5$ in modo da essere un parametro poco significativo, 17% del totale
- $w_{Ris} = 3$ in modo da essere il parametro meno importante, 10% del totale
- $w_{Pos} = 12$ in modo da essere il più importante (la distanza determina latency durante la fruizione), 40% del totale
- $w_{Carico} = 10$ In modo da tenere il carico bilanciato tra i diversi **Server**, 33% del totale

9.2 Ribilanciamento del carico

Durante lo streaming di contenuti, gli **Utenti** possono cambiare **Server** dal quale stanno erogando, su richiesta del DBMS stesso, in modo da non sovraccaricare il sistema.

Ogni 10 minuti (intervalli detti tick) il DBMS calcolerà la media di *Carico-Percentuale* dei **Server** e se tale media è sopra il 70%.

Si individuano i (massimo 3) **Server** con *CaricoPercentuale* più alto e sopra la media e, per ciascuno di essi:

1. Si leggono le **Erogazioni** attive che riguardano tale **Server** e un **File** che è presente anche in almeno un altro **Server** che non sia nei 3 ineterssati e, se è presente, dove è presente.
Le **Erogazioni** considerate non devono essere iniziate negli ultimi 3 tick e si deve prevedere (in base al *TimeStamp* di **Visualizzazione** e la *Durata* di **Edizione**) che terminino entro il prossimo.
2. Per ciascuna di queste erogazioni risalgo al **Paese** dove si trova l'Utente dall'*IP* e dal **Paese** al *ValoreDistanza* tra l'**Utente** e il **Server** attuale
3. Per ciascuna di queste erogazioni in corso individuo la migliore alternativa utilizzando lo stesso criterio di scelta iniziale dei **File** e **Server**. Si pone attenzione al fare sì, anche in questo caso, che un **Utente** non possa ricevere un **File** con qualità maggiore a quella massima del suo abbonamento. Oltretutto durante la ricerca del **File** si escludono il **Server** carichi (quelli dai quali si sta cercando di rimuovere **Erogazioni**).
Per ciascuna Alternativa individuata memorizzo anche il suo punteggio

4. Ordinando in base al punteggio, scelgo le migliori coppie Erogazione-Alternativa in numero minore al 5% del *MaxConnessioni* del **Server**: Per ogni Server scelgo le migliori N o meno Alternative, dove N è il % del *MaxConnessioni*.

Il risultato viene messo in una tabella (dal volume insignificante) che, tramite un'interrogazione avviata dai server (macchine) stessi, sarà svuotata e gli **Utenti** riassegnati, modificando solo dopo le **Erogazioni** anche nel DBMS

9.3 Caching previsionale

Descrizione: Si ipotizza che ogni **File** sia presente in un hub centrale, macchina che ha il solo scopo di trasmettere tali contenuti ai **Server**, creando così i **P.o.P.**

In modo da facilitare le operazioni agli algoritmi di *Individuazione Server* e di *Ribilanciamento di quest'ultimi* viene creato anche un *Bilanciamento Preventivo*: algoritmo che cerca di prevedere i contenuti che avranno grande domanda nel prossimo futuro e, con questi dati, cerca di indicare i **Server** nei quali è più opportuno effettuare un **P.o.P.** di tali **File** creando così un sistema di Caching previsionale.

Allo stesso modo l'algoritmo individua i **File** che avranno minore probabilità di essere visualizzati in determinati **Server**, consigliando quindi di rimuoverli per fare spazio ad altri contenuti se lo spazio sulla macchina si dovesse esaurire.

La nostra metrica misurante la probabilità che ha un certo File di essere visualizzato da un certo Utente è la seguente:

Considerati i primi 10 Film secondo la raccomandazione utente associamo ad ognuno di essi le seguenti probabilità decrescenti: 30%, 22%, 11%, 9%, 8%, 6%, 5%, 4%, 3%, 2%

Queste probabilità vanno poi ripartite tra i vari File associati al Film, considerando che si suddividano in maniera equiprobabile

La procedura di individuazione di contenuti da aggiungere funziona al seguente modo:

Input: X, N, M numeri interi non grandi

Output: Lista di spostamenti consigliati

1. Per ogni **Utente** si considera il **Paese** dal quale si connette di più e

dal Paese gli N **Server** più vicini

2. Per ogni coppia **Utente**, **Paese** si considerano gli M **File** con probabilità maggiore di essere guardati, ciascuno con la probabilità di essere guardato
3. si raggruppa in base al **Server** e ad ogni **File**, sommando, per ogni **Server-File** la probabilità che sia guardato dall'**Utente** moltiplicata per un numero che scala in maniera decrescente in base al *ValoreDistanza* tra Paese e Server
4. Si restituiscono le prime X coppie Server-File con somma maggiore per le quali non esiste già un **P.o.P.**

La procedura di individuazione di contenuti da rimuovere funziona allo stesso modo di quella di individuazione di contenuti da raggiungere ma all'ultimo punto vengono restituite le X coppie con somma più bassa dove è presente un **P.o.P.**

10 Implementazione Fisica

10.1 Note su Vincoli

Il vincolo riguardante il massimo bitarte per un file è stato implementato con un before trigger sia su inserimento che su aggiornamento. Le due ridondanze su **Film** sono mantenute con vari trigger su inserimento, modifica e cancellazione su recensione.

DistanzaPrecalcolata viene mantenuta attraverso dei trigger sia su server che su paese. **CaricoAttuale** è stato implementato grazie ad un trigger di inserimento e cancellazione su erogazione.

I vincoli di integrità referenziale sono implementati tramite delle FOREIGN KEY.

10.2 Indicizzazioni e Motori

Per ogni tabella è stato scelto il motore di memorizzazione InnoDB di MySQL, in quanto garantisce accesso mediante BTREE di default sugli indici generati dalle chiavi primarie. Questo rende la performance dei JOIN tra le Connessioni, Visualizzazioni, Erogazioni e IPRange molto veloce.

Oltretutto, quando si crea una FOREIGN KEY con InnoDB viene creato un indice secondario sulla colonna che è chiave esterna: ciò permette di accedere velocemente alle tabelle sia per chiave primaria che per chiave esterna.

10.3 Popolamento

Il popolamento del database è avvenuto attraverso l'ausilio di svariati dataset reperibili online con l'aggiunta di brevi script in python per la generazione automatica di codice in sql.

Per quanto riguarda gli **Utenti**, i codici e le email sono state selezionate da alcuni dataset esistenti.

I nomi e cognomi sono stati generati prendendo un sottoinsieme del prodot-

to cartesiano tra un insieme di nomi e cognomi anch'essi reperiti online. I restanti campi sono stati generati casualmente attraverso gli script [2], [1]. Tutta la sezione relativa a **Fattura**, **Pagamento**, **CartaDiCredito**, **Connessione** e **Visualizzazione** è stata generata in maniera aleatoria, sulla base del popolamento di utente, attraverso i suddetti script.

La tabella **Film** è stata popolata sia attraverso dataset di film italiani che mediante l'assistenza di software di intelligenza artificiale, ossia ChatGPT. La tabella **Vincita** è stata popolata sulla base di premi reali, mentre sia gli **Artisti** che i **Critici** sono stati generati grazie al sopracitato prodotto cartesiano, aggiungendo poi parametri ottenuti casualmente.

Le tabelle **Recitazione**, **Regia**, **Recensione** e **Critica** sono state popolate creando le varie corrispondenze in maniera aleatoria con degli script e includendo eventuali dati aggiuntivi o generati casualmente, come nel caso di Recensione, o presi da un ampio pool di possibilità, come nel caso di Critica.

Sia **Paese** che **Lingua** sono stati popolati utilizzando alcuni dataset, mentre **CasaProduzione** è stato generato selezionando nomi esistenti ma creando associazioni in maniera aleatoria.

Edizione e **File** sono stati generati mediante l'utilizzo che ad ogni Film, o rispettivamente Edizione, associassero un numero variabile di Edizione o File, sempre in modo da rispettare i rapporti stimati in precedenza. I restanti parametri sono stati generati casualmente scegliendoli da un pool molto ampio di opzioni.

FormatoCodifica è stato generato a partire da un elenco di codec [4].

Doppiaggio e **Sottotitolaggio** sono stati popolati con associazioni casuali mediante l'utilizzo di brevi script.

Gli **IPRange** sono realmente esistenti e sono stati ottenuti da un csv online [3].

10.4 Benchmark e Testing

Abbiamo deciso di testare rigorosamente e fare dei benchmark di alcune delle funzionalità offerte dalla piattaforma di streaming online.

I test consistono semplicemente nell'analizzare il tempo di esecuzione di una specifica funzionalità eseguita un numero N di volte, verranno quindi riportati sia le tabelle contenente il numero di esecuzioni associato all'effettivo tempo di esecuzione sia dei grafici per rendere più immediata ed intuitiva la comprensione dei risultati ottenuti.

Nei grafici sarà anche presente la funzione $Durata = m * Chiamate + q$ ottenuta adoperando una regressione lineare sui dati ottenuti attraverso il metodo dei minimi quadrati. che si basa sulla minimizzazione della seguente quantità: $L(x) = \Sigma(y_i - y(x_i))$

$$m = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\Sigma(x_i - \bar{x})^2}$$

$$q = \bar{y} - m\bar{x}$$

Per eseguire le suddette operazioni di testing e benchmark abbiamo deciso di utilizzare un costrutto di MySQL che consente di velocizzare tale procedura. `SELECT BENCHMARK(N, EXPR)` è un costrutto che prende in input il numero di esecuzioni da effettuare, ossia N , insieme all'espressione da valutare, ossia `EXPR`, l'output di questo comando sarà una tabella 1 x 1 avente come risultato 0. In fondo, sarà possibile visionare il tempo di esecuzione.

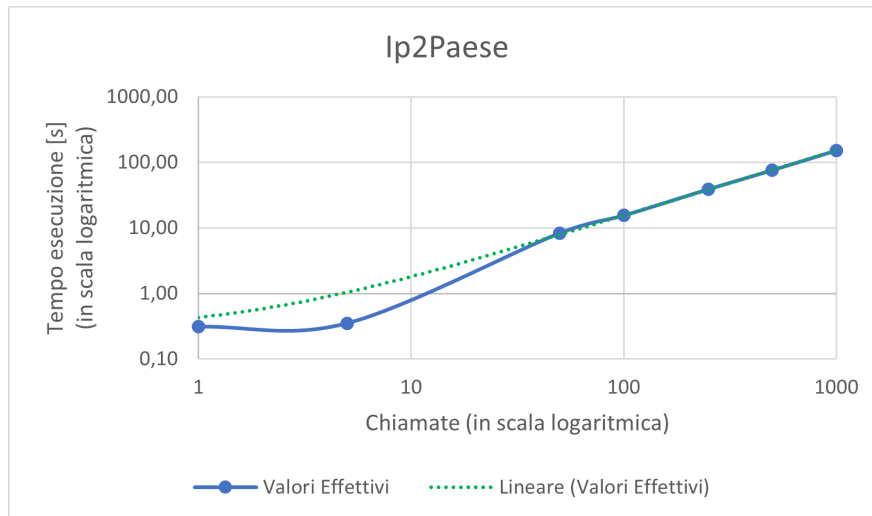
10.4.1 Ip2Paese

Il comando che è stato adoperato ai fini dell'esecuzione del test è il seguente:

```
SELECT BENCHMARK(  
    1000,  
    Ip2Paese(MathMap(RAND(), 0, 1, 16777216, 4261281279))  
);
```

MathMap(RAND(), 0, 1, 16777216, 4261281279)) assolve alla funzione di generare un IP casuale nel range [16777216, 4261281279].

Ip2Paese	
Chiamate	Durata [s]
1	0.31
5	0.35
50	8.34
100	15.51
250	39.02
500	76.42
1000	152.65



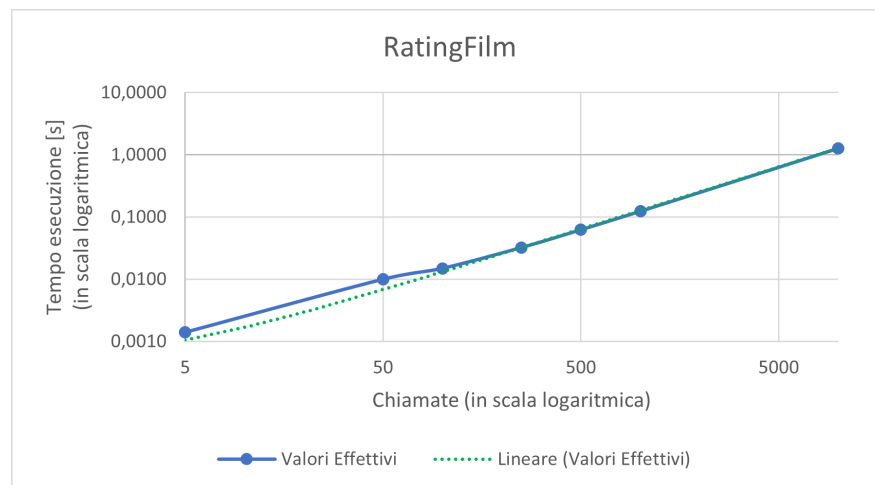
10.4.2 RatingFilm

Il comando che è stato adoperato ai fini dell'esecuzione del test è il seguente:

```
SELECT BENCHMARK(  
    10000,  
    RatingFilm( FLOOR(RAND() * 99) + 1 )  
);
```

'FLOOR(RAND() * 99) + 1' serve a generare un numero casuale compreso tra 1 e 99.

RatingFilm	
Chiamate	Durata [s]
5	0.0014
50	0.0100
100	0.0150
250	0.0324
500	0.0627
1000	0.1246
10000	1.2817



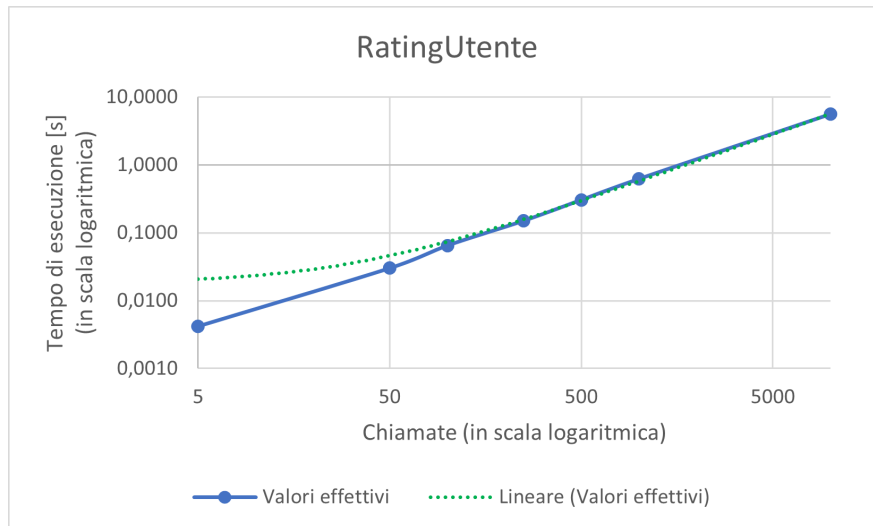
10.4.3 RatingUtente

Il comando che è stato adoperato ai fini dell'esecuzione del test è il seguente:

```
SELECT BENCHMARK(  
    10000,  
    RatingUtente( FLOOR(RAND() * 99) + 1, 'richie-314' )  
);
```

'FLOOR(RAND() * 99) + 1' serve a generare un numero casuale compreso tra 1 e 99. Rappresenterà un *ID* di un **Film**.

RatingUtente	
Chiamate	Durata [s]
5	0.0042
50	0.0302
100	0.0650
250	0.1506
500	0.3059
1000	0.6236
10000	5.6170



10.5 Trigger e Event

Nel database sono stati implementati alcuni trigger ed event, di seguito una breve lista con una descrizione sommaria.

10.5.1 AnnoEdizioneValido

Tabella: Edizione

Tipo: Before Insert

Descrizione: Controlla la validità di anno di edizione

10.5.2 DataCriticaValido

Tabella: Critica

Tipo: Before Insert

Descrizione: Controlla la validità della data di critica

10.5.3 InserimentoFile

Tabella: File

Tipo: Before Insert

Descrizione: Controlla la validità di bitrate

10.5.4 ModificaFile

Tabella: File

Tipo: Before Update

Descrizione: Controlla la validità di bitrate

10.5.5 ModificaErogazione

Tabella: Erogazione

Tipo: Before Update

Descrizione: Aggiorna il timestamp e gestisce i server

10.5.6 AggiungiErogazione

Tabella: Erogazione

Tipo: After Insert

Descrizione: Gestisce i server

10.5.7 RimuoviErogazione

Tabella: Erogazione

Tipo: After Delete

Descrizione: Gestisce i server

10.5.8 IpRangeControlloAggiornamento

Tabella: IPRange

Tipo: Before Update

Descrizione: Impedisce modifiche anomale agli IPRange

10.5.9 InserimentoRecensione

Tabella: Recensione

Tipo: After Insert

Descrizione: Gestisce la Ridondanza in Film

10.5.10 ModificaRecensione

Tabella: Recensione

Tipo: After Update

Descrizione: Gestisce la Ridondanza in Film

10.5.11 CancellazioneRecensione

Tabella: Recensione

Tipo: After Delete

Descrizione: Gestisce la Ridondanza in Film

10.5.12 GestioneVisualizzazione

Frequenza: Ogni Giorno

Descrizione: Elimina le Visualizzazioni scadute

10.5.13 GestioneConnessione

Frequenza: Ogni Giorno

Descrizione: Elimina le Connessioni scadute

10.5.14 GestioneErogazione

Frequenza: Ogni Ora

Descrizione: Elimina le Erogazioni terminate

10.6 API

Il database così strutturato oltre a fornire l'accesso ai raw data tramite interrogazioni formulate in SQL espone le seguenti API sotto forma di stored procedure o stored function:

10.6.1 AggiungiErogazioneServer

Descrizione: Incrementa il carico attuale di un server di un'unità.

Input: ID del Server

Output:

10.6.2 BilanciamentoDelCarico

Descrizione: Prevede le Edizioni che potrebbero essere richieste di frequente da un Paese, fornendo una lista di File e Server contenente gli spostamenti consigliati.

Input: N, M numeri naturali

Output: Lista di Spostamenti

10.6.3 CachingPrevisionale

Descrizione: Prevede i File che potrebbero essere richiesti di frequente da un Utente, fornendo una lista di File e Server contenente gli spostamenti consigliati.

Input: X, N, M numeri naturali

Output: Lista di Spostamenti

10.6.4 CalcolaDistanzaPaese

Descrizione: Calcola le distanze tra un **Paese** e tutti i **Server**. Gli output vengono calcolati tramite ST_DISTANCE_SPHERE e divisi per 1000 e memorizzati nella tabella DistanzaPrecalcolata. La distanza dal **Paese** fittizio 'World' è 0 verso ogni **Server**.

Input: *Codice* di **Paese**

Output:

10.6.5 CalcolaDistanzaServer

Descrizione: Calcola le distanze tra un **Server** e tutti i **Paesi**. Gli output vengono calcolati tramite ST_DISTANCE_SPHERE e divisi per 1000 e memorizzati nella tabella **DistanzaPrecalcolata**. Ogni Server ha distanza 0 verso il fittizio 'World'.

Input: *ID* di **Server**

Output:

10.6.6 CambioAbbonamento

Descrizione: Cambia l'abbonamento di un utente dopo aver controllato che sia in pari coi pagamenti

Input: Codice di Utente, Tipo di Abbonamento

Output:

10.6.7 FileMiglioreQualita

Descrizione: Restituisce i File, di uno specifico film, con le risoluzioni più alte che un utente può vedere.

Input: Codice di Utente, ID di Film

Output: ID di Film, Risoluzione

10.6.8 FilmDisponibiliInLinguaSpecifica

Descrizione: Calcola tutti i Film disponibili in una Lingua.

Input: Lingua

Output: Lista di Film

10.6.9 FilmEsclusiAbbonamento

Descrizione: Calcola il numero di Film esclusi da un abbonamento.

Input: Abbonamento

Output: Numero di Film esclusi

10.6.10 FilmPiuVistiRecentemente

Descrizione: Calcola i Film più visti di recente.

Input: Numero Massimo di Film

Output: Lista di Film

10.6.11 GeneriDiUnFilm

Descrizione: Calcola i Generi a cui appartiene un Film e determina se l'utente è abilitato o meno alla visione di quel film.

Input: Film, Abbonamento

Output: Lista di Generi, Abilitazione

10.6.12 RimuoviErogazioneServer

Descrizione: Decrementa il carico attuale di un server di un'unità.

Input: ID del Server

Output:

10.6.13 VinciteDiUnFilm

Descrizione: Calcola tutti i premi vinti da un Film.

Input: Film

Output: Lista di Premi, Numero di Premi

10.6.14 Ip2Paese

Descrizione: Chiama *Ip2PaeseStorico* passandogli come parametri l'*IP* ricevuto e *CURRENT_TIMESTAMP*

Input: L'*IP* di cui trovare il paese di appartenenza

Output: Il *Codice* del **Paese** o '??' in caso di nessun riscontro

10.6.15 Ip2PaeseStorico

Descrizione: Dato l'*IP* da cercare e l'istante temporale nel quale lo si desidera cercare cerca tra gli **IPRange** uno che soddisfi le condizioni.

Input: L'*IP* di cui trovare il paese di appartenenza

Output: Il *Codice* del **Paese** o '??' in caso di nessun riscontro

10.6.16 IpAppartieneRangeInData

Descrizione: Dati tutti gli estremi (sia numerici che temporali) di un *IPRange*, assieme ad un *IP* ed un istante temporale, controlla che l'*IP* appartenga al range dato nell'istante dato

Input: Inizio, Fine, DataInizio, DataFine, IP, DataDaControllare

Output: TRUE se l'*IP* appartiene a [Inizio, Fine] e il range è valido nell'istante richiesto, FALSE altrimenti

10.6.17 IpRangeCollidono

Descrizione: Controlla se esiste almeno un indirizzo che appartiene ad entrambi i range

Input: *Inizio1*, *Fine1*, *Inizio2*, *Fine2*. Per entrambi i range si assume *Fine* \leq *Inizio* per ipotesi

Output: TRUE in caso di collisione, FALSE altrimenti

10.6.18 IpRangeValidoInData

Descrizione: Controlla se un determinato range di ip è valido in una determinata data. Passare NULL al secondo o terzo parametro viene interpretato come *CURRENT_TIMESTAMP*

Input: InizioValidità, FineValidità, IstanteDaControllare

Output: TRUE se $\text{FineValidità} \geq \text{IstanteDaControllare} \geq \text{InizioValidità}$, FALSE altrimenti

10.6.19 IpRangePossoInserire

Descrizione: Controlla se un range è già preesistente nel DB

Input: *NewInizio, NewFine, NewDataInizio, NewPaese*

Output: TRUE se nell'istante NewDataInizio non vi è nessun Range con estremi e Paese uguali a quelli inseriti

10.6.20 IpRangeInserisciFidato

Descrizione: Inserisce un **IPRange** nel DB. Si può specificare se invalidare gli altri ip che collidono con il nuovo inserito o se non effettuare proprio tale controllo.

Input: *NewInizio, NewFine, NewDataInizio, NewDataFine, NewPaese, InvalidaCollisioni*

Output:

10.6.21 IpRangeInserisciAdessoFidato

Descrizione: Inserisce un **IPRange** nel DB. Si può specificare se invalidare gli altri ip che collidono con il nuovo inserito o se non effettuare proprio tale controllo. *DataInizio* è considerata CURRENT_TIMESTAMP e *DataFine* NULL

Input: *NewInizio, NewFine, NewPaese, InvalidaCollisioni*

Output:

10.6.22 IpRangeProvaInserire

Descrizione: Inserisce un **IPRange** nel DB solo se non è già presente uno uguale o equivalente.

Input: *NewInizio, NewFine, NewDataInizio, NewDataFine, NewPaese*

Output:

10.6.23 IpRangeProvaInserireAdesso

Descrizione: Inserisce un **IPRange** nel DB solo se non è già presente uno uguale o equivalente. *DataInizio* è considerata CURRENT_TIMESTAMP e *DataFine* NULL

Input: *NewInizio, NewFine, NewPaese*

Output:

10.6.24 VisualizzazioniGiornaliereBuild

Descrizione: Aggiunge i dati relativi alla giornata di ieri nell MV **VisualizzazioniGiornaliere**. Fa ciò solo se tali dati non sono già presenti. Viene chiamata con cadenza giornaliera dall'event **VisualizzazioniGiornaliereEvent**

Input:

Output: Solleva lo warning 'Procedura già lanciata oggi!' se i dati sono già presenti

10.6.25 VisualizzazioniGiornaliereFullBuild

Descrizione: Ricostruisce in toto la MV **VisualizzazioniGiornaliere**. Sovrascrive eventuali dati già presenti. Tale operazione può impiegare molto tempo: la procedura deve essere utilizzata solo dopo un'eventuale perdita dei dati della ridondanza.

Input:

Output:

10.6.26 MathMap

Descrizione: Porta un numero da un intervallo in un altro.

Input: X, inMin, inMax, outMin, outMax

Output: $\text{outMin} + (\text{outMax} - \text{outMin}) * (x - \text{inMin}) / (\text{inMax} - \text{inMin})$

10.6.27 StrListContains

Descrizione: Separa la stringa in base alla virgola e controlla se almeno una delle sottostringhe ottenute contiene il termine richiesto. Per le sottostringhe gli spazi iniziali e finali vengono ignorati nella comparazione

Input: *Pagliaio*, *Ago*

Output: TRUE se *Ago* è uno dei termini di *Pagliaio*

10.6.28 CalcolaDelta

Descrizione: Dato un valore massimo *Max* e un numero *Valore* restituisce *Max - Valore* se *Max* \leq *Valore* o $2(\text{Valore} - \text{Max})$ negli altri casi. NULL a *Max* viene considerato 0.

Input: *Max*, *Valore*

Output: Il Delta calcolato con il criterio sopra indicato

10.6.29 MigliorServer

Descrizione: Cerca di trovare la miglior coppia *FileID* - *ServerID* per un Utente che vorrebbe iniziare una **Visualizzazione** di un'**Edizione**. Viene controllato l'abbonamento dell'**Utente**, dalla quale si individua la massima qualità permessa. I **File** restituiti hanno codifiche Audio e Video supportate dal dispositivo del client. Il **Server** è scelto in base alla prossimità e al suo carico attuale. La posizione dell'**Utente** è estratta dal suo indirizzo IP4. Nella richiesta, *MaxBitRate* e *MaxRisoluz* devono tenere conto dell'attuale stato della connessione tra il client e il server: sarebbe inutile trasmettere al client più dati di quanti ne può ricevere. *ListaVideoEncodings* e *ListaAudioEncodings* contengono le Famiglie dei FormatiCodifica supportati dal device client (che li concatena in automatico); NULL significa che ogni codifica è supportata.

Input: id_utente, id_edizione, ip_connessione, MaxBitRate, MaxRisoluz, ListaVideoEncodings, ListaAudioEncodings

Output: FileID, ServerID

10.6.30 TrovaMigliorServer

Descrizione: Cerca di trovare il miglior trio *FileID* - *ServerID* - *Punteggio* per un'**Edizione**. I **File** restituiti hanno codifiche Audio e Video supportate dal dispositivo del client. Il **Server** è scelto in base alla prossimità e al suo carico attuale. La posizione dell'**Utente** è estratta dal suo indirizzo IP4. Nella richiesta, *MaxBitRate* e *MaxRisoluz* devono tenere conto dell'attuale stato della connessione tra il client e il server: sarebbe inutile trasmettere al client più dati di quanti ne può ricevere. *ListaVideoEncodings* e *ListaAudioEncodings* contengono le Famiglie dei FormatiCodifica supportati dal device client (che li concatena in automatico); NULL significa che ogni codifica è supportata.

Input: id_edizione, paese_utente, MaxRisoluzAbbonamento, MaxBitRate, MaxRisoluz, ListaVideoEncodings, ListaAudioEncodings

Output: FileID, ServerID, Punteggio

10.6.31 RibilanciamentoCarico

Descrizione: Individua i Server più stressati (sopra la media di carico) e, per ogni loro erogazione cerca di trovare alternative nei Server meno cari-

chi. I risultati vengono messi nella tabella **ModificaErogazioni**, in attesa che le erogazioni vengano modificate anche nella realtà (i server stessi se ne occuperanno d'ora in avanti). Vengono considerate solo le Erogazioni non spostate nell'ultima mezz'ora e che non termineranno a breve.

Input:

Output: Riferimento a Erogazione, Server (server in utilizzo), Alternativa (altro server), File (proposta di alternativa)

10.6.32 RatingUtente

Descrizione: Fornisce un Rating, in stelle, ad un film in maniera oggettiva

Input: ID di Film

Output: Valutazione

11 Codice sorgente e Bibliografia

Le repository utilizzate dagli autori di questo progetto sono:

- Ciucci Riccardo: <https://github.com/Richie314/FilmSphere>
- Gallo Simone: <https://github.com/Monesi-dev/BasiDiDati>

Le repository contengono anche il draw.io utilizzato per il diagramma E-R.

Riferimenti bibliografici

- [1] danielmiessler. *SecLists*. URL: <https://github.com/danielmiessler/SecLists>.
- [2] filippotoso. *Nomi e cognomi italiani*. URL: <https://github.com/filippotoso/nomi-cognomi-italiani/>.
- [3] sapics. *Ip ranges impiegati*. URL: <https://github.com/sapics/ip-location-db>.
- [4] Wikipedia. *Lista Codecs*. URL: https://en.wikipedia.org/wiki/List_of_codecs.