

L'algebra relazionale in RA

Gianluca Cima, Maurizio Lenzerini, Laura Papi



SAPIENZA
UNIVERSITÀ DI ROMA

Anno accademico 2024/2025

RA: Un interprete per l'Algebra Relazionale

Coloro che vogliono scaricare l'interprete dell'algebra relazionale RA possono accedere al seguente link, dove si trova anche la documentazione del software:

<https://users.cs.duke.edu/~junyang/ra2/>

In particolare, tramite il link riportato sopra è possibile fare il download del file `ra.jar` (che è proprio il programma corrispondente all'interprete per l'algebra relazionale) usato in laboratorio e del file *sample.properties*, che viene utilizzato dal software.

Come utilizzare RA

1. Per utilizzare RA bisogna preparare in una directory i seguenti file:
 - *ra.jar*,
 - *sample.properties*,
 - un file con estensione *.ra*, che contiene uno script che va a creare una base di dati, con lo schema delle relazioni e le loro istanze (nel sito di cui sopra si può scaricare un file con estensione *.ra* di esempio).

In particolare, il file *sample.properties* serve a specificare il nome del file che conterrà tutte le informazioni sulla base di dati che vogliamo usare. Ad esempio, per utilizzare una base di dati di nome “alfa”, il file *sample.properties* deve contenere la linea

```
url=jdbc:sqlite:alfa.db
```

perché questo è il meccanismo per specificare il nome della base di dati che verrà creata e che vogliamo usare.

Come utilizzare RA

2. Dirigiamoci tramite la shell del sistema operativo nella cartella dove sono presenti i tre file menzionati sopra, e diamo il seguente comando:

```
java -jar ra.jar sample.properties -i nome-file.ra
```

Questo comando legge il file *nome-file.ra* e crea il file *nome.db*, dove “*nome*” è il nome che abbiamo specificato nel file *sample.properties*.

3. A questo punto il file *nome.db* contiene tutte le informazioni sulla base di dati e la seguente istruzione permette di iniziare ad utilizzare l'interprete RA usando la base di dati appena creata:

```
java -jar ra.jar sample.properties
```

Per questa esercitazione

Per l'esercitazione odierna abbiamo preparato lo script `birre.ra` e settato il file `sample.properties`. In particolare, per svolgere l'esercitazione è sufficiente creare una nuova cartella sul Desktop e posizionare al suo interno i files “`ra.jar`”, “`birre.ra`”, “`sample.properties`” e “Queries da svolgere.pdf” prelevabili da “Esercitazione 2: esercizi di algebra relazionale in laboratorio” della sezione MOODLE del corso (dove si trova anche una copia di queste slides ed il file `basedidati.txt` con la descrizione della base di dati da usare).

Dirigendoci tramite la shell nella cartella dove sono presenti questi files, possiamo creare il file “`sample.db`” tramite il comando:

```
java -jar ra.jar sample.properties -i birra.ra
```

D'ora in poi si può iniziare ad utilizzare l'interprete RA con la base di dati appena creata tramite il comando: `java -jar ra.jar sample.properties`

L'esercitazione prevede di scrivere le query descritte in “Queries da svolgere.pdf”.

Alcuni comandi utili

D'ora in poi assumeremo di essere all'interno dell'interprete RA (**stare attenti al punto e virgola alla fine dei comandi!**).

- Il seguente comando lista tutte le relazioni presenti nel DB:

`\list;`

- Per avere la lista delle tuple della relazione di nome *relName* occorre lanciare il comando:

`relName;`

- Per uscire dal sistema occorre lanciare il comando:

`\quit;`

Nel seguito, illustriamo i comandi necessari per usare in RA gli operatori dell'algebra relazionale.

Unione

Assumiamo di avere le seguenti due relazioni nella nostra base di dati (il dominio *text* equivale a *string*):

- Laureato(matricola: *text*, nome:*text*)
 $\{\langle 7274, Rossi \rangle, \langle 7432, Neri \rangle\}$
- Quadro(matricola: *text*, nome *text*)
 $\{\langle 7432, Neri \rangle, \langle 9297, Neri \rangle\}$

Per eseguire l'unione delle due relazioni:

- Algebra relazionale sulle slide: Laureato \cup Quadro
- In RA: **Laureato \cup Quadro;**
- Risultato: schema (matricola: *text*, nome: *text*)
 $\{\langle 7274, Rossi \rangle, \langle 7432, Neri \rangle, \langle 9297, Neri \rangle\}$

Intersezione

Assumiamo di avere le seguenti due relazioni:

- Laureato(matricola: *text*, nome: *text*)
 $\{\langle 7274, Rossi \rangle, \langle 7432, Neri \rangle\}$
- Quadro(matricola: *text*, nome: *text*)
 $\{\langle 7432, Neri \rangle, \langle 9297, Neri \rangle\}$

Per eseguire l'intersezione delle due relazioni:

- Algebra relazionale sulle slide: Laureato \cap Quadro
- In RA: Laureato \backslash intersect Quadro;
- Risultato: schema (matricola: *text*, nome: *text*)
 $\{\langle 7432, Neri \rangle\}$

Differenza

Assumiamo di avere le seguenti due relazioni:

- Laureato(matricola: *text*, nome: *text*)
 $\{\langle 7274, Rossi \rangle, \langle 7432, Neri \rangle\}$
- Quadro(matricola: *text*, nome: *text*)
 $\{\langle 7432, Neri \rangle, \langle 9297, Neri \rangle\}$

Per eseguire la differenza tra le due relazioni:

- Algebra relazionale sulle slide: Laureato – Quadro
- In RA: **Laureato \diff Quadro;**
- Risultato: schema (matricola: *text*, nome: *text*)
 $\{\langle 7274, Rossi \rangle\}$

Ridenominazione

In RA la ridenominazione va fatta inserendo la lista completa di nuovi nomi separati da virgola, uno per ciascun attributo dell'operando, nell'ordine in cui compaiono nell'operando stesso.

Assumiamo di avere la seguente relazione:

- Paternità(padre: *text*, figlio : *text*)
 $\{\langle \textit{Adamo}, \textit{Abele} \rangle, \langle \textit{Adamo}, \textit{Caino} \rangle, \langle \textit{Abramo}, \textit{Isacco} \rangle\}$

Per eseguire la ridenominazione su una relazione:

- Algebra relazionale sulle slide: $\text{REN}_{\text{genitore} \leftarrow \text{padre}}(\text{Paternità})$
- In RA: $\backslash \text{rename}_{\{\text{genitore}, \text{figlio}\}}(\text{Paternità});$
- Risultato: schema (genitore: *text*, figlio: *text*)
 $\{\langle \textit{Adamo}, \textit{Abele} \rangle, \langle \textit{Adamo}, \textit{Caino} \rangle, \langle \textit{Abramo}, \textit{Isacco} \rangle\}$

Esempio di utilizzo della ridenominazione

Assumiamo di avere le seguenti due relazioni:

- Paternità(padre: *text*, figlio: *text*)
 $\{\langle \text{Adamo}, \text{Abele} \rangle, \langle \text{Adamo}, \text{Caino} \rangle, \langle \text{Abramo}, \text{Isacco} \rangle\}$
- Maternità(madre: *text*, figlio: *text*)
 $\{\langle \text{Eva}, \text{Abele} \rangle, \langle \text{Eva}, \text{Set} \rangle, \langle \text{Sara}, \text{Isacco} \rangle\}$

Supponiamo di volerne l'unione:

- Algebra relazionale sulle slide: $\text{REN}_{\text{genitore} \leftarrow \text{padre}}(\text{Paternità}) \cup \text{REN}_{\text{genitore} \leftarrow \text{madre}}(\text{Maternità})$
- In RA: $\text{\textbackslash rename_}\{\text{genitore}, \text{figlio}\}(\text{Paternità}) \text{\textbackslash union} \text{\textbackslash rename_}\{\text{genitore}, \text{figlio}\}(\text{Maternità});$
- Risultato: schema (genitore: *text*, figlio: *text*)
 $\{\langle \text{Adamo}, \text{Abele} \rangle, \langle \text{Adamo}, \text{Caino} \rangle, \langle \text{Abramo}, \text{Isacco} \rangle, \langle \text{Eva}, \text{Abele} \rangle, \langle \text{Eva}, \text{Set} \rangle, \langle \text{Sara}, \text{Isacco} \rangle\}$

Selezione

Assumiamo di avere la seguente relazione:

- Impiegato(matricola: *text*, cognome: *text*, filiale *text*, stipendio *integer*)
 $\{\langle 7309, Rossi, Roma, 55 \rangle, \langle 5998, Neri, Milano, 64 \rangle, \langle 9553, Milano, Milano, 44 \rangle, \langle 5698, Neri, Napoli, 64 \rangle\}$

Per eseguire la selezione sugli impiegati che guadagnano più di 50 e che lavorano a Milano o a Roma:

- Algebra relazionale sulle slide:
 $SEL_{\text{stipendio} > 50 \text{ AND } (\text{filiale} = 'Milano' \text{ OR } \text{filiale} = 'Roma')}(Impiegato)$
- In RA: $\backslash \text{select_}\{\text{stipendio} > 50 \text{ AND } (\text{filiale} = 'Milano' \text{ OR } \text{filiale} = 'Roma')\}(Impiegato);$
- Risultato: schema (matricola: *text*, cognome: *text*, filiale *text*, stipendio *integer*)
 $\{\langle 7309, Rossi, Roma, 55 \rangle, \langle 5998, Neri, Milano, 64 \rangle\}$

Proiezione

Assumiamo di avere la seguente relazione:

- $\text{Impiegato}(\text{matricola: text, cognome: text, filiale: text, stipendio: integer})$
 $\{\langle 7309, \text{Neri}, \text{Napoli}, 55 \rangle, \langle 5998, \text{Neri}, \text{Milano}, 64 \rangle,$
 $\langle 9553, \text{Rossi}, \text{Roma}, 44 \rangle, \langle 5698, \text{Rossi}, \text{Roma}, 64 \rangle\}$

Per eseguire la proiezione sul cognome e sulla filiale di tutti gli impiegati:

- Algebra relazionale sulle slide: $\text{PROJ}_{\text{Cognome, Filiale}}(\text{Impiegato})$
- In RA: $\backslash \text{project_}\{\text{cognome, filiale}\}(\text{Impiegato})$;
- Risultato: schema (cognome: text, filiale: text)
 $\{\langle \text{Neri}, \text{Napoli} \rangle, \langle \text{Neri}, \text{Milano} \rangle, \langle \text{Rossi}, \text{Roma} \rangle\}$

Join naturale

Assumiamo di avere le seguenti due relazioni:

- Docente(codice: *text*, corso: *text*)
 $\{\langle 1, BD \rangle, \langle 2, PS \rangle, \langle 3, Reti \rangle, \langle 1, PS \rangle\}$
- Silnsegna(corso: *text*, ling: *text*)
 $\{\langle BD, SQL \rangle, \langle BD, JAVA \rangle, \langle PS, UML \rangle, \langle KRST, DATALOG \rangle\}$

Per eseguire il join naturale delle due relazioni:

- Algebra relazionale sulle slide: Docente JOIN Silnsegna
- In RA: Docente \bowtie Silnsegna;
- Risultato: schema (codice: *text*, corso: *text*, ling: *text*)
 $\{\langle 1, BD, SQL \rangle, \langle 1, BD, JAVA \rangle, \langle 2, PS, UML \rangle, \langle 1, PS, UML \rangle\}$

Nota: ricordiamo che se le due relazioni non hanno attributi in comune, il join naturale calcola il prodotto cartesiano.

Prodotto cartesiano

Assumiamo di avere le seguenti due relazioni:

- Impiegato(imiegato: *text*, reparto: *text*)
 $\{\langle Rossi, A \rangle, \langle Neri, B \rangle, \langle Bianchi, B \rangle\}$
- Reparto(codice: *text*, capo: *text*)
 $\{\langle A, Mori \rangle, \langle B, Bruni \rangle\}$

Il prodotto cartesiano delle due relazioni si può anche ottenere così:

- Algebra relazionale sulle slide: Impiegato JOIN Reparto
- In RA: Codice \cross Reparto (in questo caso, equivalentemente, con Codice \join Reparto);
- Risultato: schema (impiegato: *text*, reparto: *text*, codice: *text*, capo: *text*)
 $\{\langle Rossi, A, A, Mori \rangle, \langle Rossi, A, B, Bruni \rangle, \langle Neri, B, A, Mori \rangle, \langle Neri, B, B, Bruni \rangle, \langle Bianchi, B, A, Mori \rangle, \langle Bianchi, B, B, Bruni \rangle\}$

Theta-join

Assumiamo di avere le seguenti due relazioni:

- Impiegato(imiegato: *text*, reparto: *text*)
 $\{\langle Rossi, A \rangle, \langle Neri, B \rangle, \langle Bianchi, B \rangle\}$
- Reparto(codice: *text*, capo: *text*)
 $\{\langle A, Mori \rangle, \langle B, Bruni \rangle\}$

Per eseguire il theta-join (in particolare l'equi-join) tra le due relazioni sulla condizione reparto=codice:

- Algebra relazionale sulle slide: Impiegato JOIN_{reparto=codice} Reparto
- In RA: Impiegato \join_{reparto=codice} Reparto;
- Risultato: schema (imiegato: *text*, reparto: *text*, codice: *text*, capo: *text*)
 $\{\langle Rossi, A, A, Mori \rangle, \langle Neri, B, B, Bruni \rangle, \langle Bianchi, B, B, Bruni \rangle\}$