CS2DB3 A4

Virendra Jethra

March 2023

1

$\pi_{comment.cid,fmention.fid}(\sigma_{comment.rid=fmention.rid}(comment \times fmention))$

2

$\pi_{fm1.fid}(\sigma_{fm1.rid \neq fm2.rid \wedge fm1.fid=fm2.fid}(\sigma(\rho_{fm1}(fmention)) \times \sigma(\rho_{fm2}(fmention))))$

3

$X : \pi_{fc1.fid}(\sigma_{fc1.fid=fc2.fid \wedge fc1.category \neq fc2.category}(\rho_{fc1}(fcategory) \times \rho_{fc2}(fcategory)))$
$Y : \pi_{fc1.fid}(\sigma_{fc1.fid=fc2.fid=fc3.fid \wedge fc1.category \neq fc2.category \neq fc3.category}(\rho_{fc1}(fcategory) \times \rho_{fc2}(fcategory) \times \rho_{fc3}(fcategory)))$
$Z : Y/X$

4

$X : \rho_{a1}(fmention) \bowtie_{a1.fid=a2.fid} \rho_{a2}(fcategory)$
$Y : \pi_{category}(fcategory)/\pi_{category}(X)$

5

$W : \pi_{rid,uid,rtime}(review) \bowtie_{review.rid=fmention.rid} \pi_{rid,fid}(fmention)$
$X : \pi_{a1.rid,a1.uid,a1.rtime}(\sigma_{a1.fid=a2.fid \wedge a1.uid=a2.uid \wedge a1.rid \neq a2.rid}(\rho_{a1}(W) \times \rho_{a2}(W)))$
$Y : \pi(\sigma_{a1.fid=a2.fid \wedge a1.uid=a2.uid \wedge a1.rtime<a2.rtime}(\rho_{a1}(X) \times \rho_{a2}(X)))$
$Z : X/Y$

6

$X : \pi_{uid}(review)$
$Y : \pi_{fm.fid,r.uid}(\sigma_{fm.rid=r.rid}(\rho_{fm}(fmention) \times \rho_{r}(review)))$
$Z : \pi_{M.fid}(Y/X(\rho(R.uid \mapsto uid)))$

7

To estimate the size of the output of all intermediate steps in the query execution plan, we need to make some assumptions and use the provided estimates:

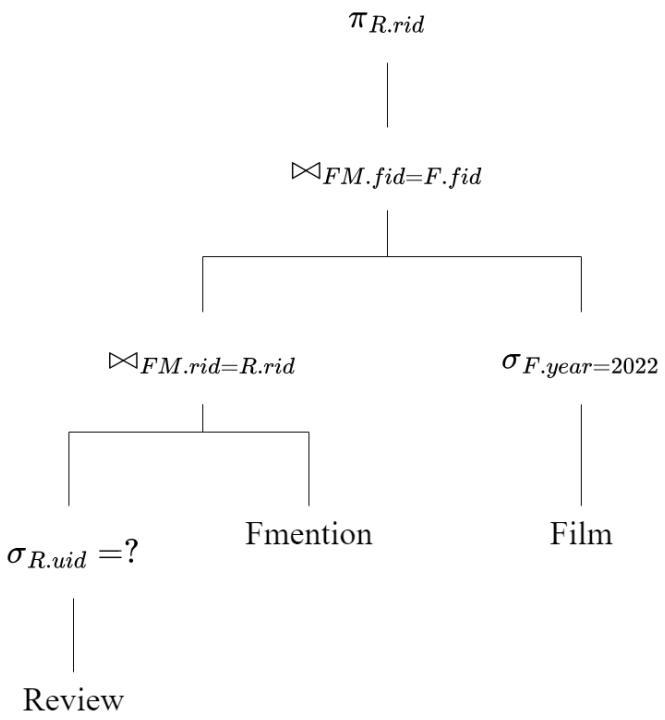
- Number of reviewers (R): 50
- Number of users (U): 10,000
- Average number of reviews per reviewers per year: 20
- Maximum number of comments per review: 1000
- Number of films released per year: 700
- Average number of categories per film: 3
- Average number of films mentioned per review: 2
- Number of years the website has been operating: 10

Based on these assumptions, we can estimate the size of the intermediate results as follows:

- ρ_R (review): There are 50 reviewers who write 20 reviews per year, so there are $50 * 20 * 10 = 10,000$ reviews in total.
- ρ_M (fmention): Each review mentions 2 films on average, so there are $2 * 10,000 = 20,000$ mentions in total.
- ρ_F (film): There are 700 films released per year, so in 10 years there are $7,000$ films in total.
- $\sigma_{R.rid=M.rid \wedge M.fid=F.fid}$: Each review mentions 2 films, so it will be the same as the fmention which is $20,000$ rows
- $\sigma_{R.uid}$: This is a selection on the user ID attribute of the review table. Assuming each user ID is unique, the output of this step has at most 20 rows per reviewer times by 10 years, we get 200 rows. Each review has 2 mentions per review so in total we have is 400 rows
- $\sigma_{F.year=2022}$: Now we take the total reviews and divide it by the number of years to get the average reviews for 2022. So we get 40 rows

Therefore, the size of the output will be around 40 rows.

8



The above plan aims to minimize the number of rows that need to be processed by pushing selection commands down to the lower level of the plan. The plan uses a natural join instead of a cross join to produce cleaner tables with fewer rows.

Additionally, the plan performs the join between the review and mention tables first, and then with the film table, so that the review only connects with the film through mentions. This approach avoids the redundant rows that would result from a Cartesian product of all three tables. Overall, this plan is designed to optimize the efficiency of the query.

9

To estimate the size of the output of the plan above. Let's estimate the output for each intermediate steps:

- The review, fmention, and films will have $10,000$, $20,000$, $7,000$ rows respectively as explained in question 7.
- $\sigma_{R.uid}$: Each user has around 20 reviews per year and the website has been running for 10 years, so we get $20 * 10 = 200$ rows.

- $\bowtie_{FM.rid=R.rid}$: Since the average films mentioned per review is 2 and there are 200 reviews, hence we get $200 * 2 = 400$
- $\sigma_{F.year=2022}$: Since the number of films released per year is 700. Therefore we have 700 rows.
- $\bowtie_{FM.fid=F.fid}$: So now we combine all the films in 2022 that are made by the user ? which is 20 as the average number reviews per reviewer is 20. Also since the average number of films that can be mentioned on a review is 2. Therefore we multiply both to get the result, $20 * 2 = 40$

Therefore, the size of the output will be around 40 rows, which is still the same as the question 7 but this time it is more efficient.

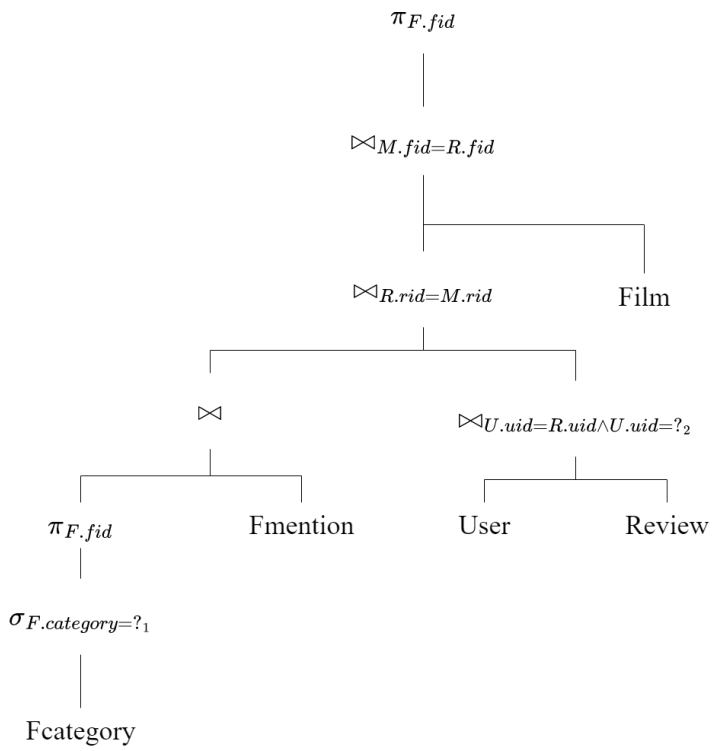
10

```
SELECT r.rid
FROM (review r CROSS JOIN fmention fm) CROSS JOIN film f
WHERE fm.rid = r.rid AND fm.fid = f.fid AND r.uid = ? AND f.year = 2022;
```

11

$X : \pi_{F.fid}(\sigma_{F.category=?_1}(\rho_F(fcategory)))$
$Y : \sigma_{U.uid=?_2 \wedge R.rid=M.rid \wedge U.uid=R.uid \wedge M.fid=F.fid \wedge M.fid \in X}(\rho_U(user) \times \rho_F(film) \times \rho_R(review) \times \rho_M(mention))$
$Z : \pi_{F.fid}(Y)$

12



The above plan aims to minimize the number of rows that need to be processed by pushing selection commands down to the lower level of the plan. The plan uses a natural join instead of a cross join to join 4 tables, which produce cleaner tables with fewer rows.

Additionally, by applying natural join at each level rather than just doing cross join for all the tables, makes it more efficient. In the plan, the USER table and the REVIEW table combined, allows to connect users with their reviews, and then checking the Fmention table directly to ensure that the film being mentioned is part of that category. Lastly, the FILM table can be also joined so that reviews are linked to the films they review. By taking these steps, the query can be executed more efficiently and with better performance.

13

To estimate the size of the output of the plan above. Let's estimate the output for each intermediate steps:

- The review and fmention will have $10,000$, $20,000$ rows respectively as explained in question 7. Also we already know that there are $10,000$

users.

- The fcategory will have $21,000$ rows as there are 700 films, each film will belong to 3 categories and the website has been running for 10 years.
- $\sigma_{F.category=?_1}$: If we make an assumption that each film will be in the specified category then we will have at most $7,000$ rows as it will be all the films
- $\bowtie$: If we still take into account that each film will be in the specified category then again it will be at most $7,000$ rows
- $\bowtie_{U.uid=R.uid \wedge U.uid=?_2}$: A single user has 20 reviews per year and it has been 10 years, so $20 * 10 = 200$ rows
- $\bowtie_{R.rid=M.rid}$: So we have 200 reviews and there can only be 2 mention of the film per review, so $200 * 2 = 400$ rows
- $\bowtie_{M.fid=R.fid}$: The amount of rows remains the same, all it does is add another column

Therefore, the size of the output will be around 400 rows at most.