CS 411 PT1 Stage 3: Database Creation and Indexing

DDL queries for all the tables:

1. Table Papers:

```
CREATE TABLE Papers (
    paper_id VARCHAR(50) PRIMARY KEY,
    paper_title VARCHAR(255) NOT NULL,
    abstract MEDIUMTEXT,
    pdf_url VARCHAR(500),
    upload_timestamp DATETIME,
    status VARCHAR(50),
    venue_id VARCHAR(10),
    project_id VARCHAR(50),
    dataset_id VARCHAR(50),q
    FOREIGN KEY (venue_id) REFERENCES Venues(venue_id),
    FOREIGN KEY (project_id) REFERENCES Projects(project_id),
    FOREIGN KEY (dataset_id) REFERENCES Datasets(dataset_id)
);
```

Note: We can also add On Delete Cascade, however we haven't done it for this stage, since some of the data was manually added.

2. Table Projects:

```
CREATE TABLE Projects (
    project_id VARCHAR(50) PRIMARY KEY,
```

```
    project_title VARCHAR(255) NOT NULL,
    description TEXT,
    project_date DATE
);
```

3. Table Venues

```
CREATE TABLE Venues (
    venue_id VARCHAR(10) PRIMARY KEY,
    venue_name VARCHAR(255) NOT NULL,
    venue_type VARCHAR(50),
    publisher VARCHAR(100),
    year SMALLINT
);
```

4. Table Datasets:

```
CREATE TABLE Datasets (
    dataset_id VARCHAR(50) PRIMARY KEY,
    dataset_name VARCHAR(255) NOT NULL,
    dataset_url VARCHAR(500),
    domain VARCHAR(100),
    access_type VARCHAR(50)
);
```

5. Table Users:

```
CREATE TABLE Users (
  user_id      VARCHAR(50)  NOT NULL,
```

```
  user_name    VARCHAR(255) NOT NULL,
  email        VARCHAR(320) UNIQUE NOT NULL,
  affiliation  VARCHAR(255),
  profile_url  VARCHAR(500),
  is_reviewer TINYINT(1) NOT NULL DEFAULT 0,
  PRIMARY KEY (user_id),
) ENGINE=InnoDB;
```

6. Table Reviews:

```
CREATE TABLE Reviews (
  review_id        VARCHAR(50)  NOT NULL,
  user_id          VARCHAR(50)  NOT NULL,
  paper_id         VARCHAR(50)  NOT NULL,
  comment          TEXT,
  review_timestamp  DATETIME,
  PRIMARY KEY (review_id),
  CONSTRAINT fk_reviews_user   FOREIGN KEY (user_id)  REFERENCES Users(user_id),
  CONSTRAINT fk_reviews_paper  FOREIGN KEY (paper_id) REFERENCES Papers(paper_id),
) ENGINE=InnoDB;
```

7. Table Authorship:

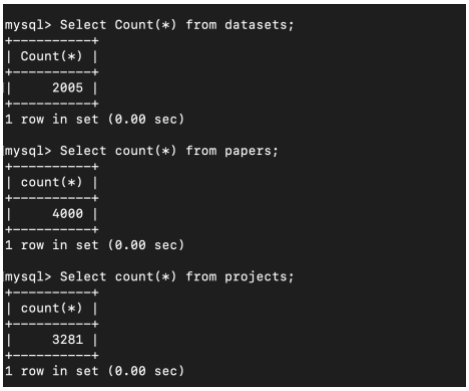
```
CREATE TABLE Authorship (
  user_id    VARCHAR(50) NOT NULL,
  paper_id   VARCHAR(50) NOT NULL,
  is_primary TINYINT(1)  NOT NULL DEFAULT 0,
  PRIMARY KEY (user_id, paper_id),
```

```
  CONSTRAINT fk_authorship_user  FOREIGN KEY (user_id)  REFERENCES Users(user_id),
  CONSTRAINT fk_authorship_paper FOREIGN KEY (paper_id) REFERENCES Papers(paper_id)
) ENGINE=InnoDB;
```

8. Table RelatedPapers:

```
CREATE TABLE RelatedPapers (
  paper_id        VARCHAR(50) NOT NULL,
  related_paper_id VARCHAR(50) NOT NULL,
  PRIMARY KEY (paper_id, related_paper_id),
  CONSTRAINT chk_not_self CHECK (paper_id <> related_paper_id),
  CONSTRAINT fk_relpapers_left   FOREIGN KEY (paper_id)       REFERENCES Papers(paper_id),
  CONSTRAINT fk_relpapers_right  FOREIGN KEY (related_paper_id) REFERENCES
Papers(paper_id)
) ENGINE=InnoDB;
```

Count of tables where rows>1000



```
mysql> Select Count(*) from datasets;
+----------+
| Count(*) |
+----------+
|     2005 |
+----------+
1 row in set (0.00 sec)

mysql> Select count(*) from papers;
+----------+
| count(*) |
+----------+
|     4000 |
+----------+
1 row in set (0.00 sec)

mysql> Select count(*) from projects;
+----------+
| count(*) |
+----------+
|     3281 |
+----------+
1 row in set (0.00 sec)
```

Count(*) for other tables:

```
mysql> Select count(*) from Venues;
+----------+
| count(*) |
+----------+
|      100 |
+----------+
1 row in set (0.00 sec)

mysql> Select count(*) from Users;
+----------+
| count(*) |
+----------+
|       40 |
+----------+
1 row in set (0.00 sec)

mysql> Select count(*) from Authorship;
+----------+
| count(*) |
+----------+
|     2564 |
+----------+
1 row in set (0.00 sec)

mysql> Select count(*) from Reviews;
+----------+
| count(*) |
+----------+
|      570 |
+----------+
1 row in set (0.00 sec)

mysql> Select count(*) from RelatedPapers;
+----------+
| count(*) |
+----------+
|       10 |
+----------+
1 row in set (0.00 sec)
```

Advanced Queries:

-- Q1 --

-- Lists all projects and papers authored by a selected user(U005) since a given date,

-- showing how many reviews each paper has received.

```
EXPLAIN SELECT
    pr.project_id,
    pr.project_title,
    p.paper_id,
    p.paper_title,
    p.upload_timestamp,
    COUNT(r.review_id) AS review_count
FROM Authorship a
```

```
JOIN Papers p
    ON a.paper_id = p.paper_id
JOIN Projects pr
    ON p.project_id = pr.project_id
LEFT JOIN Reviews r
    ON p.paper_id = r.paper_id
WHERE a.user_id = 'U005'
  AND p.upload_timestamp >= '2018-01-01'
GROUP BY pr.project_id, pr.project_title, p.paper_id, p.paper_title, p.upload_timestamp
ORDER BY p.upload_timestamp DESC, review_count DESC
LIMIT 15;
```

Output:

project_id	project_title	paper_id	paper_title	upload_timestamp	review_count
8c53e26c6973dabc	gated path planning networks	ee34b900db20f6bd	Temporal Difference Variational Auto-Encoder	2018-06-08 00:00:00	0
d3b0aac7098ffad5	low shot learning with large scale diffusion	53ddebc92aba3210	BOCK : Bayesian Optimization with Cylindrical…	2018-06-05 00:00:00	0
54f45c0fbfba9c11	selfless sequential learning	9c7a6cf8da44f6e8	A Survey of Domain Adaptation for Neural Mach…	2018-06-01 00:00:00	0
7ecd8d51d64e5aa7	wikiref wikilinks as a route to recommending ap…	14b5c13a85d1f781	Approximate Knowledge Compilation by Online…	2018-05-31 00:00:00	0
883f3f83cebef926	to understand deep learning we need to underst…	9525f4a1587ab8eb	CRRN: Multi-Scale Guided Concurrent Reflectio…	2018-05-30 00:00:00	0
515d6cb116c675b3	learning deep resnet blocks sequentially using b…	96e8d75c74d892ce	Polyglot Semantic Role Labeling	2018-05-29 00:00:00	3
00dbf4e597303af6	learning in pomdps with monte carlo tree search	30928cec4b22ae91	Sigsoftmax: Reanalysis of the Softmax Bottleneck	2018-05-28 00:00:00	3
23b1b59fd3fe031e	teaching multiple concepts to a forgetful learner	5f8380ddc9c21f5c	Lipschitz regularity of deep neural networks: an…	2018-05-28 00:00:00	3
24080539b4589947	a survey on open information extraction	f2db5fb0e9e19dd7	Reliability and Learnability of Human Bandit Fee…	2018-05-27 00:00:00	0
fb0a1d686ffc86f7	entity commonsense representation for neural a…	3723fd9a9f84a718	Towards More Efficient Stochastic Decentralize…	2018-05-25 00:00:00	0
41fd68588b0d1939	semaxis a lightweight framework to characterize…	8b228622f75db232	Heterogeneous Bitwidth Binarization in Convolu…	2018-05-25 00:00:00	0
fb0a1d686ffc86f7	entity commonsense representation for neural a…	fd51028e3fd8728c	Robust Distant Supervision Relation Extraction…	2018-05-24 00:00:00	2
134db750be9874c6	smhd a large scale resource for exploring online…	73fedf627e66a4fd	Optimizing the F-measure for Threshold-free Sa…	2018-05-19 00:00:00	0
349d93e371a67885	bringing replication and reproduction together wi…	0e089ac357993330	PG-TS: Improved Thompson Sampling for Logis…	2018-05-18 00:00:00	0
966bba78b68db3f9	generative neural machine translation	c1ca887445078048	Extrapolation in NLP	2018-05-17 00:00:00	3

```
-- Q2 --
-- Displays all venues with the count of published papers in or after 2018.
-- Helps identify recent publication activity per venue across years.
EXPLAIN SELECT
```

```
    v.venue_id,
    v.venue_name,
    v.year,
    COUNT(p.paper_id) AS total_papers
FROM Venues v
JOIN Papers p
    ON v.venue_id = p.venue_id
WHERE v.year >= 2018
  AND p.status IN ('Published')
GROUP BY v.venue_id, v.venue_name, v.year
ORDER BY v.year DESC, total_papers DESC
LIMIT 15;
```

Output:

	venue_id	venue_name	year	total_pape...
	V00027	evaluation-of-unsupervised-compositional-1	2018	51
	V0000R	gated-path-planning-networks-1	2018	51
	V0000M	Unknown Conference	2018	50
	V0000G	constraining-the-dynamics-of-deep-1	2018	49
	V0001E	unsupervised-training-for-3d-morphable-model-1	2018	49
	V0002L	on-accurate-evaluation-of-gans-for-language-1	2018	49
	V0001M	a-dataset-for-building-code-mixed-goal-2	2018	48
	V0000W	minimal-i-map-mcmc-for-scalable-structure-1	2018	48
	V0000A	snap-ml-a-hierarchical-framework-for-machine-1	2018	48
	V0000H	deforming-autoencoders-unsupervised-1	2018	48
	V0000S	multimodal-grounding-for-language-processing-1	2018	48
	V0000K	ncrf-an-open-source-neural-sequence-labeling-1	2018	47
	V00011	learning-towards-minimum-hyperspherical-1	2018	46
	V0002D	there-are-many-consistent-explanations-of-1	2018	46
	V0000Z	gile-a-generalized-input-label-embedding-for-1	2018	45

```
-- Q3 --
-- Ranks reviewers based on the number of reviews they provided within a single day.
-- Filters users marked as reviewers and counts reviews within a time range.
SELECT
    u.user_id,
    u.user_name,
    u.affiliation,
    COUNT(DISTINCT a.paper_id) AS total_papers_authored,
    COUNT(r.review_id) AS total_reviews_received
FROM Users u
JOIN Authorship a
    ON u.user_id = a.user_id
JOIN Reviews r
    ON a.paper_id = r.paper_id
WHERE u.is_reviewer = TRUE
  AND r.review_timestamp BETWEEN '2024-02-15 00:00:00' AND '2024-05-15 23:00:00'
GROUP BY u.user_id, u.user_name, u.affiliation
HAVING COUNT(r.review_id) > 0
ORDER BY total_reviews_received DESC
LIMIT 15;
```

Output: The number of records are less than 15. This if for 3 months in the year 2024, hence the limited amount of output data.

user_id	user_name	affiliation	total_papers_author...	total_reviews_receiv...
U006	Farhan Malik	University of Toronto	21	48
U004	David Patel	UC Berkeley	19	47
U005	Elena Garcia	Carnegie Mellon University	19	47
U002	Brian Chen	Stanford University	18	43
U008	Henry Nguyen	ETH Zurich	19	38
U003	Catherine Li	MIT CSAIL	15	35
U010	Jack Miller	Harvard SEAS	16	35
U007	Grace Zhou	Oxford University	13	31
U001	Alice Kim	UIUC	13	28
U009	Isha Sharma	IIT Delhi	12	25

-- Q4 --

-- Lists all papers authored by a given user and reports both:

-- (a) how many total reviews each paper has, and

-- (b) the most recent review timestamp.

-- Allows tracking which of the author's works are actively discussed or recently reviewed.

```
EXPLAIN SELECT
    p.paper_id,
    p.paper_title,
    COUNT(r.review_id) AS review_count,
    MAX(r.review_timestamp) AS last_review_at
FROM Authorship a
JOIN Papers p
    ON a.paper_id = p.paper_id
LEFT JOIN Reviews r
```

```
    ON p.paper_id = r.paper_id
WHERE a.user_id = 'U010'
GROUP BY p.paper_id, p.paper_title
ORDER BY review_count DESC, last_review_at DESC
LIMIT 15;
```

Output:

paper_id	paper_title	review_count	last_review_at
a024340399f521d0	Net2Vec: Quantifying and Explaining how Conc…	3	2024-06-29 17:07:00
507e6eeab1ddee7c	Human Pose Estimation using Global and Local…	3	2024-06-26 11:14:00
a7490d5461da5afa	Coloring with Words: Guiding Image Colorizatio…	3	2024-06-03 12:31:00
6f72b53719831b73	Cut, Paste and Learn: Surprisingly Easy Synthe…	3	2024-04-14 10:17:00
a3c38391d3f7d130	Seq2SQL: Generating Structured Queries from…	3	2024-03-23 16:11:00
7ebeb187f78dfbf6	Predict Responsibly: Improving Fairness and Ac…	2	2024-06-23 17:41:00
d4e496e148d2839e	Efficient Video Object Segmentation via Networ…	2	2024-05-29 16:43:00
9085c520c1fdabee	Importance Weighted Transfer of Samples in Re…	2	2024-05-27 10:04:00
0ecfab1ecb257327	Autoregressive Convolutional Neural Networks f…	2	2024-05-26 15:36:00
9568a08410f139a7	ISO-Standard Domain-Independent Dialogue Ac…	2	2024-05-15 12:02:00
2bee38f1ba7075cb	COCO-Stuff: Thing and Stuff Classes in Context	2	2024-05-13 17:45:00
c4528cdae0dcb06c	Gradient Estimators for Implicit Models	2	2024-05-04 10:55:00
3242cd37c75f7607	MAP inference via Block-Coordinate Frank-Wolf…	2	2024-04-08 08:01:00
a565fe77cb1693ad	Learning Structure and Strength of CNN Filters f…	2	2024-03-07 11:19:00
12289695e3f51161	Probabilistic Model-Agnostic Meta-Learning	1	2024-04-03 14:25:00

Part B. Indexing(Redone)

Q1.

```
Explain Analyze

SELECT pr.project_id, pr.project_title, p.paper_id, p.paper_title, p.upload_timestamp,

     COUNT(r.review_id) AS review_count

FROM Authorship a

JOIN Papers p      ON a.paper_id = p.paper_id

JOIN Projects pr   ON p.project_id = pr.project_id

LEFT JOIN Reviews r ON p.paper_id = r.paper_id

WHERE a.user_id = 'U005'

  AND p.upload_timestamp >= '2018-01-01'

GROUP BY pr.project_id, pr.project_title, p.paper_id, p.paper_title, p.upload_timestamp

ORDER BY p.upload_timestamp DESC, review_count DESC

LIMIT 15;   -- record cost_info.query_cost
```

Result before any indexing:

'-> Limit: 15 row(s) (actual time=14.8..14.8 rows=15 loops=1)\n   -> Sort: p.upload_timestamp DESC, review_count DESC, limit input to 15 row(s) per chunk (actual time=14.8..14.8 rows=15 loops=1)\n   -> Table scan on <temporary> (actual time=14.7..14.7 rows=37 loops=1)\n   -> Aggregate using temporary table (actual time=14.7..14.7 rows=37 loops=1)\n   -> Nested loop left join (cost=54.1 rows=51) (actual time=1.07..14.3 rows=51 loops=1)\n   -> Nested loop inner join (cost=42.7 rows=24) (actual time=1.06..14.1 rows=37 loops=1)\n   -> Nested loop inner join (cost=34.3 rows=24) (actual time=1.04..13.6 rows=37 loops=1)\n   -> Covering index lookup on a using PRIMARY (user_id = \'U005\') (cost=9.06 rows=72) (actual time=0.408..0.486 rows=72 loops=1)\n   -> Filter: ((p.upload_timestamp >= TIMESTAMP\'2018-01-01 00:00:00\') and (p.project_id is not null)) (cost=0.25 rows=0.333) (actual time=0.181..0.181

rows=0.514 loops=72)\n -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.25 rows=1) (actual time=0.181..0.181 rows=1 loops=72)\n -> Single-row index lookup on pr using PRIMARY (project_id = p.project_id) (cost=0.254 rows=1) (actual time=0.0129..0.013 rows=1 loops=37)\n -> Covering index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.272 rows=2.13) (actual time=0.00555..0.0061 rows=0.676 loops=37)\n'

Index 1: **Filter + ORDER BY on Papers**

CREATE INDEX idx_papers_upload ON Papers(upload_timestamp DESC);
DROP INDEX idx_papers_upload ON Papers; // Executed this after running the query

Result after Index 1:
'-> Limit: 15 row(s) (actual time=2.56..2.56 rows=15 loops=1)\n -> Sort: p.upload_timestamp DESC, review_count DESC, limit input to 15 row(s) per chunk (actual time=2.55..2.56 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=2.12..2.13 rows=37 loops=1)\n -> Aggregate using temporary table (actual time=2.12..2.12 rows=37 loops=1)\n -> Nested loop left join (cost=73.3 rows=100) (actual time=0.161..1.29 rows=51 loops=1)\n -> Nested loop inner join (cost=50.8 rows=47.2) (actual time=0.114..1.12 rows=37 loops=1)\n -> Nested loop inner join (cost=34.3 rows=47.2) (actual time=0.0807..0.691 rows=37 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U005\') (cost=9.06 rows=72) (actual time=0.0466..0.0993 rows=72 loops=1)\n -> Filter: ((p.upload_timestamp >= TIMESTAMP\'2018-01-01 00:00:00\') and (p.project_id is not null)) (cost=0.251 rows=0.655) (actual time=0.00796..0.00802 rows=0.514 loops=72)\n -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.00765..0.00767 rows=1 loops=72)\n -> Single-row index lookup on pr using PRIMARY (project_id = p.project_id) (cost=0.252 rows=1) (actual time=0.0113..0.0114 rows=1 loops=37)\n -> Covering index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.268 rows=2.13) (actual time=0.00413..0.00438 rows=0.676 loops=37)\n'

Index 2: **Cover more on Papers (filter + join + select list)**

CREATE INDEX idx_papers_upload_proj_paper ON Papers(upload_timestamp DESC, project_id, paper_id);

DROP INDEX idx_papers_upload_proj_paper ON Papers;

Result after Index 2:
'-> Limit: 15 row(s) (actual time=1.98..1.99 rows=15 loops=1)\n   -> Sort: p.upload_timestamp DESC, review_count DESC, limit input to 15 row(s) per chunk (actual time=1.98..1.98 rows=15 loops=1)\n   -> Table scan on <temporary> (actual time=1.92..1.93 rows=37 loops=1)\n -> Aggregate using temporary table (actual time=1.91..1.91 rows=37 loops=1)\n   -> Nested loop left join (cost=73.3 rows=100) (actual time=0.0951..1.56 rows=51 loops=1)\n -> Nested loop inner join (cost=50.8 rows=47.2) (actual time=0.0808..1.33 rows=37 loops=1)\n -> Nested loop inner join (cost=34.3 rows=47.2) (actual time=0.0655..0.947 rows=37 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U005\') (cost=9.06 rows=72) (actual time=0.0288..0.0972 rows=72 loops=1)\n   -> Filter: ((p.upload_timestamp >= TIMESTAMP\'2018-01-01 00:00:00\') and (p.project_id is not null)) (cost=0.251 rows=0.655) (actual time=0.0114..0.0115 rows=0.514 loops=72)\n   -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.0109..0.011 rows=1 loops=72)\n   -> Single-row index lookup on pr using PRIMARY (project_id = p.project_id) (cost=0.252 rows=1) (actual time=0.00982..0.00989 rows=1 loops=37)\n   -> Covering index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.268 rows=2.13) (actual time=0.00542..0.0059 rows=0.676 loops=37)\n'

Index 3:
CREATE INDEX idx_reviews_paper_review ON Reviews(paper_id, review_id);

DROP INDEX idx_reviews_paper_review ON Reviews;

Result after Index 3:
'-> Sort: p.upload_timestamp DESC, review_count DESC (actual time=1.59..1.59 rows=37 loops=1)\n   -> Table scan on <temporary> (actual time=1.47..1.48 rows=37 loops=1)\n   -> Aggregate using temporary table (actual time=1.46..1.46 rows=37 loops=1)\n   -> Nested loop left join (cost=54.1 rows=51) (actual time=0.0806..1.22 rows=51 loops=1)\n   -> Nested loop inner join (cost=42.7 rows=24) (actual time=0.0687..1.05 rows=37 loops=1)\n -> Nested loop inner join (cost=34.3 rows=24) (actual time=0.0559..0.751 rows=37 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U005\') (cost=9.06 rows=72) (actual time=0.0245..0.0765 rows=72 loops=1)\n   -> Filter: ((p.upload_timestamp >= TIMESTAMP\'2018-01-01 00:00:00\') and (p.project_id is not null)) (cost=0.25 rows=0.333) (actual time=0.0091..0.00916 rows=0.514 loops=72)\n   -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.25 rows=1) (actual time=0.00873..0.00877 rows=1 loops=72)\n   -> Single-row index lookup on pr using

PRIMARY (project_id = p.project_id) (cost=0.254 rows=1) (actual time=0.00785..0.0079 rows=1 loops=37)\n -> Covering index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.272 rows=2.13) (actual time=0.00392..0.00422 rows=0.676 loops=37)\n'

Index 4:
CREATE INDEX idx_papers_upload_paper ON Papers(upload_timestamp DESC, paper_id);
Drop INDEX idx_papers_upload_paper ON Papers;

Result after index 4:
'-> Limit: 15 row(s) (actual time=1.88..1.89 rows=15 loops=1)\n -> Sort: p.upload_timestamp DESC, review_count DESC, limit input to 15 row(s) per chunk (actual time=1.88..1.89 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=1.78..1.8 rows=37 loops=1)\n -> Aggregate using temporary table (actual time=1.77..1.77 rows=37 loops=1)\n -> Nested loop left join (cost=68.5 rows=88.3) (actual time=0.149..1.46 rows=51 loops=1)\n -> Nested loop inner join (cost=48.8 rows=41.5) (actual time=0.0933..1.24 rows=37 loops=1)\n -> Nested loop inner join (cost=34.3 rows=41.5) (actual time=0.064..0.784 rows=37 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U005\') (cost=9.06 rows=72) (actual time=0.0308..0.0877 rows=72 loops=1)\n -> Filter: ((p.upload_timestamp >= TIMESTAMP\'2018-01-01 00:00:00\') and (p.project_id is not null)) (cost=0.251 rows=0.576) (actual time=0.00935..0.00942 rows=0.514 loops=72)\n -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.00897..0.00901 rows=1 loops=72)\n -> Single-row index lookup on pr using PRIMARY (project_id = p.project_id) (cost=0.252 rows=1) (actual time=0.0121..0.0122 rows=1 loops=37)\n -> Covering index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.269 rows=2.13) (actual time=0.00508..0.00545 rows=0.676 loops=37)\n'

Analysis:

Index designs tested

1. D1 — Papers(upload_timestamp)
2. D2 — Papers(upload_timestamp, project_id, paper_id)
3. D3 — Reviews(paper_id, review_id)

What happened

- D1: No improvement by the optimizer’s cost; plan shape unchanged.

- D2: Same as D1—no cost benefit over baseline.
- D3: No meaningful change; plan and cost matched baseline.
- D4: Improves vs. baseline/D1, but similar to D2 and slower than D3

Why the indexes didn't help

- The plan starts from Authorship using its PK (user_id, paper_id) (filtered by user_id='U005') and then does primary-key lookups into Papers by paper_id. Because access to Papers is by PK, an index on upload_timestamp (even in a composite) isn't used to find rows.
- The ORDER BY p.upload_timestamp, review_count happens after aggregation, so MySQL still performs a sort; the timestamp index cannot remove that sort here.
- For Reviews, we already join on paper_id and have an index on paper_id; adding (paper_id, review_id) doesn't change the join strategy or estimates.

Final index design (selected)

- Keep the baseline only (existing PK/FK/UNIQUE indexes).
- Do not add any new indexes for this query.

Trade-off note

Adding the above indexes would increase storage and write overhead without improving this query's planned cost, so we did not adopt them.

Q2.

Explain Analyze

SELECT

v.venue_id,

v.venue_name,

```
    v.year,
    COUNT(p.paper_id) AS total_papers
FROM Venues v
JOIN Papers p
    ON v.venue_id = p.venue_id
WHERE v.year >= 2018
  AND p.status IN ('Published')
GROUP BY v.venue_id, v.venue_name, v.year
ORDER BY v.year DESC, total_papers DESC
LIMIT 15 ;
```

Baseline Result:

```
'-> Limit: 15 row(s)  (actual time=30.3..30.3 rows=15 loops=1)\n    -> Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk  (actual time=30.3..30.3 rows=15 loops=1)\n        -> Table scan on <temporary>  (actual time=29.7..29.7 rows=81 loops=1)\n -> Aggregate using temporary table  (actual time=29.7..29.7 rows=81 loops=1)\n             -> Nested loop inner join  (cost=586 rows=260) (actual time=0.485..23.9 rows=3231 loops=1)\n -> Filter: ((p.`status` = \'Published\') and (p.venue_id is not null))  (cost=474 rows=322) (actual time=0.396..7.81 rows=3999 loops=1)\n                 -> Table scan on p  (cost=474 rows=3216) (actual time=0.381..6.28 rows=4000 loops=1)\n                 -> Filter: (v.`year` >= 2018)  (cost=0.25 rows=0.81) (actual time=0.00369..0.0038 rows=0.808 loops=3999)\n                 -> Single-row index lookup on v using PRIMARY (venue_id = p.venue_id)  (cost=0.25 rows=1) (actual time=0.00345..0.0035 rows=1 loops=3999)\n'
```

Index 1: **Year filter on Venues (add back what we dropped)**

```
CREATE INDEX idx_venues_year ON Venues(year DESC);
ANALYZE TABLE Venues;
EXPLAIN FORMAT=JSON <query>
```

DROP INDEX idx_venues_year ON Venues;

Result after Index 1: '{\n \"query\": \"/* select#1 */ select `v`.`venue_id` AS `venue_id`,`v`.`venue_name` AS `venue_name`,`v`.`year` AS `year`,count(`p`.`paper_id`) AS `total_papers` from `papersdb`.`venues` `v` join `papersdb`.`papers` `p` where ((`v`.`venue_id` = `p`.`venue_id`) and (`p`.`status` = \'Published\') and (`v`.`year` >= 2018)) group by `v`.`venue_id`,`v`.`venue_name`,`v`.`year` order by `v`.`year` desc,`total_papers` desc limit 15\",\n \"query_plan\": {\n \"limit\": 15,\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"alias\": \"p\",\n \"operation\": \"Table scan on p\",\n \"table_name\": \"papers\",\n \"access_type\": \"table\",\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"status\",\n \"venue_id\"\n],\n \"estimated_rows\": 3216.0,\n \"estimated_total_cost\": 473.85\n }\n],\n \"condition\": \"((p.`status` = \'Published\') and (p.venue_id is not null))\",\n \"operation\": \"Filter: ((p.`status` = \'Published\') and (p.venue_id is not null))\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 321.60000479221344,\n \"filter_columns\": [\n \"p.`status`\",\n \"p.venue_id\"\n],\n \"estimated_total_cost\": 473.85\n },\n {\n \"inputs\": [\n {\n \"alias\": \"v\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on v using PRIMARY (venue_id = p.venue_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"venues\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"venue_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"venue_id\",\n \"venue_name\",\n \"year\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"venue_id = p.venue_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.p.venue_id\"\n],\n \"estimated_total_cost\": 0.25025186566863\n }\n],\n \"condition\": \"(v.`year` >= 2018)\",\n \"operation\": \"Filter: (v.`year` >= 2018)\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 0.8100000023841858,\n \"filter_columns\": [\n \"v.`year`\"\n],\n \"estimated_total_cost\": 0.25025186566863\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 260.49600464844707,\n

\"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 586.4100016772747\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"v.`year` DESC\",\n \"total_papers DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Explain Analyze on Index 1:
'-> Limit: 15 row(s) (actual time=27..27 rows=15 loops=1)\n -> Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk (actual time=27..27 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=27..27 rows=81 loops=1)\n -> Aggregate using temporary table (actual time=27..27 rows=81 loops=1)\n -> Nested loop inner join (cost=586 rows=260) (actual time=0.129..21.4 rows=3231 loops=1)\n -> Filter: ((p.`status` = \'Published\') and (p.venue_id is not null)) (cost=474 rows=322) (actual time=0.101..5.06 rows=3999 loops=1)\n -> Table scan on p (cost=474 rows=3216) (actual time=0.0922..3.5 rows=4000 loops=1)\n -> Filter: (v.`year` >= 2018) (cost=0.25 rows=0.81) (actual time=0.00376..0.00388 rows=0.808 loops=3999)\n -> Single-row index lookup on v using PRIMARY (venue_id = p.venue_id) (cost=0.25 rows=1) (actual time=0.00351..0.00356 rows=1 loops=3999)\n'

Index 2:
CREATE INDEX idx_papers_venue_status ON Papers(venue_id, status);

ANALYZE TABLE Papers;

EXPLAIN FORMAT=JSON <query2>
DROP INDEX idx_papers_venue_status ON Papers;

Result after Index 2:
'{\n \"query\": \"/* select#1 */ select `v`.`venue_id` AS `venue_id`,`v`.`venue_name` AS `venue_name`,`v`.`year` AS `year`,count(`p`.`paper_id`) AS `total_papers` from `papersdb`.`venues` `v` join `papersdb`.`papers` `p` where ((`p`.`venue_id` = `v`.`venue_id`) and

(`p`.`status` = \'Published\') and (`v`.`year` >= 2018)) group by `v`.`venue_id`,`v`.`venue_name`,`v`.`year` order by `v`.`year` desc,`total_papers` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"alias\": \"v\",\n   \"operation\": \"Table scan on v\",\n   \"table_name\": \"venues\",\n   \"access_type\": \"table\",\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"venue_id\",\n   \"venue_name\",\n   \"year\"\n   ],\n   \"estimated_rows\": 100.0,\n   \"estimated_total_cost\": 10.25\n   }\n   ],\n   \"condition\": \"(v.`year` >= 2018)\",\n   \"operation\": \"Filter: (v.`year` >= 2018)\",\n   \"access_type\": \"filter\",\n   \"estimated_rows\": 81.00000023841858,\n   \"filter_columns\": [\n   \"v.`year`\"\n   ],\n   \"estimated_total_cost\": 10.25\n   },\n   {\n   \"alias\": \"p\",\n   \"covering\": true,\n   \"operation\": \"Covering index lookup on p using idx_papers_venue_status (venue_id = v.venue_id, status = \'Published\')\",\n   \"index_name\": \"idx_papers_venue_status\",\n   \"table_name\": \"papers\",\n   \"access_type\": \"index\",\n   \"key_columns\": [\n   \"venue_id\",\n   \"status\"\n   ],\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"paper_id\",\n   \"status\",\n   \"venue_id\"\n   ],\n   \"estimated_rows\": 32.56435775756836,\n   \"lookup_condition\": \"venue_id = v.venue_id, status = \'Published\'\",\n   \"index_access_type\": \"index_lookup\",\n   \"lookup_references\": [\n   \"papersdb.v.venue_id\",\n   \"const\"\n   ],\n   \"estimated_total_cost\": 2.501888748890885\n   }\n   ],\n   \"join_type\": \"inner join\",\n   \"operation\": \"Nested loop inner join\",\n   \"access_type\": \"join\",\n   \"estimated_rows\": 2637.712986126985,\n   \"join_algorithm\": \"nested_loop\",\n   \"estimated_total_cost\": 473.4178520936001\n   }\n   ],\n   \"operation\": \"Aggregate using temporary table\",\n   \"access_type\": \"temp_table_aggregate\"\n   }\n   ],\n   \"operation\": \"Table scan on <temporary>\",\n   \"table_name\": \"<temporary>\",\n   \"access_type\": \"table\"\n   }\n   ],\n   \"operation\": \"Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk\",\n   \"access_type\": \"sort\",\n   \"sort_fields\": [\n   \"v.`year` DESC\",\n   \"total_papers DESC\"\n   ],\n   \"per_chunk_limit\": 15\n   }\n   ],\n   \"operation\": \"Limit: 15 row(s)\",\n   \"access_type\": \"limit\",\n   \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Analyze Explain after Index 2:

'-> Limit: 15 row(s) (actual time=15.2..15.2 rows=15 loops=1)\n   -> Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk (actual time=15.2..15.2 rows=15 loops=1)\n   -> Table scan on <temporary> (actual time=15.1..15.2 rows=81 loops=1)\n   -> Aggregate using temporary table (actual time=15.1..15.1 rows=81 loops=1)\n   -> Nested loop inner join (cost=473 rows=2638) (actual time=0.117..7 rows=3231 loops=1)\n   -> Filter: (v.`year` >= 2018) (cost=10.2 rows=81) (actual time=0.069..0.204 rows=81 loops=1)\n   -> Table scan on v (cost=10.2 rows=100) (actual time=0.0663..0.177 rows=100 loops=1)\n   -> Covering index lookup on p using idx_papers_venue_status (venue_id = v.venue_id, status = \'Published\') (cost=2.5 rows=32.6) (actual time=0.0161..0.0775 rows=39.9 loops=81)\n'

Index 3:

```
CREATE INDEX idx_papers_status_venue ON Papers(status, venue_id);

ANALYZE TABLE Papers;

EXPLAIN FORMAT=JSON <query3>


DROP INDEX idx_papers_status_venue ON Papers;
```

Result after Index 3:

'{\n \"query\": \"/* select#1 */ select `v`.`venue_id` AS `venue_id`,`v`.`venue_name` AS `venue_name`,`v`.`year` AS `year`,count(`p`.`paper_id`) AS `total_papers` from `papersdb`.`venues` `v` join `papersdb`.`papers` `p` where ((`p`.`venue_id` = `v`.`venue_id`) and (`p`.`status` = \'Published\') and (`v`.`year` >= 2018)) group by `v`.`venue_id`,`v`.`venue_name`,`v`.`year` order by `v`.`year` desc,`total_papers` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"alias\": \"v\",\n   \"operation\": \"Table scan on v\",\n   \"table_name\": \"venues\",\n   \"access_type\": \"table\",\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"venue_id\",\n   \"venue_name\",\n   \"year\"\n   ],\n   \"estimated_rows\": 100.0,\n   \"estimated_total_cost\": 10.25\n   }\n   ],\n   \"condition\": \"(v.`year` >= 2018)\",\n   \"operation\": \"Filter: (v.`year` >= 2018)\",\n   \"access_type\": \"filter\",\n   \"estimated_rows\": 81.00000023841858,\n   \"filter_columns\": [\n   \"v.`year`\"\n   ],\n

\"estimated_total_cost\": 10.25\n },\n {\n \"alias\": \"p\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on p using idx_papers_status_venue (status = \'Published\', venue_id = v.venue_id)\",\n \"index_name\": \"idx_papers_status_venue\",\n \"table_name\": \"papers\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"status\",\n \"venue_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"status\",\n \"venue_id\"\n],\n \"estimated_rows\": 40.0990104675293,\n \"lookup_condition\": \"status = \'Published\', venue_id = v.venue_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"const\",\n \"papersdb.v.venue_id\"\n],\n \"estimated_total_cost\": 2.8780094080205867\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 3248.019857430222,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 564.1608474321077\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"v.`year` DESC\",\n \"total_papers DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Explain Analyze after Index 3:

'-> Limit: 15 row(s) (actual time=12..12 rows=15 loops=1)\n -> Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk (actual time=12..12 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=12..12 rows=81 loops=1)\n -> Aggregate using temporary table (actual time=12..12 rows=81 loops=1)\n -> Nested loop inner join (cost=564 rows=3248) (actual time=0.116..5.56 rows=3231 loops=1)\n -> Filter: (v.`year` >= 2018) (cost=10.2 rows=81) (actual time=0.0705..0.183 rows=81 loops=1)\n -> Table scan on v (cost=10.2 rows=100) (actual time=0.0686..0.16 rows=100 loops=1)\n -> Covering index lookup on p using idx_papers_status_venue (status = \'Published\', venue_id = v.venue_id) (cost=2.88 rows=40.1) (actual time=0.0141..0.0613 rows=39.9 loops=81)\n'

Index 4:
CREATE INDEX idx_papers_status ON Papers(status);

ANALYZE TABLE Papers;

EXPLAIN FORMAT=JSON <query>
DROP INDEX idx_papers_status ON Papers;

Result after Index 4:
'{\n \"query\": \"/* select#1 */ select `v`.`venue_id` AS `venue_id`,`v`.`venue_name` AS `venue_name`,`v`.`year` AS `year`,count(`p`.`paper_id`) AS `total_papers` from `papersdb`.`venues` `v` join `papersdb`.`papers` `p` where ((`p`.`venue_id` = `v`.`venue_id`) and (`p`.`status` = \'Published\') and (`v`.`year` >= 2018)) group by `v`.`venue_id`,`v`.`venue_name`,`v`.`year` order by `v`.`year` desc,`total_papers` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"alias\": \"v\",\n \"operation\": \"Table scan on v\",\n   \"table_name\": \"venues\",\n \"access_type\": \"table\",\n   \"schema_name\": \"papersdb\",\n \"used_columns\": [\n   \"venue_id\",\n   \"venue_name\",\n \"year\"\n   ],\n   \"estimated_rows\": 100.0,\n \"estimated_total_cost\": 10.25\n   }\n   ],\n   \"condition\": \"(v.`year` >= 2018)\",\n   \"operation\": \"Filter: (v.`year` >= 2018)\",\n \"access_type\": \"filter\",\n   \"estimated_rows\": 81.00000023841858,\n \"filter_columns\": [\n   \"v.`year`\"\n   ],\n \"estimated_total_cost\": 10.25\n   },\n   {\n   \"inputs\": [\n {\n   \"alias\": \"p\",\n   \"covering\": false,\n \"operation\": \"Index lookup on p using idx_papers_venue (venue_id = v.venue_id)\",\n \"index_name\": \"idx_papers_venue\",\n   \"table_name\": \"papers\",\n \"access_type\": \"index\",\n   \"key_columns\": [\n \"venue_id\"\n   ],\n   \"schema_name\": \"papersdb\",\n \"used_columns\": [\n   \"paper_id\",\n   \"status\",\n \"venue_id\"\n   ],\n   \"estimated_rows\": 36.94059371948242,\n \"lookup_condition\": \"venue_id = v.venue_id\",\n   \"index_access_type\": \"index_lookup\",\n   \"lookup_references\": [\n \"papersdb.v.venue_id\"\n   ],\n   \"estimated_total_cost\": 9.280754100994988\n   }\n   ],\n   \"condition\": \"(p.`status` = \'Published\')\",\n   \"operation\": \"Filter: (p.`status` =

\'Published\')\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 36.94059371948242,\n \"filter_columns\": [\n \"p.`status`\"\n],\n \"estimated_total_cost\": 9.280754100994988\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 2992.1881000854,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 1057.51583502989\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"v.`year` DESC\",\n \"total_papers DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Explain Analyze after Index 4:

'-> Limit: 15 row(s) (actual time=24..24 rows=15 loops=1)\n -> Sort: v.`year` DESC, total_papers DESC, limit input to 15 row(s) per chunk (actual time=24..24 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=23.9..23.9 rows=81 loops=1)\n -> Aggregate using temporary table (actual time=23.9..23.9 rows=81 loops=1)\n -> Nested loop inner join (cost=1058 rows=2992) (actual time=0.147..18.8 rows=3231 loops=1)\n -> Filter: (v.`year` >= 2018) (cost=10.2 rows=81) (actual time=0.0535..0.15 rows=81 loops=1)\n -> Table scan on v (cost=10.2 rows=100) (actual time=0.0511..0.128 rows=100 loops=1)\n -> Filter: (p.`status` = \'Published\') (cost=9.28 rows=36.9) (actual time=0.0305..0.226 rows=39.9 loops=81)\n -> Index lookup on p using idx_papers_venue (venue_id = v.venue_id) (cost=9.28 rows=36.9) (actual time=0.03..0.212 rows=39.9 loops=81)\n'

Analysis:

Query (parsed):

Venues joined to Papers on venue_id. Filters: v.year >= 2018 and p.status = 'Published'.

Grouped by venue and year; ordered by v.year DESC, total_papers DESC; LIMIT 15.

Baseline (only PK/FK/UNIQUE)

- Plan scans Papers first, filters status='Published', then does PK lookups into Venues.
- Aggregates into a temp table, then sorts for the final order and limit.

Index designs tested and what happened

D1 — Venues(year)

- Plan stayed the same: still scanned Papers first and only checked v.year after the PK lookup.
- No cost benefit; still sorts after aggregation.
- Decision: Not adopted.

D2 — Papers(venue_id, status)

- Plan improved: starts with Venues and, for each venue row (filtered later by year), performs a covering index lookup on Papers using (venue_id, status='Published').
- Fewer rows pulled from Papers; actual time dropped noticeably; optimizer cost decreased.
- Decision: Adopt this index.

D3 — Papers(status, venue_id)

- EXPLAIN shows the optimizer still using idx_papers_venue_status (from D2), not this new index.
- That means D3 wasn't isolated (the D2 index was still present or preferred), so these results don't tell us anything about D3.
- Expectation (if re-run correctly): usually worse than D2 for this query shape, because the join is on venue_id first and the leftmost column matters.
- Decision: Not adopted.

D4 — Papers(status)

- EXPLAIN again shows the plan using idx_papers_venue_status, so D4 wasn't isolated either.
- Expectation (if isolated): typically weaker than D2 since the join still needs venue_id.
- Decision: Not adopted.

Final index design (selected)

- Papers(venue_id, status)

Why:

- Joins on venue_id and filters status='Published' in the same lookup.
- Acts as a covering path for counting paper_id, so fewer table reads.
- Gave a cheaper plan and faster runtime in our tests.

Why not Venues(year) here:

- The optimizer didn't use it for this query—it still scanned Venues and filtered year afterward.
- No change in cost or plan shape. It might help other queries that start from Venues and filter by year, but not this one.

Q3.

```
SELECT
  u.user_id,
  u.user_name,
  u.affiliation,
  COUNT(DISTINCT a.paper_id) AS total_papers_authored,
  COUNT(r.review_id) AS total_reviews_received
```

```
FROM Users u
JOIN Authorship a
   ON u.user_id = a.user_id
JOIN Reviews r
   ON a.paper_id = r.paper_id
WHERE u.is_reviewer = TRUE
  AND r.review_timestamp BETWEEN '2024-02-15 00:00:00' AND '2024-05-15 23:00:00'
GROUP BY u.user_id, u.user_name, u.affiliation
HAVING COUNT(r.review_id) > 0
ORDER BY total_reviews_received DESC
LIMIT 15;
```

Baseline Result:

'{\n \"query\": \"/* select#1 */ select `u`.`user_id` AS `user_id`,`u`.`user_name` AS `user_name`,`u`.`affiliation` AS `affiliation`,count(distinct `papersdb`.`authorship`.`paper_id`) AS `total_papers_authored`,count(`papersdb`.`reviews`.`review_id`) AS `total_reviews_received` from `papersdb`.`users` `u` join `papersdb`.`authorship` `a` join `papersdb`.`reviews` `r` where ((`u`.`user_id` = `a`.`user_id`) and (`a`.`paper_id` = `r`.`paper_id`) and (`u`.`is_reviewer` = true) and (`r`.`review_timestamp` between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')) group by `u`.`user_id`,`u`.`user_name`,`u`.`affiliation` having (count(`papersdb`.`reviews`.`review_id`) > 0) order by `total_reviews_received` desc limit 15\",\n \"query_plan\": {\n \"limit\": 15,\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"alias\": \"r\",\n \"operation\": \"Table scan on r\",\n \"table_name\": \"reviews\",\n \"access_type\": \"table\",\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n

\"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 570.0,\n \"estimated_total_cost\": 58.5\n }\n],\n \"condition\": \"(r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n \"operation\": \"Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 63.32700178027153,\n \"filter_columns\": [\n \"r.review_timestamp\"\n],\n \"estimated_total_cost\": 58.5\n },\n {\n \"alias\": \"a\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id)\",\n \"index_name\": \"ix_authorship_paper\",\n \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 2.499025344848633,\n \"lookup_condition\": \"paper_id = r.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.r.paper_id\"\n],\n \"estimated_total_cost\": 0.2807145337327799\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 158.25578246217304,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 91.85248548917629\n },\n {\n \"inputs\": [\n {\n \"alias\": \"u\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on u using PRIMARY (user_id = a.user_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"users\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"user_name\",\n \"affiliation\",\n \"is_reviewer\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"user_id = a.user_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.user_id\"\n],\n \"estimated_total_cost\": 0.2500631888452569\n }\n],\n \"condition\": \"(u.is_reviewer = true)\",\n \"operation\": \"Filter: (u.is_reviewer = true)\",\n \"access_type\":

\"filter\",\n \"estimated_rows\": 0.10000000149011612,\n \"filter_columns\": [\n \"u.is_reviewer\"\n],\n \"estimated_total_cost\": 0.2500631888452569\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 15.825578482036796,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 147.24200935093685\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\",\n \"estimated_rows\": 15.825578482036796,\n \"estimated_total_cost\": 147.24200935093685\n }\n],\n \"operation\": \"Sort: u.user_id, u.user_name, u.affiliation\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"u.user_id\",\n \"u.user_name\",\n \"u.affiliation\"\n]\n }\n],\n \"group_by\": true,\n \"functions\": [\n \"count(reviews.review_id)\",\n \"count(distinct authorship.paper_id)\",\n \"count(reviews.review_id)\"\n],\n \"operation\": \"Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id)\",\n \"access_type\": \"aggregate\"\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\"\n }\n],\n \"condition\": \"(count(reviews.review_id) > 0)\",\n \"operation\": \"Filter: (count(reviews.review_id) > 0)\",\n \"access_type\": \"filter\",\n \"filter_columns\": [\n \"papersdb.reviews.review_id\"\n]\n }\n],\n \"operation\": \"Sort: total_reviews_received DESC\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"total_reviews_received DESC\"\n]\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Index 1:

CREATE INDEX idx_users_is_reviewer ON Users(is_reviewer);

ANALYZE TABLE Users;

EXPLAIN FORMAT=JSON <query>

DROP INDEX idx_users_is_reviewer ON Users;

Result after Index 1:

'{\n \"query\": \"/* select#1 */ select `u`.`user_id` AS `user_id`,`u`.`user_name` AS `user_name`,`u`.`affiliation` AS `affiliation`,count(distinct `papersdb`.`authorship`.`paper_id`) AS `total_papers_authored`,count(`papersdb`.`reviews`.`review_id`) AS `total_reviews_received` from `papersdb`.`users` `u` join `papersdb`.`authorship` `a` join `papersdb`.`reviews` `r` where ((`u`.`user_id` = `a`.`user_id`) and (`a`.`paper_id` = `r`.`paper_id`) and (`u`.`is_reviewer` = true) and (`r`.`review_timestamp` between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')) group by `u`.`user_id`,`u`.`user_name`,`u`.`affiliation` having (count(`papersdb`.`reviews`.`review_id`) > 0) order by `total_reviews_received` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"alias\": \"r\",\n   \"operation\": \"Table scan on r\",\n   \"table_name\": \"reviews\",\n   \"access_type\": \"table\",\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"review_id\",\n   \"paper_id\",\n   \"review_timestamp\"\n   ],\n   \"estimated_rows\": 570.0,\n   \"estimated_total_cost\": 58.5\n   }\n   ],\n   \"condition\": \"(r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n   \"operation\": \"Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n   \"access_type\": \"filter\",\n   \"estimated_rows\": 63.32700178027153,\n   \"filter_columns\": [\n   \"r.review_timestamp\"\n   ],\n   \"estimated_total_cost\": 58.5\n   },\n   {\n   \"alias\": \"a\",\n   \"covering\": true,\n   \"operation\": \"Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id)\",\n   \"index_name\": \"ix_authorship_paper\",\n   \"table_name\": \"authorship\",\n   \"access_type\": \"index\",\n   \"key_columns\": [\n   \"paper_id\"\n   ],\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"user_id\",\n   \"paper_id\"\n   ],\n   \"estimated_rows\": 2.499025344848633,\n   \"lookup_condition\": \"paper_id = r.paper_id\",\n   \"index_access_type\": \"index_lookup\",\n   \"lookup_references\": [\n   \"papersdb.r.paper_id\"\n   ],\n   \"estimated_total_cost\": 0.2807145337327799\n

```
}\n ],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 158.25578246217304,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 91.85248548917629\n },\n {\n \"inputs\": [\n {\n \"alias\": \"u\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on u using PRIMARY (user_id = a.user_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"users\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n ],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"user_name\",\n \"affiliation\",\n \"is_reviewer\"\n ],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"user_id = a.user_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.user_id\"\n ],\n \"estimated_total_cost\": 0.2501579721107883\n }\n ],\n \"condition\": \"(u.is_reviewer = true)\",\n \"operation\": \"Filter: (u.is_reviewer = true)\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 0.25,\n \"filter_columns\": [\n \"u.is_reviewer\"\n ],\n \"estimated_total_cost\": 0.2501579721107883\n }\n ],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 39.56394561554326,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 147.24200935093685\n }\n ],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\",\n \"estimated_rows\": 39.56394561554326,\n \"estimated_total_cost\": 147.24200935093685\n }\n ],\n \"operation\": \"Sort: u.user_id, u.user_name, u.affiliation\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"u.user_id\",\n \"u.user_name\",\n \"u.affiliation\"\n ]\n }\n ],\n \"group_by\": true,\n \"functions\": [\n \"count(reviews.review_id)\",\n \"count(distinct authorship.paper_id)\",\n \"count(reviews.review_id)\"\n ],\n \"operation\": \"Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id)\",\n \"access_type\": \"aggregate\"\n }\n ],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\"\n }\n ],\n \"condition\":
```

\"(count(reviews.review_id) > 0)\",\n \"operation\": \"Filter: (count(reviews.review_id) > 0)\",\n \"access_type\": \"filter\",\n \"filter_columns\": [\n \"papersdb.reviews.review_id\"\n]\n }\n],\n \"operation\": \"Sort: total_reviews_received DESC\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"total_reviews_received DESC\"\n]\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:

'-> Limit: 15 row(s) (actual time=12.6..12.6 rows=10 loops=1)\n -> Sort: total_reviews_received DESC (actual time=12.6..12.6 rows=10 loops=1)\n -> Filter: (`count(reviews.review_id)` > 0) (actual time=12.1..12.5 rows=10 loops=1)\n -> Stream results (actual time=12.1..12.5 rows=10 loops=1)\n -> Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id) (actual time=12.1..12.5 rows=10 loops=1)\n -> Sort: u.user_id, u.user_name, u.affiliation (actual time=12..12.1 rows=237 loops=1)\n -> Stream results (cost=147 rows=39.6) (actual time=0.17..11.7 rows=237 loops=1)\n -> Nested loop inner join (cost=147 rows=39.6) (actual time=0.165..11.3 rows=237 loops=1)\n -> Nested loop inner join (cost=91.9 rows=158) (actual time=0.147..5.99 rows=854 loops=1)\n -> Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\') (cost=58.5 rows=63.3) (actual time=0.111..1.31 rows=361 loops=1)\n -> Table scan on r (cost=58.5 rows=570) (actual time=0.102..0.645 rows=570 loops=1)\n -> Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id) (cost=0.281 rows=2.5) (actual time=0.0104..0.0123 rows=2.37 loops=361)\n -> Filter: (u.is_reviewer = true) (cost=0.25 rows=0.25) (actual time=0.00581..0.00589 rows=0.278 loops=854)\n -> Single-row index lookup on u using PRIMARY (user_id = a.user_id) (cost=0.25 rows=1) (actual time=0.00528..0.00536 rows=1 loops=854)\n'

Index 2:

```
CREATE INDEX idx_reviews_ts_paper ON Reviews(review_timestamp, paper_id);

ANALYZE TABLE Reviews;

EXPLAIN FORMAT=JSON <query>
DROP INDEX idx_reviews_ts_paper ON Reviews;
```

Result after Index 1:

'{\n \"query\": \"/* select#1 */ select `u`.`user_id` AS `user_id`,`u`.`user_name` AS `user_name`,`u`.`affiliation` AS `affiliation`,count(distinct `a`.`paper_id`) AS `total_papers_authored`,count(`r`.`review_id`) AS `total_reviews_received` from `papersdb`.`users` `u` join `papersdb`.`authorship` `a` join `papersdb`.`reviews` `r` where ((`a`.`user_id` = `u`.`user_id`) and (`r`.`paper_id` = `a`.`paper_id`) and (`u`.`is_reviewer` = true) and (`r`.`review_timestamp` between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')) group by `u`.`user_id`,`u`.`user_name`,`u`.`affiliation` having (count(`r`.`review_id`) > 0) order by `total_reviews_received` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n {\n   \"inputs\": [\n   {\n   \"inputs\": [\n {\n   \"inputs\": [\n   {\n   \"alias\": \"u\",\n   \"operation\": \"Table scan on u\",\n \"table_name\": \"users\",\n   \"access_type\": \"table\",\n \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n \"user_id\",\n   \"user_name\",\n   \"affiliation\",\n \"is_reviewer\"\n   ],\n   \"estimated_rows\": 40.0,\n \"estimated_total_cost\": 4.25\n   }\n   ],\n \"condition\": \"(u.is_reviewer = true)\",\n   \"operation\": \"Filter: (u.is_reviewer = true)\",\n   \"access_type\": \"filter\",\n \"estimated_rows\": 40.0,\n   \"filter_columns\": [\n \"u.is_reviewer\"\n   ],\n   \"estimated_total_cost\": 4.25\n   }\n   ],\n   \"operation\": \"Sort: u.user_id, u.user_name, u.affiliation\",\n   \"access_type\": \"sort\",\n \"sort_fields\": [\n   \"u.user_id\",\n   \"u.user_name\",\n \"u.affiliation\"\n   ],\n   \"estimated_rows\": 40.0,\n \"estimated_total_cost\": 4.25\n   },\n   {\n \"alias\": \"a\",\n   \"covering\": true,\n   \"operation\": \"Covering index lookup on a using PRIMARY (user_id = u.user_id)\",\n \"index_name\": \"PRIMARY\",\n   \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n   \"key_columns\": [\n \"user_id\"\n   ],\n   \"schema_name\": \"papersdb\",\n \"used_columns\": [\n   \"user_id\",\n   \"paper_id\"\n],\n   \"estimated_rows\": 64.0999984741211,\n \"lookup_condition\": \"user_id = u.user_id\",\n   \"index_access_type\":

\"index_lookup\",\n \"lookup_references\": [\n \"papersdb.u.user_id\"\n],\n \"estimated_total_cost\": 3.2865908123857603\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 2563.9999389648438,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 267.3863574945114\n },\n {\n \"inputs\": [\n {\n \"alias\": \"r\",\n \"covering\": false,\n \"operation\": \"Index lookup on r using idx_reviews_paper (paper_id = a.paper_id)\",\n \"index_name\": \"idx_reviews_paper\",\n \"table_name\": \"reviews\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n \"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 2.1268656253814697,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.5322417631671948\n }\n],\n \"condition\": \"(r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n \"operation\": \"Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 1.3470148791721215,\n \"filter_columns\": [\n \"r.review_timestamp\"\n],\n \"estimated_total_cost\": 0.5322417631671948\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 3453.746067982056,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 949.0467762340818\n }\n],\n \"group_by\": true,\n \"functions\": [\n \"count(r.review_id)\",\n \"count(distinct a.paper_id)\",\n \"count(r.review_id)\"\n],\n \"operation\": \"Group aggregate: count(r.review_id), count(distinct a.paper_id), count(r.review_id)\",\n \"access_type\": \"aggregate\",\n \"estimated_rows\": 473.6310714527134,\n \"estimated_total_cost\": 1744.8408011147399\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\",\n \"estimated_rows\": 473.6310714527134,\n \"estimated_total_cost\": 1744.8408011147399\n }\n],\n \"condition\": \"(count(r.review_id) > 0)\",\n \"operation\": \"Filter: (count(r.review_id) > 0)\",\n \"access_type\": \"filter\",\n \"filter_columns\": [\n \"r.review_id\"\n]\n }\n],\n

\"operation\": \"Sort: total_reviews_received DESC\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"total_reviews_received DESC\"\n]\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:
'-> Limit: 15 row(s) (actual time=8.84..8.84 rows=10 loops=1)\n -> Sort: total_reviews_received DESC (actual time=8.84..8.84 rows=10 loops=1)\n -> Filter: (`count(r.review_id)` > 0) (actual time=0.949..8.81 rows=10 loops=1)\n -> Stream results (cost=1745 rows=474) (actual time=0.946..8.8 rows=10 loops=1)\n -> Group aggregate: count(r.review_id), count(distinct a.paper_id), count(r.review_id) (cost=1745 rows=474) (actual time=0.942..8.78 rows=10 loops=1)\n -> Nested loop inner join (cost=949 rows=3454) (actual time=0.266..8.14 rows=237 loops=1)\n -> Nested loop inner join (cost=267 rows=2564) (actual time=0.175..1.13 rows=669 loops=1)\n -> Sort: u.user_id, u.user_name, u.affiliation (cost=4.25 rows=40) (actual time=0.135..0.142 rows=10 loops=1)\n -> Filter: (u.is_reviewer = true) (cost=4.25 rows=40) (actual time=0.0453..0.103 rows=10 loops=1)\n -> Table scan on u (cost=4.25 rows=40) (actual time=0.043..0.0936 rows=40 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = u.user_id) (cost=3.29 rows=64.1) (actual time=0.0197..0.0879 rows=66.9 loops=10)\n -> Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\') (cost=0.532 rows=1.35) (actual time=0.00953..0.0101 rows=0.354 loops=669)\n -> Index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.532 rows=2.13) (actual time=0.00878..0.00921 rows=0.564 loops=669)\n'

Index 3:
CREATE INDEX idx_reviews_paper_ts ON Reviews(paper_id, review_timestamp);

ANALYZE TABLE Reviews;

EXPLAIN FORMAT=JSON <query>

DROP INDEX idx_reviews_paper_ts ON Reviews;

Result after Index 3:

'{\n \"query\": \"/* select#1 */ select `u`.`user_id` AS `user_id`,`u`.`user_name` AS `user_name`,`u`.`affiliation` AS `affiliation`,count(distinct `papersdb`.`authorship`.`paper_id`) AS `total_papers_authored`,count(`papersdb`.`reviews`.`review_id`) AS `total_reviews_received` from `papersdb`.`users` `u` join `papersdb`.`authorship` `a` join `papersdb`.`reviews` `r` where ((`u`.`user_id` = `a`.`user_id`) and (`a`.`paper_id` = `r`.`paper_id`) and (`u`.`is_reviewer` = true) and (`r`.`review_timestamp` between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')) group by `u`.`user_id`,`u`.`user_name`,`u`.`affiliation` having (count(`papersdb`.`reviews`.`review_id`) > 0) order by `total_reviews_received` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"alias\": \"r\",\n   \"covering\": true,\n   \"operation\": \"Covering index scan on r using idx_reviews_paper_ts\",\n   \"index_name\": \"idx_reviews_paper_ts\",\n   \"table_name\": \"reviews\",\n   \"access_type\": \"index\",\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"review_id\",\n   \"paper_id\",\n   \"review_timestamp\"\n   ],\n   \"estimated_rows\": 570.0,\n   \"index_access_type\": \"index_scan\",\n   \"estimated_total_cost\": 58.5\n   }\n   ],\n   \"condition\": \"(r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n   \"operation\": \"Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n   \"access_type\": \"filter\",\n   \"estimated_rows\": 63.32700178027153,\n   \"filter_columns\": [\n   \"r.review_timestamp\"\n   ],\n   \"estimated_total_cost\": 58.5\n   },\n   {\n   \"alias\": \"a\",\n   \"covering\": true,\n   \"operation\": \"Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id)\",\n   \"index_name\": \"ix_authorship_paper\",\n   \"table_name\": \"authorship\",\n   \"access_type\": \"index\",\n   \"key_columns\": [\n   \"paper_id\"\n   ],\n   \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n   \"user_id\",\n   \"paper_id\"\n   ],\n   \"estimated_rows\": 2.499025344848633,\n   \"lookup_condition\": \"paper_id = r.paper_id\",\n   \"index_access_type\": \"index_lookup\",\n

\"lookup_references\": [\n \"papersdb.r.paper_id\"\n],\n \"estimated_total_cost\": 0.2807145337327799\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 158.25578246217304,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 91.85248548917629\n },\n {\n \"inputs\": [\n {\n \"alias\": \"u\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on u using PRIMARY (user_id = a.user_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"users\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"user_name\",\n \"affiliation\",\n \"is_reviewer\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"user_id = a.user_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.user_id\"\n],\n \"estimated_total_cost\": 0.2500631888452569\n }\n],\n \"condition\": \"(u.is_reviewer = true)\",\n \"operation\": \"Filter: (u.is_reviewer = true)\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 0.10000000149011612,\n \"filter_columns\": [\n \"u.is_reviewer\"\n],\n \"estimated_total_cost\": 0.2500631888452569\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 15.825578482036796,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 147.24200935093685\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\",\n \"estimated_rows\": 15.825578482036796,\n \"estimated_total_cost\": 147.24200935093685\n }\n],\n \"operation\": \"Sort: u.user_id, u.user_name, u.affiliation\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"u.user_id\",\n \"u.user_name\",\n \"u.affiliation\"\n]\n }\n],\n \"group_by\": true,\n \"functions\": [\n \"count(reviews.review_id)\",\n \"count(distinct authorship.paper_id)\",\n \"count(reviews.review_id)\"\n],\n \"operation\": \"Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id)\",\n \"access_type\":

\"aggregate\"\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\"\n }\n],\n \"condition\": \"(count(reviews.review_id) > 0)\",\n \"operation\": \"Filter: (count(reviews.review_id) > 0)\",\n \"access_type\": \"filter\",\n \"filter_columns\": [\n \"papersdb.reviews.review_id\"\n]\n }\n],\n \"operation\": \"Sort: total_reviews_received DESC\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"total_reviews_received DESC\"\n]\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:

'-> Limit: 15 row(s) (actual time=10.1..10.1 rows=10 loops=1)\n -> Sort: total_reviews_received DESC (actual time=10.1..10.1 rows=10 loops=1)\n -> Filter: (`count(reviews.review_id)` > 0) (actual time=9.58..10 rows=10 loops=1)\n -> Stream results (actual time=9.58..10 rows=10 loops=1)\n -> Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id) (actual time=9.58..10 rows=10 loops=1)\n -> Sort: u.user_id, u.user_name, u.affiliation (actual time=9.53..9.58 rows=237 loops=1)\n -> Stream results (cost=147 rows=15.8) (actual time=0.215..9.24 rows=237 loops=1)\n -> Nested loop inner join (cost=147 rows=15.8) (actual time=0.212..8.84 rows=237 loops=1)\n -> Nested loop inner join (cost=91.9 rows=158) (actual time=0.126..4.62 rows=854 loops=1)\n -> Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\') (cost=58.5 rows=63.3) (actual time=0.0863..1.01 rows=361 loops=1)\n -> Covering index scan on r using idx_reviews_paper_ts (cost=58.5 rows=570) (actual time=0.0772..0.459 rows=570 loops=1)\n -> Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id) (cost=0.281 rows=2.5) (actual time=0.00796..0.00949 rows=2.37 loops=361)\n -> Filter: (u.is_reviewer = true) (cost=0.25 rows=0.1) (actual time=0.00464..0.0047 rows=0.278 loops=854)\n -> Single-row index lookup on u using PRIMARY (user_id = a.user_id) (cost=0.25 rows=1) (actual time=0.00422..0.00428 rows=1 loops=854)\n'

Index 4:

```
CREATE INDEX idx_reviews_paper_ts_id ON Reviews(paper_id, review_timestamp, review_id);

ANALYZE TABLE Reviews;
```

DROP INDEX idx_reviews_paper ON Reviews; -- FK still covered (paper_id is leftmost)

ANALYZE TABLE Reviews;
EXPLAIN FORMAT=JSON <query>

CREATE INDEX idx_reviews_paper ON Reviews(paper_id);

DROP INDEX idx_reviews_paper_ts_id ON Reviews;

ANALYZE TABLE Reviews;

Result after Index 4:

'{\n \"query\": \"/* select#1 */ select `u`.`user_id` AS `user_id`,`u`.`user_name` AS `user_name`,`u`.`affiliation` AS `affiliation`,count(distinct `papersdb`.`authorship`.`paper_id`) AS `total_papers_authored`,count(`papersdb`.`reviews`.`review_id`) AS `total_reviews_received` from `papersdb`.`users` `u` join `papersdb`.`authorship` `a` join `papersdb`.`reviews` `r` where ((`u`.`user_id` = `a`.`user_id`) and (`a`.`paper_id` = `r`.`paper_id`) and (`u`.`is_reviewer` = true) and (`r`.`review_timestamp` between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')) group by `u`.`user_id`,`u`.`user_name`,`u`.`affiliation` having (count(`papersdb`.`reviews`.`review_id`) > 0) order by `total_reviews_received` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n {\n   \"inputs\": [\n   {\n   \"inputs\": [\n {\n   \"inputs\": [\n   {\n \"inputs\": [\n   {\n \"alias\": \"r\",\n   \"covering\": true,\n \"operation\": \"Covering index scan on r using idx_reviews_paper_ts_id\",\n \"index_name\": \"idx_reviews_paper_ts_id\",\n   \"table_name\": \"reviews\",\n   \"access_type\": \"index\",\n \"schema_name\": \"papersdb\",\n   \"used_columns\": [\n \"review_id\",\n   \"paper_id\",\n \"review_timestamp\"\n   ],\n \"estimated_rows\": 570.0,\n   \"index_access_type\": \"index_scan\",\n \"estimated_total_cost\": 58.5\n   }\n   ],\n \"condition\": \"(r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n   \"operation\": \"Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\')\",\n \"access_type\": \"filter\",\n   \"estimated_rows\": 63.32700178027153,\n

\"filter_columns\": [\n \"r.review_timestamp\"\n],\n \"estimated_total_cost\": 58.5\n },\n {\n \"alias\": \"a\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id)\",\n \"index_name\": \"ix_authorship_paper\",\n \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 2.499025344848633,\n \"lookup_condition\": \"paper_id = r.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.r.paper_id\"\n],\n \"estimated_total_cost\": 0.2807145337327799\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 158.25578246217304,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 91.85248548917629\n },\n {\n \"inputs\": [\n {\n \"alias\": \"u\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on u using PRIMARY (user_id = a.user_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"users\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"user_name\",\n \"affiliation\",\n \"is_reviewer\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"user_id = a.user_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.user_id\"\n],\n \"estimated_total_cost\": 0.2500631888452569\n }\n],\n \"condition\": \"(u.is_reviewer = true)\",\n \"operation\": \"Filter: (u.is_reviewer = true)\",\n \"access_type\": \"filter\",\n \"estimated_rows\": 0.10000000149011612,\n \"filter_columns\": [\n \"u.is_reviewer\"\n],\n \"estimated_total_cost\": 0.2500631888452569\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 15.825578482036796,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\":

147.24200935093685\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\",\n \"estimated_rows\": 15.825578482036796,\n \"estimated_total_cost\": 147.24200935093685\n }\n],\n \"operation\": \"Sort: u.user_id, u.user_name, u.affiliation\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"u.user_id\",\n \"u.user_name\",\n \"u.affiliation\"\n]\n }\n],\n \"group_by\": true,\n \"functions\": [\n \"count(reviews.review_id)\",\n \"count(distinct authorship.paper_id)\",\n \"count(reviews.review_id)\"\n],\n \"operation\": \"Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id)\",\n \"access_type\": \"aggregate\"\n }\n],\n \"operation\": \"Stream results\",\n \"access_type\": \"stream\"\n }\n],\n \"condition\": \"(count(reviews.review_id) > 0)\",\n \"operation\": \"Filter: (count(reviews.review_id) > 0)\",\n \"access_type\": \"filter\",\n \"filter_columns\": [\n \"papersdb.reviews.review_id\"\n]\n }\n],\n \"operation\": \"Sort: total_reviews_received DESC\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"total_reviews_received DESC\"\n]\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:

'-> Limit: 15 row(s) (actual time=12.6..12.6 rows=10 loops=1)\n -> Sort: total_reviews_received DESC (actual time=12.6..12.6 rows=10 loops=1)\n -> Filter: (`count(reviews.review_id)` > 0) (actual time=12..12.6 rows=10 loops=1)\n -> Stream results (actual time=12..12.6 rows=10 loops=1)\n -> Group aggregate: count(reviews.review_id), count(distinct authorship.paper_id), count(reviews.review_id) (actual time=12..12.6 rows=10 loops=1)\n -> Sort: u.user_id, u.user_name, u.affiliation (actual time=11.9..12 rows=237 loops=1)\n -> Stream results (cost=147 rows=15.8) (actual time=0.218..11.5 rows=237 loops=1)\n -> Nested loop inner join (cost=147 rows=15.8) (actual time=0.214..11.1 rows=237 loops=1)\n -> Nested loop inner join (cost=91.9 rows=158) (actual time=0.103..5.75 rows=854 loops=1)\n -> Filter: (r.review_timestamp between \'2024-02-15 00:00:00\' and \'2024-05-15 23:00:00\') (cost=58.5 rows=63.3) (actual time=0.0732..1.22 rows=361 loops=1)\n -> Covering index scan on r using idx_reviews_paper_ts_id (cost=58.5 rows=570) (actual time=0.0661..0.543 rows=570 loops=1)\n -> Covering index lookup on a using ix_authorship_paper (paper_id = r.paper_id) (cost=0.281 rows=2.5) (actual

time=0.0099..0.0119 rows=2.37 loops=361)\n -> Filter: (u.is_reviewer = true) (cost=0.25 rows=0.1) (actual time=0.00593..0.006 rows=0.278 loops=854)\n
-> Single-row index lookup on u using PRIMARY (user_id = a.user_id) (cost=0.25 rows=1) (actual time=0.0054..0.00548 rows=1 loops=854)\n'

Analysis:

Query

Ranks reviewers (users with is_reviewer = TRUE) by number of reviews in a date range, joining:

- Users → Authorship on user_id
- Authorship → Reviews on paper_id
- Date filter on Reviews.review_timestamp
- Aggregates per user: COUNT(DISTINCT a.paper_id), COUNT(r.review_id)

Baseline behavior (no helpful secondary indexes)

- The optimizer either starts from Reviews (scan + timestamp filter) or from Users (scan reviewers → join).
- Joins use existing PK/unique indexes (Authorship.PRIMARY(user_id,paper_id), Users.PRIMARY(user_id), Reviews(paper_id)).
- Runtime observed in the “no real help” case is ~12.6s.

Index experiments

D1 —

Users(is_reviewer)

Plan used: table scan on Users, then filter is_reviewer = TRUE.

Effect: No change. The table is tiny and the predicate is low-selectivity; scanning 40 rows is cheaper than using a separate index and then fetching rows.

Keep? No.

- Result: Slower.
- Why: The index's leading column is review_timestamp, but the join is on paper_id. MySQL can't do a selective seek for the join, so it still scans many Reviews rows and then filters by time. That extra scanning/maintenance makes it slower than using the paper_id-first path.
- So we don't keep it.

D3 —

Reviews(paper_id, review_timestamp)

Plan used: Covering index scan on Reviews(paper_id, review_timestamp); then join to Authorship via its paper_id index; then to Users.

Effect: Moderate improvement vs ~12.6s baseline → about 10.1s.

Why only modest? The filter is on timestamp, but paper_id is the first key, so MySQL can't do a tight timestamp range seek; it scans the index and filters by time. Being covering avoids table reads, which is where the small win comes from.

Keep? Possible, see Final pick below.

D4 —

Reviews(paper_id, review_timestamp, review_id)

("Index 4" as idx_reviews_paper_ts_id, with Reviews(paper_id) dropped to isolate)

Plan used: Covering index scan on the 3-column index.

Effect: Slower (~12.6s) than D3.

Why? Wider index entries (extra review_id) increase I/O but still can't seek on timestamp (wrong leading column), so pay more to scan roughly the same rows.

Keep? No.

Final index design (selected)

Reviews(paper_id, review_timestamp)

Why this one

- It is actually used by the optimizer in our run.
- It gives a consistent, modest speedup (~12.6s → ~10.1s) by making the Reviews access covering, reducing table reads during the join to Authorship.
- It doesn't bloat the index more than needed (unlike adding review_id).

Why not the others

- Users(is_reviewer): table is small; optimizer prefers a scan. No measurable benefit.
- Reviews(review_timestamp, paper_id): not chosen in plan; any improvement we saw was due to join order, not the index.
- Reviews(paper_id, review_timestamp, review_id): larger index, same scan pattern, slower.

Q4.

-- Lists all papers authored by a given user and reports both:

-- (a) how many total reviews each paper has, and

-- (b) the most recent review timestamp.

-- Allows tracking which of the author’s works are actively discussed or recently reviewed.

```
SELECT
   p.paper_id,
   p.paper_title,
   COUNT(r.review_id) AS review_count,
   MAX(r.review_timestamp) AS last_review_at
FROM Authorship a
JOIN Papers p
   ON a.paper_id = p.paper_id
LEFT JOIN Reviews r
   ON p.paper_id = r.paper_id
WHERE a.user_id = 'U010'
GROUP BY p.paper_id, p.paper_title
ORDER BY review_count DESC, last_review_at DESC
LIMIT 15;
```

Baseline Result:

'{\n \"query\": \"/* select#1 */ select `p`.`paper_id` AS `paper_id`,`p`.`paper_title` AS `paper_title`,count(`r`.`review_id`) AS `review_count`,max(`r`.`review_timestamp`) AS `last_review_at` from `papersdb`.`authorship` `a` join `papersdb`.`papers` `p` left join `papersdb`.`reviews` `r` on((`r`.`paper_id` = `a`.`paper_id`)) where ((`p`.`paper_id` = `a`.`paper_id`) and (`a`.`user_id` = \'U010\')) group by `p`.`paper_id`,`p`.`paper_title` order by `review_count` desc,`last_review_at` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n

\"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"alias\": \"a\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on a using PRIMARY (user_id = \'U010\')\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 70.0,\n \"lookup_condition\": \"user_id = \'U010\'\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"const\"\n],\n \"estimated_total_cost\": 8.818181818181818\n },\n {\n \"alias\": \"p\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on p using PRIMARY (paper_id = a.paper_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"papers\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"paper_title\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.25142857142857145\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 70.0,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 33.31818181818182\n },\n {\n \"alias\": \"r\",\n \"covering\": false,\n \"operation\": \"Index lookup on r using idx_reviews_paper (paper_id = a.paper_id)\",\n \"index_name\": \"idx_reviews_paper\",\n \"table_name\": \"reviews\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n \"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 2.1268656253814697,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.5347547858101981\n }\n],\n \"join_type\": \"left join\",\n \"operation\": \"Nested loop left join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 148.88059377670288,\n \"join_algorithm\": \"nested_loop\",\n

\"estimated_total_cost\": 85.42638964002782\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"review_count DESC\",\n \"last_review_at DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Index 1:

CREATE INDEX idx_reviews_paper_ts ON Reviews(paper_id, review_timestamp);

ANALYZE TABLE Reviews;

EXPLAIN FORMAT=JSON <query>
DROP INDEX idx_reviews_paper_ts ON Reviews;

Result after Index 1:

'{\n \"query\": \"/* select#1 */ select `p`.`paper_id` AS `paper_id`,`p`.`paper_title` AS `paper_title`,count(`r`.`review_id`) AS `review_count`,max(`r`.`review_timestamp`) AS `last_review_at` from `papersdb`.`authorship` `a` join `papersdb`.`papers` `p` left join `papersdb`.`reviews` `r` on((`r`.`paper_id` = `a`.`paper_id`)) where ((`p`.`paper_id` = `a`.`paper_id`) and (`a`.`user_id` = \'U010\')) group by `p`.`paper_id`,`p`.`paper_title` order by `review_count` desc,`last_review_at` desc limit 15\",\n \"query_plan\": {\n \"limit\": 15,\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"alias\": \"a\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on a using PRIMARY (user_id = \'U010\')\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 70.0,\n \"lookup_condition\": \"user_id = \'U010\'\",\n \"index_access_type\": \"index_lookup\",\n

\"lookup_references\": [\n \"const\"\n],\n \"estimated_total_cost\": 8.818181818181818\n },\n {\n \"alias\": \"p\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on p using PRIMARY (paper_id = a.paper_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"papers\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"paper_title\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.25142857142857145\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 70.0,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 33.31818181818182\n },\n {\n \"alias\": \"r\",\n \"covering\": false,\n \"operation\": \"Index lookup on r using idx_reviews_paper (paper_id = a.paper_id)\",\n \"index_name\": \"idx_reviews_paper\",\n \"table_name\": \"reviews\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n \"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 2.1268656253814697,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.5347547858101981\n }\n],\n \"join_type\": \"left join\",\n \"operation\": \"Nested loop left join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 148.88059377670288,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 85.42638964002782\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"review_count DESC\",\n \"last_review_at DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:

'-> Limit: 15 row(s) (actual time=3.54..3.55 rows=15 loops=1)\n   -> Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk (actual time=3.54..3.54 rows=15 loops=1)\n   -> Table scan on <temporary> (actual time=3.41..3.44 rows=70 loops=1)\n -> Aggregate using temporary table (actual time=3.41..3.41 rows=70 loops=1)\n   -> Nested loop left join (cost=85.4 rows=149) (actual time=0.206..2.9 rows=89 loops=1)\n -> Nested loop inner join (cost=33.3 rows=70) (actual time=0.151..2.12 rows=70 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U010\') (cost=8.82 rows=70) (actual time=0.0635..0.149 rows=70 loops=1)\n   -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.0276..0.0277 rows=1 loops=70)\n   -> Index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.535 rows=2.13) (actual time=0.01..0.0105 rows=0.5 loops=70)\n'

Index 2:

CREATE INDEX idx_reviews_paper_ts_id ON Reviews(paper_id, review_timestamp, review_id);

ANALYZE TABLE Reviews;

EXPLAIN FORMAT=JSON <query>

DROP INDEX idx_reviews_paper_ts_id ON Reviews;

Result after Index 1:

'{\n \"query\": \"/* select#1 */ select `p`.`paper_id` AS `paper_id`,`p`.`paper_title` AS `paper_title`,count(`r`.`review_id`) AS `review_count`,max(`r`.`review_timestamp`) AS `last_review_at` from `papersdb`.`authorship` `a` join `papersdb`.`papers` `p` left join `papersdb`.`reviews` `r` on((`r`.`paper_id` = `a`.`paper_id`)) where ((`p`.`paper_id` = `a`.`paper_id`) and (`a`.`user_id` = \'U010\')) group by `p`.`paper_id`,`p`.`paper_title` order by `review_count` desc,`last_review_at` desc limit 15\",\n \"query_plan\": {\n   \"limit\": 15,\n \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"inputs\": [\n   {\n   \"alias\": \"a\",\n   \"covering\": true,\n \"operation\": \"Covering index lookup on a using PRIMARY (user_id = \'U010\')\",\n \"index_name\": \"PRIMARY\",\n   \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n   \"key_columns\": [\n \"user_id\"\n   ],\n   \"schema_name\": \"papersdb\",\n

\"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 70.0,\n \"lookup_condition\": \"user_id = \'U010\'\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"const\"\n],\n \"estimated_total_cost\": 8.818181818181818\n },\n {\n \"alias\": \"p\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on p using PRIMARY (paper_id = a.paper_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"papers\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"paper_title\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.25142857142857145\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 70.0,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 33.31818181818182\n },\n {\n \"alias\": \"r\",\n \"covering\": false,\n \"operation\": \"Index lookup on r using idx_reviews_paper (paper_id = a.paper_id)\",\n \"index_name\": \"idx_reviews_paper\",\n \"table_name\": \"reviews\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n \"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 2.1268656253814697,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.5347547858101981\n }\n],\n \"join_type\": \"left join\",\n \"operation\": \"Nested loop left join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 148.88059377670288,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 85.42638964002782\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n

\"review_count DESC\",\n \"last_review_at DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:
'-> Limit: 15 row(s) (actual time=1.18..1.18 rows=15 loops=1)\n -> Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk (actual time=1.18..1.18 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=1.13..1.14 rows=70 loops=1)\n -> Aggregate using temporary table (actual time=1.13..1.13 rows=70 loops=1)\n -> Nested loop left join (cost=85.4 rows=149) (actual time=0.0799..0.928 rows=89 loops=1)\n -> Nested loop inner join (cost=33.3 rows=70) (actual time=0.0493..0.549 rows=70 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U010\') (cost=8.82 rows=70) (actual time=0.0281..0.0668 rows=70 loops=1)\n -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.00661..0.00665 rows=1 loops=70)\n -> Index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.535 rows=2.13) (actual time=0.00492..0.00515 rows=0.5 loops=70)\n'

Index 3:

CREATE INDEX idx_papers_id_title ON Papers(paper_id, paper_title);

ANALYZE TABLE Papers;

EXPLAIN FORMAT=JSON <query4>
DROP INDEX idx_papers_id_title ON Papers;

Result after Index 3:
'{\n \"query\": \"/* select#1 */ select `p`.`paper_id` AS `paper_id`,`p`.`paper_title` AS `paper_title`,count(`r`.`review_id`) AS `review_count`,max(`r`.`review_timestamp`) AS `last_review_at` from `papersdb`.`authorship` `a` join `papersdb`.`papers` `p` left join `papersdb`.`reviews` `r` on((`r`.`paper_id` = `a`.`paper_id`)) where ((`p`.`paper_id` = `a`.`paper_id`) and (`a`.`user_id` = \'U010\')) group by `p`.`paper_id`,`p`.`paper_title` order by `review_count` desc,`last_review_at` desc limit 15\",\n \"query_plan\": {\n \"limit\": 15,\n \"inputs\": [\n {\n \"inputs\": [\n

{\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"alias\": \"a\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on a using PRIMARY (user_id = \'U010\')\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 70.0,\n \"lookup_condition\": \"user_id = \'U010\'\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"const\"\n],\n \"estimated_total_cost\": 8.818181818181818\n },\n {\n \"alias\": \"p\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on p using PRIMARY (paper_id = a.paper_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"papers\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"paper_title\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.25142857142857145\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 70.0,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 33.31818181818182\n },\n {\n \"alias\": \"r\",\n \"covering\": false,\n \"operation\": \"Index lookup on r using idx_reviews_paper (paper_id = a.paper_id)\",\n \"index_name\": \"idx_reviews_paper\",\n \"table_name\": \"reviews\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n \"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 2.1268656253814697,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.5347547858101981\n }\n],\n \"join_type\": \"left join\",\n \"operation\": \"Nested loop left join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 148.88059377670288,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 85.42638964002782\n }\n

],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"review_count DESC\",\n \"last_review_at DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:
'-> Limit: 15 row(s) (actual time=1.35..1.35 rows=15 loops=1)\n -> Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk (actual time=1.35..1.35 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=1.25..1.28 rows=70 loops=1)\n -> Aggregate using temporary table (actual time=1.24..1.24 rows=70 loops=1)\n -> Nested loop left join (cost=85.4 rows=149) (actual time=0.0709..1.03 rows=89 loops=1)\n -> Nested loop inner join (cost=33.3 rows=70) (actual time=0.0465..0.614 rows=70 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U010\') (cost=8.82 rows=70) (actual time=0.0289..0.0706 rows=70 loops=1)\n -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.00745..0.0075 rows=1 loops=70)\n -> Index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.535 rows=2.13) (actual time=0.00535..0.00561 rows=0.5 loops=70)\n'

Index 4:

CREATE INDEX idx_reviews_paper_id ON Reviews(paper_id, review_id);

ANALYZE TABLE Reviews;

EXPLAIN FORMAT=JSON <query4>

DROP INDEX idx_reviews_paper_id ON Reviews;

Result after Index 4:
'{\n \"query\": \"/* select#1 */ select `p`.`paper_id` AS `paper_id`,`p`.`paper_title` AS `paper_title`,count(`r`.`review_id`) AS `review_count`,max(`r`.`review_timestamp`) AS `last_review_at` from `papersdb`.`authorship` `a` join `papersdb`.`papers` `p`

left join `papersdb`.`reviews` `r` on((`r`.`paper_id` = `a`.`paper_id`)) where ((`p`.`paper_id` = `a`.`paper_id`) and (`a`.`user_id` = \'U010\')) group by `p`.`paper_id`,`p`.`paper_title` order by `review_count` desc,`last_review_at` desc limit 15\",\n \"query_plan\": {\n \"limit\": 15,\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"inputs\": [\n {\n \"alias\": \"a\",\n \"covering\": true,\n \"operation\": \"Covering index lookup on a using PRIMARY (user_id = \'U010\')\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"authorship\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"user_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"user_id\",\n \"paper_id\"\n],\n \"estimated_rows\": 70.0,\n \"lookup_condition\": \"user_id = \'U010\'\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"const\"\n],\n \"estimated_total_cost\": 8.818181818181818\n },\n {\n \"alias\": \"p\",\n \"covering\": false,\n \"operation\": \"Single-row index lookup on p using PRIMARY (paper_id = a.paper_id)\",\n \"index_name\": \"PRIMARY\",\n \"table_name\": \"papers\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"paper_id\",\n \"paper_title\"\n],\n \"estimated_rows\": 1.0,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\": 0.25142857142857145\n }\n],\n \"join_type\": \"inner join\",\n \"operation\": \"Nested loop inner join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 70.0,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 33.31818181818182\n },\n {\n \"alias\": \"r\",\n \"covering\": false,\n \"operation\": \"Index lookup on r using idx_reviews_paper (paper_id = a.paper_id)\",\n \"index_name\": \"idx_reviews_paper\",\n \"table_name\": \"reviews\",\n \"access_type\": \"index\",\n \"key_columns\": [\n \"paper_id\"\n],\n \"schema_name\": \"papersdb\",\n \"used_columns\": [\n \"review_id\",\n \"paper_id\",\n \"review_timestamp\"\n],\n \"estimated_rows\": 2.1268656253814697,\n \"lookup_condition\": \"paper_id = a.paper_id\",\n \"index_access_type\": \"index_lookup\",\n \"lookup_references\": [\n \"papersdb.a.paper_id\"\n],\n \"estimated_total_cost\":

0.5347547858101981\n }\n],\n \"join_type\": \"left join\",\n \"operation\": \"Nested loop left join\",\n \"access_type\": \"join\",\n \"estimated_rows\": 148.88059377670288,\n \"join_algorithm\": \"nested_loop\",\n \"estimated_total_cost\": 85.42638964002782\n }\n],\n \"operation\": \"Aggregate using temporary table\",\n \"access_type\": \"temp_table_aggregate\"\n }\n],\n \"operation\": \"Table scan on <temporary>\",\n \"table_name\": \"<temporary>\",\n \"access_type\": \"table\"\n }\n],\n \"operation\": \"Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk\",\n \"access_type\": \"sort\",\n \"sort_fields\": [\n \"review_count DESC\",\n \"last_review_at DESC\"\n],\n \"per_chunk_limit\": 15\n }\n],\n \"operation\": \"Limit: 15 row(s)\",\n \"access_type\": \"limit\",\n \"limit_offset\": 0\n },\n \"query_type\": \"select\",\n \"json_schema_version\": \"2.0\"\n}'

Result after Explain Analyze:

'-> Limit: 15 row(s) (actual time=1.93..1.93 rows=15 loops=1)\n -> Sort: review_count DESC, last_review_at DESC, limit input to 15 row(s) per chunk (actual time=1.93..1.93 rows=15 loops=1)\n -> Table scan on <temporary> (actual time=1.86..1.88 rows=70 loops=1)\n -> Aggregate using temporary table (actual time=1.85..1.85 rows=70 loops=1)\n -> Nested loop left join (cost=85.4 rows=149) (actual time=0.108..1.55 rows=89 loops=1)\n -> Nested loop inner join (cost=33.3 rows=70) (actual time=0.075..0.996 rows=70 loops=1)\n -> Covering index lookup on a using PRIMARY (user_id = \'U010\') (cost=8.82 rows=70) (actual time=0.042..0.102 rows=70 loops=1)\n -> Single-row index lookup on p using PRIMARY (paper_id = a.paper_id) (cost=0.251 rows=1) (actual time=0.0124..0.0124 rows=1 loops=70)\n -> Index lookup on r using idx_reviews_paper (paper_id = a.paper_id) (cost=0.535 rows=2.13) (actual time=0.00712..0.00746 rows=0.5 loops=70)\n'

Analysis:

Query 4 — Design & Analysis

What the query does

Find all papers authored by a given user (a.user_id='U010') and, for each paper, report:

- total number of reviews, and
- the most recent review time,

then order by review count and last review time.

Baseline behavior (shape of the plan)

- Filter Authorship by user_id via its PRIMARY(user_id, paper_id).
- Join to Papers by PRIMARY(paper_id).
- LEFT JOIN to Reviews by paper_id using the existing idx_reviews_paper(paper_id).
- Aggregate (COUNT(review_id), MAX(review_timestamp)) and sort results.
- No filter on review_timestamp, so the plan must read all reviews per paper anyway.

Index experiments

D1 —

Reviews(paper_id, review_timestamp)

- Expectation: could help MAX(review_timestamp) and keep rows ordered per paper.
- What happened: the optimizer still used idx_reviews_paper(paper_id); plan shape unchanged.
- Effect vs baseline: no meaningful improvement (minor timing noise only).

Why: the join is on paper_id and there is no time-range filter; MySQL prefers the cheaper single-column lookup.

D2 —

Reviews(paper_id, review_timestamp, review_id)

- Expectation: fully covering for both MAX(timestamp) and COUNT(review_id).

- What happened: optimizer still chose idx_reviews_paper(paper_id); plan unchanged.
- Effect vs baseline: no consistent gain; any small speed change is likely caching, not indexing.
- Takeaway: without a time predicate, the composite key doesn't get picked.

D3 —

Papers(paper_id, paper_title)

- Expectation: possibly avoid extra lookups for the selected columns.
- What happened: access to Papers is already via PRIMARY(paper_id); this composite didn't change the plan.
- Effect vs baseline: no benefit.

D4 —

Reviews(paper_id, review_id)

- Expectation: helps counting reviews per paper; not helpful for MAX(timestamp).
- What happened: plan still used idx_reviews_paper(paper_id); even if chosen, it would still need table access for review_timestamp.
- Effect vs baseline: slightly worse in practice; not helpful for this query.

Final index design (selected)

- Keep: existing Reviews(paper_id) (already in use by the optimizer).
- Do not add a new index specifically for this query.

Why this choice

- The query joins Reviews only on paper_id and has no timestamp filter.

- MySQL continues to prefer the simple paper_id lookup and then aggregates; composite indexes with review_timestamp were not used and did not change the plan shape.
- Adding more keys increased index size without a clear gain.

Why not the others

- Reviews(paper_id, review_timestamp[, review_id]): helpful mainly when filter by time or can exploit an index for groupwise max. Here, the engine still scans all reviews per paper, so it didn't help.
- Papers(paper_id, paper_title): the join already uses the PRIMARY key; no extra benefit.
- Reviews(paper_id, review_id): may help counting, but hurts the MAX(timestamp) part (and wasn't chosen anyway).

----------xxx----------

Local Connection with mySql using Terminal: