**Query Optimization**

**Problem statement :** optimize Query in question no2 to finddonation amount, ads spending, total donation, pageview, conversion rate percentage (no of donation/pageview), % spending per donation amount per campaign each day on postgres SQL.

Query

SELECT created\_dt,campaign\_id,title,URL,donation\_amount,total\_donation,

ads\_spending,total\_pageview,(total\_donation/total\_pageview::FLOAT)\*100 AS convertion\_rate,(ads\_spending/donation\_amount::FLOAT)\* 100 AS spending\_per\_donation\_rate

FROM (SELECT don.created\_dt,campaign\_id,title,don.URL,donation\_amount,total\_donation,

SUM(spend) OVER(PARTITION BY don.created\_dt,campaign\_id) AS ads\_spending,

SUM(V.pageview) OVER(PARTITION BY don.created\_dt,campaign\_id) AS total\_pageview

FROM (SELECT date\_trunc('day', D.created) AS created\_dt,D.campaign\_id,C.title,C.URL,

SUM(D.amount) OVER (PARTITION BY date\_trunc('day', D.created),campaign\_id) AS donation\_amount,

COUNT(D.user\_id) OVER (PARTITION BY date\_trunc('day', D.created),campaign\_id) AS total\_donation

FROM donations D

INNER JOIN campaign C ON D.campaign\_id = C.ID) don

LEFT JOIN ads\_spent ads ON don.created\_dt = ads.date\_id AND don.URL = ads.short\_url

LEFT JOIN visit V ON don.created\_dt = V.date\_id AND don.URL = V.campaign\_url

ORDER BY 1,2) FINAL

GROUP BY 1,2,3,4,5,6,7,8 ORDER BY 9;

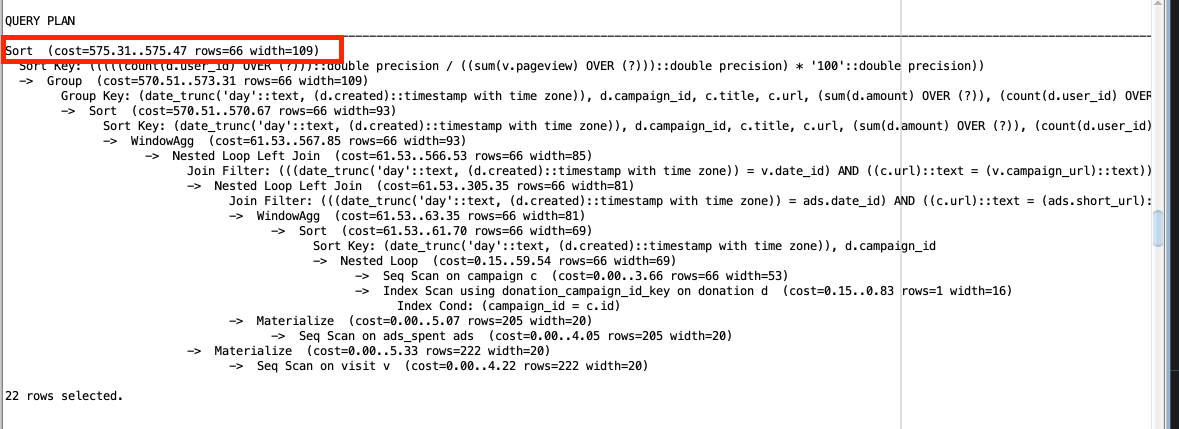
1. Create Index

With creating index prove can speeds up the data retrieval, there several index that we can implemented to optimize the query cost as below :

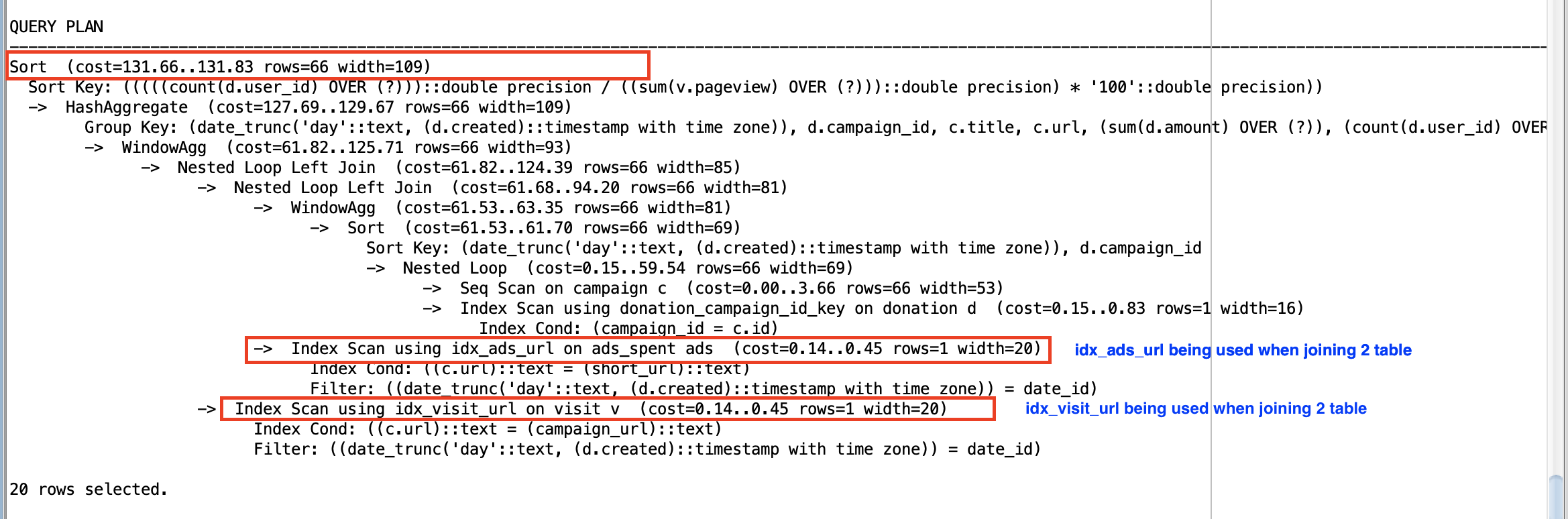
|  |  |  |
| --- | --- | --- |
| Table Name | Index Name | Index Definition |
| ads\_spent | idx\_ads\_url | CREATE INDEX idx\_ads\_url ON ads\_spent USING btree **(short\_url)** |
| campaign | idx\_campaign\_url | CREATE INDEX idx\_campaign\_url ON campaign USING btree (**url**) |
| visit | idx\_visit\_url | CREATE INDEX idx\_visit\_url ON visit USING btree (**campaign**\_**url**) |

Please find below query explain plan before and after we implement index.

**Before :**



**After**



As seen on the image obove that the query cost improve by 77% after index being implemented.