Data Warehousing

Politecnico di Torino

Versioning services

Conceptual design
Logical design

Primary keys are underlined.

GIT (Bid, Tid, Cid, hour, #commits, #additions, #deletions)

BRANCH (Bid, branchname, creationdate, creationyear, repositoryname, company, visibility, category, scala, python, java, ...)
TIME (Tid, date, month, bimester, 4-months, trimester, year, monthofyear, 4-monthsofyear)

COLLABORATOR (Cid, role, yearsexperience, workteam)

Queries

Query A

Consider only private repositories. Separately for month and repository name, analyze: the number of commits made on average in a day, the number of commits made on average in a branch, the monthly cumulative number of commits from the beginning of the year.

SELECT
  repositoryname, month,
  sum(#commits)/count(distinct date),
  sum(#commits)/count(distinct bid),
  sum(sum(#commits)) over (partition by repositoryname, year
  order by month rows unbounded preceding)
FROM
  git g, time t, branch b
WHERE
  g.bid = b.bid and g.tid = t.tid
  AND private = True
GROUP BY
  repositoryname, month, year
Query B

Consider data related to repositories which include the ‘Scala’ language. Separately for branch and work team, analyze: the ratio between the number of additions and the number of deletions, the percentage of additions with respect to the total of the repository the branch belongs to, assign a rank to work teams based on the ratio between the number of additions and the number of deletions, separately for each branch.

```sql
SELECT
    bid, branchname, repositoryname, workteam,
    sum(#additions)/sum(#deletions),
    sum(#additions)/sum(sum(#deletions))
        over (partition by repository, workteam),
    rank() over (partition by bid
        order by sum(#additions)/sum(#deletions))
FROM
    git g, branch b, collaborator c
WHERE
    g.bid = b.bid and g.cid = c.cid
    and scala_language = true
GROUP BY
    bid, branchname, repositoryname, workteam
```