Università degli Studi di Roma “Tor Vergata”
Dipartimento di Ingegneria Civile e Ingegneria Informatica

Project 2

Corso di Sistemi e Architetture per Big Data

A.A. 2017/18

Valeria Cardellini, Matteo Nardelli

Project delivery

- Submission deadline
  - July 6, 2018
  - After the deadline, the maximum achievable score will be decreased by 2 points for each week of delay
- Your presentation
  - July 11, 2018
- What to deliver
  - Link to cloud storage or repository containing the project code
  - *Optional*: project report composed by 4-6 pages in ACM or IEEE proceedings format
  - Slides of your presentation (max. **15 minutes** per group), to be delivered after the presentation
- Team
  - Target: 2 students per team
  - Also possible 1 student or 3 students per team

Dataset

- You will use a real dataset from ACM DEBS 2016 Grand Challenge
- Social analysis
  - Goal: real-time analytics over high volume data streams in the context of graph models
  - Data set available at https://www.dropbox.com/s/vo83ohrgcgfqq27/data.tar.bz2

Dataset

- Data organized in 4 separate streams
  - We will use only 3 of them
- The first input stream indicates when two users enter a “friendship” relationship
  - Contained in friendship.dat
  - Each stream:

    ts, user_id_1, user_id_2

    Example: 2010-02-22T16:53:02.979+0000|2783|4555

Attribute	Description
ts	timestamp indicating when a friendship was established
user_id_1	id of the first user
user_id_2	id of the second user

Dataset

- The second input stream indicates when a user creates a new post
  - Contained in posts.dat
  - Each stream:

    ts, post_id, user_id, post, user

    Example: 2010-02-02T19:53:55.226+0000|299113|4661|photo299113.jpg|Michael Wang

Attribute	Description
ts	timestamp indicating when a post was created
post_id	unique id of the post
user_id	unique id of the user who created the post
post	string containing the post's content
user	string containing the user name of the post creator

Dataset

- The third input stream indicates when a user comments on a post
  - Contained in comments.dat
  - Each stream:

    ts, comment_id, user_id, comment, user, comment_replied, post_commented

    Example: 2010-08-26T10:21:58.862+0000|25770896734|1099511631482|great|Javed Ahmed|25770896731|

Dataset

- The third input stream indicates when a user comments on a post

Attribute	Description
ts	timestamp indicating when a comment was created
comment_id	unique id of the comment
user_id	unique id of the user who created the comment
comment	string containing the comment's content
user	string containing the user name of the comment creator
comment_replied	id of the comment being commented (*empty* if this is a comment to a post)
post_commented	id of the post being commented (*empty* if this is a comment to a comment)

Queries with Storm/Flink

- Use the Apache Storm framework or alternatively the Apache Flink framework
- Include in your report/slides the queries' latency time and throughput on your reference architecture

1. Analyze the friendship relationships that are created in the social network to obtain statistics about the time slot during which friendship relationships are created
   - Query output:
     ts , count_h00 , count_h01 , . . . , count_h22 , count_h23
   - Time window:
     - 24 hours (event time)
     - 7 days (event time)
     - From the beginning

Queries with Storm/Flink

2. Determine the top-10 posts that receive the largest number of (direct) comments
   - Query output:

     ts , post_id_1 , num_comments_1, post_id_2, num_comments_2 , ... , post_id_10, num_comments_10
   - Time window:
     - 1 hour (event time)
     - 24 hours (event time)
     - 7 days (event time)

Queries with Storm/Flink

3. Determine the top-10 users that are the most active in the social network
   - The user total score is given by a + b + c, where:
     - a: number of relationships in which the user is involved
     - b: number of posts created by the user
     - c: number of comments written by the user
   - Order the users according to their total score: if user U_1 has (a_1, b_1, c_1) and user U_2 has (a_2, b_2, c_2), then

     $$U_1 > U_2 \quad \text{iff} \quad a_1 + b_1 + c_1 > a_2 + b_2 + c_2$$
   - Query output:

     ts, user_1, rating_1, user_2, rating_2, ..., user_10, rating_10
   - Time window:
     - 1 hour (event time)
     - 24 hours (event time)
     - 7 days (event time)

Optional part

- Compulsory for team composed of 3 students
- Use Kafka Streams for query 1
- Use the other framework (Flink or Storm) to answer query 2

Queries for the team

- 1 student in the team: queries 1 and 2
- 2 students in the team: all the three queries
- 3 students in the team: all the three queries plus optional part