Outfittery Data Challenge

Below you will find the coding problem we ask you to solve. Please read carefully and implement a solution for it. We don't ask for a production level (over-engineered) solution, but be prepared to explain how you would adapt your solution for a real-life scenario.

Things we value:

- Optimal design decisions.
- A working solution or a prototype.
- A working build process.
- A readme file that contains the information necessary to understand and build the solution.
- Evidence that you have thought about errors and edge case situations, either in the code or in the readme.
- In general, a solution that shows your skills using state of the art big data tools.

We don't set a time limit for the challenge, but it might take you about 8 hours or work to finish it. You can commit your work to a free private repo in GitLab, for example.

The Challenge

You have the task to prepare the data helping analysts to segment the users of the popular Q/A site stackoverflow. You can use the available data (https://archive.org/download/stackexchange), for example you can start with a German site file (german.stackexchange.com.7z). Note that there is a README file available explaining the data structure. Treat the comments and posts history as streaming "events" and note that posts therefore contain different revisions of the same entity. You can also try to load the data by parts, if you prefer.

For further analysis, please provide a published **Users** table (in any relational DB or Hive table) with some additional columns per user (**please only choose and implement a couple of them, to save your time!**):

- Total number of posts created
- Creation date of the last post which has comments linked to it
- Number of posts in the last 30 days
- Average number of comments per month
- Two separate columns for badges "Critic" and "Editor" with boolean flags (1/0 or true/false), i.e. "is_critic" and "is_editor"

Don't hesitate to ask for clarification if necessary.

Good luck!

P.S.: If you are using spark, take into account such important issues:

- https://github.com/databricks/spark-xml/issues/109