PS Non-Standard Database Systems

Summer term 2020

Checkpoint 01

Planing Phase

Due date: Wed, May 20, 2020

General

Submit your checkpoint report until **Wed, May 20, 2020** using our submission system [1]. You may repeatedly upload new versions: only the **latest submission** is assessed.

Support

The preferred way of communication is the Slack channel `#nsdb-ps-2020ss` in the `dbteaching` workspace[2].

You may use the channel to get support if the instructions are unclear or if you run into problems in the course of your project. The channel is also open for topic-related discussions among students.

1 Task Description

For this checkpoint, you will specify the project you are going to work on throughout this semester. This includes (1) a survey of different types of non-standard database systems, (2) the choice of the database/processing system you would like to work with, (3) providing a concise, motivating application that is a good fit for the chosen database/processing system, (4) finding data sets to process and/or load into your database, and (5) a roadmap for your project (*optional*).

[1] `https://abgaben.cosy.sbg.ac.at`

[2] `https://dbteaching.slack.com`

1.1 Survey

The first subtask is to explore literature and other trustworthy (online) resources for information on different non-standard database systems. In our context, *non-standard* database systems include all databases except relational, disk-based systems like PostgreSQL. Non-standard database systems also include processing frameworks that are closely related to databases (e.g., Big Data processing systems). After this literature survey, you should have a good overview of available non-standard database systems.

Enumerate *four* references in the report (0.125 points each; max. 0.5 points in total) and briefly summarize the main insight they provided (0.125 points each; max. 0.5 points in total).

Note: You may want to check out the suggested readings section on our course website as a starting point for your investigation. Especially the *overview on non-standard database systems* may be helpful to get an overview on available *types of database/processing systems*.

1.2 Choose a System

After reviewing the literature and (some of) the respective database/processing systems, you will choose one (open-source/freely-available) system to base your project on.

Motivate your choice: Why do you find the chosen database system interesting? Also name and discuss *four* key features/properties of your system (0.25 points each; max. 1 point in total). Please focus on the underlying database technology rather than, for example, tools for designing nice user interfaces.

1.3 Application Description

Now that you have chosen a database system, you need a suitable application that benefits from this kind of system. Try to come up with such an application. This can be anything, for example, an everyday problem you always wanted to solve or a problem you encountered at work. However, make sure it is meaningful and not just a dummy application. Try to define the workloads your application has to deal with (e.g., OLTP, OLAP, ...). Describe your application (1 point) and justify why this application/workload fits your database/processing system of choice (1 point).

Architectural Overview If possible, also provide an architectural overview: Which programming language will be used? Do you plan to use any additional frameworks/libraries (e.g., for visualization)? If this is the case, briefly describe their role in your application pipeline. Will you deploy your application in a real/simulated distributed environment?

Experimental Data You will also need some data to test and evaluate your application. Basically, you have two choices here:

1. Find data; most probably online (e.g., Twitter [3], Kaggle [4], Google Data Set Search [5], ...)
2. Generate synthetic data that fits your needs.

In both cases, you are required to describe important properties of the experimental data and why this data set is good to test and evaluate your application (1 point). If you find multiple interesting data sets, you can receive a bonus point (1 bonus point).

[3] https://developer.twitter.com/en/docs.html

[4] https://www.kaggle.com/

[5] https://toolbox.google.com/datasetsearch

You are not required to use data sets that are too large for a single machine. However, for the sake of a meaningful evaluation of your application, the data sets should be large enough to challenge the systems.

1.4 Roadmap

Note: This subtask is optional. However, you can earn a bonus point if you work on this subtask.

Provide a roadmap for your project (1 bonus point): Think about the actual implementation and try to identify necessary steps in order to reach the goal. The roadmap does *not* necessarily need to state explicit dates. In any case, you will benefit from planning your project before you start working on it.

2 Grading Scheme

	Category	**Max. points**
1.1	References	0.5 (0.125 each)
	Short summaries	0.5 (0.125 each)
1.2	Key features of chosen database/processing system	1.0 (0.25 each)
1.3	Description of application	1.0
	Reasoning	1.0
	Description of experimental dataset	1.0
Bonus	Additional datasets	1.0
	Roadmap	1.0
	Max. points	**5+2**

3 Feedback

You can help us to improve this class (even for the current semester). Therefore, please answer the following questions.

- How much time did each of the group members spend on this assignment? Please use the Effort Collector tool [6] to anonymously answer this question; you find the access data in the slack channel of the course.
- Are there any hints/references we should provide for future students? Did you find any of our guidance misleading or ambiguous?
- Do you have any suggestions for the instructors to support students more effectively?
- Any other comments?

Note: Feedback is optional. Your answers will have no impact on the grade.

[6] https://aufwand.cosy.sbg.ac.at