database
research group

PS Non-Standard Database Systems

Summer term 2018

Checkpoint 02

Implementation

Due date: Friday, 2018-05-25, 23:55

---

1 General

Submit your checkpoint report until **Friday, 2018-05-25, 23:55** using our submission system [1]. Please be aware that only the **last** submission is assessed.

1.1 Support

If there are any ambiguities or problems of understanding regarding the checkpoint, you have the following possibilities to clarify them:

1. Slack channel: **#nsdb** [2] (preferred way of communication)
2. Office hours: Wednesday, 10AM - 11AM, room 0.26 (ground floor)

If you run into a problem, first, *try to resolve it yourself (as a group)*. If the problem remains unresolved, you should use one of the above possibilities *in time* in order to allow best possible support by the instructor.

2 Task Description

This checkpoint consists of two parts: (1) the actual implementation of the project specified in the previous checkpoint and (2) the corresponding report. In the report, you are required to provide precise information on

1. helpful resources on techniques/systems/tools/. . . you use,
2. how to set up a machine such that your application can be deployed,

[1] `https://abgaben.cosy.sbg.ac.at`
[2] `https://dbteaching.slack.com`

3. how your dataset import/generation is implemented,
4. the implementation of the application itself,
5. problems encountered on the way (*not graded*), and
6. an alternative implementation based upon a relational database system (*optional*)

2.1 Resources

Most probably, you will consult some (online) resources while you are implementing your project. Reference at least *four* and briefly discuss in which respect the respective resources were useful.

Note: For future development of this class, it would be helpful to the instructor if you list *all* resources that were useful to you (even without a description of the reference).

2.2 Setup

Describe your setup thoroughly. This subtask is supposed to provide precise information about the architecture/pipeline of your project.

Essentially, this section serves as a documentation of your setup. Using this documentation, it should be possible to reproduce your setup and experiments.

Write down important facts of the (virtual) machine you are using, for example, the operating system, the prerequisites in order to reproduce your setup on a different machine (third-party tools/libraries, . . .). If you adapted configuration files, write down your changes to these files (provide code snippets of the changes) and why the respective adaptation was necessary.

2.3 Datasets

If you found additional or more interesting datasets than the dataset(s) described in the previous checkpoint, you may use them. However, describe the characteristics of the new/additional dataset(s) in your report (as required for the previous checkpoint).

Analogously, if you generate synthetic datasets on your own, you may adjust the dataset generation process described in the previous checkpoint. In this case, describe and justify your changes in the report.

2.3.1 Generation

Note: This part only applies to groups that generate their own synthetic datasets.

Explain in detail how your dataset generation is implemented (provide code snippets). This includes (but is not limited to)

- the programming languages/tools you use, and
- the enforcement of the characteristics your dataset is required to satisfy.

2.3.2 Import

Describe the process of importing the dataset(s) (provide code snippets). Did you apply any transformations before the data is stored in the underlying database system? Which information is (not) stored and how is the information represented? For example, if the data would be stored in a traditional relational database system, you would define the schema of the relation(s) here.

For static datasets (rather than stream-based datasets): How long did it take you to import the dataset(s)? Did you put some effort into optimizing the import process? If so, what optimizations did you apply and which approach did (not) improve the import process?

2.4 Implementation

Describe all important parts of the actual implementation of your application (provide code snippets). This section is supposed to constitute the majority of the report (this is also reflected in the grading scheme).

2.4.1 Key System Features

Discuss at least two key features of your underlying (database) system that you use to implement (or optimize) your application. For example, indexes: type of the index, purpose of the index, and what is indexed?

To retrieve a bonus point, discuss at least two other (system-specific) features you found useful, e.g., views, replication/sharding, transactions, specialized algorithms/data structures, aggregations, ...

2.5 Problems Encountered

Note: This part does not influence your grade whatsoever.

If you had to make design decisions, summarize them here and justify your decision(s) briefly, e.g., by exposing possible trade-offs. If any, state problems you encountered while implementing your application/working with your system of choice. Briefly describe how you resolved the problem(s).

2.6 Alternative Implementation

Note: This part is optional. However, you will receive a bonus point if you work on this subtask.

Based on the application description in the previous checkpoint: try to come up with an alternative implementation of your application that is based on a different (database/processing) system and describe/specify the most important parts of it. Compare the alternative implementation to your implementation and discuss the main differences. This includes (but is not limited to) limitations, advantages/disadvantages, and any other interesting aspect with respect to performance, scalability, flexibility, . . . In any case, focus on the underlying (database) systems and their characteristics.

This subtask is *conceptual* (rather than pratical). Although you are free to implement your alternative approach, you are *not* required to do so.

3 Grading

	Category	**Max. points**
2.1	Resources	0.25 each (max. 1)
2.2	Setup description/documentation	2
2.3	Dataset import process	2
2.4	Implementation details	3
	Key system features	1 each (max. 2)
Bonus	Additional system features	0.5 each (max. 1)
	Alternative implementation	1
	Max. points	**10 + 2**

4 Feedback

Note: Answering the feedback questions is optional.

You can help us to improve this class (even for this semester). Therefore, you are asked to answer the following questions:

- Was this checkpoint too easy/hard in any regard? Did it take too much/little time? Please indicate the average time per group member that was spent working on this checkpoint. The time that you indicate will have *no* impact on your grade.
- Are there any hints/references we should give future students? Conversely, did you find any of our guidance misleading?
- Do you have any suggestions for the instructors to more effectively support students?
- Any other comments?

If you would like to provide your feedback anonymously, you may also answer these questions in the evaluation at the end of the semester.