NATIONAL RESEARCH UNIVERSITY

Faculty of Computer Science

SQLITE RDBMS EXTENSION FOR DATA INDEXING USING B-TREE MODIFICATIONS

Bachelor Thesis Project Proposal

Student: Anton Rigin, group BSE153
Academic Supervisor: Sergey Shershakov,
Senior Lecturer at School of Software Engineering

Moscow, 2019

OUTLINE

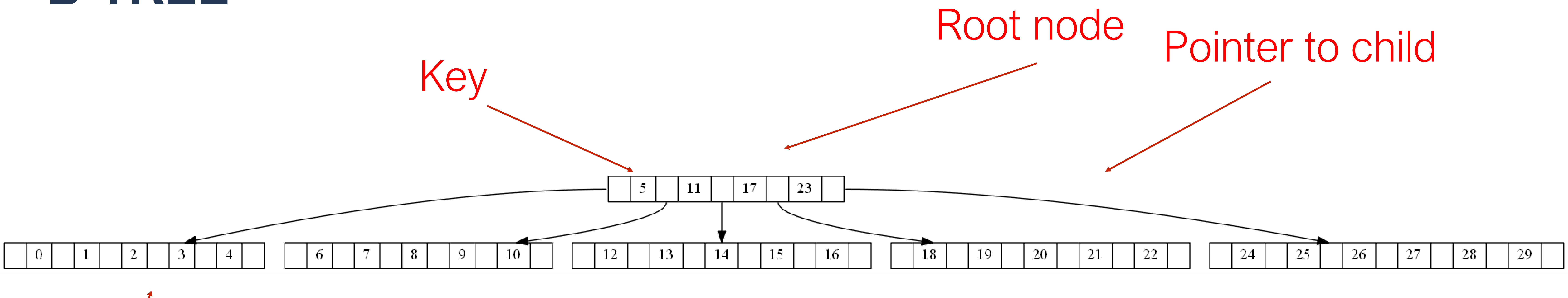
- B-tree and modifications
- SQLite and extensions
- B-trees modifications C++ library
- Selecting the best index structure
- SQLite GUI managers

B-TREE

- Balanced search tree [1]
- Nodes may contain more than 1 key and more than 2 pointers to the children nodes [1]
- If some node contains k keys than it contains $k + 1$ pointers to the children nodes [1]
- **B-tree order** is such a t number that:
  - ✓for each non-root node: $t - 1 \leq k \leq 2t - 1$
  - ✓for root node in the non-empty tree: $1 \leq k \leq 2t - 1$
  - ✓for root node in the empty tree: $k = 0$ [1]
- Usually used as the data index [1]

[1] R. Bayer and E. McCreight, "Organization and maintenance of large ordered indexes," *Acta Informatica*, vol. 1, no. 3, pp. 173 – 189, 1972.

B-TREE



Key

Root node

Pointer to child

Leaf node

The B-tree example, t = 6

B-TREE MODIFICATIONS

- **B^+-tree**
  - ✓Only leaf nodes contain real keys (real data), other nodes contain router keys [2]
  - ✓Deletion is probably faster than in B-tree
- **B^*-tree**
  - ✓Each node (except of the root node) is filled at least by 2/3 not 1/2 [3]
  - ✓Keys insertion in B^*-tree is expected to be faster than in B-tree
- **B^{*+}-tree**
  - ✓Combines the main B^+-tree and B^*-tree features together

[2] K. Pollari-Malmi. (2010). B+-trees [PDF paper]. Available: https://www.cs.helsinki.fi/u/mluukkai/tirak2010/B-tree.pdf

[3] "B^*-tree." NIST Dictionary of Algorithms and Data Structures. Available: https://xlinux.nist.gov/dads/HTML/bstartree.html (accessed Dec. 24, 2018).

SQLITE EXTENSIONS

- Popular open-source embedded relational DBMS
- Written in the C language
- Uses the B-tree as the default index
- SQLite extensions are the dynamically linked libraries [4]

[4] “Run-Time Loadable Extensions.” SQLite.org. Available: https://www.sqlite.org/loadext.html (accessed Jan. 20, 2019).

B-TREES MODIFICATIONS C++ LIBRARY

- Developed previously [5]
- Contains B-tree, B$^+$-tree, B*-tree and B^{*+}-tree implementations
- In the current work connected to the SQLite as the **run-time loadable extension**

[5] A. Rigin, “On the Performance of Multiway Trees in the Problem of Structured Data Indexing,” (in Russian), coursework, Dept. Soft. Eng., HSE, Moscow, Russia, 2018.

ALGORITHM OF SELECTING THE BEST INDEX STRUCTURE

- Uses the multiclass linear classification:

$$y_i = w_i^1 rows\_count + w_i^2 columns\_count + w_i^3 index\_size + \beta_i,$$

$$i = 1, 2, 3, 4$$

- The *i*-th classifier's tree is chosen as index structure for the table if and only if: $y_i = max\{y_1, y_2, y_3, y_4\}$
- The classifiers are trained using the Python 2 and the Sci-Kit Learn library

SQLITE GUI MANAGERS

SUMMARY

- **B-tree** is the balanced search tree for data indexing
- **B$^+$-tree, B*-tree and B^{*+}-tree** are the B-tree modifications
- **SQLite** RDBMS supports adding new features using **extensions**
- **B-trees modifications library** is connected to the SQLite as an extension
- **The best index structure** is selected using multiclass linear classification
- **SQLite GUI managers** support developing plugins

REFERENCES

[1] R. Bayer and E. McCreight, “Organization and maintenance of large ordered indexes,” *Acta Informatica*, vol. 1, no. 3, pp. 173 – 189, 1972.

[2] K. Pollari-Malmi. (2010). “B$^+$-trees,” *University of Helsinki*. [PDF paper]. Available: https://www.cs.helsinki.fi/u/mluukkai/tirak2010/B-tree.pdf

[3] “B*-tree.” NIST Dictionary of Algorithms and Data Structures. Available: https://xlinux.nist.gov/dads/HTML/bstartree.html (accessed Dec. 24, 2018).

[4] “Run-Time Loadable Extensions.” SQLite.org. Available: https://www.sqlite.org/loadext.html (accessed Jan. 20, 2019).

[5] A. Rigin, “On the Performance of Multiway Trees in the Problem of Structured Data Indexing,” (in Russian), coursework, Dept. Soft. Eng., HSE, Moscow, Russia, 2018.

HIGHER SCHOOL OF ECONOMICS

NATIONAL RESEARCH
UNIVERSITY

Thank you for your attention!

amrigin@edu.hse.ru
anton19979@yandex.ru
anton19979@yandex-team.ru