Database Systems on Modern CPU Architectures

Adrian Riedl, Philipp Fent

Technical University of Munich

Chair for Database Systems
Lecture Overview

Database Systems on Modern CPU Architectures
Lecture Overview

Database Systems on Modern CPU Architectures

Database Systems and Modern CPU Architectures
Lecture Overview

Implementation of Database Systems
Lecture Overview

Implementation of Database Systems

(on Modern CPU Architectures)
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

2. Efficient Query Processing
   2.1. Set oriented query processing
   2.2. Algebraic operators
   2.3. Code generation
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

2. Efficient Query Processing
   2.1. Set oriented query processing
   2.2. Algebraic operators
   2.3. Code generation

3. Designing a DBMS for Modern Hardware
   3.1. Re-designing storage
   3.2. Optimizing cache locality
   3.3. Main memory databases
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

   External Sorting
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

   - External Sorting
   - Buffer Manager
Lecture Overview

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

Implementation of Database Systems

- External Sorting
- Buffer Manager
- Slotted Pages
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

   B+-Tree
Lecture Overview

Implementation of Database Systems

1. The Classical Architecture
   1.1. Storage
   1.2. Access paths
   1.3. Transactions & recovery

2. Efficient Query Processing
   2.1. Set oriented query processing
   2.2. Algebraic operators
   2.3. Code generation
<table>
<thead>
<tr>
<th>Lecture Overview</th>
<th>Implementation of Database Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The Classical Architecture</td>
<td>2. Efficient Query Processing</td>
</tr>
<tr>
<td>1.1. Storage</td>
<td>2.1. Set oriented query processing</td>
</tr>
<tr>
<td>1.2. Access paths</td>
<td>2.2. Algebraic operators</td>
</tr>
<tr>
<td>1.3. Transactions &amp; recovery</td>
<td>2.3. Code generation</td>
</tr>
</tbody>
</table>
Exercises

- Sessions: Tuesdays 15:30 – 17:00
- Programming assignments every 2 weeks, starting today
- Announcements on website & Mattermost
- Implementation assignment tasks on GitLab
  - Submit via git
  - Due two weeks later, Tuesdays @15:30
- No Teams. We will check for copied code!
- Bonus System:
  - .3/.4 grade bonus on final exam (≥ % exercises passed)
  - Passed: Green GitLab CI (build, lint, test)
  - Fail: CI pipeline failed, skipped/disabled tests
GitLab & Mattermost

● Register:  https://gitlab.db.in.tum.de/
● Join Group:  https://gitlab.db.in.tum.de/moderndbs-2023
● Fork first task External Sort
● Clone & Push your solution
● Announcements / Questions:
  https://mattermost.db.in.tum.de/moderndbs23