

## 3. Übungsblatt

Ausgabe: 20. Oktober 2014 · Besprechung: 3. und 5. November 2014

### 1 Verdrängungsstrategien — Implementation

Implementieren Sie eine Verdrängungsstrategie Ihrer Wahl (mit einer Programmiersprache Ihrer Wahl).

Schreiben Sie Ihr Programm so, dass es aus einer Datei oder von der Konsole jeweils `pin()`- und `unpin()`-Aufrufe entgegennimmt. Das Programm soll dann Buch führen und jeweils entscheiden, welche Seite gegebenenfalls verdrängt wird.

Jede Verdrängungsstrategie wird in der Praxis zu einer gewissen Hitrate führen. Ausgabe Ihres Programms soll sein, wie viele `pin()`-Aufrufe ein “Hit” waren und für wie viele Aufrufe die Seite neu von der Festplatte geladen werden musste.

### 2 Buffer Manager und Strategien in der Praxis

Auf der Vorlesungswebseite finden Sie den Sourcecode für eine modifizierte Version von PostgreSQL 9.3.1. Dabei wurde der *Buffer Manager* so verändert, dass er `pin()`- und `unpin()`-Aufrufe jeweils protokolliert.<sup>1</sup>

1. Installieren Sie die so modifizierte PostgreSQL-Version auf Ihrem System. Aktivieren Sie das Protokollieren von Meldungen ab dem `DEBUG1`-Level, um die genannten Aufrufe sichtbar zu machen.
2. Modifizieren Sie Ihr Programm aus Aufgabe 1 so, dass es die `pin()`- und `unpin()`-Sequenz von PostgreSQL interpretieren und simulieren kann.
3. Lassen Sie Anfragen des TPC-H-Benchmarks im resultierenden System laufen. Untersuchen Sie die Effektivität (Hitrate) Ihrer gewählten Verdrängungsstrategie.

---

<sup>1</sup>Tatsächlich finden Sie nur die modifizierte Datei `bufmgr.c`. Diese wurde auf dem Webserver als `bufmgr.txt` abgelegt (weil das Content Management System des ITMC leider keine `.c`-Dateien erlaubt). Tauschen Sie diese bitte einfach in den Original-Quellen aus.