

**Managing Massive Multiplayer Online Games**  
SS 2013

**Übungsblatt 4: Persistenz**

Besprechung: 6.05.2013 und 07.06.2013

**Aufgabe 4-1**     *Persistenz*

Betrachten Sie im Folgenden ein abstraktes Spiel, bei dem Spielinformationen serverseitig gespeichert werden. Nehmen Sie dabei an, dass Daten auf dem Server in Objekten  $O_1, \dots, O_3$  gespeichert werden. Zu Beginn hat jedes Objekt  $O_i$  den Wert  $o_i$ . Um den Server vor Datenverlust bei einem Systemfehler zu schützen, sollen beginnend bei  $t_{10}$  alle zehn 10 Ticks Spielinformation persistent auf Festplatte gespeichert werden. Nehmen Sie an, dass das Ausschreiben eines Objektes auf Festplatte zwei Ticks Zeit benötigt.

Der Server führt dabei folgende Änderungen an der Datenbasis durch:

Zeitpunkt	Seite	Neuer Wert
$t_6$	$O_1$	$o'_1$
$t_9$	$O_2$	$o'_2$
$t_{12}$	$O_3$	$o'_3$
$t_{15}$	$O_1$	$o''_1$
$t_{16}$	$O_3$	$o''_3$
$t_{22}$	$O_2$	$o''_2$
$t_{22}$	$O_3$	$o'''_3$

- (a) Skizzieren Sie, wie der Logging Algorithmus *Naive Snapshot* abläuft.
- (b) Skizzieren Sie, wie der Logging Algorithmus *Copy-on-Update* abläuft.
- (c) Skizzieren Sie, wie der Logging Algorithmus *Wait-Free Zigzag* abläuft.
- (d) Skizzieren Sie, wie der Logging Algorithmus *Wait-Free Ping-Pong* abläuft.
- (e) Diskutieren Sie Vor- und Nachteile dieser Verfahren.