Inhaltsverzeichnis

Gliederung

Grundbegriffe

Thread, Nicht-Determinismus, kritische Bereiche, Sperren

Verifikation

Zeitliche Abläufe, serielle Abläufe, faire Mischung, Sicherheits- und Liveness-Eigenschaften, Modellierung

Synchronisation

Signale, Beispiel: Erzeuger/Verbraucher (1), Semaphore, Beispiel: Erzeuger/Verbraucher (2), bedingte kritische Bereiche, Beispiel: Erzeuger/Verbraucher (3), wiederbetretbare Sperren, Leser/Schreiber-Problem

Feinkörnige Nebenläufigkeit

Methoden, Beispiel: Mengen, grobkörnig, Beispiel: Mengen, feinkörnig, Beispiel: Mengen, optimistisch, Beispiel: Mengen, faul

Implementierung

Atomare Befehle, Konsenszahlen, Zwischenspeicher, Bäckerei-Algorithmus

Transactional Memory

Probleme mit Sperren, Transaktionen, Software Transactional Memory (STM): Transaktionsstatus, Transactional Thread, 2 Implementierungen

Literatur

Maurice Herlihy, Nir-Shavit: The Art of Multiprocessor Programming (Morgan Kaufmann, 2008)

Kalvin Lin, Larry Snyder: Principles of Parallel Programming (Addison Wesley)

Greg Andrews: Concurrent Programming (Addison Wesley, 1991)

Brian Goetz, u.a.: Java Concurrency in Practice (Addison Wesley)

1 Grundbegriffe

1.1 Threads

Definition 1 (Prozess)**.** *Sequentieller Rechenvorgang*

Definition 2 (sequentiell)**.** *Alle Rechenschritte laufen nacheinander in einer vorgegebenen Reihenfolge ab.*

Definition 3 (Thread)**.** *„leichte“ Variante eines Prozesses*

Allgemeine Tendenz:

1. Systemkern möglichst „schlank“ halten
2. Systemkern möglichst selten betreten

Unterschied zu Prozess:

- Kein eigener Speicherbereich
- Üblicherweise nicht vom Systemkern verwaltet („leight-weight process“), vom Systemkern verwaltet

Vorteile:

- Wechsel zwischen Threads weniger aufwändig als Wechsel zwischen Prozessen
- Threads benötigen weniger Speicher
- Man kann viel mehr Threads (≈ 10.000) als Prozesse (≈ 100) laufen lassen.

Nachteil:
Anwendungsprogrammierer muss sich um Verwaltung der Threads kümmern.

2 Verifikation

3 Synchronisation

4 Feinkoernige Nebenlaeufigkeit

5 Implementierung

6 Transactional Memory