Laborbericht Container:

Ein Docker-Container ist eine isolierte Umgebung, in der eine Anwendung innerhalb eines Docker-Images ausgeführt wird. Ein Docker-Image ist eine Vorlage zur Erstellung von Containern. Es enthält alle Komponenten, die benötigt werden, um eine Anwendung oder einen Service lauffähig zu machen z.B. Betriebssystem, die Laufzeitumgebung, die Bibliotheken, die Anwendungsdateien, die Konfigurationsdateien).   
Der Container verwendet das zugrundeliegende Betriebssystem des Hosts und teilt dessen Kernel mit anderen Containern und dem Host-Betriebssystem.

Ein Docker-Container aus einem Docker-Image erstellt, indem es gestartet und ausgeführt wird. Jeder Container läuft in einer isolierten Umgebung, in der er seine eigenen Prozesse, Dateisysteme und Netzwerkschnittstellen hat.

Vorbereitung:

Für die Durchführung des Praktikums ist zuerst die Installation von Docker nötig. Es wurde Version 24.0.2 verwendet. Insgesamt steht der virtuellen Ubuntu Maschine, auf der Docker installiert wurde ca 3GB Speicher zur Verfügung.

Als Base-Image wurde die aktuellste Ubuntu-Version verwendet.

Anschließend wird ein Programm erstellt, dass Speicher allokiert und diesen verwendet. In diesem Praktikum wurde ein Long-Long Array verwendet. Jedes Feld im Array belegt etwa 8 Byte. Das Array ist 67.108.864 Felder groß (insgesamt ca 500MB). Das Programm benötigt allerdings zusätzlichen Speicher für andere Operationen. Deswegen ist der Gesamtbedarf etwas größer als 500MB.

Durchführung:

Das Programm wurde jeweils 5 Mal in einem Container ausgeführt, um die mittlere Laufzeit zu ermitteln. Dabei wurden verschiedene Speicher-Limitierungen verwendet (s. Tabelle). Swap wurde auf 1GB gesetzt.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Memory  Limit (MB)** | **Durchschnitt Runtime (ms)** | **Time Run 1** | **Time Run 2** | **Time Run 3** | **Time Run 4** | **Time Run 5** |
| **0** | 436,2 | 470 | 419 | 422 | 419 | 451 |
| **50** | 5999,6 | 6195 | 5910 | 5782 | 5894 | 6217 |
| **100** | 5412,2 | 5252 | 5392 | 5292 | 5588 | 5537 |
| **150** | 4948,4 | 4878 | 5081 | 4944 | 5020 | 4819 |
| **200** | 4371,6 | 4357 | 4246 | 4345 | 4482 | 4428 |
| **250** | 3696,8 | 3469 | 4026 | 3563 | 3818 | 3608 |
| **300** | 3322,6 | 3291 | 3320 | 3330 | 3347 | 3325 |
| **350** | 2785 | 3023 | 2947 | 2574 | 2749 | 2632 |
| **400** | 2225,4 | 2253 | 2052 | 2045 | 2205 | 2572 |
| **450** | 1551,8 | 1811 | 1453 | 1517 | 1479 | 1499 |
| **500** | 755,8 | 785 | 808 | 736 | 658 | 792 |
| **550** | 522,8 | 478 | 512 | 513 | 550 | 561 |

Die Belegung des Speichers wurde mit *docker stats* überwacht.

Tabelle1: ohne Speicher limitierung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Container ID** | **Memory** | **CPU** | **Memory** | **Block I/O** |
| b2a72ba3abd2 | 1.012MiB / 2.84GiB | 0.00% | 0.03% | 573kB / 0B |
| b2a72ba3abd2 | 1.012MiB / 2.84GiB | 0.01% | 0.03% | 573kB / 0B |
| b2a72ba3abd2 | 23.94MiB / 2.84GiB | 0.66% | 0.82% | 573kB / 0B |
| b2a72ba3abd2 | 39.23MiB / 2.84GiB | 0.74% | 1.35% | 573kB / 0B |
| b2a72ba3abd2 | 62.16MiB / 2.84GiB | 0.63% | 2.14% | 573kB / 0B |
| b2a72ba3abd2 | 77.45MiB / 2.84GiB | 0.62% | 2.66% | 573kB / 0B |
| b2a72ba3abd2 | 100.4MiB / 2.84GiB | 0.65% | 3.45% | 573kB / 0B |
| b2a72ba3abd2 | 115.7MiB / 2.84GiB | 0.59% | 3.98% | 573kB / 0B |
| b2a72ba3abd2 | 138.6MiB / 2.84GiB | 0.62% | 4.77% | 573kB / 0B |
| b2a72ba3abd2 | 161.5MiB / 2.84GiB | 0.67% | 5.56% | 573kB / 0B |
| b2a72ba3abd2 | 176.8MiB / 2.84GiB | 0.61% | 6.08% | 573kB / 0B |
| b2a72ba3abd2 | 199.8MiB / 2.84GiB | 0.73% | 6.87% | 573kB / 0B |
| b2a72ba3abd2 | 215.1MiB / 2.84GiB | 0.67% | 7.40% | 573kB / 0B |
| b2a72ba3abd2 | 238MiB / 2.84GiB | 0.64% | 8.18% | 573kB / 0B |
| b2a72ba3abd2 | 253.3MiB / 2.84GiB | 0.80% | 8.71% | 573kB / 0B |
| b2a72ba3abd2 | 276.2MiB / 2.84GiB | 0.70% | 9.50% | 573kB / 0B |
| b2a72ba3abd2 | 291.5MiB / 2.84GiB | 0.62% | 10.02% | 573kB / 0B |
| b2a72ba3abd2 | 314.4MiB / 2.84GiB | 0.87% | 10.81% | 573kB / 0B |
| b2a72ba3abd2 | 329.7MiB / 2.84GiB | 0.65% | 11.34% | 573kB / 0B |
| b2a72ba3abd2 | 352.6MiB / 2.84GiB | 0.66% | 12.13% | 573kB / 0B |
| b2a72ba3abd2 | 367.9MiB / 2.84GiB | 0.67% | 12.65% | 573kB / 0B |
| b2a72ba3abd2 | 390.9MiB / 2.84GiB | 0.66% | 13.44% | 573kB / 0B |
| b2a72ba3abd2 | 413.8MiB / 2.84GiB | 0.64% | 14.23% | 573kB / 0B |
| b2a72ba3abd2 | 429MiB / 2.84GiB | 0.67% | 14.75% | 573kB / 0B |
| b2a72ba3abd2 | 452.1MiB / 2.84GiB | 0.65% | 15.55% | 705kB / 0B |
| b2a72ba3abd2 | 467.3MiB / 2.84GiB | 0.68% | 16.07% | 705kB / 0B |
| b2a72ba3abd2 | 490.2MiB / 2.84GiB | 0.68% | 16.86% | 705kB / 0B |
| b2a72ba3abd2 | 505.5MiB / 2.84GiB | 0.70% | 17.38% | 705kB / 0B |

Tabelle2: Memory auf 200MB limitiert (Swapping unbegrenzt)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Container ID** | **Memory** | **CPU** | **Memory** | **Block I/O** |
| 9a9ac0f1629e | 1.277MiB / 200MiB | 0.00% | 0.64% | 868kB / 0B |
| 9a9ac0f1629e | 1.277MiB / 200MiB | 0.01% | 0.64% | 868kB / 0B |
| 9a9ac0f1629e | 24.21MiB / 200MiB | 0.64% | 12.10% | 868kB / 0B |
| 9a9ac0f1629e | 39.5MiB / 200MiB | 0.60% | 19.75% | 868kB / 0B |
| 9a9ac0f1629e | 62.43MiB / 200MiB | 0.67% | 31.21% | 868kB / 0B |
| 9a9ac0f1629e | 77.72MiB / 200MiB | 0.60% | 38.86% | 868kB / 0B |
| 9a9ac0f1629e | 100.6MiB / 200MiB | 0.61% | 50.32% | 868kB / 0B |
| 9a9ac0f1629e | 115.9MiB / 200MiB | 0.61% | 57.97% | 868kB / 0B |
| 9a9ac0f1629e | 138.9MiB / 200MiB | 0.66% | 69.44% | 868kB / 0B |
| 9a9ac0f1629e | 154.3MiB / 200MiB | 0.58% | 77.13% | 868kB / 0B |
| 9a9ac0f1629e | 177.1MiB / 200MiB | 0.63% | 88.55% | 868kB / 0B |
| 9a9ac0f1629e | 192.4MiB / 200MiB | 0.60% | 96.19% | 868kB / 0B |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 5.13% | 99.96% | 868kB / 26.5MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 4.01% | 99.93% | 868kB / 45.6MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 3.62% | 99.94% | 868kB / 72.3MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 3.84% | 99.91% | 868kB / 90.7MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 3.63% | 99.96% | 868kB / 120MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 5.38% | 99.94% | 868kB / 141MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 5.31% | 99.88% | 868kB / 164MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 8.04% | 99.93% | 868kB / 190MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 3.60% | 99.94% | 868kB / 207MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 1.36% | 99.90% | 868kB / 208MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 1.20% | 99.92% | 868kB / 208MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 3.42% | 99.90% | 868kB / 234MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 3.81% | 99.91% | 868kB / 250MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 4.40% | 99.93% | 868kB / 278MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 6.35% | 99.93% | 868kB / 295MB |
| 9a9ac0f1629e | 199.9MiB / 200MiB | 5.03% | 99.96% | 868kB / 316MB |
| 9a9ac0f1629e | 199.8MiB / 200MiB | 3.75% | 99.91% | 868kB / 341MB |

Tabelle 3: Memory auf 200 MB limitiert (Swapping auf 100MB limitiert)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Container ID** | **Memory** | **CPU** | **Memory** | **Block I/O** |
| 49db876c1f59 | 508KiB / 200MiB | 0.00% | 0.25% | 590kB / 0B |
| 49db876c1f59 | 508KiB / 200MiB | 0.01% | 0.25% | 590kB / 0B |
| 49db876c1f59 | 23.43MiB / 200MiB | 0.78% | 11.71% | 590kB / 0B |
| 49db876c1f59 | 38.71MiB / 200MiB | 0.64% | 19.36% | 590kB / 0B |
| 49db876c1f59 | 61.65MiB / 200MiB | 0.63% | 30.83% | 590kB / 0B |
| 49db876c1f59 | 76.94MiB / 200MiB | 0.65% | 38.47% | 590kB / 0B |
| 49db876c1f59 | 99.87MiB / 200MiB | 1.07% | 49.94% | 590kB / 0B |
| 49db876c1f59 | 115.2MiB / 200MiB | 0.60% | 57.58% | 590kB / 0B |
| 49db876c1f59 | 138.1MiB / 200MiB | 0.62% | 69.05% | 590kB / 0B |
| 49db876c1f59 | 153.4MiB / 200MiB | 0.63% | 76.69% | 590kB / 0B |
| 49db876c1f59 | 176.3MiB / 200MiB | 0.68% | 88.16% | 590kB / 0B |
| 49db876c1f59 | 199.2MiB / 200MiB | 0.87% | 99.62% | 590kB / 0B |
| 49db876c1f59 | 199.9MiB / 200MiB | 6.58% | 99.93% | 606kB / 27.9MB |
| 49db876c1f59 | 199.9MiB / 200MiB | 4.92% | 99.96% | 606kB / 49.2MB |
| 49db876c1f59 | 199.8MiB / 200MiB | 5.56% | 99.88% | 606kB / 78.5MB |
| 49db876c1f59 | 199.9MiB / 200MiB | 4.31% | 99.94% | 606kB / 99.6MB |

Fazit:

Die Laufzeiten der Container bei einer Limitierung des Speichers auf weniger als den tatsächlichen Bedarf steigen. Das ist darauf zurückzuführen, dass durch Swapping der Speicher auf die Festplatte ausgelagert wird. Die Auslagerung kann in Tabelle 2 (Spalte Block I/O) beobachtet werden.   
In dem Versuch ist festzustellen, dass die Festplatte wesentlich langsamer ist, als der Speicher.  
Wenn wie in Tabelle 3 zu sehen auch die Größe des Swappings limitiert wird und die Summe von Memoy und Swapping . (Memory 200MB + Swapping 100MB; Bedarf: ca. 500MB) geringer als der Bedarf ist, crasht das Programm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Container ID | IMAGE | COMMAND | CREATED | STATUS |
| 49db876c1f59 | Memory\_05sleep | "./05gb\_sleep" | About a minute ago | Exited (137) 36 seconds ago |

Die Meldung „Exited (137)“ bedeutet, dass Docker nicht genügend RAM zur Beendigung der Aufgabe besitzt.