Simulationsbasierte Bewertung von maschinellen Lernverfahren für Vorablade-Strategien in Linux

Abschlussvortrag

Alexander Lochmann

08. März 2011

Alexander.Lochmann@tu-dortmund.de

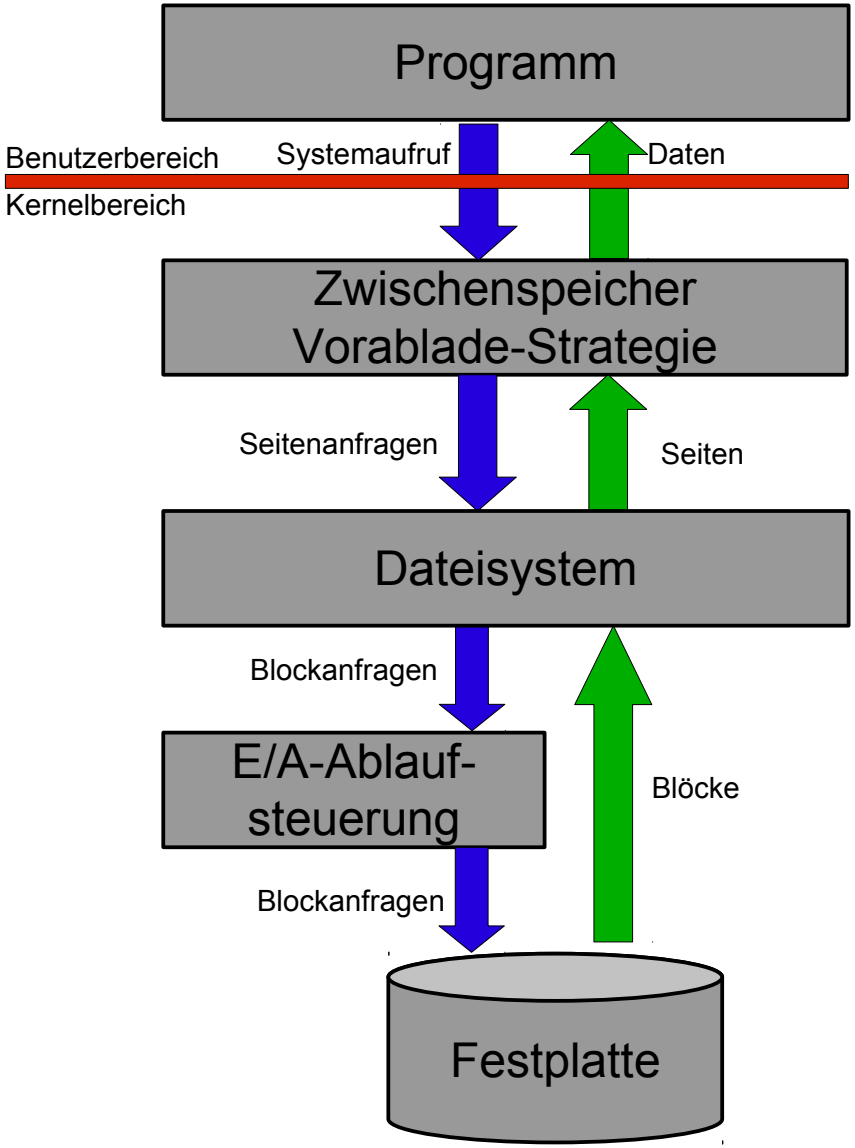
Agenda

- Ziele
- Linux E/A-Stapel
- Linux-Vorablade-Strategie
- Open-Vorablade-Strategie
- Simulationsmodell
- Evaluation
- Fazit
- Ausblick
- Neue Erkenntnisse

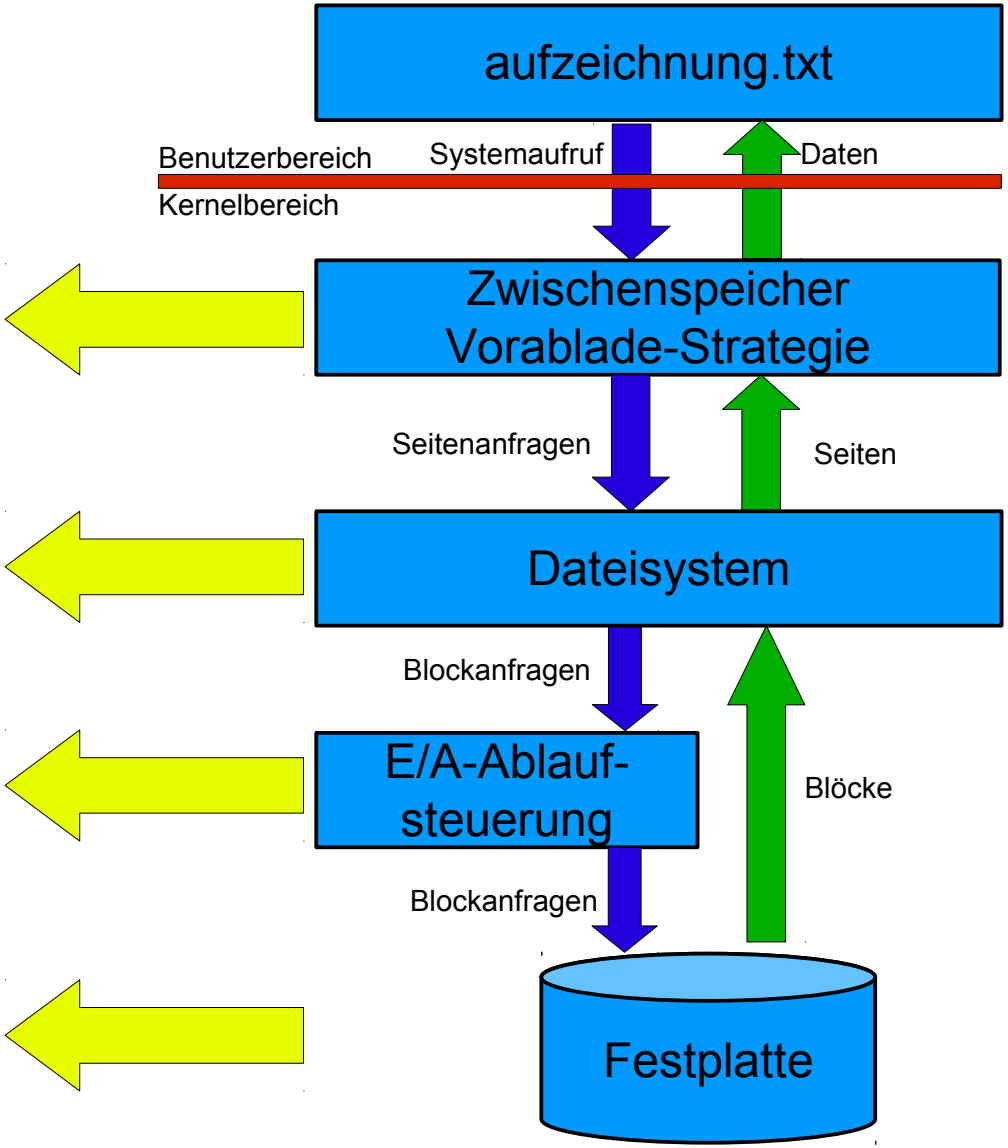
Ziele

- Entwicklung einer Evaluationsplattform
  - Abbildung der einzelnen Schichten des E/A-Stapels
  - Eingabe für die Simulation
  - Simulationsmodell
- Evaluation zweier Vorablade-Strategien
  - Linux-Vorablade-Strategie
  - Open-Vorablade-Strategie
    - Conditional Random Fields (CRFs)
    - Naives Bayes
    - Orakel

Linux E/A-Stapel



Simulierter E/A-Stapel



- LRU und LFU
- Linux- und Open-Strategie

- SimpleFilesystem

- FiFo
- NOOP

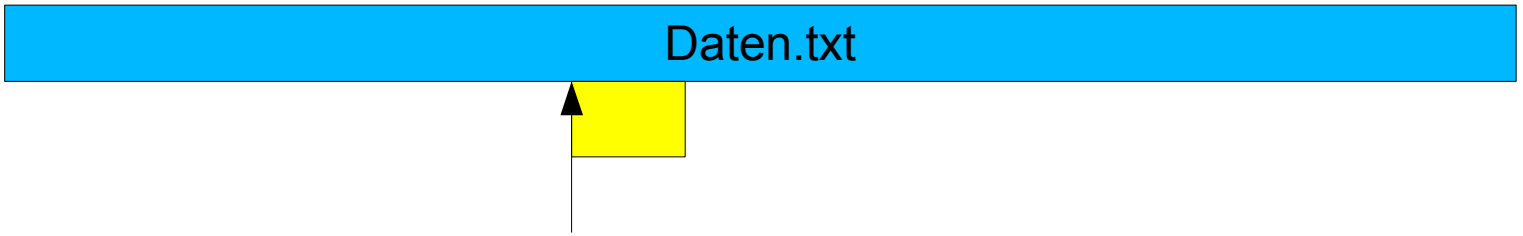
- NotSim
- DiskSim [7]

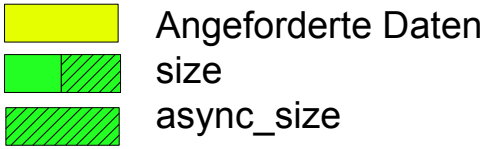
Linux-Vorablade-Strategie

Daten.txt

- Linux-Vorablade-Strategie
  - Arbeitet seitenbasiert
  - Untersucht die Zugriffsmuster auf die Seiten
  - Bei erfolgreichem Vorabladen wird aggressiver vorausgeladen
  - Bei Zugriff auf den ersten Block einer Datei wird von einem sequentiellen Zugriff ausgegangen
  - Weiteres Vorausladen bei *async_size* verbleibenden Seiten
  - Max. Parallelität bei *async_size* = *size*
  - Ziel: Reduktion der Zugriffszeiten auf Dateien

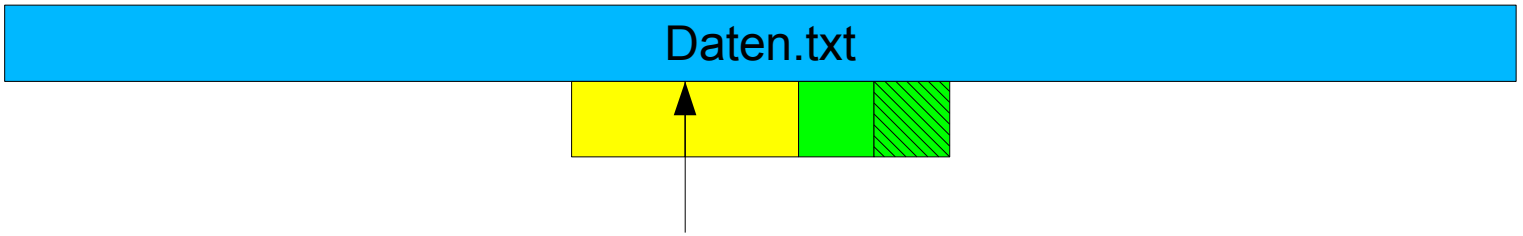
Linux-Vorablade-Strategie

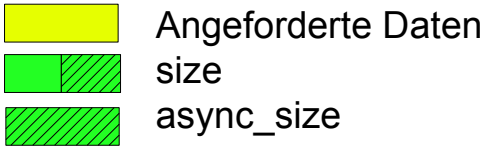




- Linux-Vorablade-Strategie
  - Arbeitet seitenbasiert
  - Untersucht die Zugriffsmuster auf die Seiten
  - Bei erfolgreichem Vorabladen wird aggressiver vorausgeladen
  - Bei Zugriff auf den ersten Block einer Datei wird von einem sequentiellen Zugriff ausgegangen
  - Weiteres Vorausladen bei *async_size* verbleibenden Seiten
  - Max. Parallelität bei *async_size* = *size*
  - Ziel: Reduktion der Zugriffszeiten auf Dateien

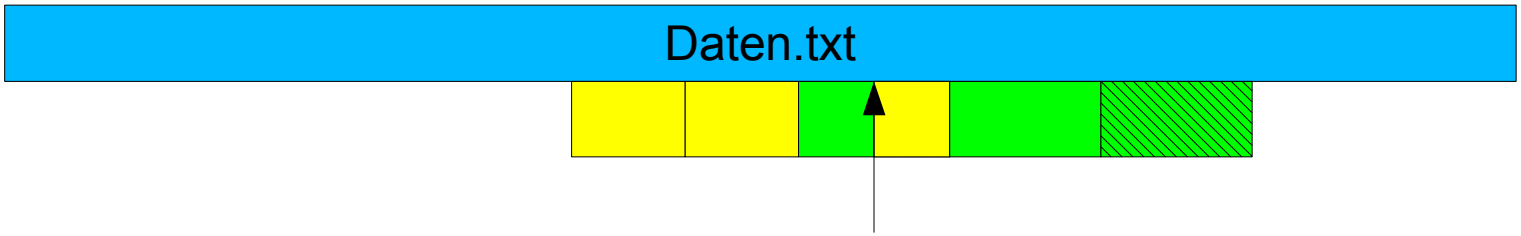
Linux-Vorablade-Strategie

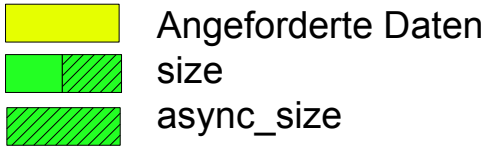




- Linux-Vorablade-Strategie
  - Arbeitet seitenbasiert
  - Untersucht die Zugriffsmuster auf die Seiten
  - Bei erfolgreichem Vorabladen wird aggressiver vorausgeladen
  - Bei Zugriff auf den ersten Block einer Datei wird von einem sequentiellen Zugriff ausgegangen
  - Weiteres Vorausladen bei *async_size* verbleibenden Seiten
  - Max. Parallelität bei *async_size* = *size*
  - Ziel: Reduktion der Zugriffszeiten auf Dateien

Linux-Vorablade-Strategie





- Linux-Vorablade-Strategie
  - Arbeitet seitenbasiert
  - Untersucht die Zugriffsmuster auf die Seiten
  - Bei erfolgreichem Vorabladen wird aggressiver vorausgeladen
  - Bei Zugriff auf den ersten Block einer Datei wird von einem sequentiellen Zugriff ausgegangen
  - Weiteres Vorausladen bei *async_size* verbleibenden Seiten
  - Max. Parallelität bei *async_size* = *size*
  - Ziel: Reduktion der Zugriffszeiten auf Dateien

Linux-Vorablade-Strategie

Daten.txt

- Linux-Vorablade-Strategie
  - Arbeitet seitenbasiert
  - Untersucht die Zugriffsmuster auf die Seiten
  - Bei erfolgreichem Vorabladen wird aggressiver vorausgeladen
  - Bei Zugriff auf den ersten Block einer Datei wird von einem sequentiellen Zugriff ausgegangen
  - Weiteres Vorausladen bei *async_size* verbleibenden Seiten
  - Max. Parallelität bei *async_size* = *size*
  - Ziel: Reduktion der Zugriffszeiten auf Dateien

Linux-Vorablade-Strategie

- Linux-Vorablade-Strategie
  - Arbeitet seitenbasiert
  - Untersucht die Zugriffsmuster auf die Seiten
  - Bei erfolgreichem Vorabladen wird aggressiver vorausgeladen
  - Bei Zugriff auf den ersten Block einer Datei wird von einem sequentiellen Zugriff ausgegangen
  - Weiteres Vorausladen bei *async_size* verbleibenden Seiten
  - Max. Parallelität bei *async_size* = *size*
  - Ziel: Reduktion der Zugriffszeiten auf Dateien

Open-Vorablade-Strategie

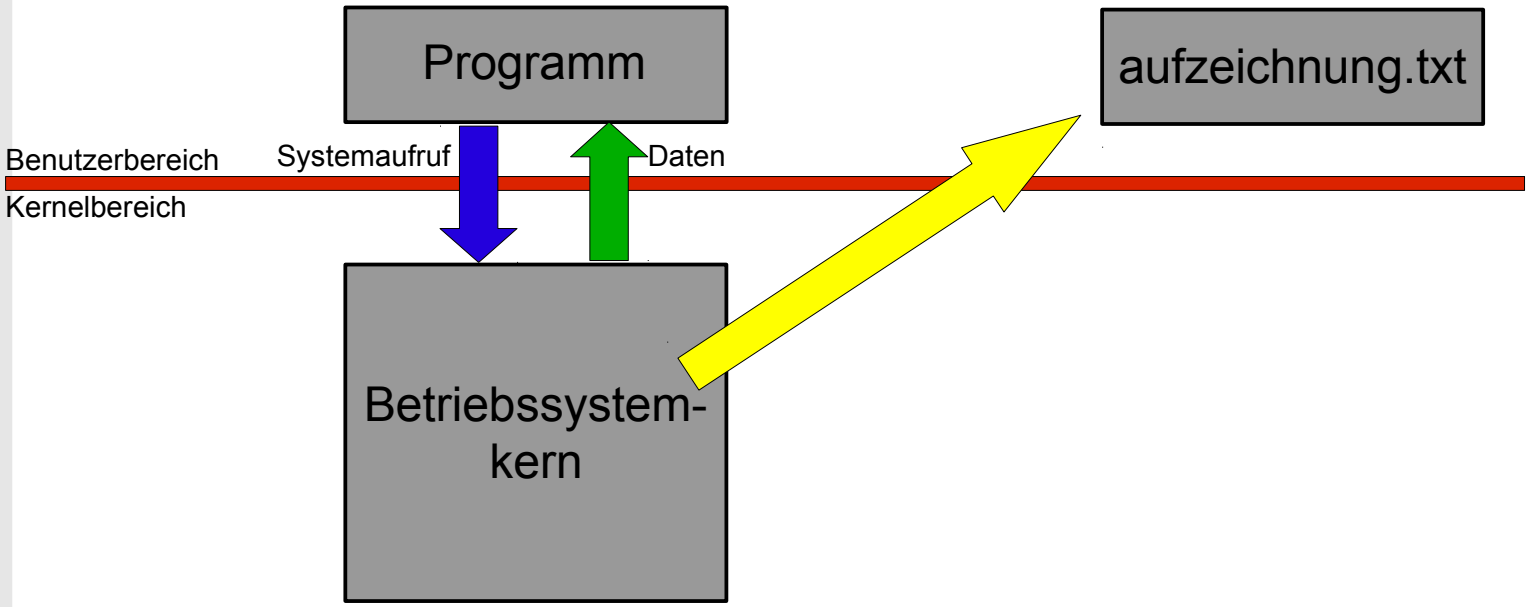
- Entscheidung über Vorausladen **nur** beim Öffnen einer Datei
- Vorhersage erfolgt durch:
  - CRFs
  - Naive Bayes
  - Orakel
- Mögliche Vorhersagen:
  - *ZERO*
  - *READ / RANDOM*

  } Kein Vorausladen
  - *FULL*

  } Komplette Datei in den Zwischenspeicher laden

Eingabe für den Simulator

- Mitschneiden aller relevanten Systemaufrufe im Betriebssystemkern mittels SystemTap [6]
- Verwendet C-ähnliche Syntax
- Zugriff auf alle Datenstrukturen im Kern



Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Mikrosekunden seit Start der Aufzeichnung

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Millisekunden im Benutzer-bereich seit letztem Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Name des Systemaufrufes

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Prozess-ID des aktuellen Prozesses

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Thread-ID des aktuellen Fadens

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Prozess-ID des Vaters

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

Benutzer- bzw. Gruppen-ID

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

Name des Programms (max. 15 Zeichen)

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

Dateideskriptor; prozessweite Identifikation der Datei

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

Dateigröße in Bytes zum Zeitpunkt des Systemaufrufes

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

Eindeutige Identifizierung der Datei im System

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

Parameters des System-aufrufes

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eingabe für den Simulator

- XML-basiertes Format → Einfache Validierung der Eingabe
- Struktur der Eingabe
  - Informationen für jeden Systemaufruf

```
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
```

  - Systemaufruf-spezifische Informationen – hier: open()

```
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path
>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
```

Eindeutige Identifizierung des open-Systemaufrufes innerhalb der Aufzeichnung

Eingabe für den Simulator

- Größe hängt von der Aktivität der Programme ab

- Benchmark
  - 291186 Systemaufrufe
  - 74 MB
  - Dauer: ~ 4 h

- Glimpse
  - 1460464 Systemaufrufe
  - 389 MB
  - Dauer: ~ 4 Min

```
<?xml version="1.0" encoding="UTF-8"?>
<trace xmlns="http://www.w3.org/2001/XMLSchema">
<event>
<time>13220137</time>
<usertime>0</usertime>
<syscallname>open</syscallname>
<pid>3392</pid>
<tid>3392</tid>
<ppid>3390</ppid>
<uid>1000</uid>
<gid>1000</gid>
<execname>acroread</execname>
<fd>10</fd>
<fsize>601684</fsize>
<inode>1839478</inode>
<path>/home/sfb/10-Fadensync.pdf</path>
<mode>384</mode>
<flags>0</flags>
<id>589</id>
</event>
[…]
</trace>
```

Eingabe für den Simulator

- Dateioperationen
  - open / creat
  - lseek / llseek
  - read / write
  - pread / pwrite
  - close
  - unlink
- Prozessverwaltung
  - fork / vfork
  - clone
  - execve
  - exit
  - wait4

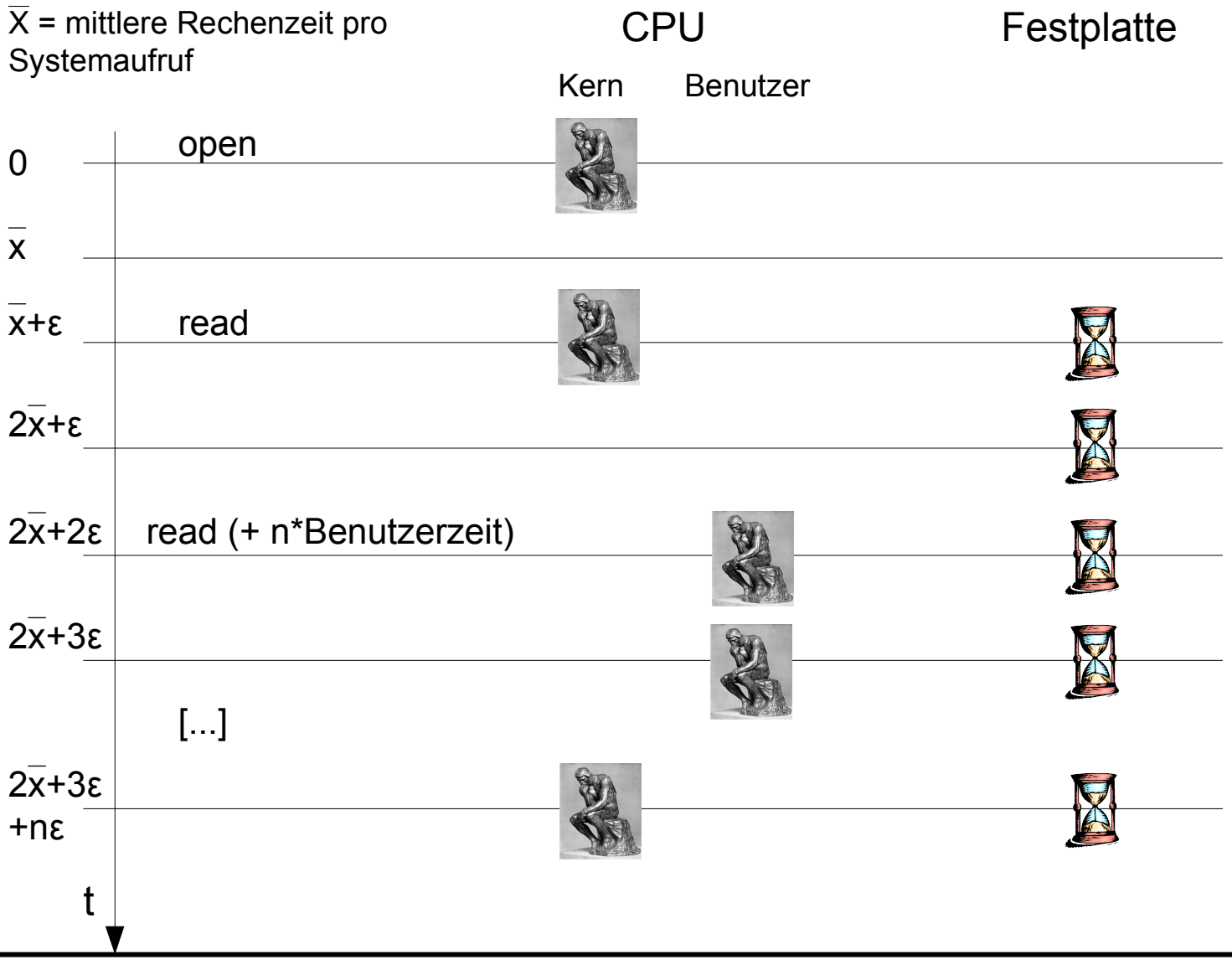
Nötig zur Einhaltung der aufgezeichneten Prozessabfolge

Simulationsmodell [4]

- Ereignisorientierte Sicht
  - Zeit schreitet sprunghaft voran
  - Simulation terminiert bei Eintreten einer Bedingung oder eines festen Zeitpunktes
  - Zeit verändert sich während der Bearbeitung
- Prozessorientierte Sicht
  - Zeit schreitet schrittweise voran (zeitdiskret)
  - „Arbeit“ geschieht in der passiven Phase
  - Zeit steht während der aktiven Phase still

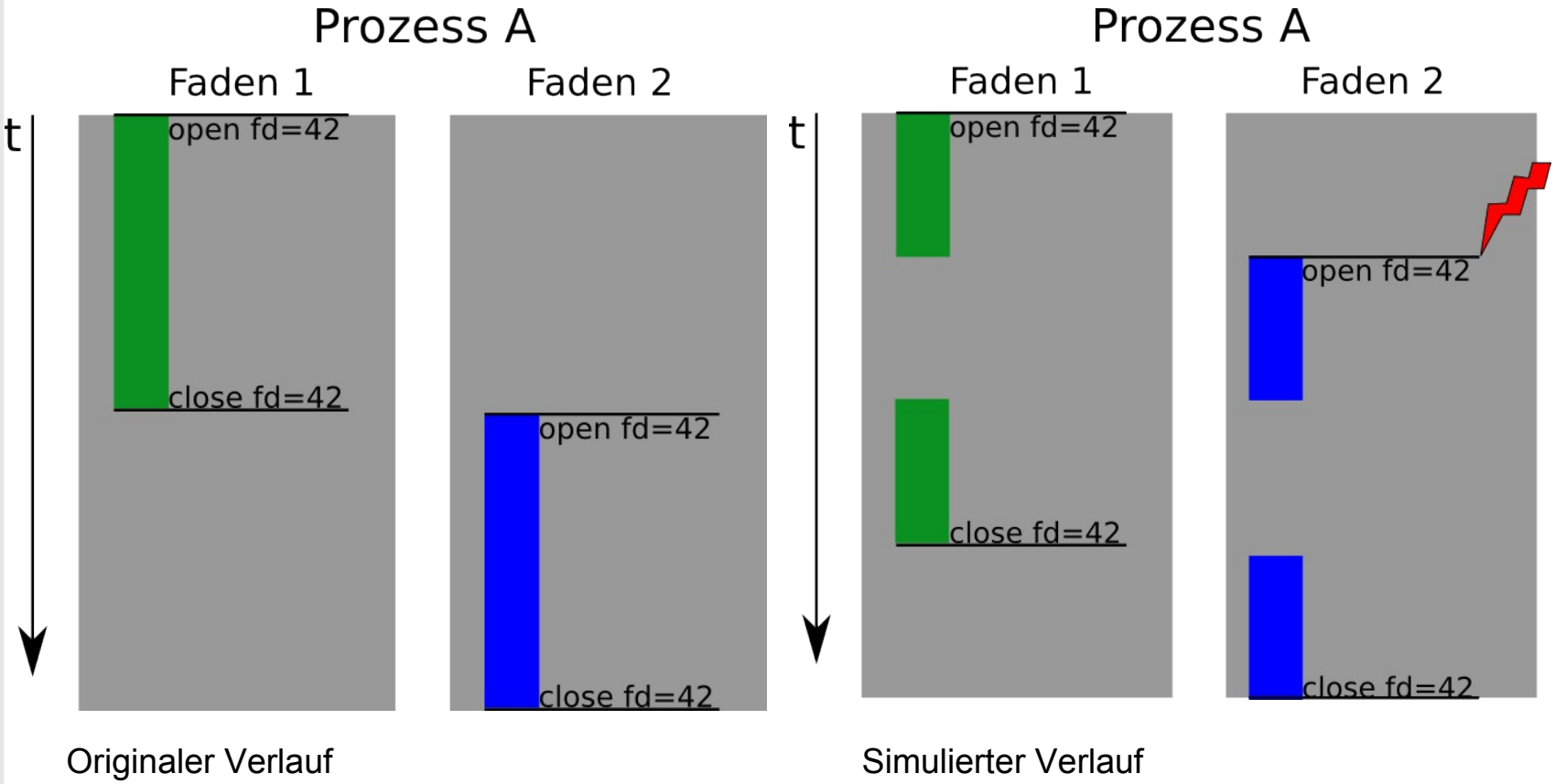
Simulationsmodell

$\overline{X}$ = mittlere Rechenzeit pro Systemaufruf



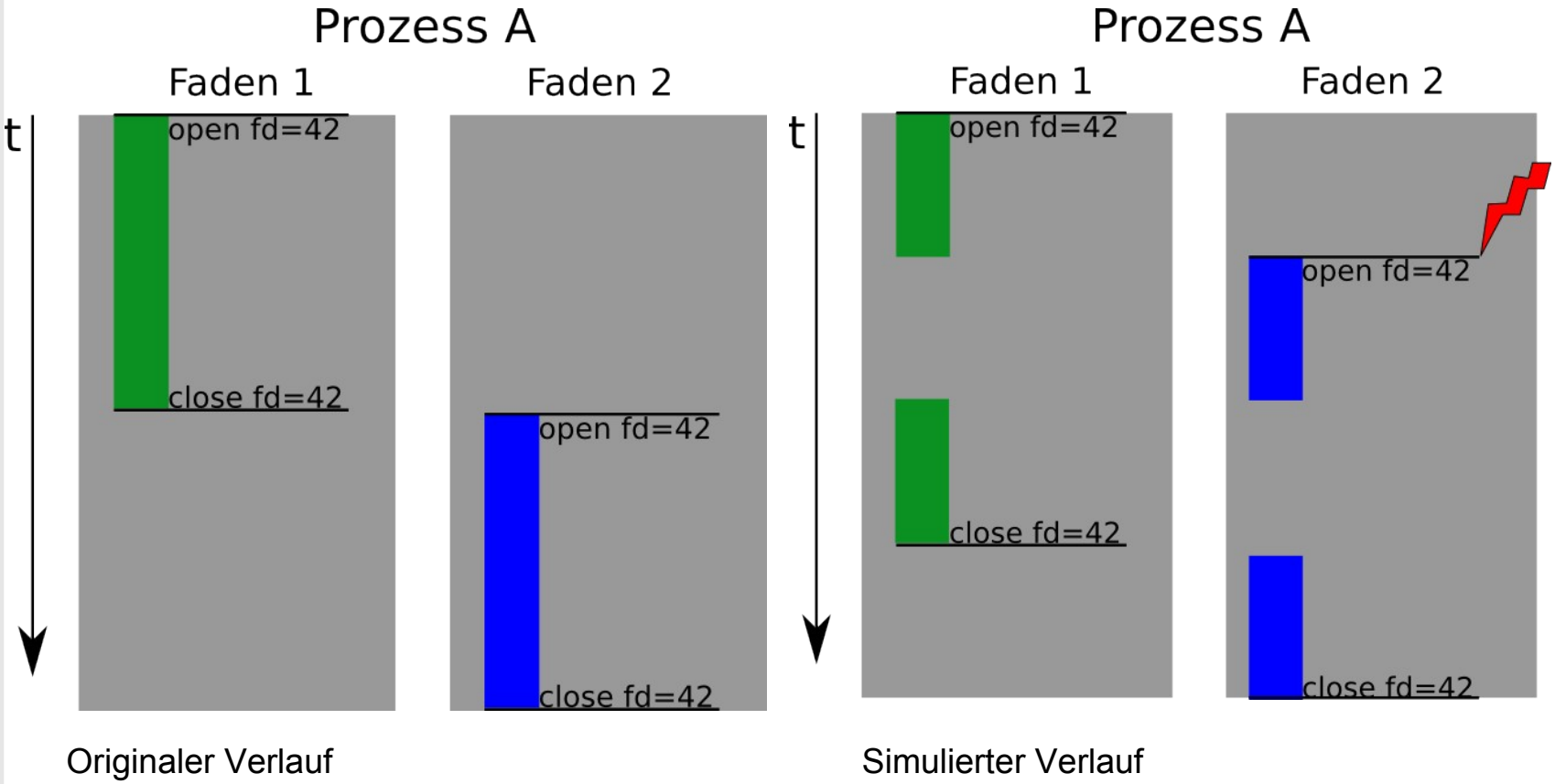
Probleme bei der Simulation

- **Problem:**



Probleme bei der Simulation

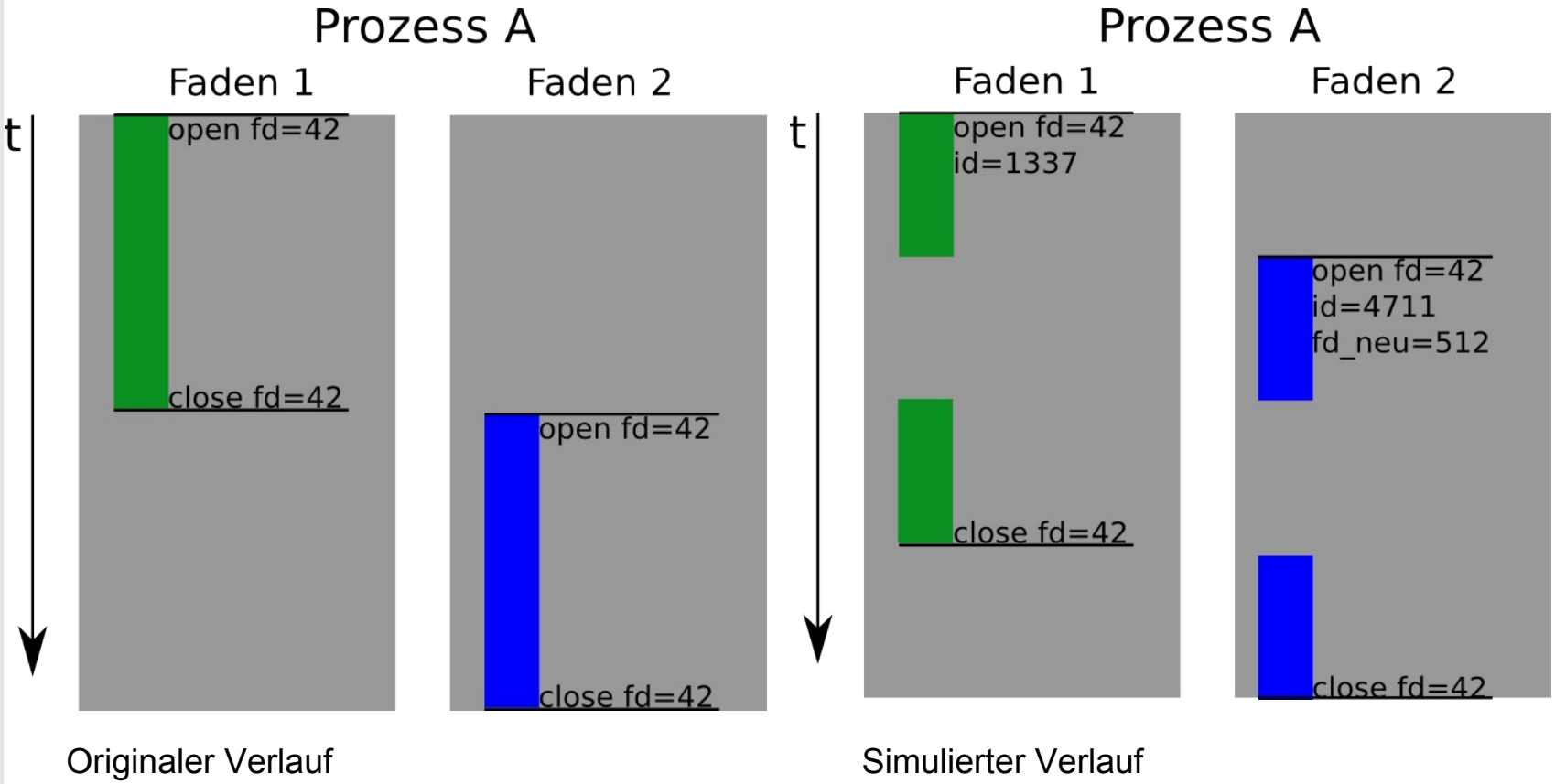
- **Problem:**



Lösung: Dateideskriptoren umlenken und das Öffnen mit einer ID versehen

Probleme bei der Simulation

- **Lösung:**



Evaluation

- Ziel
  - Reduktion der Gesamt-E/A-Dauer
    → Energieeinsparung bei der E/A-Einheit
  - Unnötige E/A-Anfragen vermeiden

- Messgrößen
  - Gesamte E/A-Dauer (ms)
  - Anzahl vorausgeladenen Seiten
  - Anzahl unnötig vorausgeladenen Seiten
  - Güte: $1 - \frac{unnötige\ Seiten}{vorausgeladenen\ Seiten}$

Evaluation

- Welche Werte liefert der Simulator?

Info:
Inputfile: /fs/students/lochmann/bachelorarbeit/logs/alex.v6.txt_benchmark_1
Paramfile: ../cheetah9LP.parv
One tick: 100
Cache size: 8.00 MiB
Cache replacement: Least Recently Used
Filesystem: SimpleFilesystem
IOScheduler: FIFOScheduler
Disk: DiskSim
Processscheduler: RoundRobin
Prefetching: Linux prefetching
[…]
Info: Overriding systime with max_continue_time. Some process still waiting for io completion but there are no events left.
Info: Time for forced writebacks:10357, syscalls:291186
Info: CPU: **execution time:349366554**, **idle time:55787300,** user time:292420000
Info: Disk: Response time(in ms):**n=57231, total time:67692.4**, average time:1.18279, std. deviation:1.13241, bytes transfered:232395776
Info: Cache: **Total accesses:291177, Hitratio:91.4392% (266250)**, Prefetch ratio:99.8284% of total hits (265793 of 266250), **#prefetched pages:55537, #prefetched but not accessed:6630**

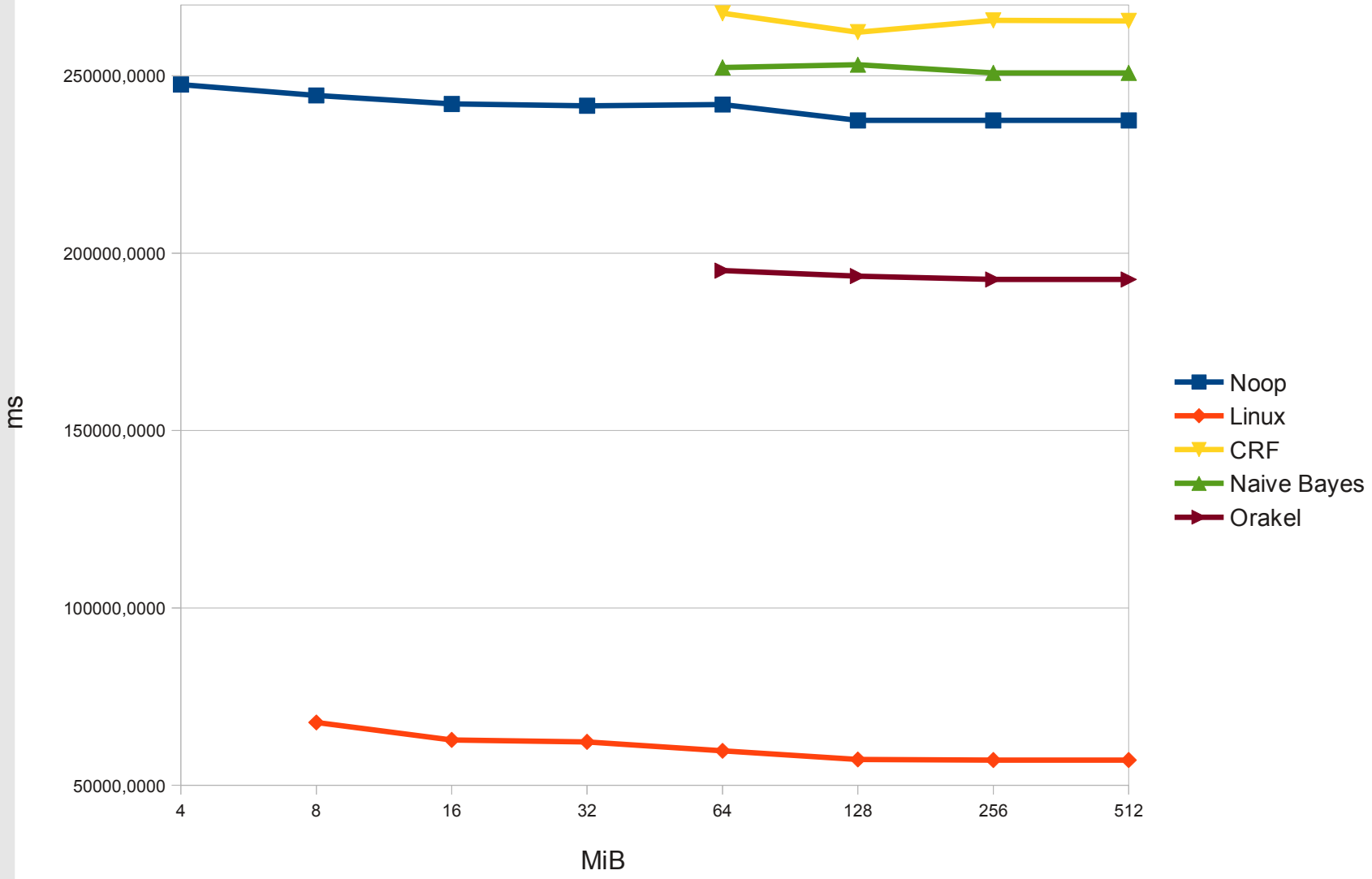
Evaluation

- Was wurde aufgezeichnet?

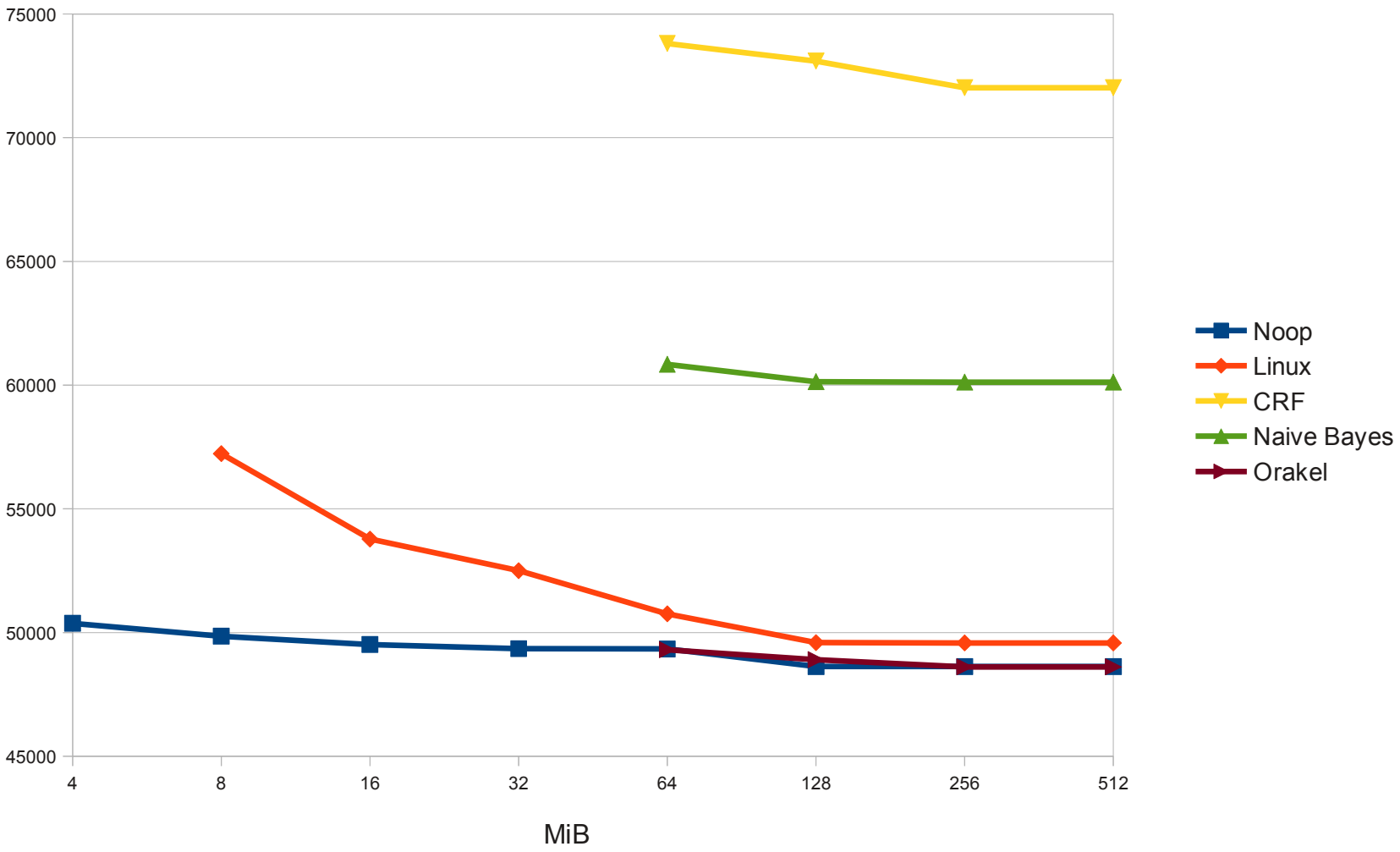
- Benchmark
  - Acrobat Reader → Betrachtung mehrerer PDF-Dateien
  - VLC → sequentielle Wiedergabe einer Wiedergabeliste
  - VLC → zufällige Wiedergabe einer Wiedergabeliste
  - Eye of Gnome → Bildbetrachter

- Plattform
  - Virtuelle Maschine (1 Prozessor) mit 1 GB Arbeitsspeicher
  - Ubuntu 10.10, 32 bit
  - SystemTap 1.4
  - Linux 2.6.35.7

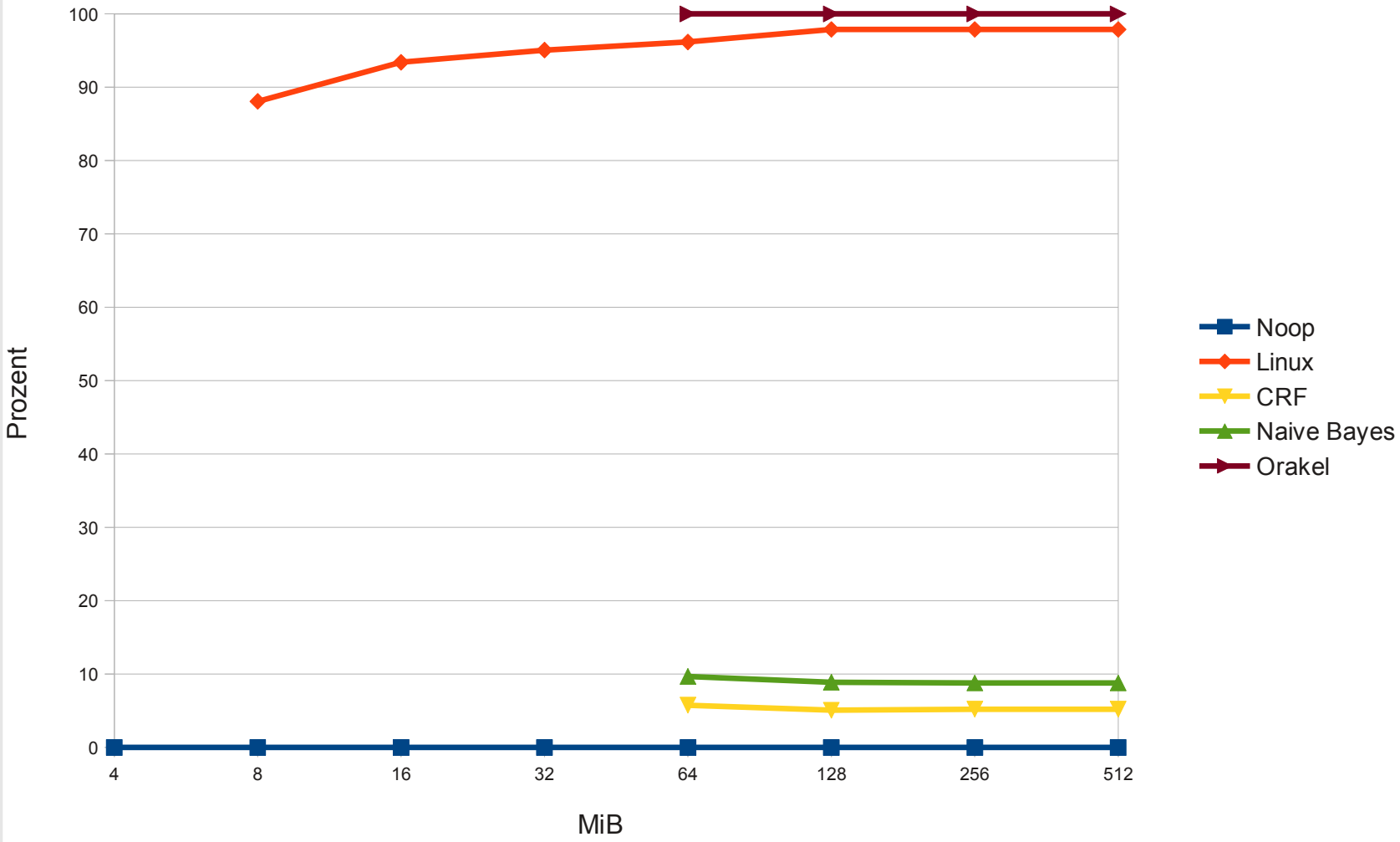
Evaluation – Gesamt-E/A-Dauer



Evaluation – E/A-Anfragen



Evaluation - Güte



Fazit

- Simulator funktioniert
  - Veranschaulicht gut Effekte in einem realen BS
  - Alle Komponenten sind einfach austauschbar
  - Laufzeit hängt stark von der Anzahl Prozesse bzw. Systemaufrufe ab → Datenstrukturen sind zu langsam
- Open-Vorablade-Strategie
  - Aktueller Ansatz skaliert nicht gut
    → Alles oder nichts vorausladen
  - Linux ist in allen Disziplinen besser
    - Passt sich während des Lesens an
    - Lädt nur kleine Blöcke aus der Datei voraus
  - Selbst ein Orakel bringt keine Verbesserung

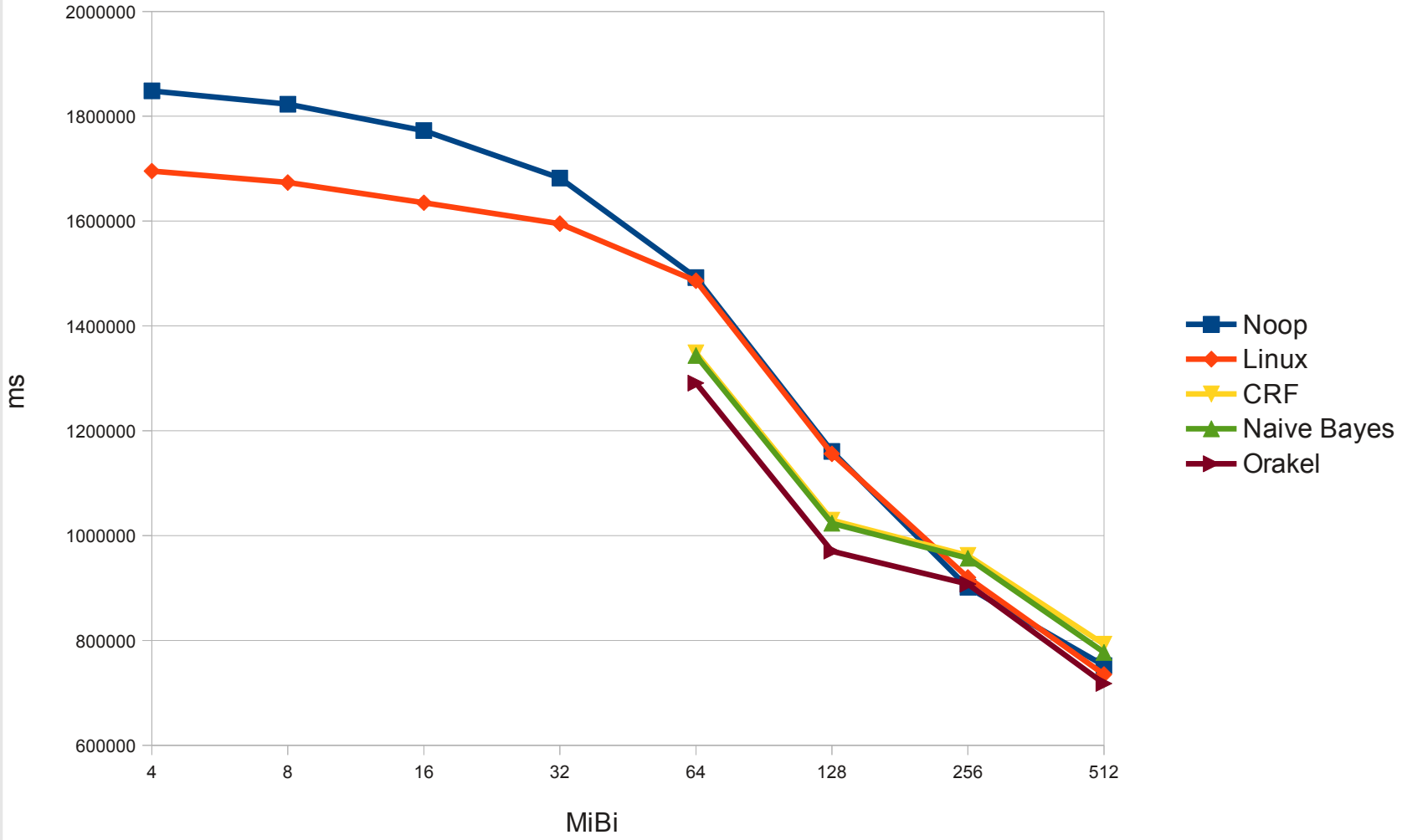
Ausblick

- Simulator / Evaluation:
  - Vertreter für sequentielle Zugriffe, z.B. Datenbank-Benchmark
  - Simulator für Flashspeicher
  - Verteilung der E/A-Anfragen
  - Echtes Dateisystem bzw. echte E/A-Ablaufsteuerung implementieren
- Vorablade-Strategie:
  - Linux-Strategie unterstützen statt zu konkurrieren
  - Das gesamte System beobachten
  - Vorhersagen eher treffen

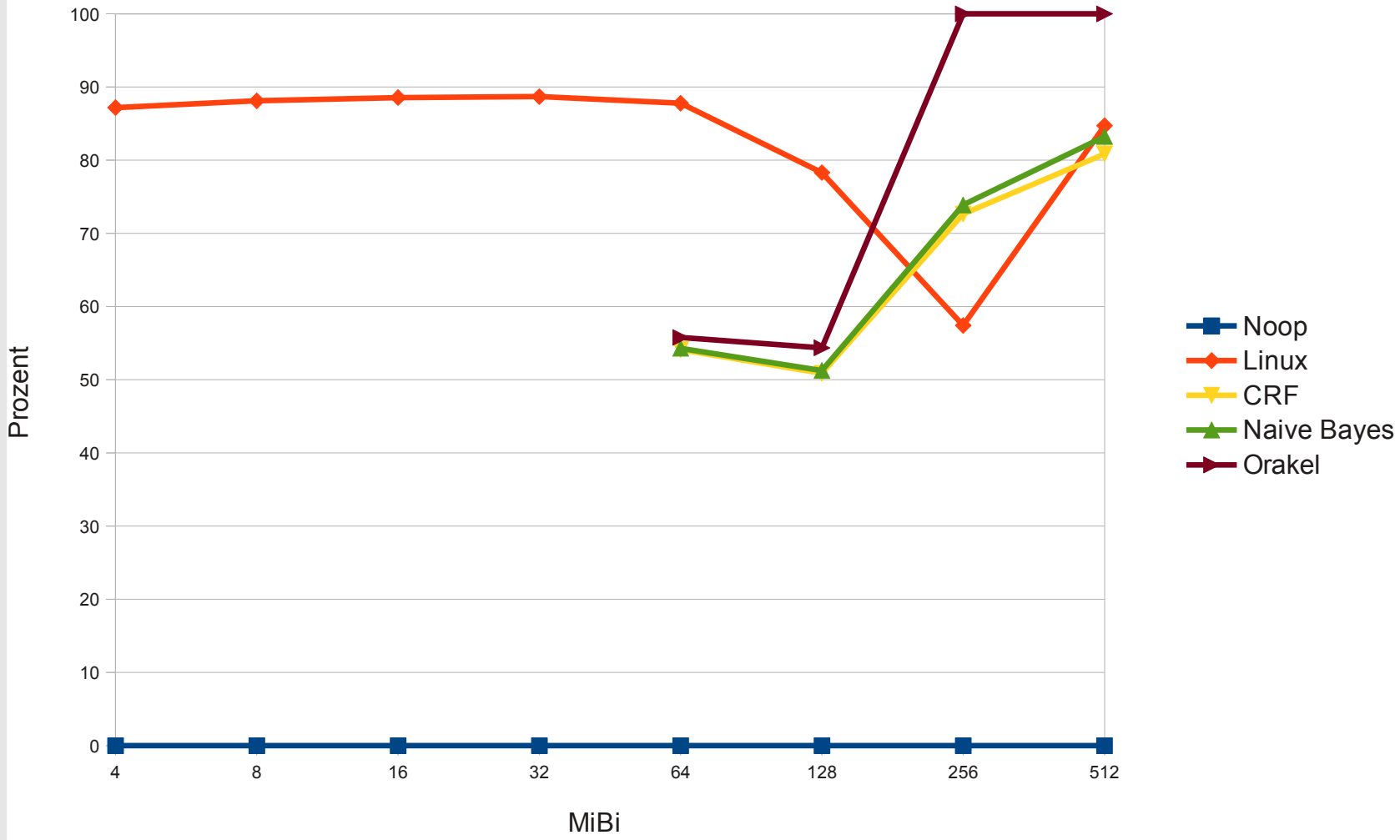
Neue Erkenntnisse – Teil 1

- Weitere Simulationsläufe mit einer Aufzeichnung von Glimpse[8]
- Werkzeug zur Indezierung von Verzeichnisbäumen

Gesamt-E/A-Dauer



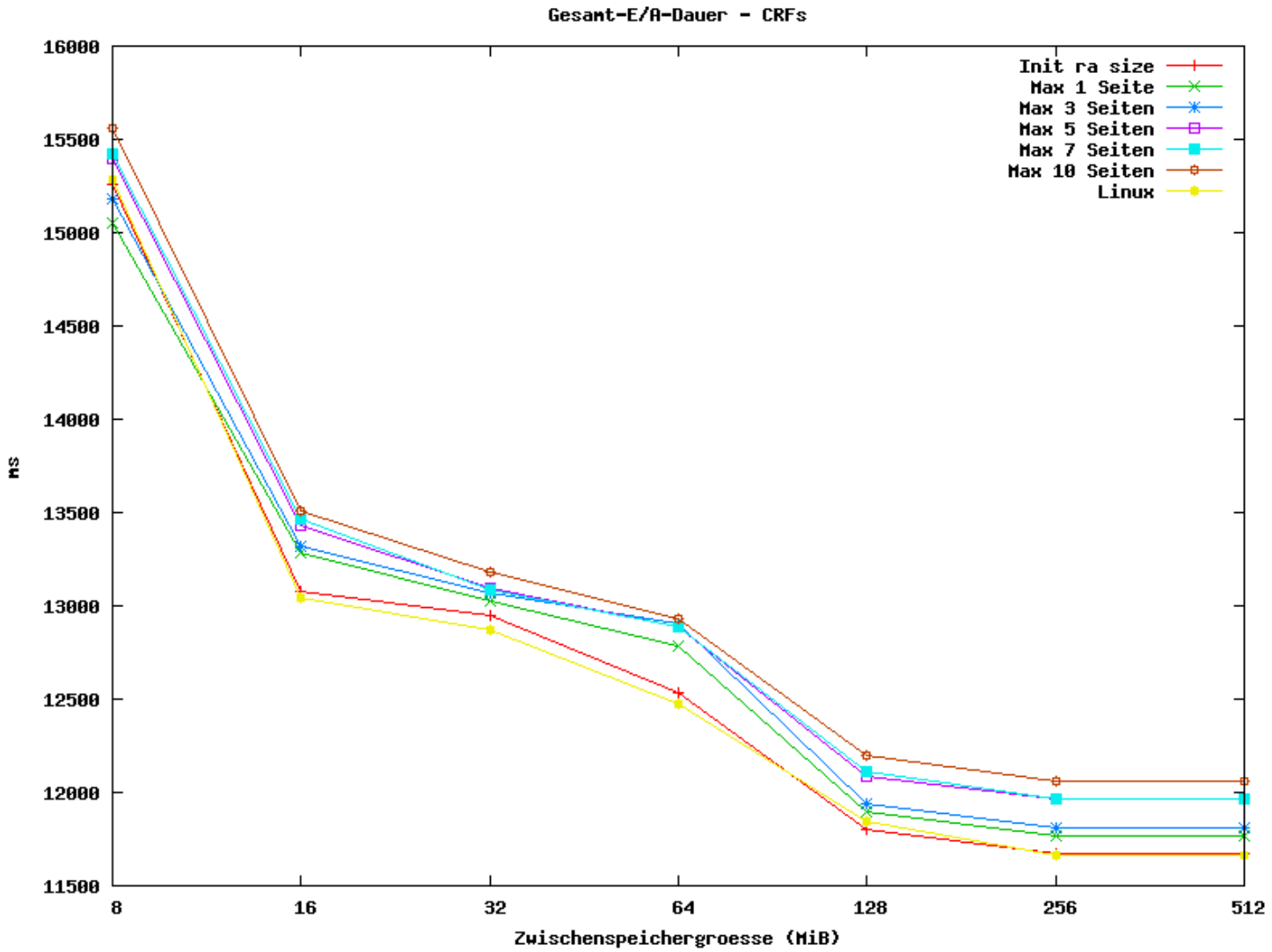
Güte


100
90
80
70
60
50
40
30
20
10
0
Prozent
4
8
16
32
64
128
256
512
MiBi
Noop
Linux
CRF
Naive Bayes
Orakel

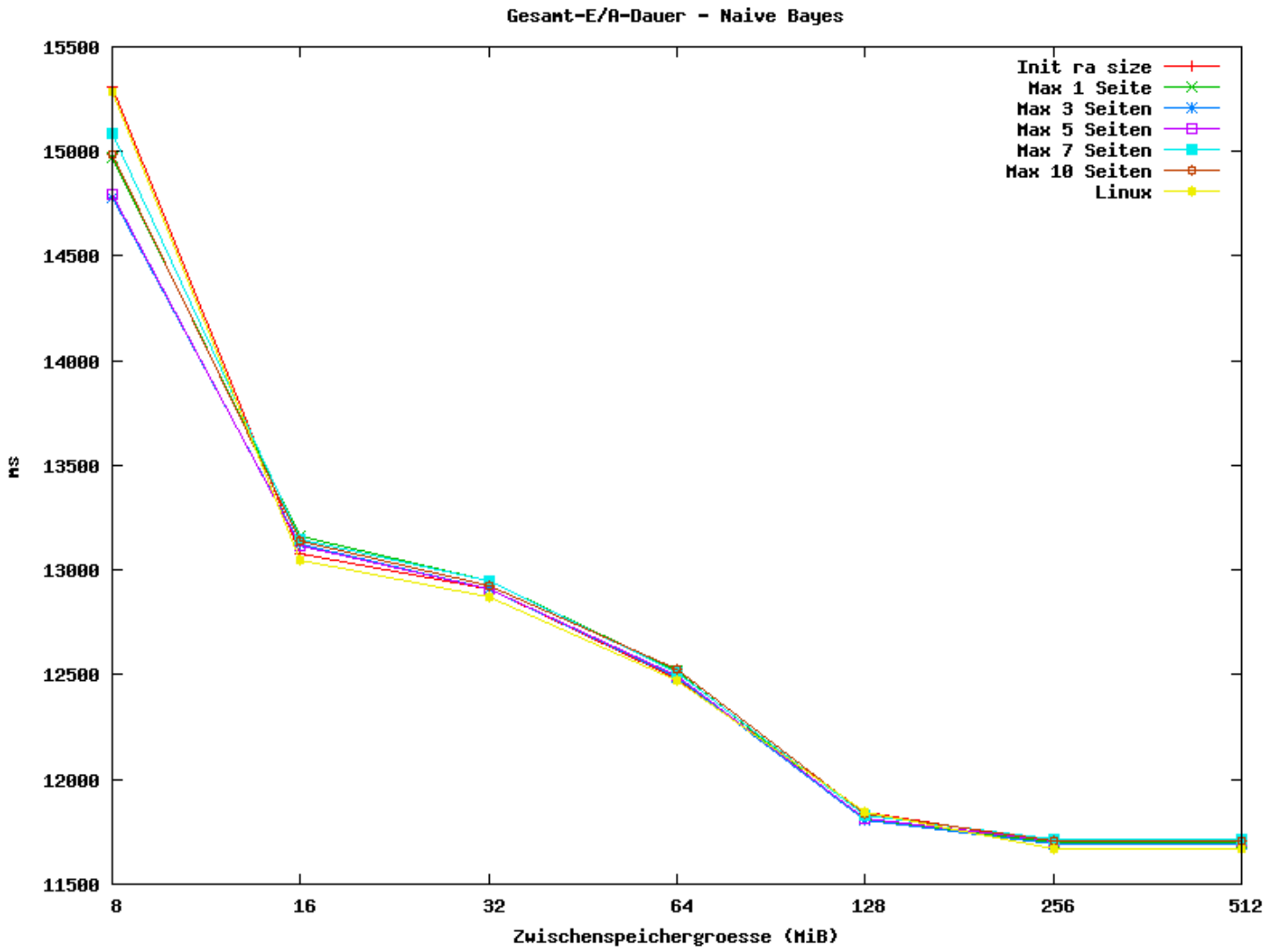
Neue Erkenntnisse – Teil 2

- Modifikation der Open-Strategie
  - Grundlage ist die Linux-Strategie
  - Verdopplung des initialen Fensters beim Öffnen bei Vorhersage *FULL* – kein Vorausladen
  - Erweiterung: Vorausladen von 1-10 Seiten bereits beim Öffnen

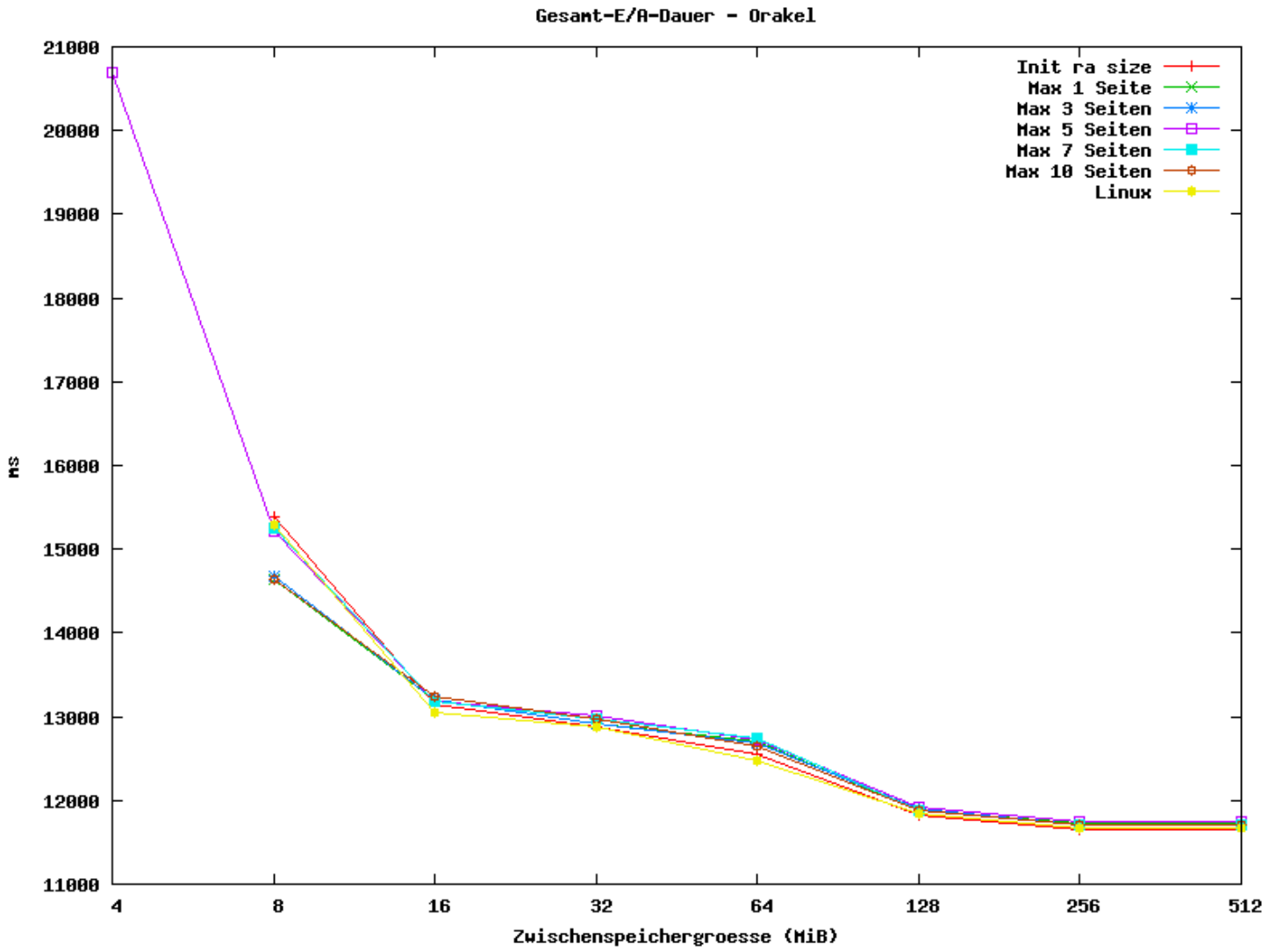
Gesamt-E/A-Dauer - CRF



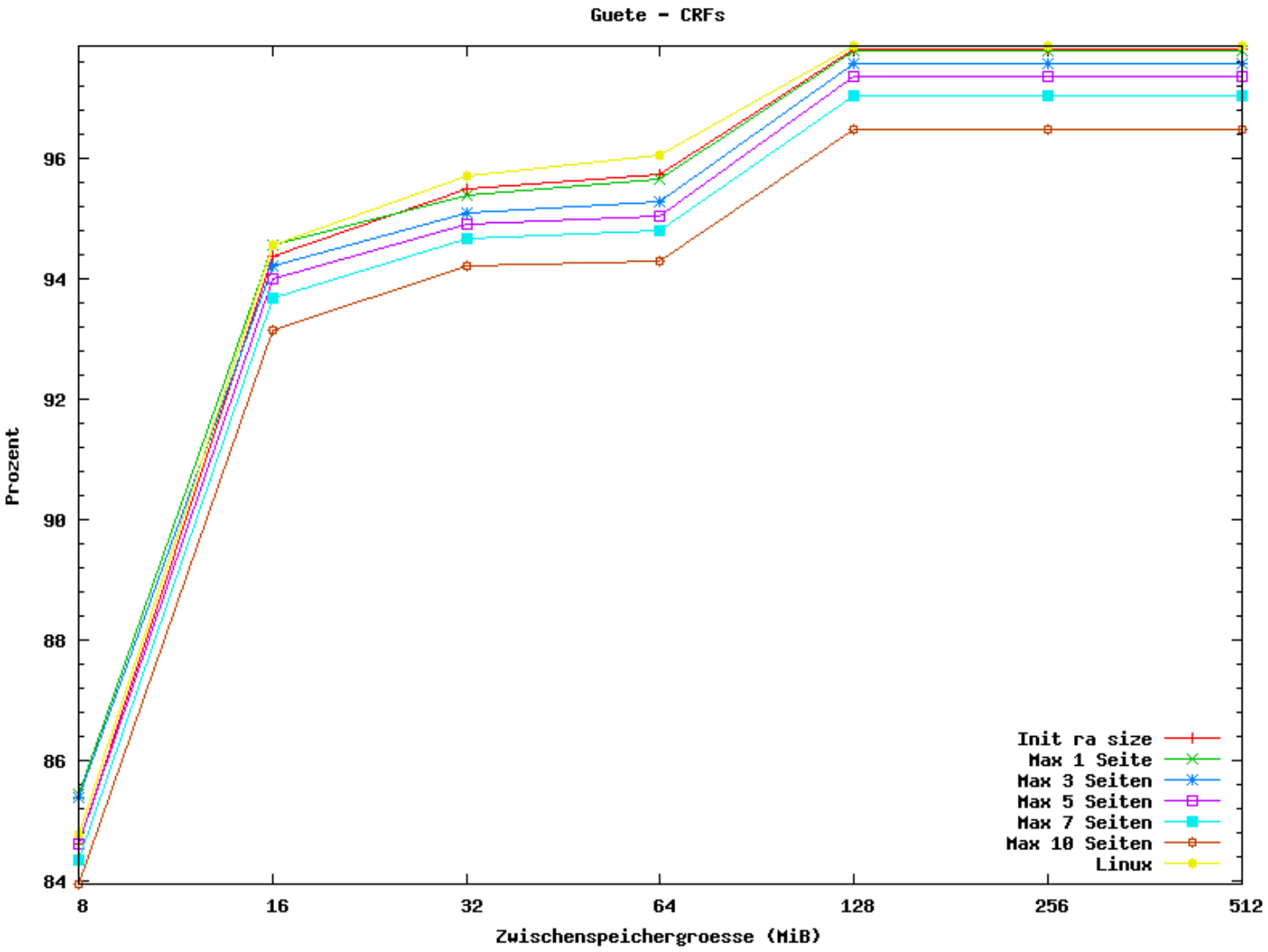
Gesamt-E/A-Dauer - Naive Bayes



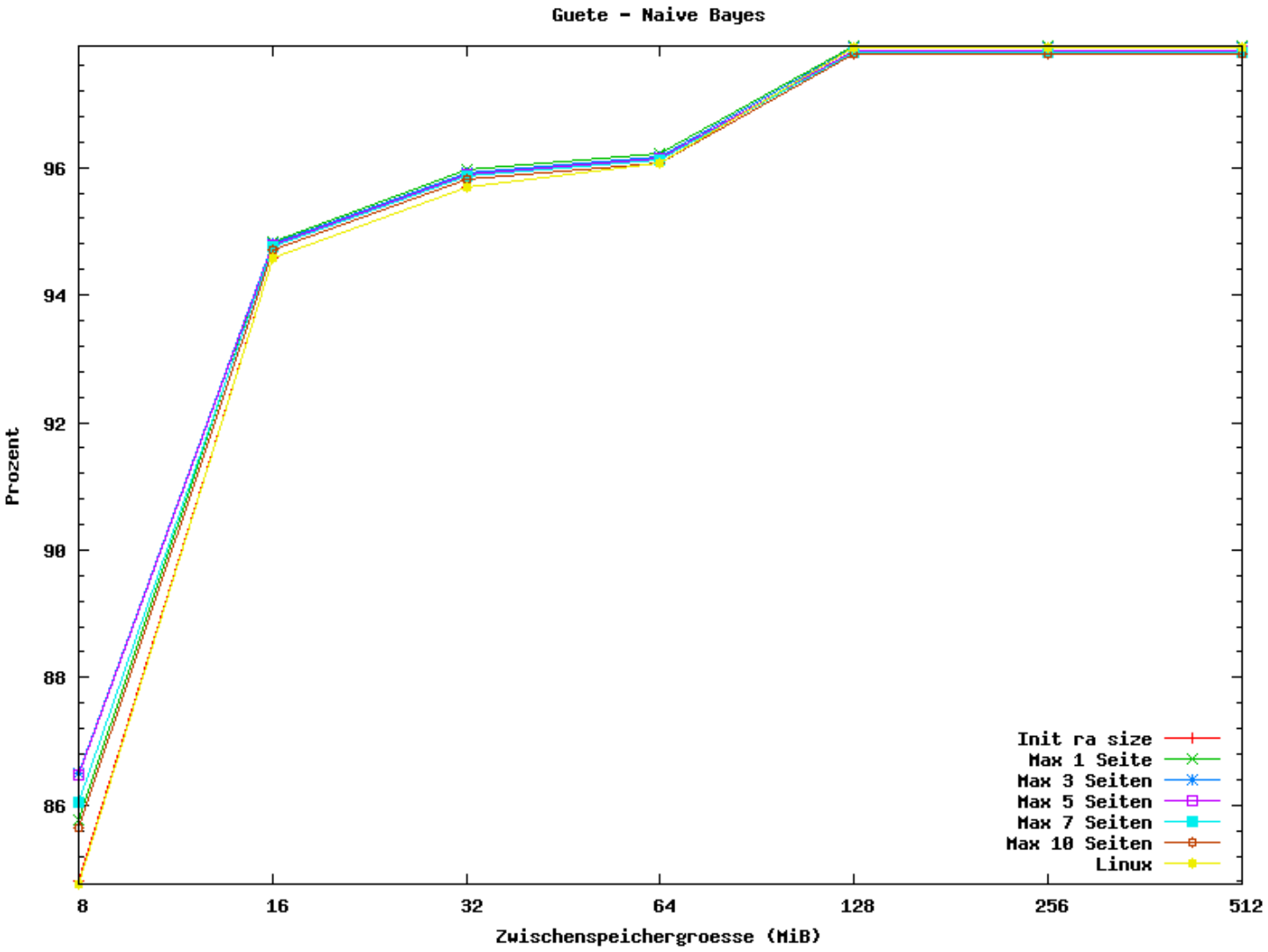
Gesamt-E/A-Dauer - Orakel



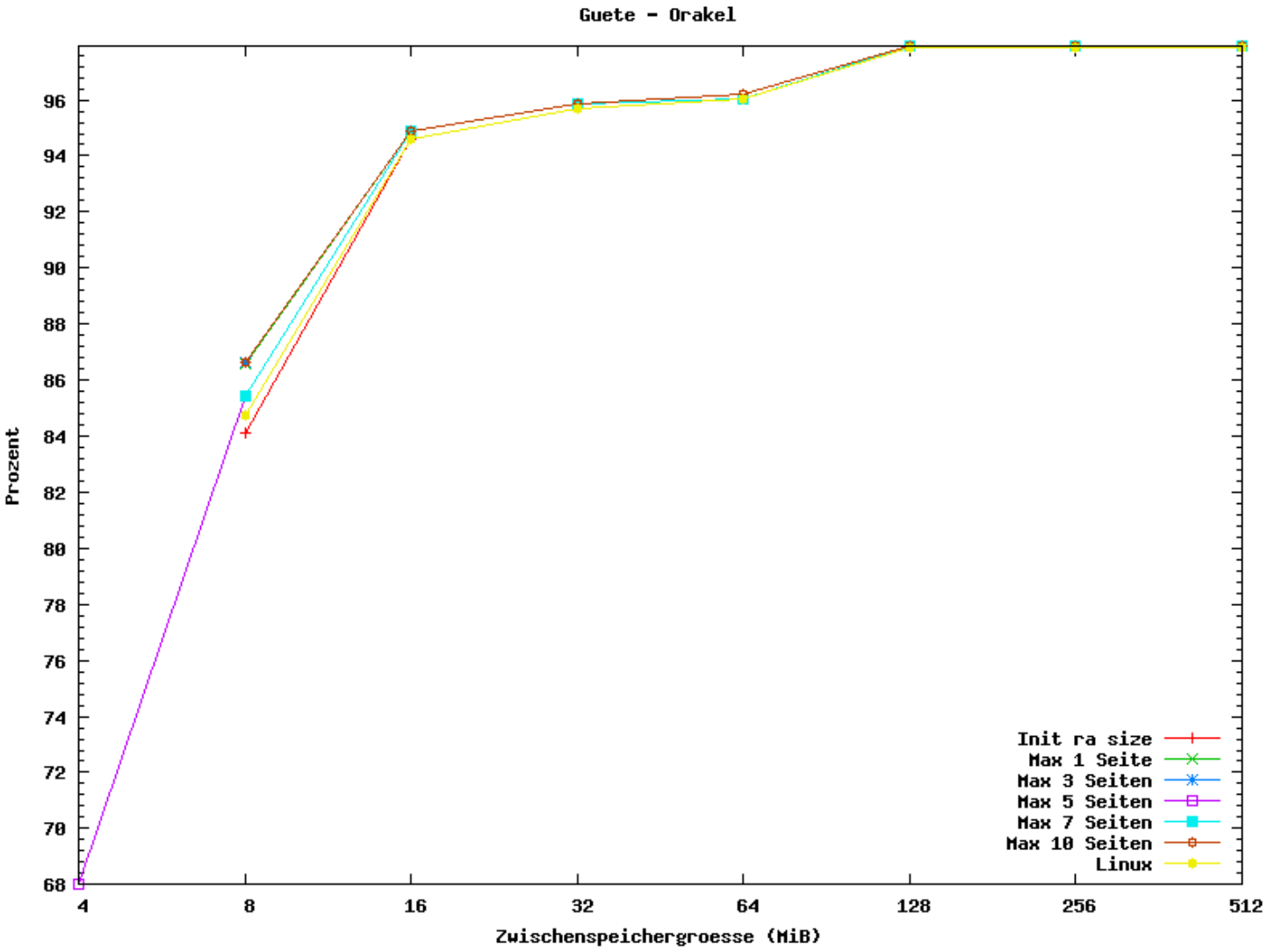
Güte - CRF


Guete - CRFs
Prozent
84
86
88
90
92
94
96
8
16
32
64
128
256
512
Zwischenspeichergroesse (MiB)
Init ra size
Max 1 Seite
Max 3 Seiten
Max 5 Seiten
Max 7 Seiten
Max 10 Seiten
Linux

Güte - Naive Bayes



Güte - Orakel


Guete - Orakel
Prozent
96
94
92
90
88
86
84
82
80
78
76
74
72
70
68
4
8
16
32
64
128
256
512
Zwischenspeichergroesse (MiB)
Init ra size
Max 1 Seite
Max 3 Seiten
Max 5 Seiten
Max 7 Seiten
Max 10 Seiten
Linux

Ende

Vielen Dank für Ihre Aufmerksamkeit!

Quellen

- [1] *Linux Kernel Development – Third Edition*, Robert Love, Addison-Wesley, 2010
- [2] *Understanding the Linux Kernel – Third Edition*, Daniel P. Bovet; Marco Cesati, O'Reily, 2005
- [3] Linux Quellcode, Version 2.6.35.7
- [4] *Diskrete Simulation – Eine Einführung in Modula 2*, Bernd Page, Springer-Verlage, 1991
- [5] *Data Mining – Praktische Werkzeuge und Techniken für das maschinelle Lernen*, Ian H. Witten; Eibe Frank, Hanser, 2001
- [6] http://sourceware.org/systemtap/
- [7] http://www.pdl.cmu.edu/DiskSim/
- [8] http://webglimpse.net/