

[1]

# **Leistungsanalyse mit Vampir**

Alexis Engelke

[2]

# Vampir

- "Visualisation and Analysis of MPI Programs"
- Offline-Werkzeug
  - Daten werden während Programmlauf aufgezeichnet,
  - Analyse der Daten danach
- Sehr weit verbreitetes Werkzeug auf Clustern

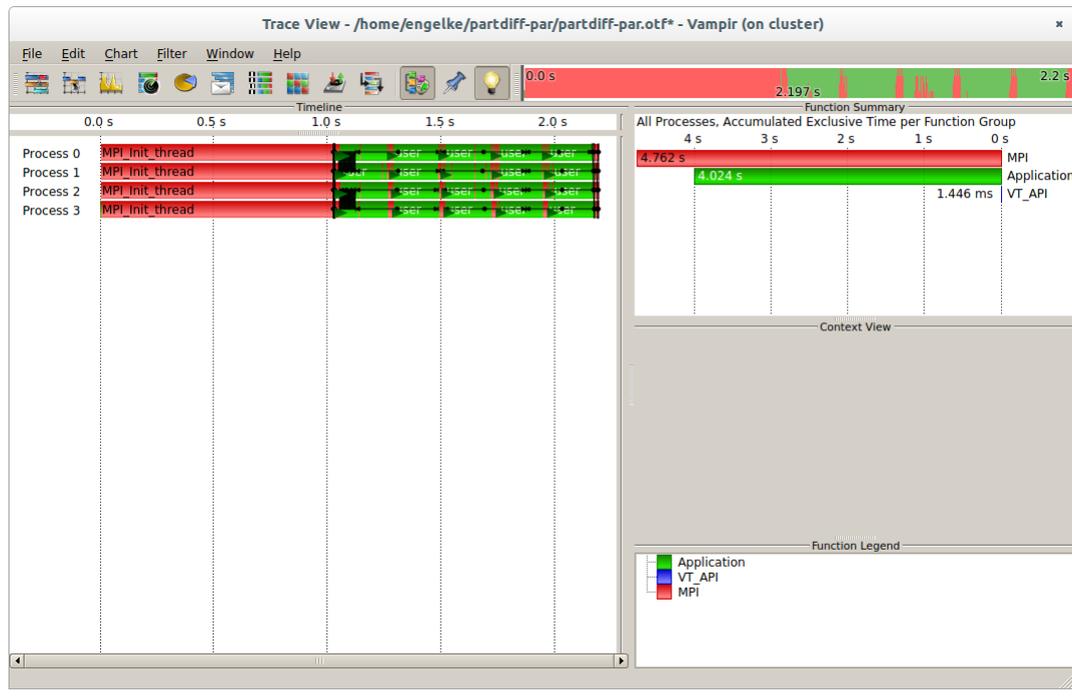
# Instrumentierung

- Vampir fügt in bestimmte Funktionen Code zum Tracing ein (Instrumentierung)
- Jeder Prozess schreibt Aufrufe in eigene Datei
- Kompilierung:
  - Ursprünglich: `mpicc ...`
  - Mit Vampir: `vtcc -vt:cc mpicc ...`
- Ansicht: `vampir <program-name>.otf`

Anzumerken ist, dass das Programm mit der Vampir-Instrumentierung langsamer ist. Außerdem sollte man darauf achten, dass die Laufzeit nicht zu lang ist, da alle Funktionsaufrufe protokolliert werden.

[4]

## Start-Ansicht

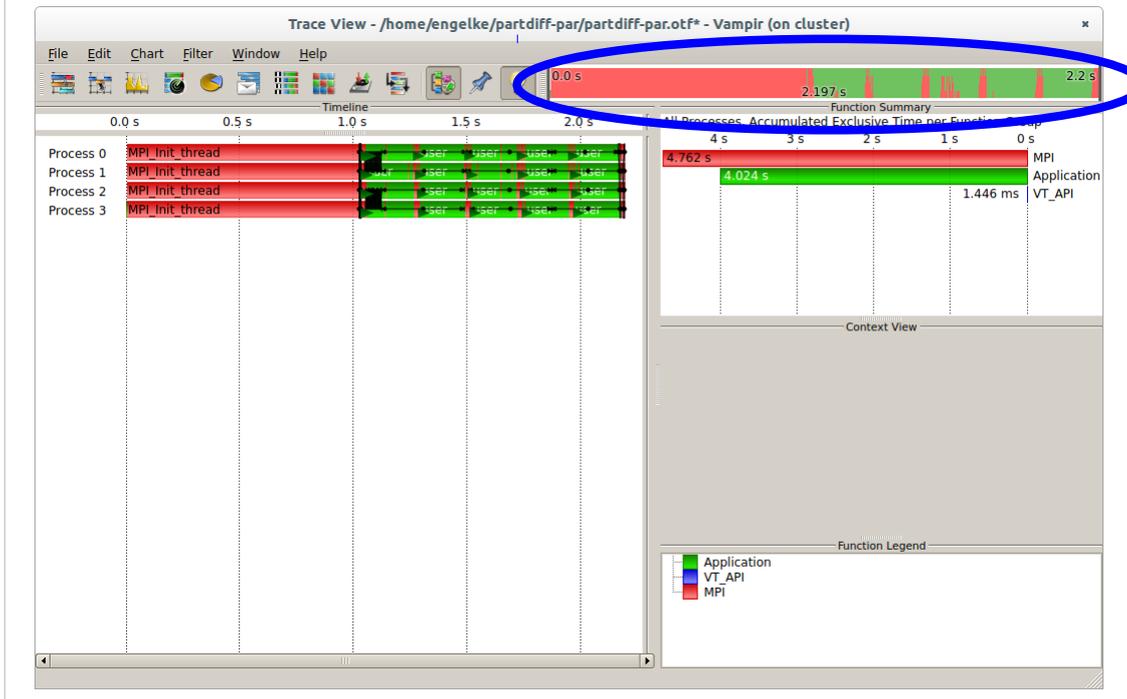


MPI-Funktionen sind rot, Vampir-Routinen sind blau, die eigenen Funktionen sind grün.

Das Beispiel, was im Folgenden verwendet wird, ist das Programm "partdiff-par". Dieses Programm löst partielle Differentialgleichungen und arbeitet auf einer Matrix.

[5]

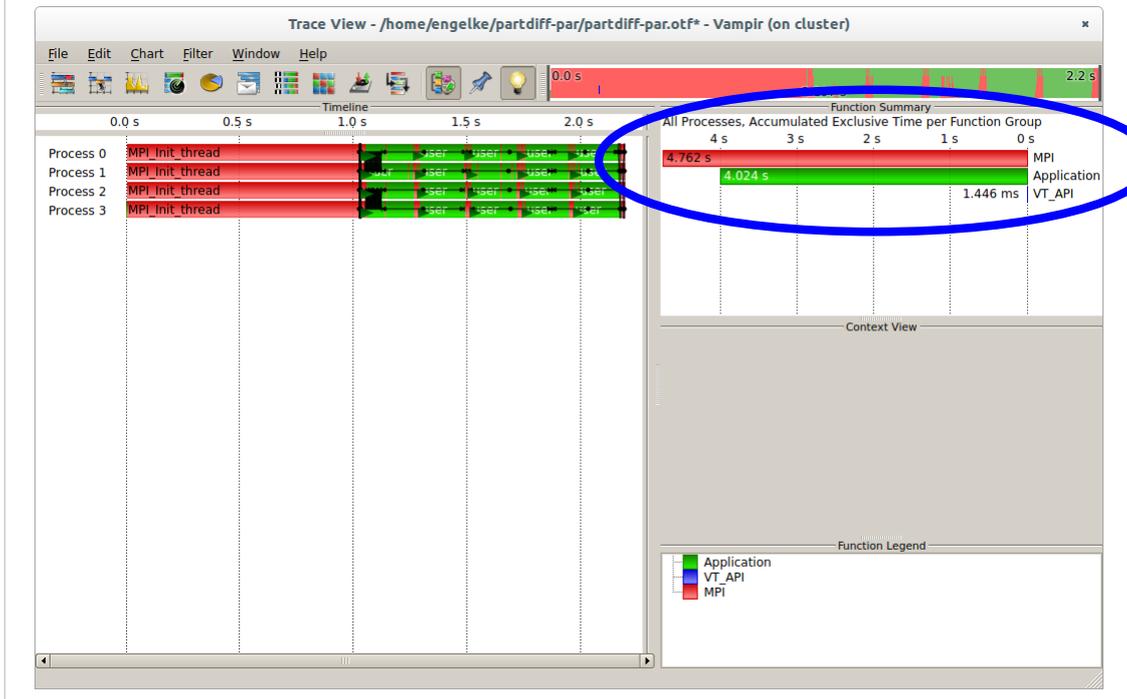
# Miniview



Der Miniview zeigt an, wann wieviele Prozesse rechnen bzw. Kommunizieren.

[6]

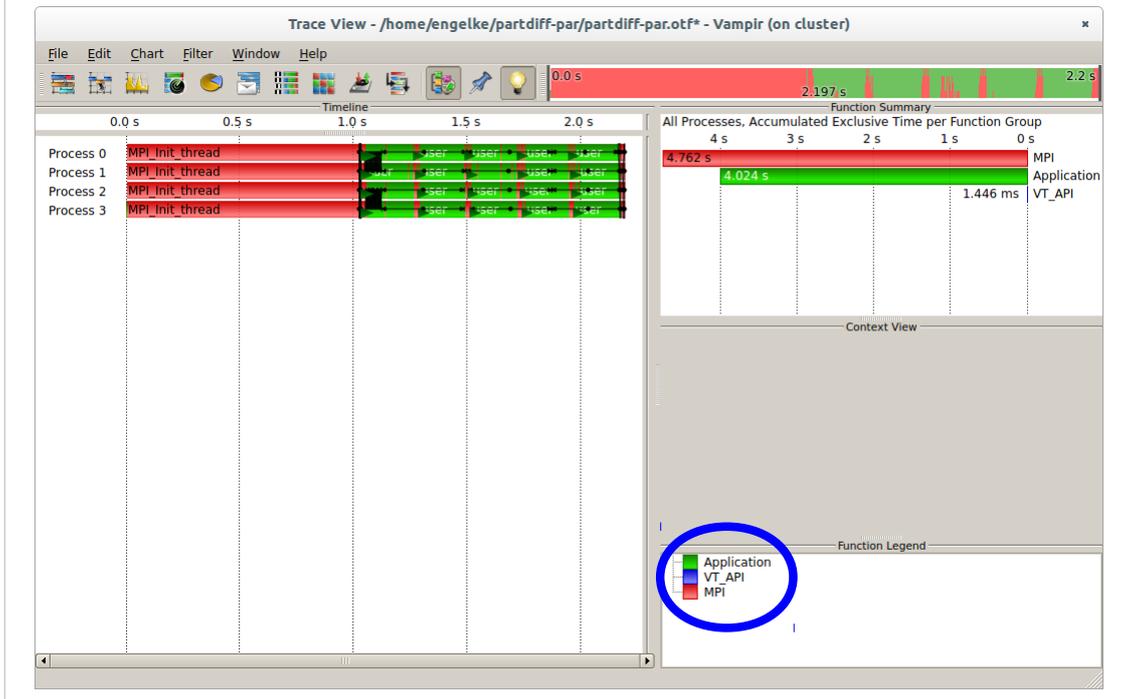
# Function-Summary



Die Function Summary zeigt, wieviel Zeit die Prozesse mit MPI Funktionen oder Berechnungen verbringen.

[7]

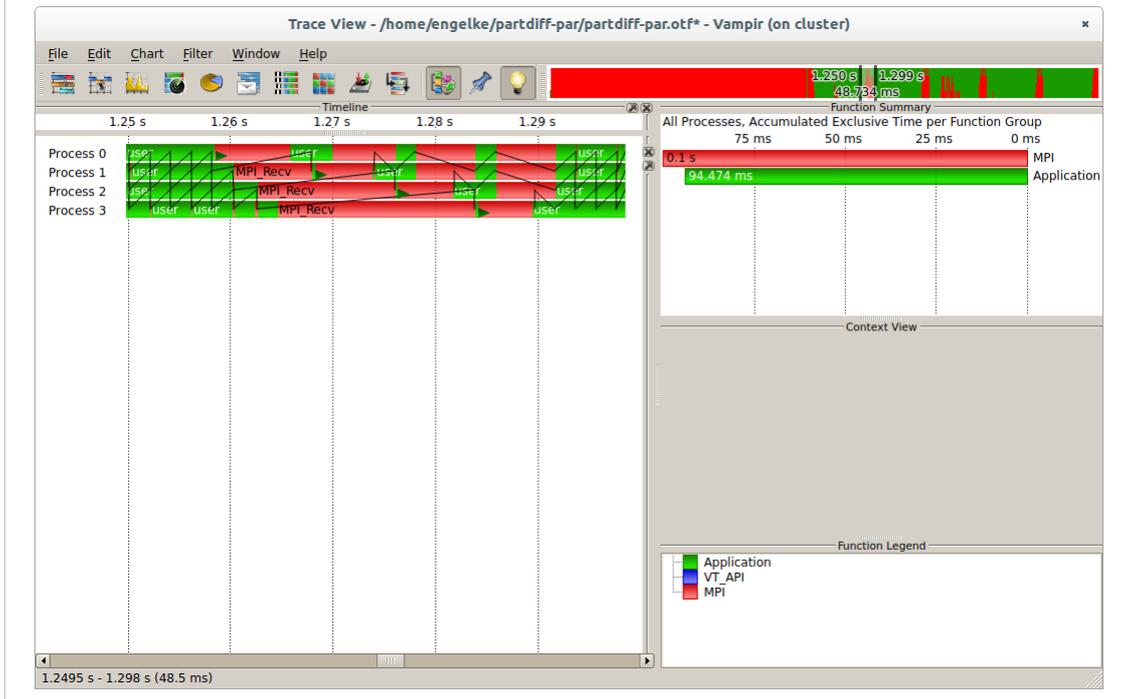
# Legende



MPI-Funktionen sind rot, Vampir-Routinen sind blau, die eigenen Funktionen sind grün.

[8]

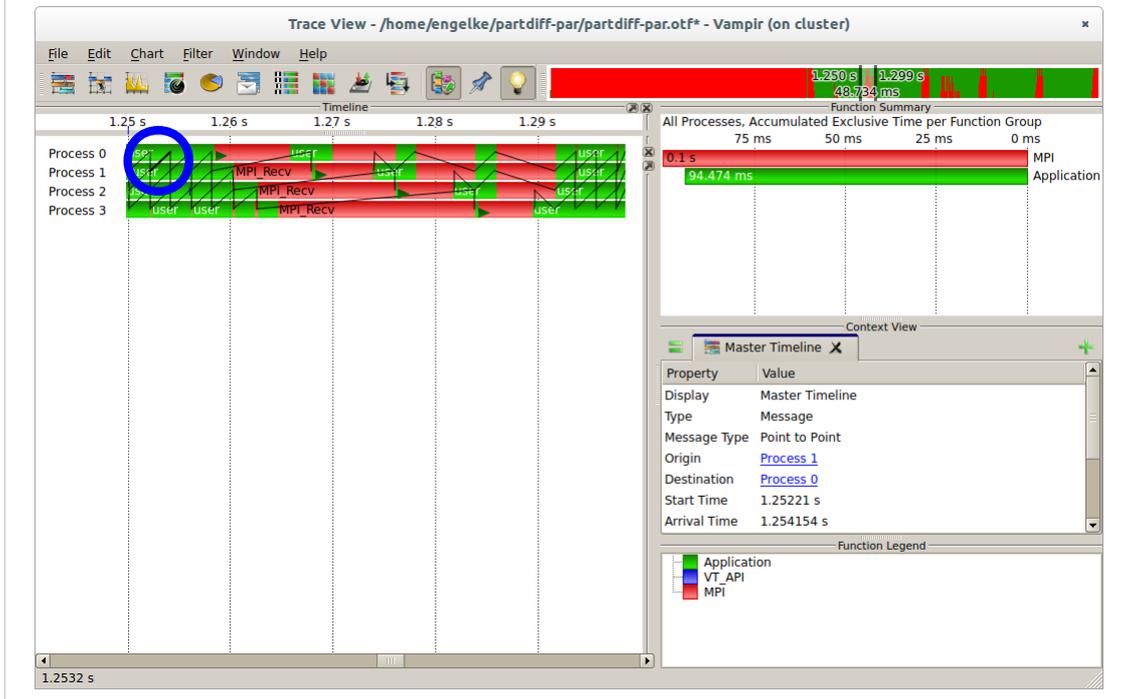
# Zoom



Durch horizontales Ziehen eines horizontalen Bereiches mit der Maus wird an einen Bereich herangezoomt.

[9]

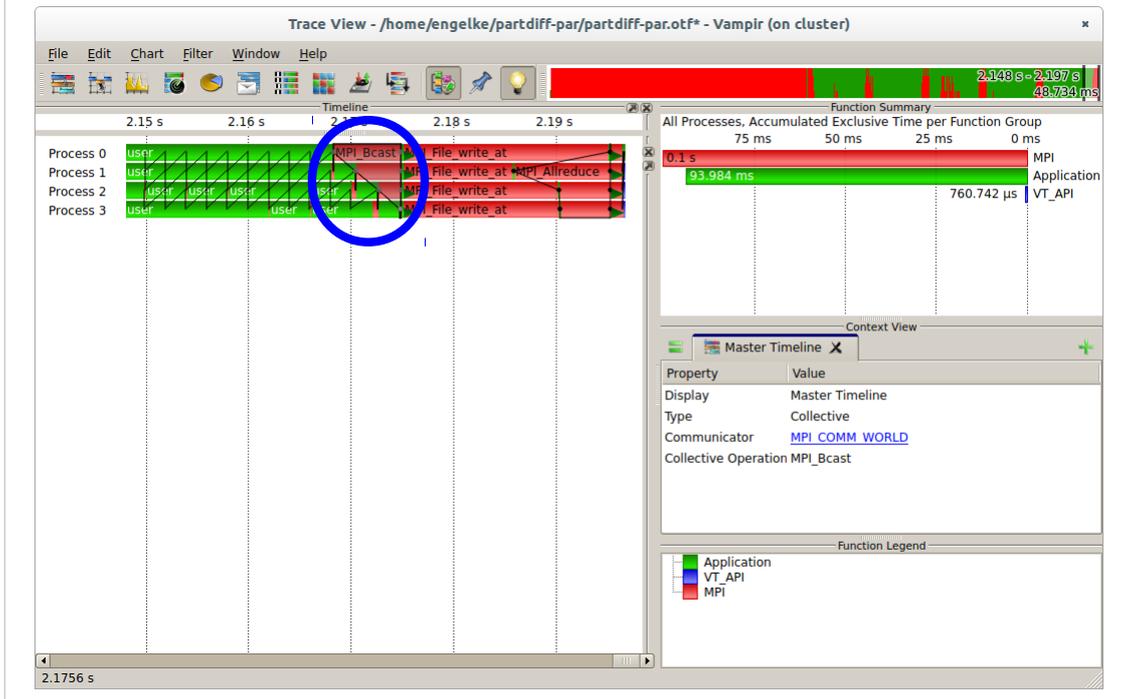
# Point-to-Point Message



Informationen erhält man durch Anklicken der Linie.

[10]

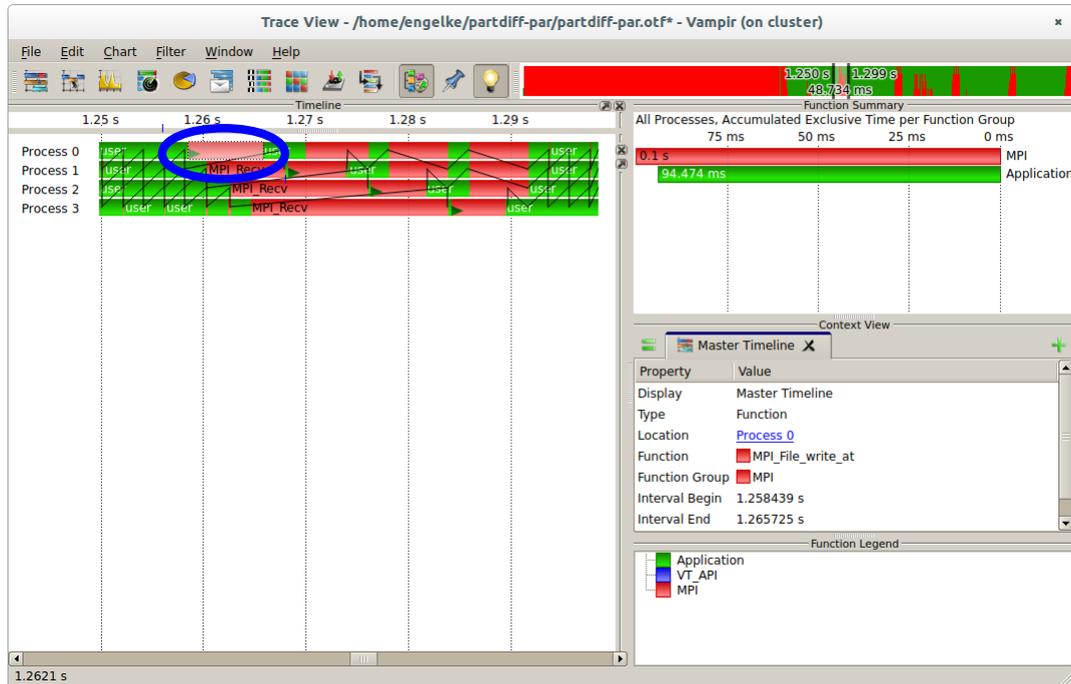
# Kollektive Operation



Kollektive Operationen sind immer durch kleine Kreise gekennzeichnet.

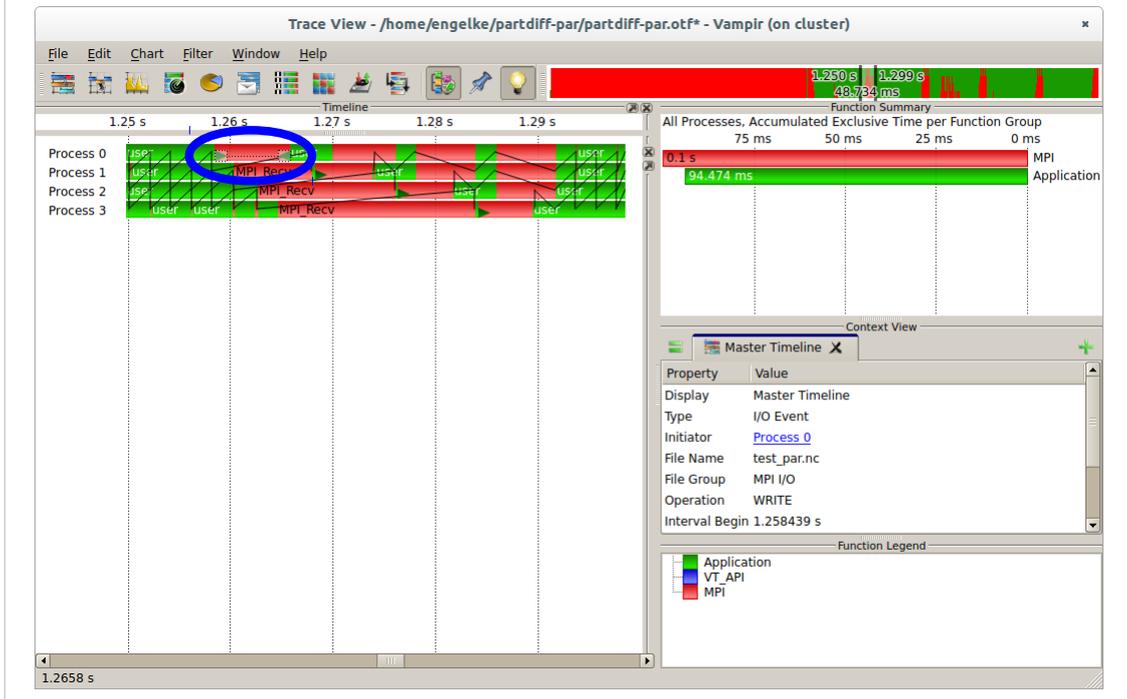
[11]

# Aufruf Information



[12]

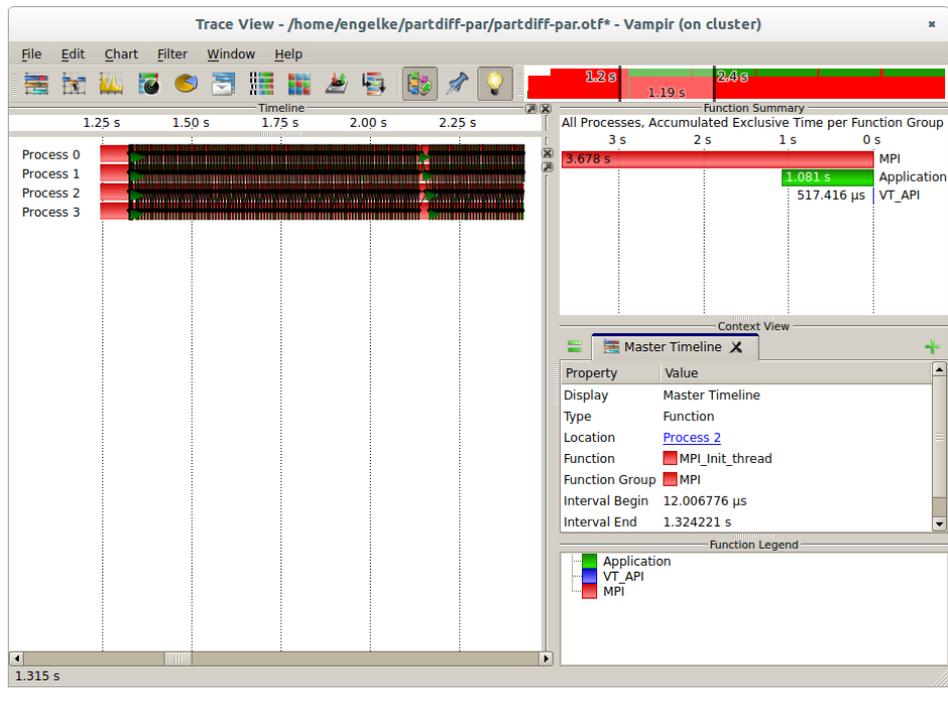
# MPI I/O



Diese Information erhält man durch Anklicken des grünen Pfeils.

[13]

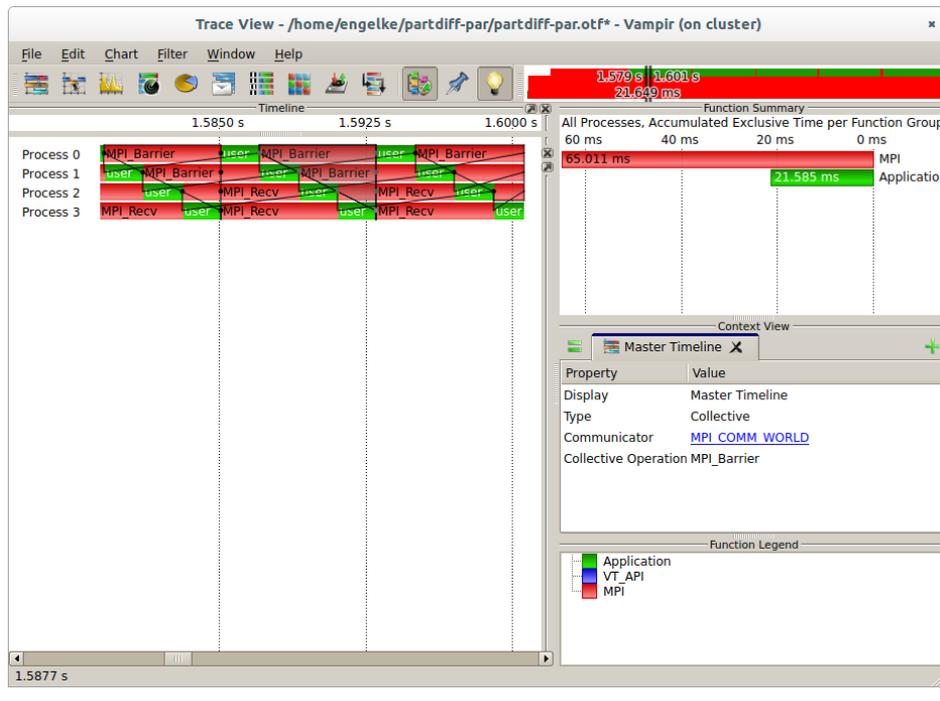
## Schlechte Parallelisierung



Bei dieser Parallelisierung arbeitet immer nur einer der vier Prozesse gleichzeitig. Dies kann man in dem Bereich oben recht erkennen.

[14]

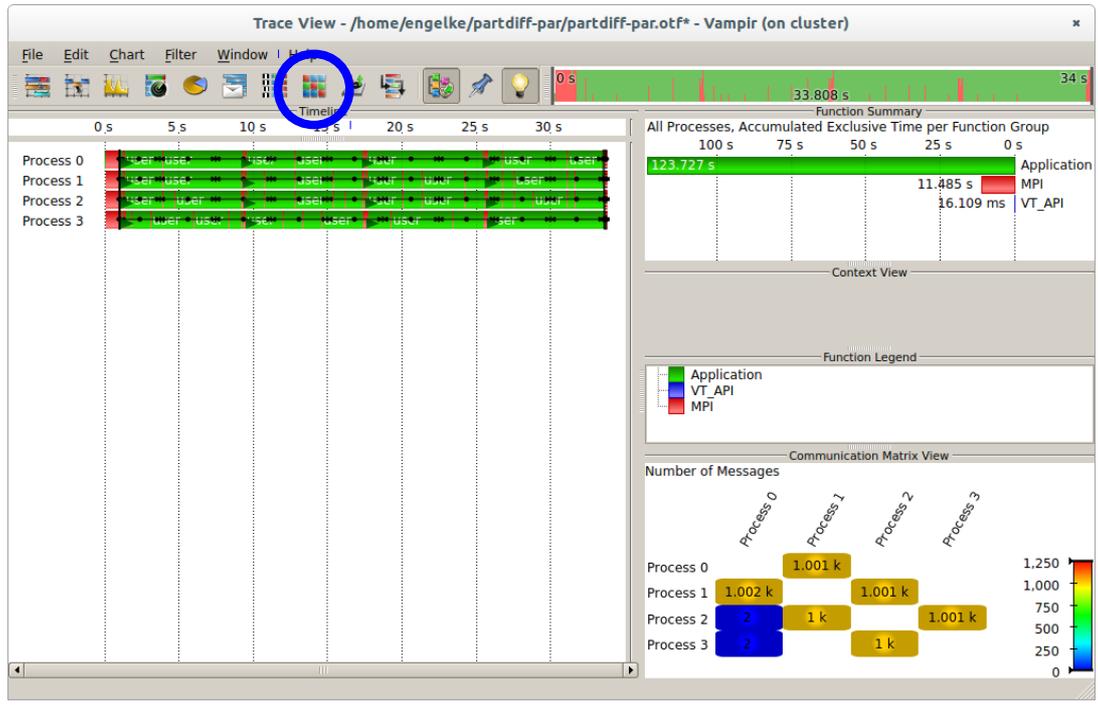
## Schlechte Parallelisierung



Bei genauem Hinsehen erkennt man, dass dies durch eine MPI\_Barrier verursacht wird.

[15]

# Kommunikationsmatrix



# Hardware Performance Counter

- Messen Leistung von bestimmten Komponenten der CPU
  - Cache Hits/Misses
  - Speicherdurchsatz
  - FLOPS
  - u.v.m.

## Hardware Counter mit Vampir

- Setzen einer Umgebungsvariable vor Programmstart:

```
export VT_METRICS='PAPI_FP_OPS:!PAPI_TOT_CYC'
```

- In Vampir:
  - Performance Radar
  - Counter Data Timeline

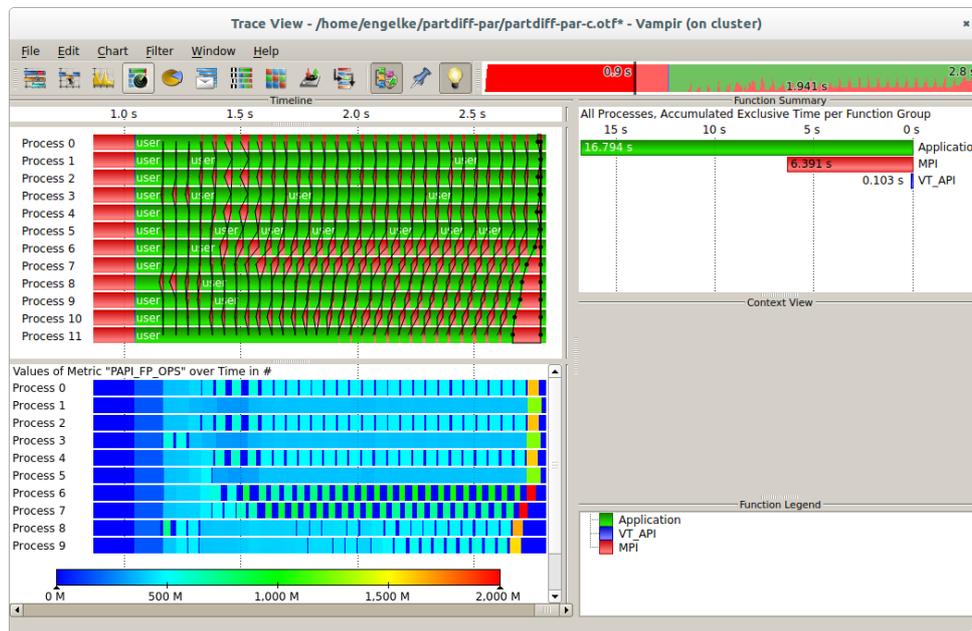
Das Ausrufungszeichen (“!”) bedeutet, dass die Werte addiert werden.

Auf dem WR-Cluster kann man so eine Übersicht aller unterstützten Counter erhalten:

```
$ /opt/papi/5.0.0/bin/papi_avail
```

[18]

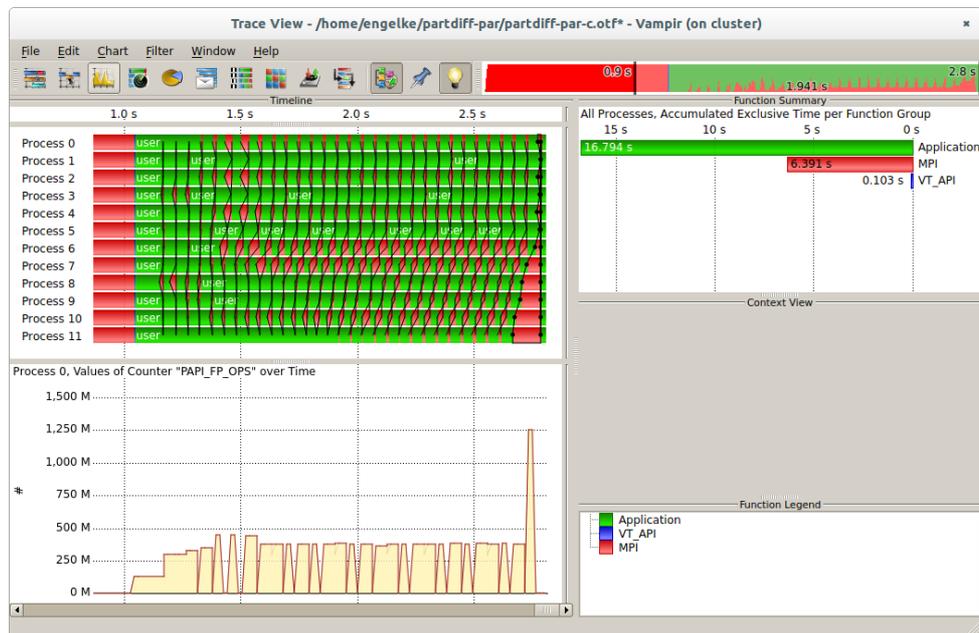
# Performance Radar



Über die Option "Select Metric..." im Kontextmenü kann man die Metrik ändern.

[19]

# Performance Radar



Über die Option "Select Metric..." im Kontextmenü kann man die Metrik ändern.

[20]