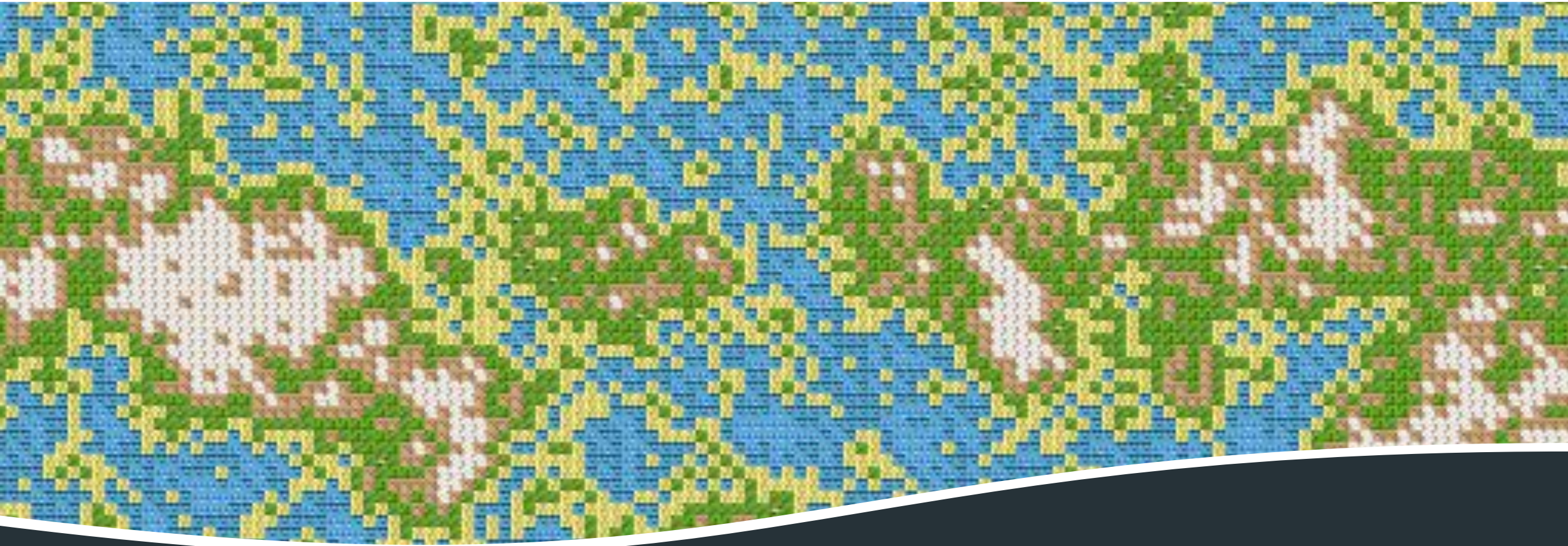






Parallelisierung des Wave Function Collapse Algorithmus in einem Kubernetes Cluster



Generierung von Landkarten







Motivation

-  Prozedurale Generierung in Games & Simulationen
-  WFC erzeugt lokal konsistente, global plausible Karten
-  Ziel: effiziente, skalierbare Lösung durch Parallelisierung
-  Einsatz in Kubernetes-Cluster



Zielsetzung der Arbeit

-  Skalierbare Softwarelösung in Kubernetes entwickeln
-   WFC-Algorithmus parallelisieren
-  Effizienz, Skalierbarkeit und Vielfalt evaluieren



Technologien

- Python & Flask
- Kubernetes & Docker
- RabbitMQ (Message Queue)
- MySQL & Oracle VirtualBox





Wave Function Collapse Algorithmus

- Idee basierend auf Superposition und Entropie
- Kollaps des wahrscheinlichsten Potential → Constraint Propagation
- Anwendungsbereiche: Texturen, Karten, 2D/3D-Welten



Beispiel: Sudoku

	4		
	2	3	
			3
4	3		2

3	4	2	1
1	2	3	4
2	1	4	3
4	3	1	2



Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

12 34	4	12 34	12 34
12 34	2	3	12 34
12 34	12 34	12 34	3
4	3	12 34	2



Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

12 3	4	12 3	12 3
12 34	2	3	12 34
12 34	12 34	12 34	3
4	3	12 34	2



Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

1 3	4	1 2	1
1	2	3	1 4
1 2	1	1 4	3
4	3	1	2



Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

1 3	4	1 2	1
1	2	3	1 4
1 2	1	1 4	3
4	3	1	2



Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

3	4	2	1
1	2	3	4
2	1	4	3
4	3	1	2



Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

			6					3
8					5			
			4			5	2	
				7	2			
	7	6			4			
5	4	2	3			8		
	3	8	1	4			9	5
7				3				
	2			6	8	3		7

Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

² 4 9	⁵ 9	^{4 5} 7 9	6	² 8 9	1	⁴ 7 9	⁴ 7 8	3
8	¹ 6 9	^{1 3} 4 7 9	⁷ 9	² 9	5	¹ 4 6 7 9	¹ 4 6 7	¹ 4 6 9
¹ 9	¹ 6 9	¹ 7 9	4	⁸ 9	3	5	2	¹ 6 8 9
^{1 3} 9	¹ 8 9	^{1 3} 9	⁸ 9	7	2	¹ 4 6 9	^{1 3} 4 5 6	¹ 4 6 9
^{1 3} 9	7	6	⁸ 9	¹ 5 8 9	4	¹ 9	¹ 5 3	^{1 2} 9
5	4	2	3	¹ 9	6	8	¹ 7	¹ 9
6	3	8	1	4	7	2	9	5
7	¹ 5	^{1 4 5} 4 5	2	3	9	¹ 4 6	¹ 4 6 8	¹ 4 6 8
¹ 4 9	2	¹ 4 9	5	6	8	3	¹ 4	7

Beispiel: Sudoku

Potential: ein Feld

Entropie: Anzahl der möglichen Zahlen in einem Feld

Constraint: keine wiederholenden Zahlen in:

- Reihe
- Spalte
- Block

² 4 9	5 9	^{4 5} 7 9	6	↯	1	⁴ 7 9	⁴ 7 8	3
8	¹ 6 9	^{1 3} 4 7 9	⁷ 9	2	5	¹ 4 7 9	¹ 4 6 7	¹ 4 6 9
¹ 9	¹ 6 9	¹ 7 9	4	8	3	5	2	¹ 6 8 9
^{1 3} 9	¹ 8 9	^{1 3} 9	⁸ 9	7	2	¹ 4 6 9	^{1 3} 4 5 6	¹ 4 6 9
^{1 3} 9	7	6	⁸ 9	1	4	¹ 9	¹ 5 3	^{1 2} 9
5	4	2	3	9	6	8	¹ 7	¹ 9
6	3	8	1	4	7	2	9	5
7	¹ 5	^{1 4 5}	2	3	9	¹ 4 6	¹ 4 6 8	¹ 4 6 8
¹ 4 9	2	¹ 4 9	5	6	8	3	¹ 4	7

Anwendungsbereiche

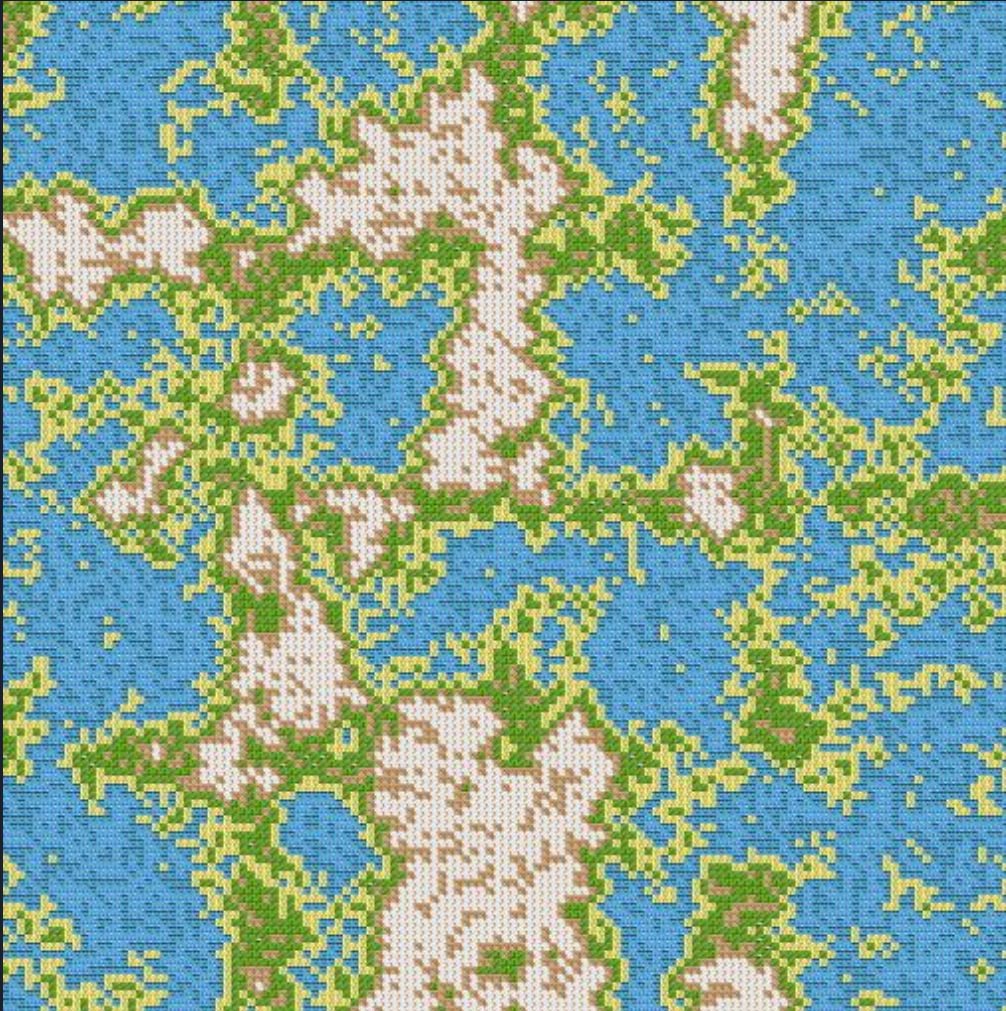
Sample =



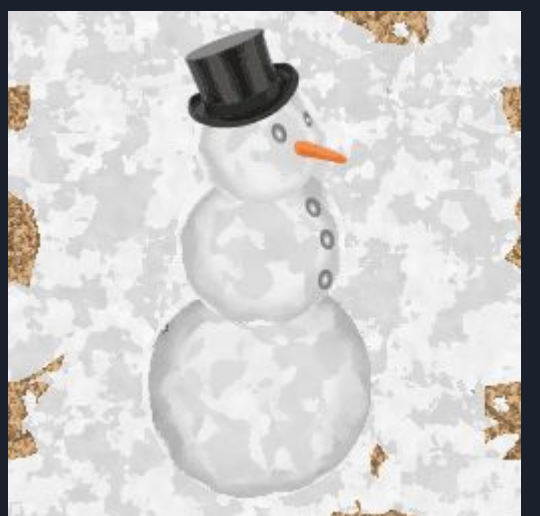
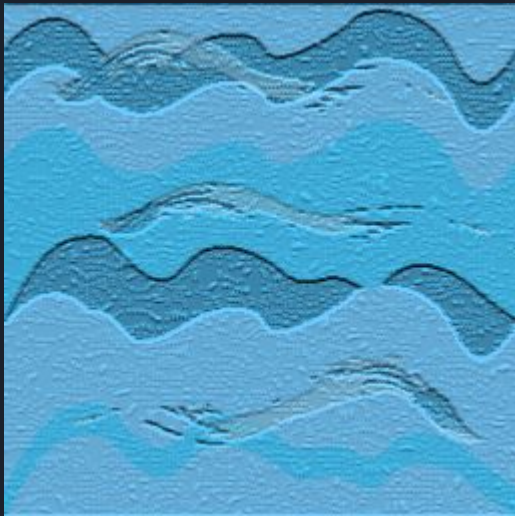
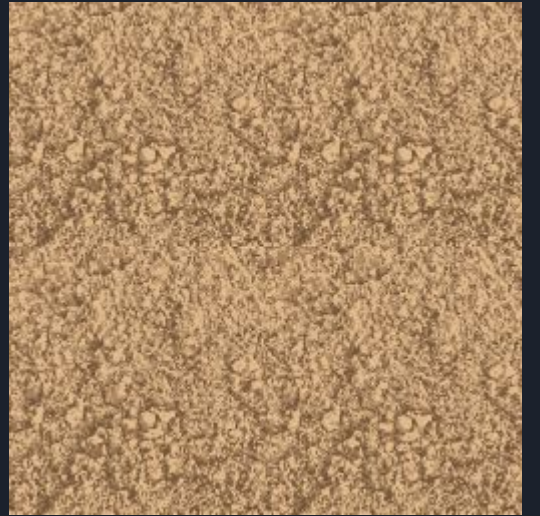
, $N = 3$

Max Gumin:
<https://github.com/mxgmn/WaveFunctionCollapse>

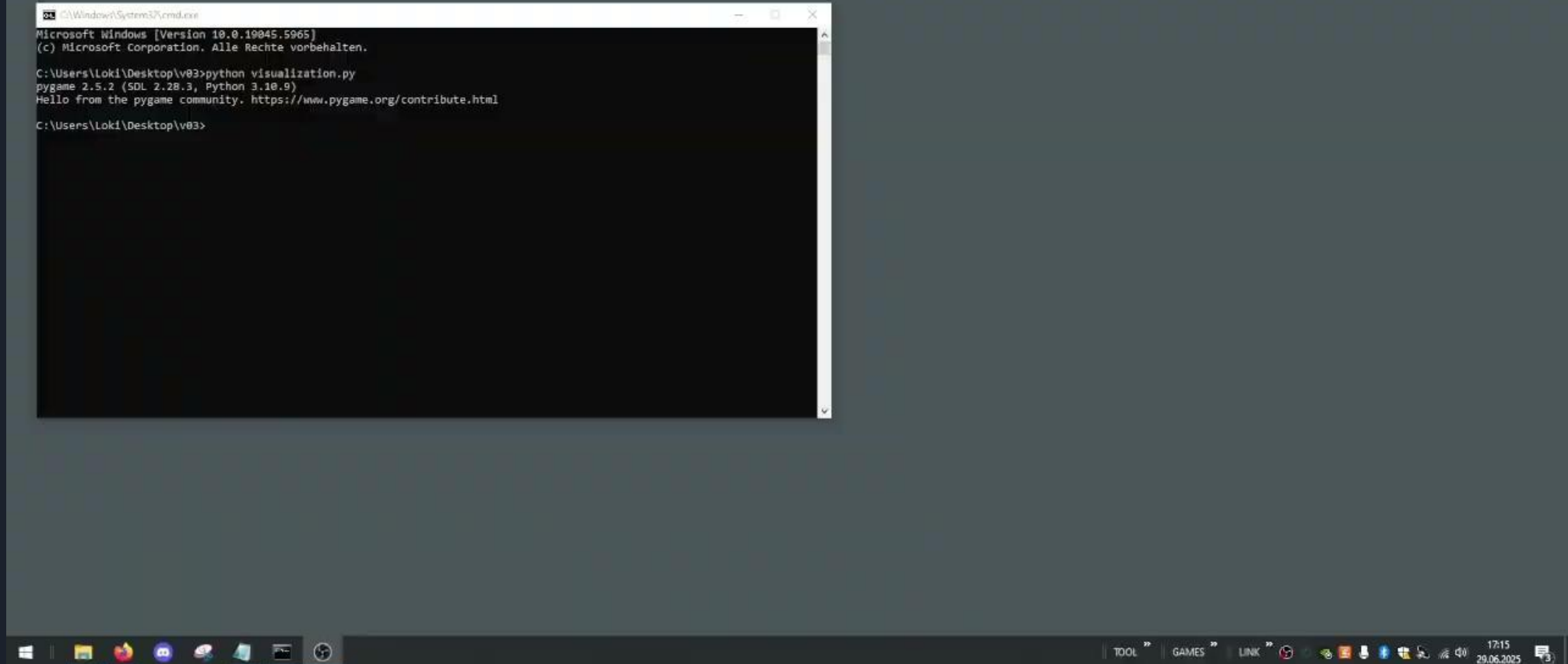
Maps / Tiles



Sprites



Video Demo



A screenshot of a Windows desktop environment. In the center, a command prompt window is open, displaying the following text:

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5965]
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Loki\Desktop\vb3>python visualization.py
pygame 2.5.2 (SDL 2.28.3, Python 3.10.9)
Hello from the pygame community. https://www.pygame.org/contribute.html

C:\Users\Loki\Desktop\vb3>
```

The taskbar at the bottom of the screen shows several application icons on the left, including the Start button, File Explorer, and various utility programs. On the right side of the taskbar, there are system tray icons for network, volume, and power, along with the date and time (17:15, 29.06.2025). Above the taskbar, there are some window management buttons labeled "TOOL", "GAMES", and "LINK".

Video Demo

```
C:\Windows\System32\cmd.exe

C:\Users\Loki\Desktop\v03>python visualization.py
pygame 2.5.2 (SDL 2.28.3, Python 3.10.9)
Hello from the pygame community. https://www.pygame.org/contribute.html

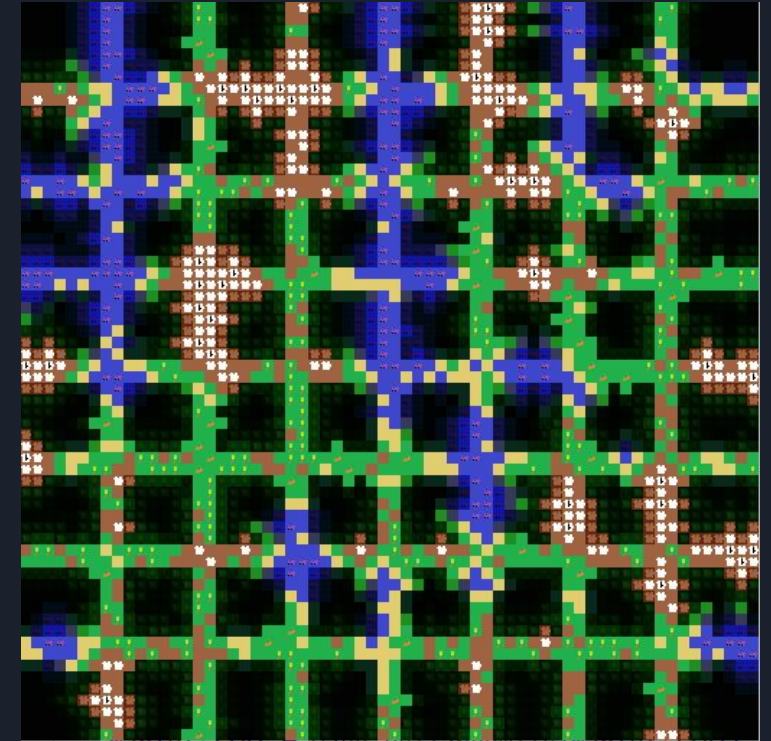
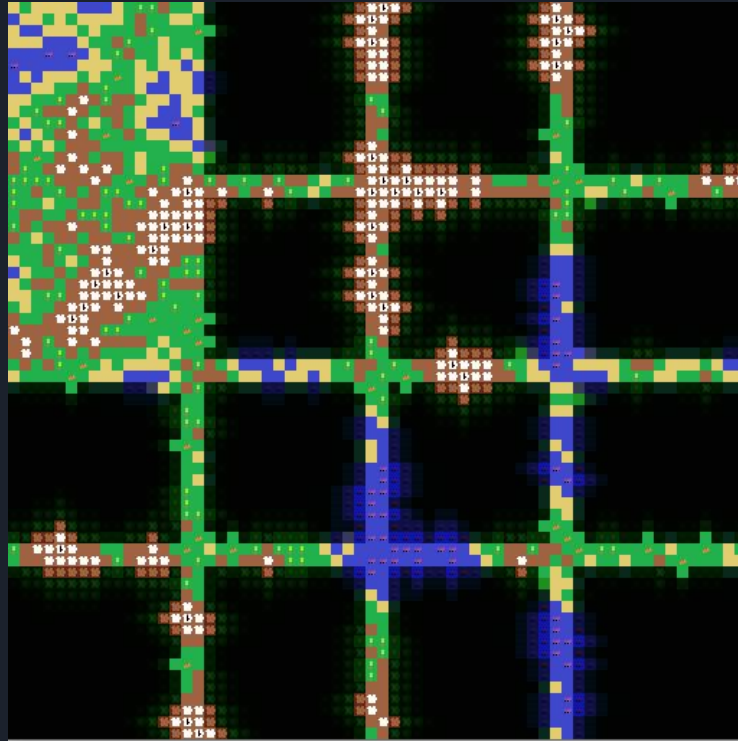
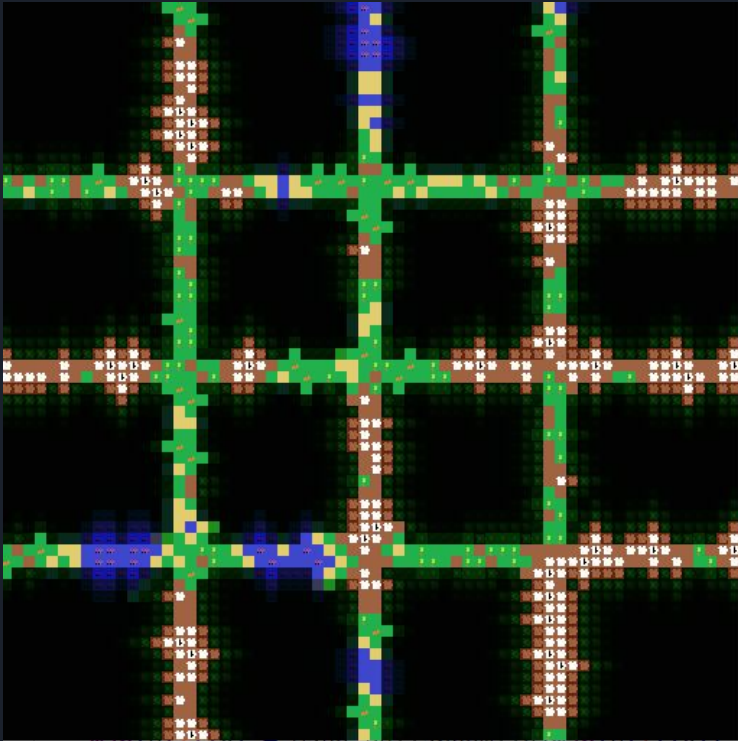
C:\Users\Loki\Desktop\v03>python visualization.py
pygame 2.5.2 (SDL 2.28.3, Python 3.10.9)
Hello from the pygame community. https://www.pygame.org/contribute.html

C:\Users\Loki\Desktop\v03>python visualization.py
pygame 2.5.2 (SDL 2.28.3, Python 3.10.9)
Hello from the pygame community. https://www.pygame.org/contribute.html

C:\Users\Loki\Desktop\v03>
```


Parallelisierung

- Maps werden in Chunks unterteilt
- Chunks mit vorberechneten Rändern unabhängig berechnet
- Verarbeitung durch mehrere Worker parallel
- Zusammenführung zu konsistenter Map





Parallelisierung - Limitation

Maps können nicht in beliebig viele Parts geteilt werden:

$$p = n^2 \quad \text{mit} \quad n \in \mathbb{N} \quad \text{und} \quad p \mid k^2$$

Rules

- Number of Tiles
- Number of Parts
- Entropy Tolerance
- Number of Workers



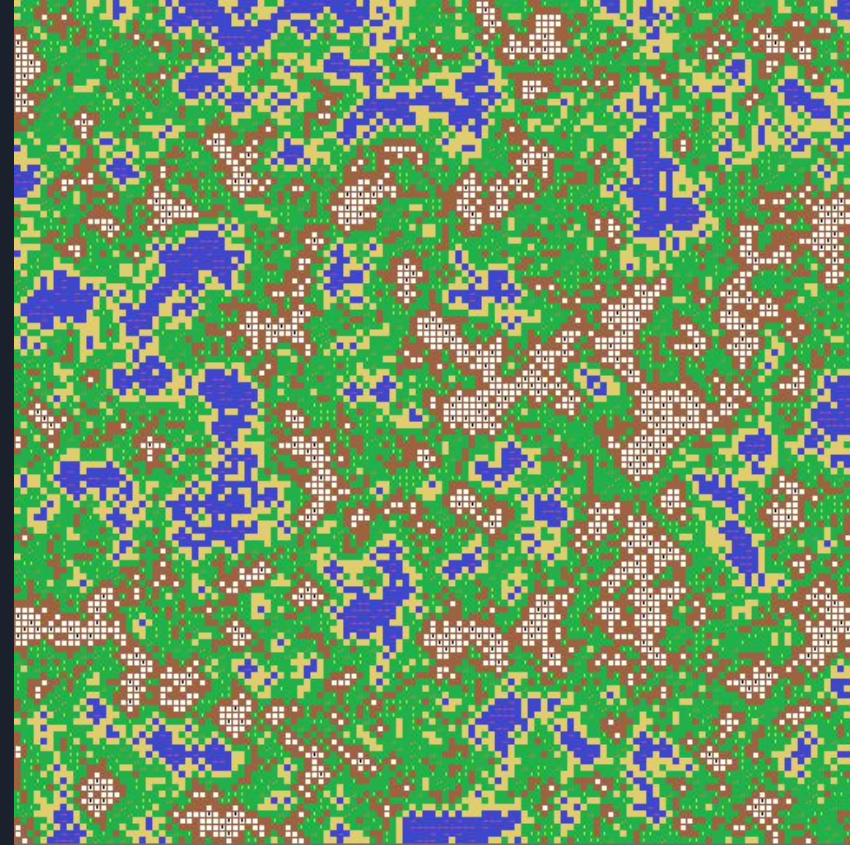
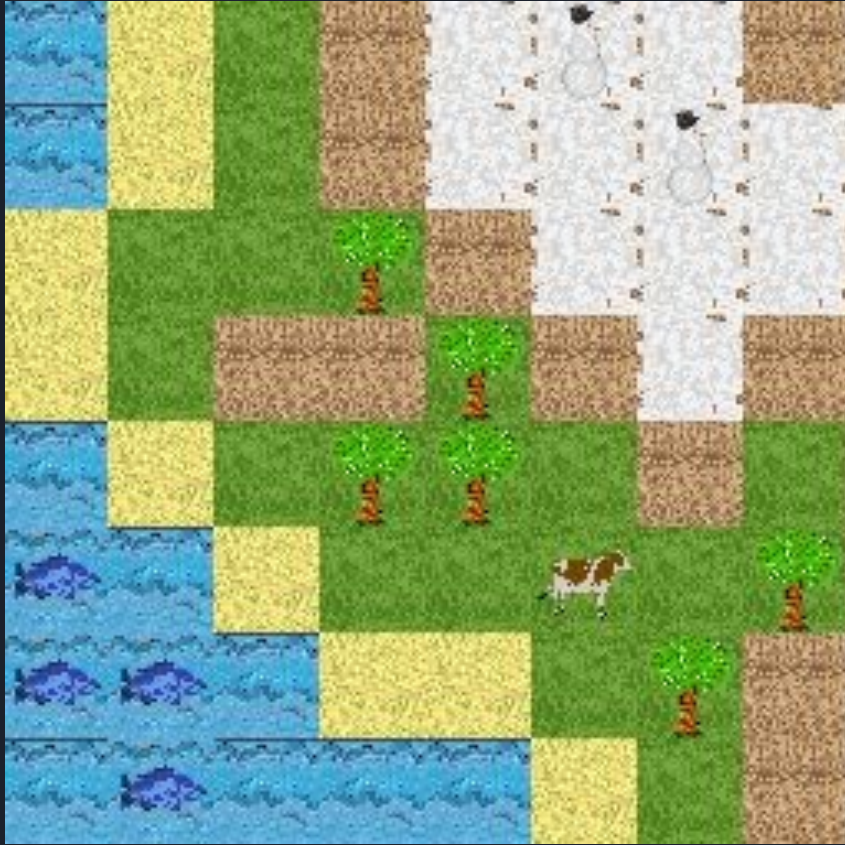


Restrictions

	Gras	Wald	Kuh	Strand	Wasser	Fisch	Berg	Schnee	Mann
Gras	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Wald	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Kuh	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Strand	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Wasser	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Fisch	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Berg	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Schnee	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Mann	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

	Binär
Gras	0b001001111
Wald	0b001000011
Kuh	0b000000001
Strand	0b000011001
Wasser	0b000111000
Fisch	0b000110000
Berg	0b011000011
Schnee	0b111000000
Mann	0b010000000

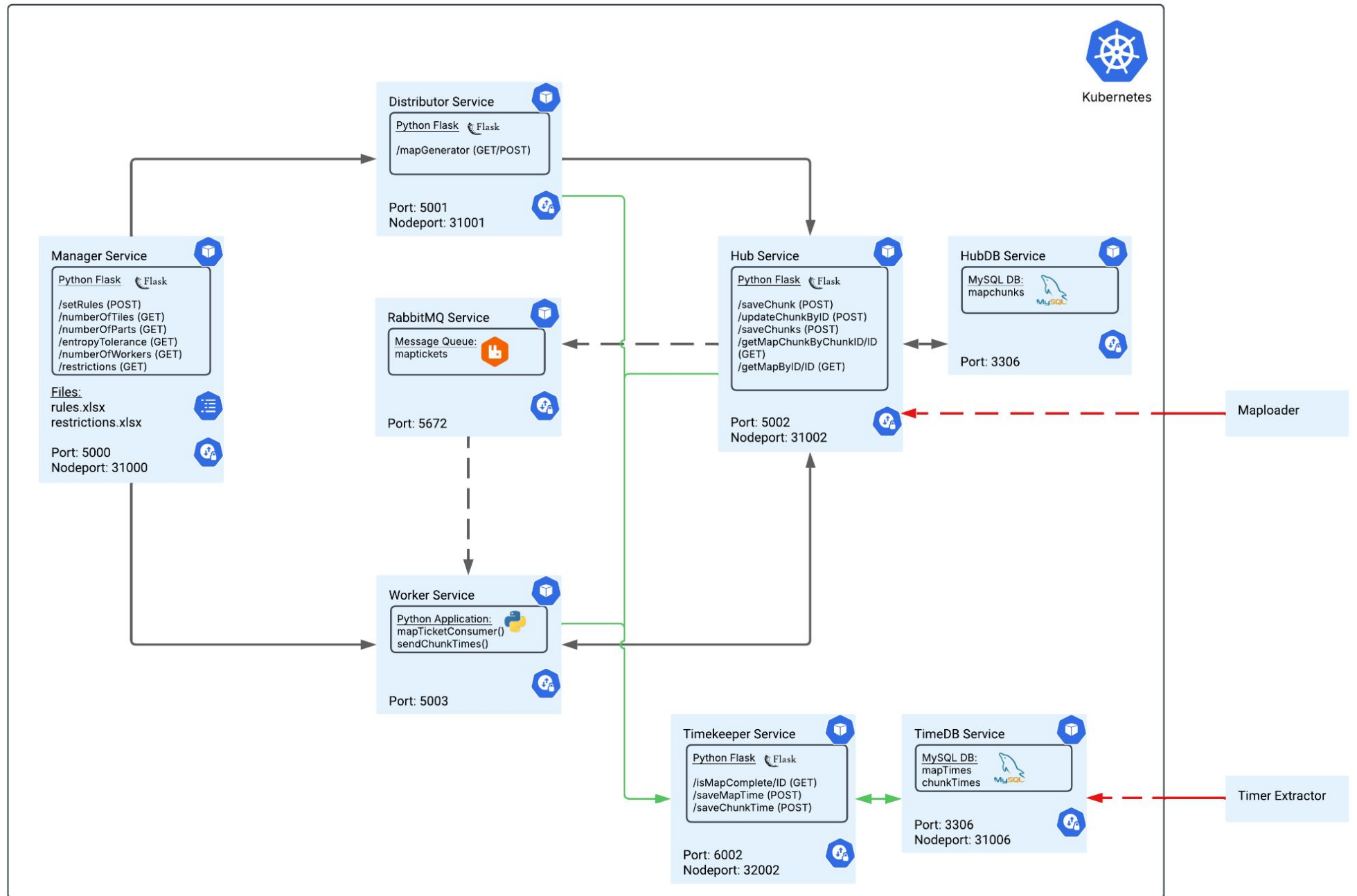
Restricted Maps



Architektur



WFC Kubernetes Cluster



Evaluation - Aufbau und Methodik

Systeme:

- Cluster: 8x Linux Ubuntu Server 22.04
- Extern: 1x MacOS 12.6.6





Evaluation - Aufbau und Methodik

Parameter:

- Mapgröße (16, 32, 48, ..., 256)
- Anzahl Parts (1, 4, 16, ..., 256)
- Anzahl Worker (1, 2, 4, ..., 256)
- (Entropietoleranz)

Jede Messung 3x

676 Maps / 50540 Chunks

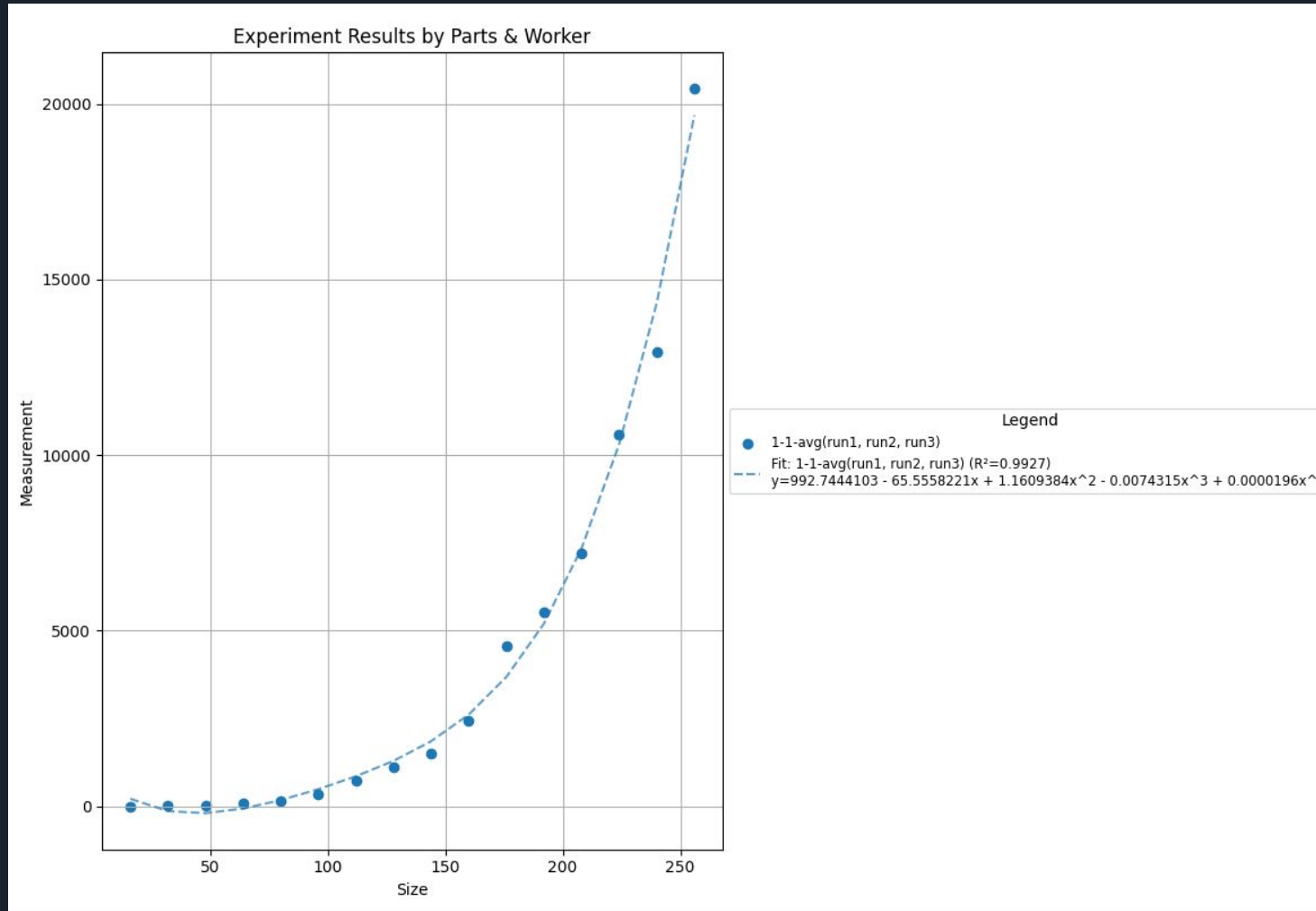


Evaluation - Messreihe

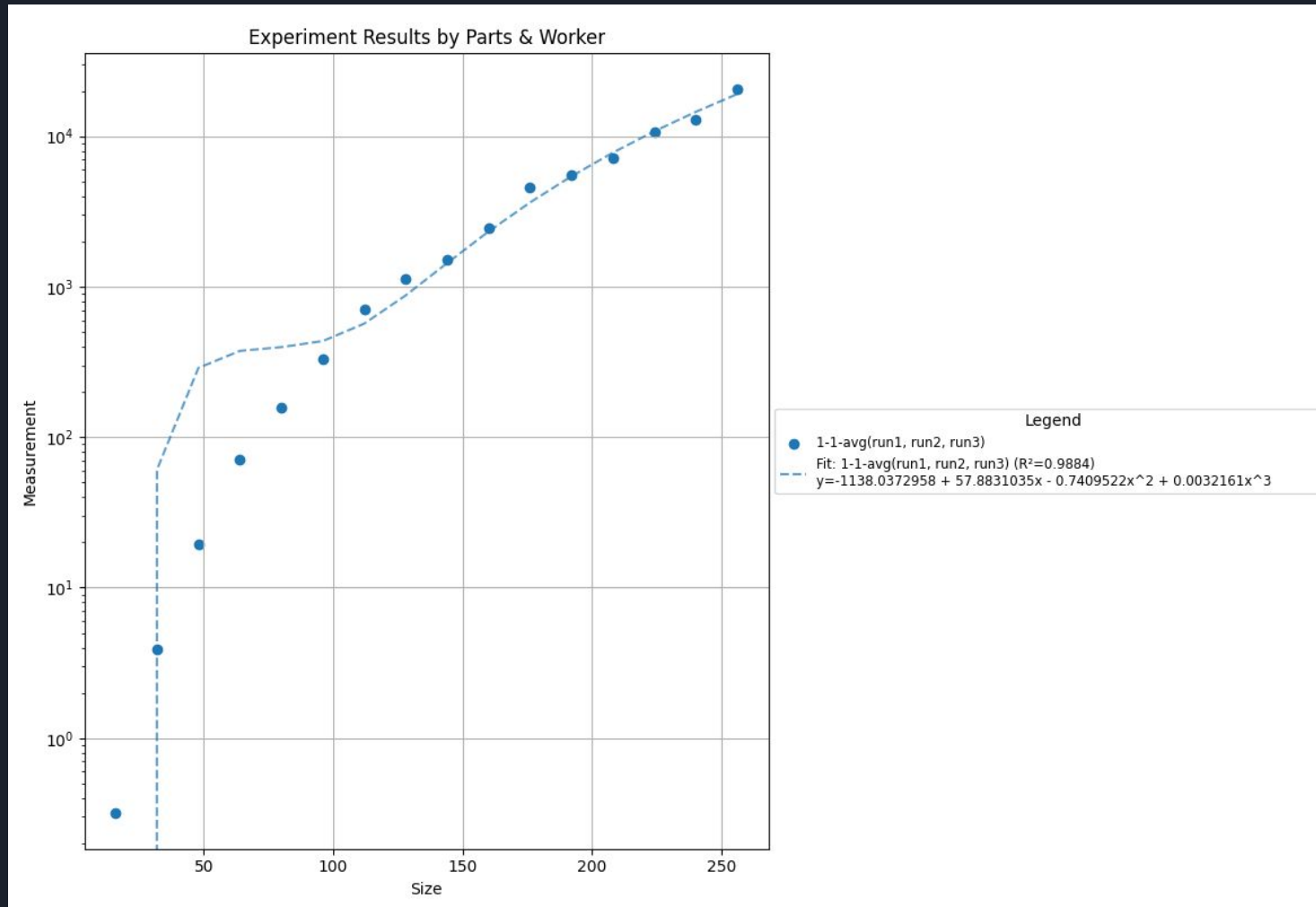
Row	SIZE	PARTS	WORKER	Run1	Run2	Run3
201	256	64	32	15058	15609	15856
202	256	64	64	10680	10553	10652
203	256	256	64	51584	55462	55294
204	256	256	128	56363	57736	59080
205	256	256	256	49638	47459	47535

→ messreihen.csv

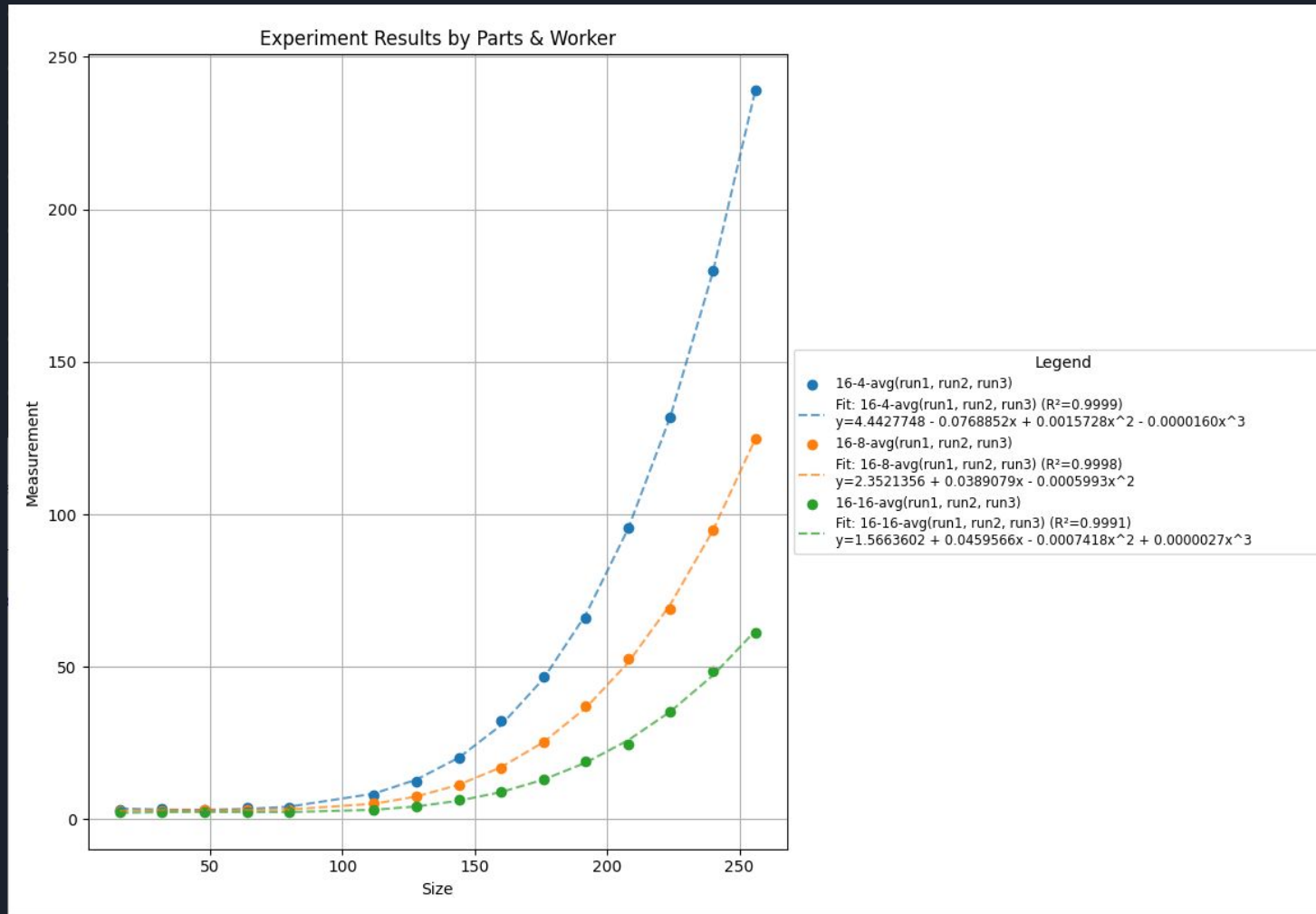
Evaluation - Auswertung



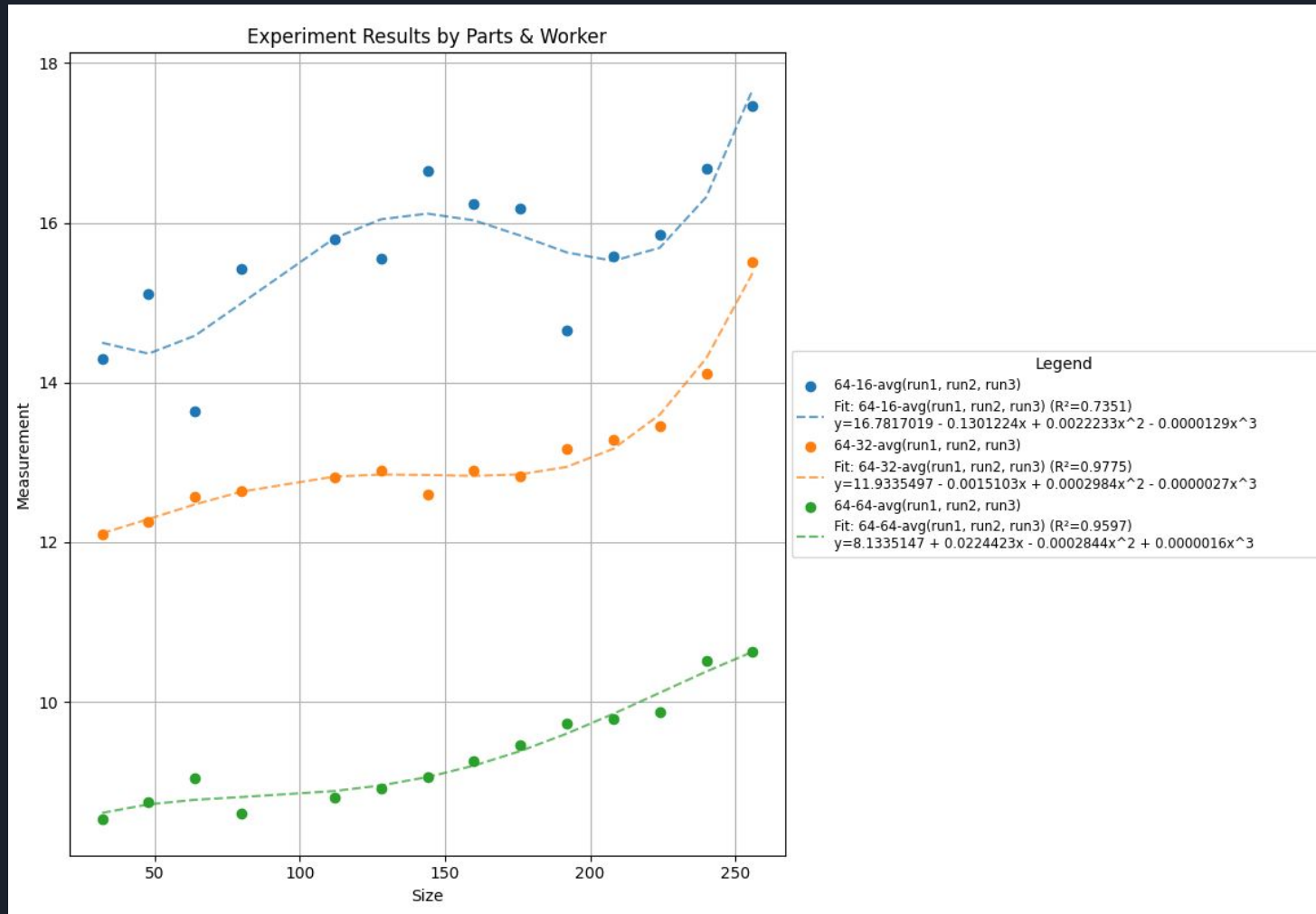
Evaluation - Auswertung



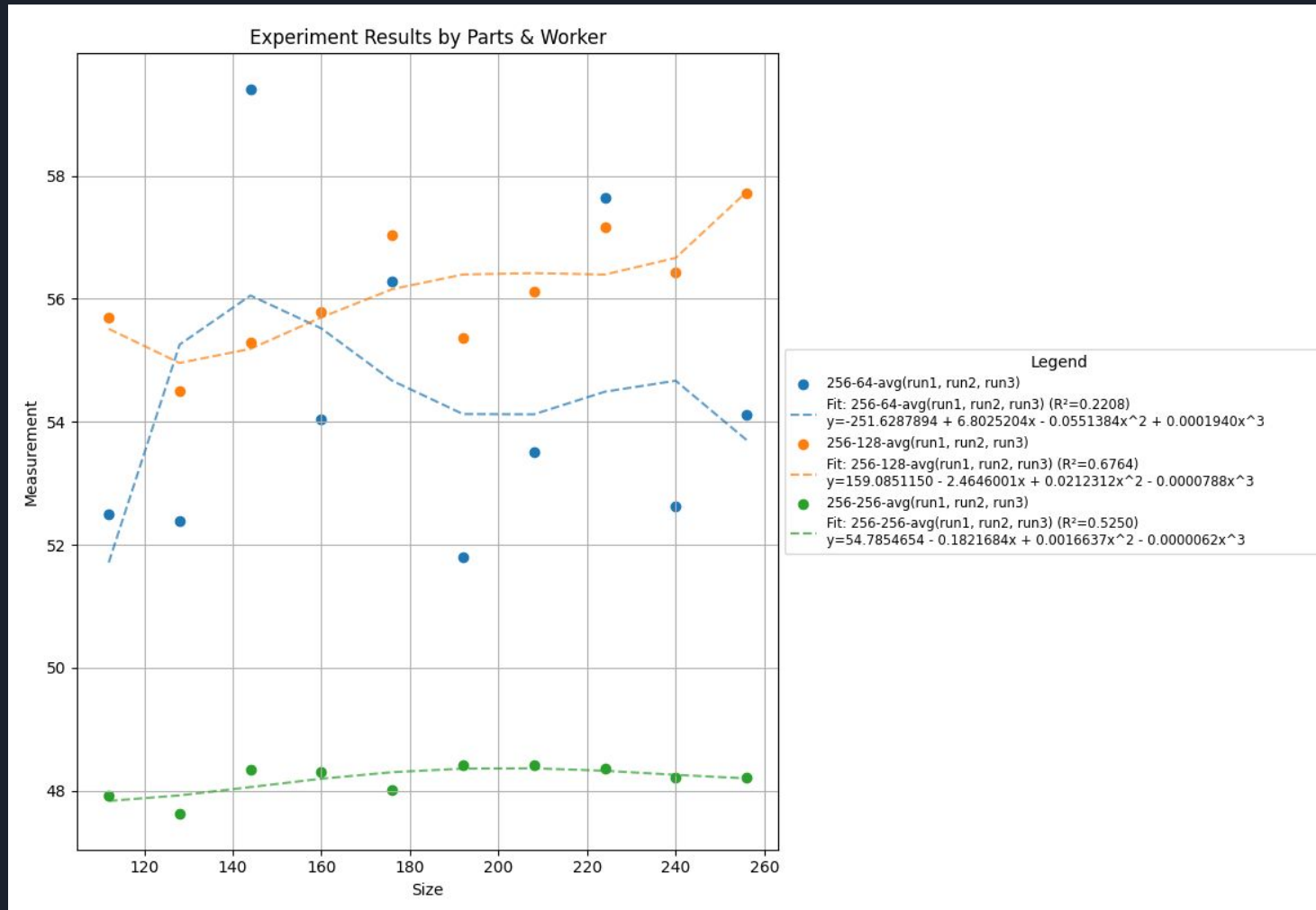
Evaluation - Auswertung



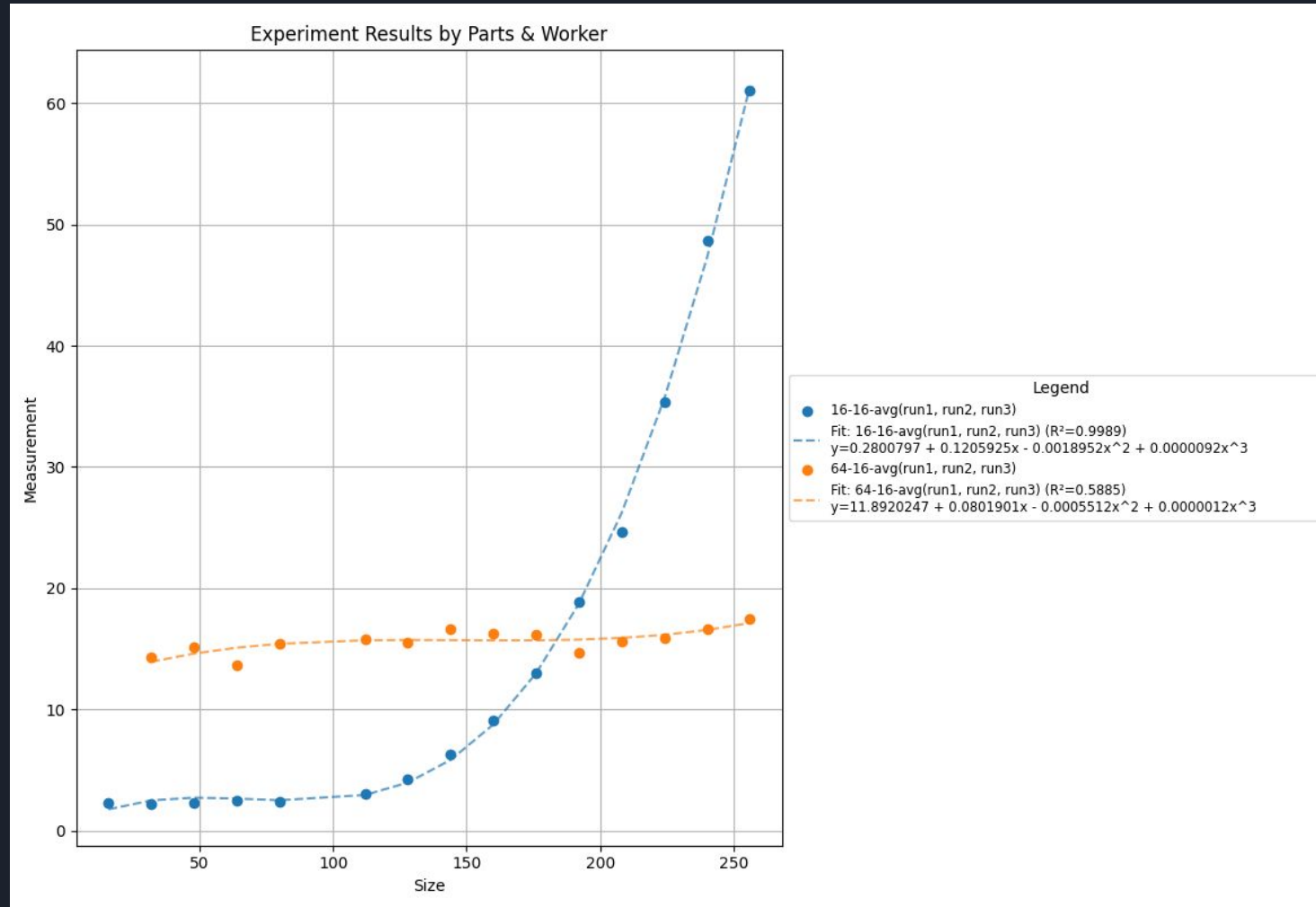
Evaluation - Auswertung



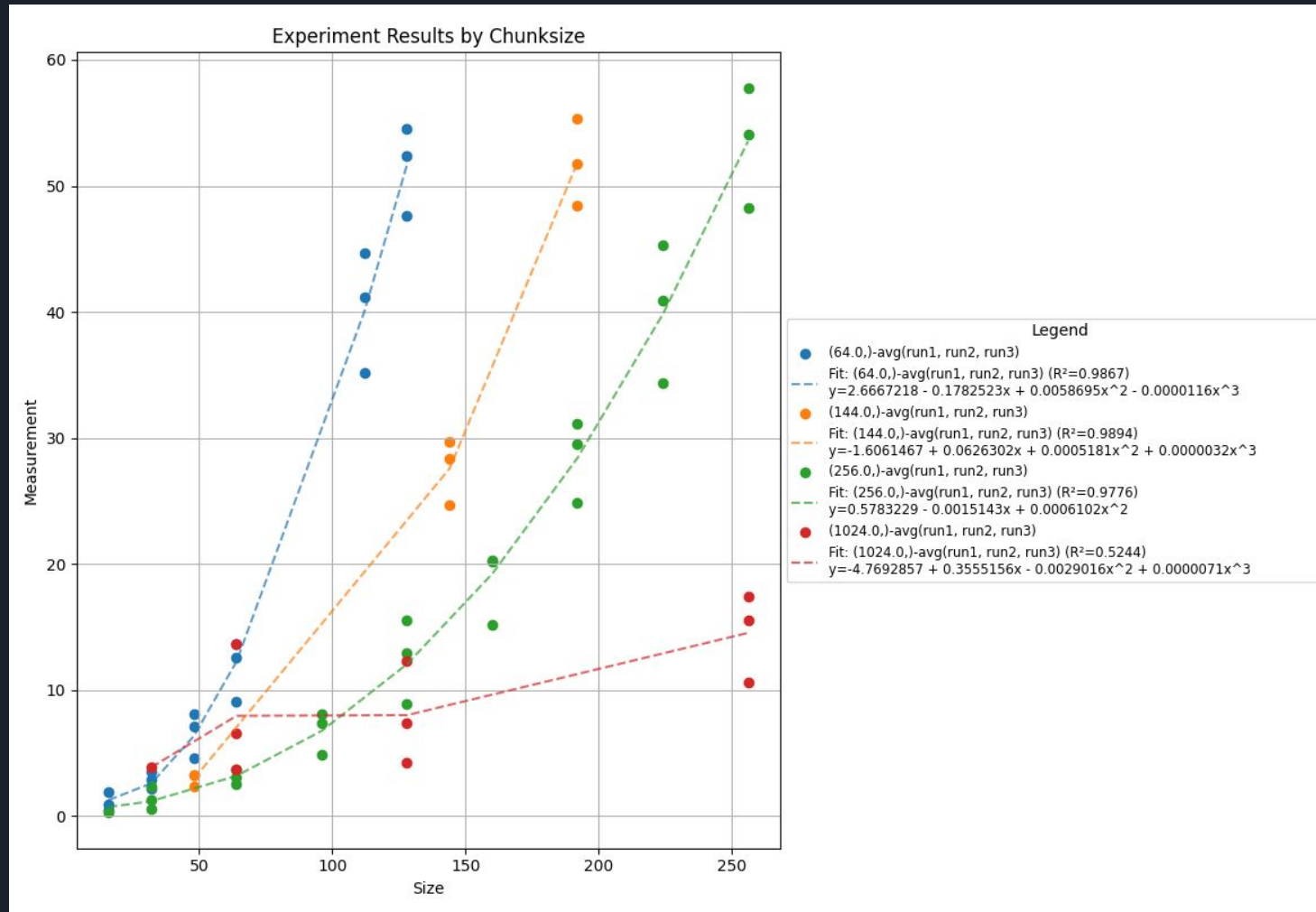
Evaluation - Auswertung



Evaluation - Auswertung



Evaluation - Auswertung





Evaluation - Ergebnisse

- Signifikante Beschleunigung durch Parallelisierung
- Größere Maps profitieren stärker
- Entropie-Toleranz beeinflusst Vielfalt & Konsistenz
- Diagramme zeigen lineare bis polynomiale Optimierung



Herausforderungen und Limitationen

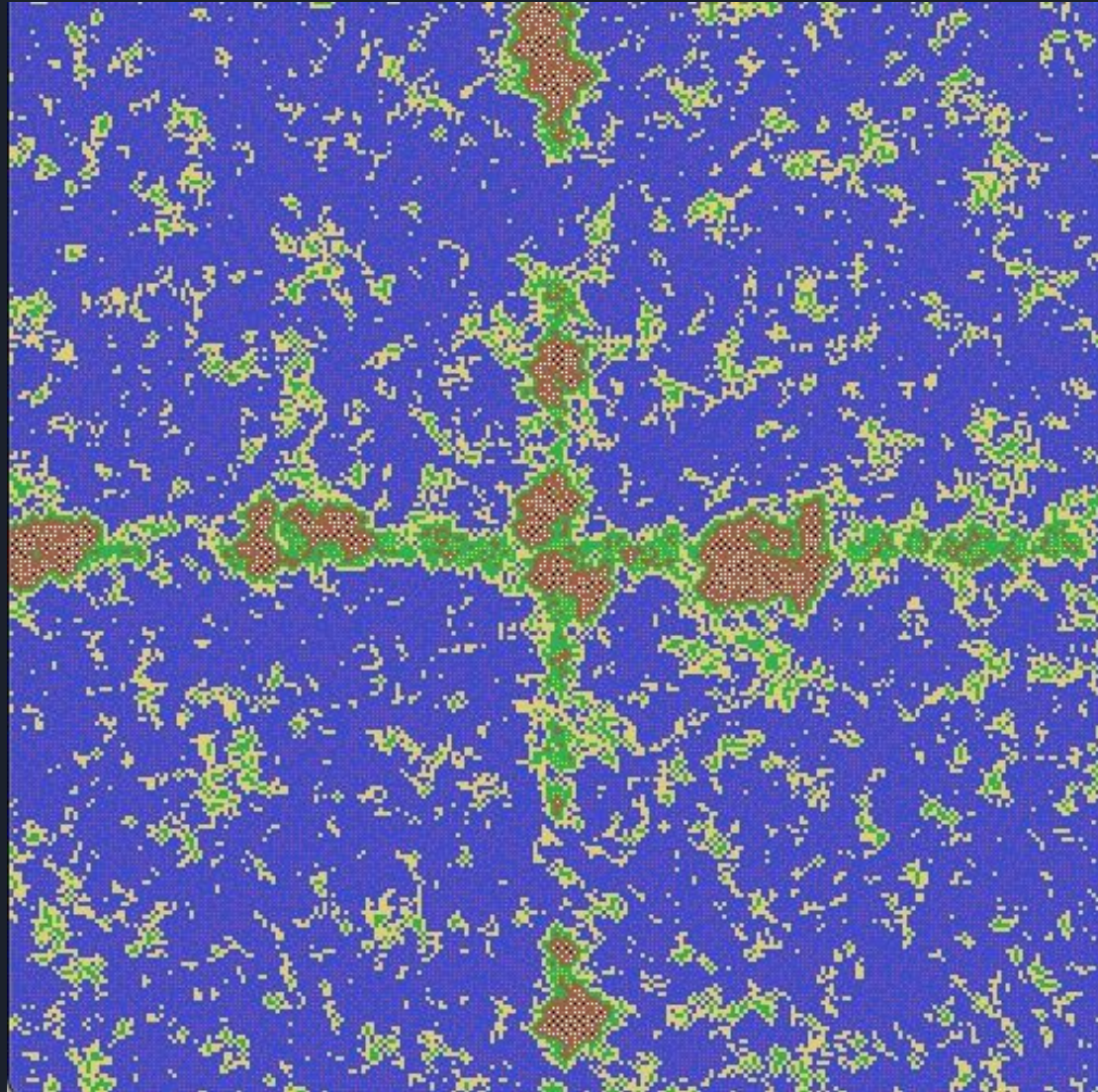
- Broker vs. Message-Queue
- Dockerfile Layer
- Robustheit Worker/RabbitMQ
- Invalide Messungen



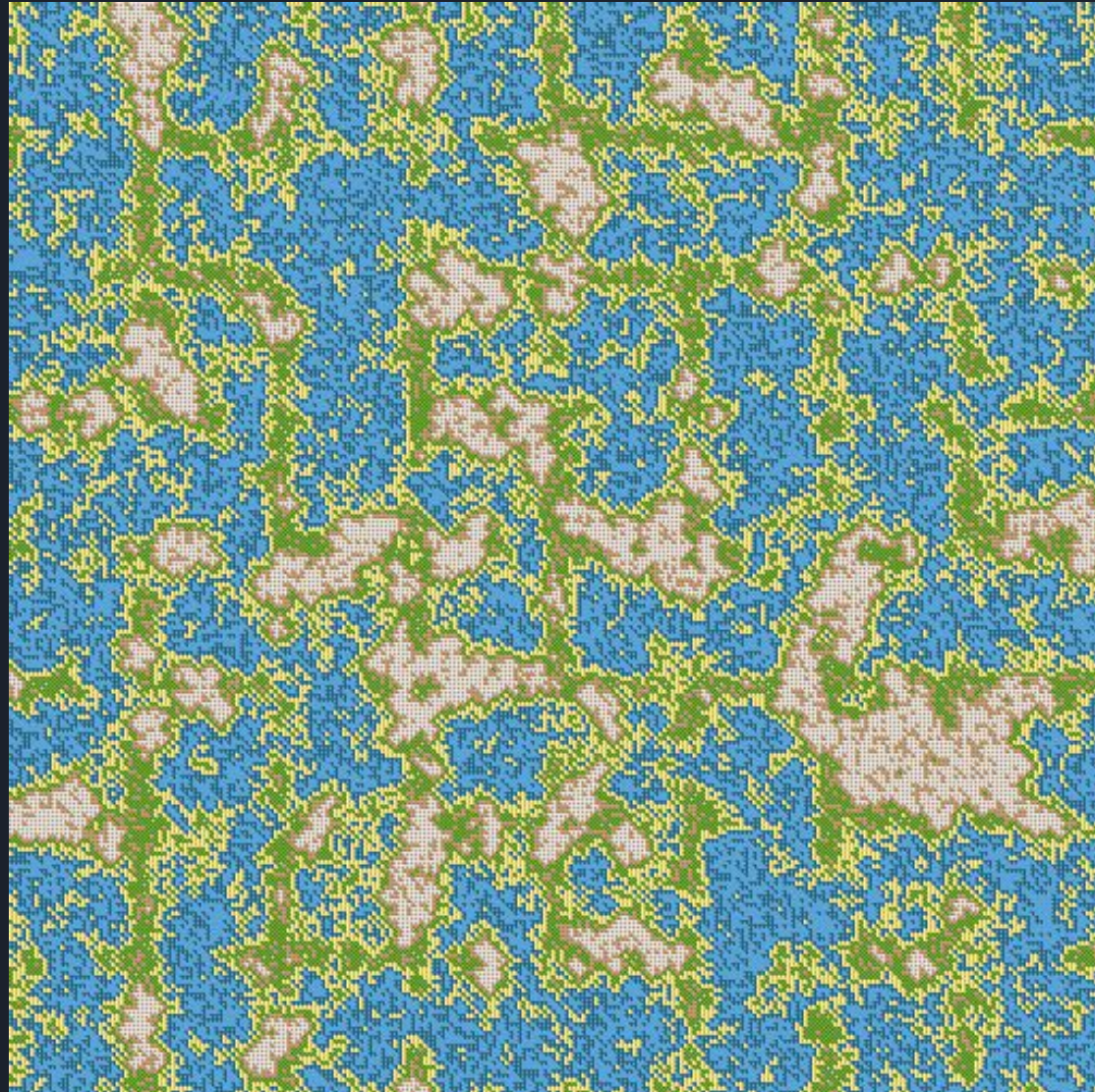
Herausforderungen und Limitationen

- Distributor
- Quadratische Maps
- Parallelisierung / Entropy Tolerance

Herausforderungen und Limitationen



Herausforderungen und Limitationen





Fazit

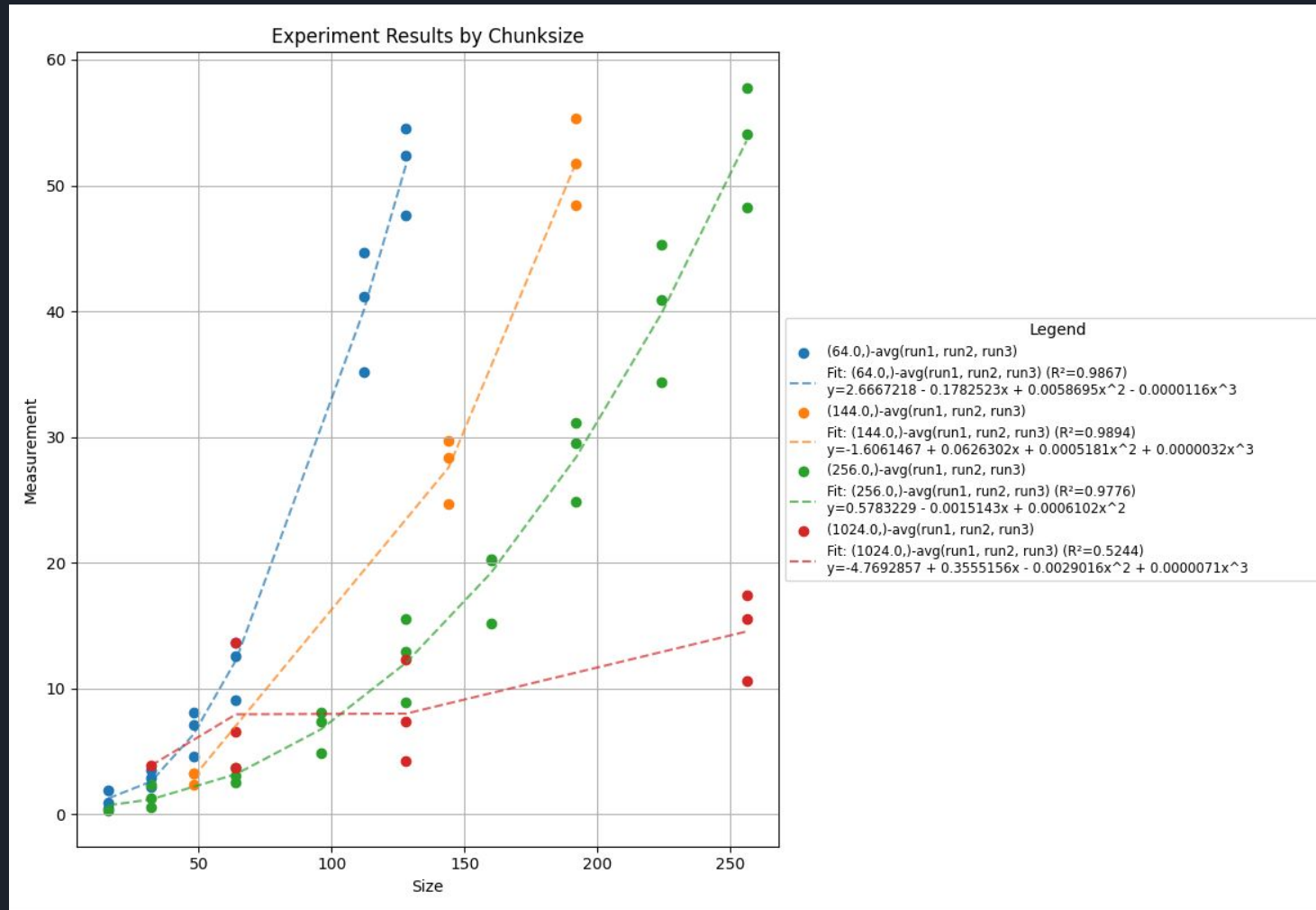
- WFC erfolgreich parallelisiert
- Kubernetes als geeignete Infrastruktur
- Hohe Skalierbarkeit & Effizienz
- Gute Basis für zukünftige Forschung



Ausblick

- Nicht-quadratische Maps / Adaptive Chunkgröße
- Erweiterung des Tilesets
- Verschachtelung des Distributors
- GPU-basierte Beschleunigung
- Einsatz in Echtzeitumgebungen (Games, Simulationen)
- 3D-Map-Generierung mit WFC
- Erweiterung der Evaluation

Ausblick





Vielen Dank

es folgt nun eine Live-Demonstration

