

JUNE 03, 2025 | AI ENGINEER WORLD'S FAIR

# Introduction to LLM serving with SGLang

YINENG ZHANG & PHILIP KIELY

# Welcome to the workshop!

Thank you for choosing to spend the morning with us working on inference optimization.

Let's have a great time and make some fast models.



**Yineng Zhang**

Core maintainer of SGLang, member LMSYS Org, model performance engineer at Baseten. Contributor to 3 papers including FlashInfer. Previously held engineering roles at Baidu and Meituan.



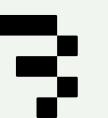
**Philip Kiely**

Head of developer relations at Baseten. B+ in linear algebra.



# Agenda

1. Introductions and setup
2. History of SGLang
3. Deploying your first model with SGLang
4. SGLang performance optimizations
5. SGLang community and codebase



# Introduction

A 10x10 grid of letters (N, B, S, T) arranged in a repeating pattern. The letters are organized into four groups: N (top-left), B (top-right), S (bottom-left), and T (bottom-right). A thick green diagonal band runs from the top-left corner (N) towards the bottom-right corner (T). The letters are arranged as follows:

	N	B	S	T	N	B	S	T	N
N	N	B	S	T	N	B	S	T	N
B	N	B	S	T	N	B	S	T	N
S	N	B	S	T	N	B	S	T	N
T	N	B	S	T	N	B	S	T	N
N	N	B	S	T	N	B	S	T	N
B	N	B	S	T	N	B	S	T	N
S	N	B	S	T	N	B	S	T	N
T	N	B	S	T	N	B	S	T	N
N	N	B	S	T	N	B	S	T	N

B S N B N B  
B S T B S B S  
S T S T S T  
T N T N S T N  
T N T N T N  
N B N B N B  
N B N B N B  
B N B N B N B  
B S B S B S  
B S B S B S  
B S B S B N B  
B S N B N B N B  
N B N B N B N B  
N B N B T N  
T N T N T N T N  
T N S T N S T  
S T B S T B S T  
B S N B S N B S  
B T N B T N B  
B S T N B S T N  
T N B S T N B S  
B S T N B S T N B S  
B S T N B S T N B S  
R S T N R S T

# What is SGLang?

SGLang is an open-source fast serving framework for large language models and vision language models

The screenshot shows the GitHub repository page for 'sgl-project/sglang'. The repository name is 'sglang' and it is described as 'Public'. Key statistics shown are 105 Watchers, 1.9k Forks, and 14.8k Stars. The repository has 29 Branches and 89 Tags. The 'Code' tab is selected, showing a list of recent commits from the 'main' branch. The commits are as follows:

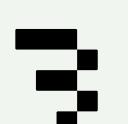
- fzyzcyj Speed up rebalancing when using non-static dispatch algorithms (#6812) · df7f61e · 1 hour ago · 3,525 Commits
- .devcontainer update toc for doc and dockerfile code style format (#64... · last week
- .github [CI] update verlengine ci to 4-gpu test (#6007) · last week
- 3rdparty/amd Revert "fix some typos" (#6244) · 3 weeks ago
- assets Add OpenAI backend to the CI test (#689) · 11 months ago
- benchmark [test] add ut and bm for get\_last\_loc (#6746) · 4 days ago
- docker chore: update blackwell docker (#6800) · yesterday
- docs Improve profiler and integrate profiler in bench\_one\_batc... · 2 days ago
- examples update llama4 chat template and pythonic parser (#6679) · 3 days ago
- python Speed up rebalancing when using non-static dispatch alg... · 1 hour ago
- scripts Correctly abort the failed grammar requests & improve th... · 17 hours ago
- sgl-kernel [EP] Add cuda kernel for moe\_ep\_pre\_reorder (#6699) · 16 hours ago
- sgl-pdlb PD Rust LB (PO2) (#6437) · 4 days ago
- sgl-router Sgl-router Prometheus metrics endpoint and usage track ... · last week
- test Add draft extend CUDA graph for flashinference backend (#68... · 10 hours ago
- .clang-format-ignore add tensorrt\_llm common and cutlass\_extensions as 3rd... · 5 months ago
- .editorconfig minor: Add basic editorconfig and pre-commit hooks to e... · 7 months ago
- .gitignore Support Phi-4 Multi-Modal (text + vision only) (#6494) · last week

On the right side, there is an 'About' section which states: 'SGLang is a fast serving framework for large language models and vision language models.' It also includes a link to 'docs.sglang.ai/' and a list of supported models: cuda, inference, pytorch, transformer, moe, llama, vlm, llm, llm-serving, llava, deepseek-llm, deepseek, llama3, llama3-1, deepseek-v3, deepseek-r1, deepseek-r1-zero, qwen3, llama4. Below the 'About' section are links to 'Readme', 'Apache-2.0 license', 'Activity', 'Custom properties', '14.8k stars', '105 watching', '1.9k forks', and 'Report repository'. The 'Releases' section shows 'Release v0.4.6 (Latest)' from April 27, and '+ 22 releases'. The 'Packages' section lists various packages: ST, BST, B S T, B S, N B S, N B S, B, T N B, T N B, B S T N, B S T N, T N B S, T N B S, B S T N, N B S T N, B S T N B S T N B S, B S T N B S T.



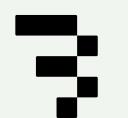
# Why SGLang

- Highly performant inference runtime
- Day zero support for releases from Qwen, DeepSeek, and more
- Welcoming community and strong open-source ethos + enterprise adoption



# Who uses SGLang?

Inference providers, foundation model labs, research institutions, and AI product companies all use SGLang to power production workloads



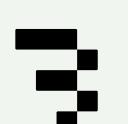
# History

The figure shows a 10x10 grid of letters. The letters are arranged in a repeating pattern of B, S, T, and N. The pattern forms a large diagonal band from the top-left to the bottom-right. The grid is partially shaded in light green.

	B	S	T	N	B	S	T	N	B
	S	T	N	B	S	T	N	B	S
	T	N	B	S	T	N	B	S	T
	N	B	S	T	N	B	S	T	N
	B	S	T	N	B	S	T	N	B
	S	T	N	B	S	T	N	B	S
	T	N	B	S	T	N	B	S	T
	N	B	S	T	N	B	S	T	N
	B	S	T	N	B	S	T	N	B
	S	T	N	B	S	T	N	B	S

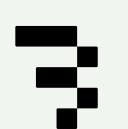
# History of SGLang

- Dec 2023: Arxiv paper published
- Sep 2024: DeepSeek MLA release
- Oct 2024: First official meetup
- Dec 2024: DeepSeek-V3 support
- Jan 2025: Day-one DeepSeek-R1 support
- Mar 2025: Join PyTorch ecosystem
- June 2025: 15K GitHub stars (soon!)



# How I got involved with SGLang

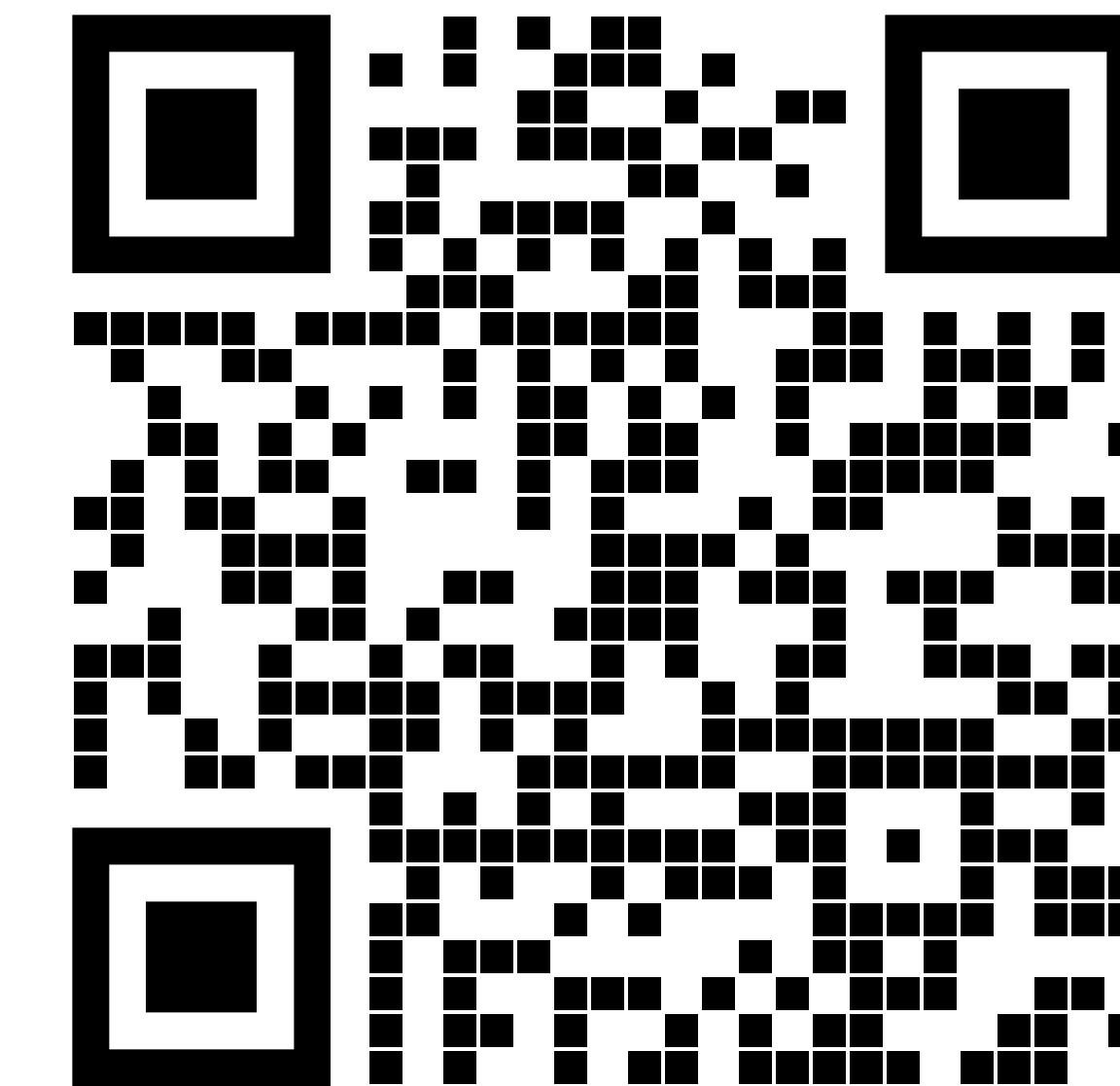
- Work with Lianmin Zheng and Ying Sheng on SGLang
- Work with Zihao Ye on the FlashInfer project
- Core maintainer of SGLang with LMSYS Org



# Setup

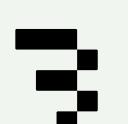
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
T N S T N S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
S T N T N B  
S T N T N B  
S T N T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
B S T T N B  
N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
T N B T N B  
T N B T N B  
T N B T N B  
B S T T N B  
B S T T N B  
T N B T N B  
T N B T N B  
T N B S

github.com/basetenlabs/  
SGLang-Workshop



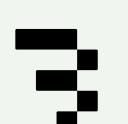
# Deploy your first model

- SGLang launch\_server
- Bundle model weights in image
- Everything is a flag
- Our example: Llama 3.1 8B on L4



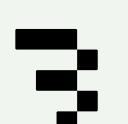
# Quantization with FP8

- Use pre-built model weights
- Switch to FlashAttention backend for L4 compatibility
- Lovelace and Hopper FP8 support



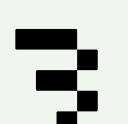
# Eagle 3 speculation

- SGLang supports many speculators
- EAGLE 3 creates draft model from target model layers
- Multi-layer fusion massively increases draft token acceptance rate



# CUDA graph max batch

- By default, cuda-graph-max-bs is 8 on L4
- Updating to 32 improves perf massively
- We want CUDA graph to be True in decode



# Community

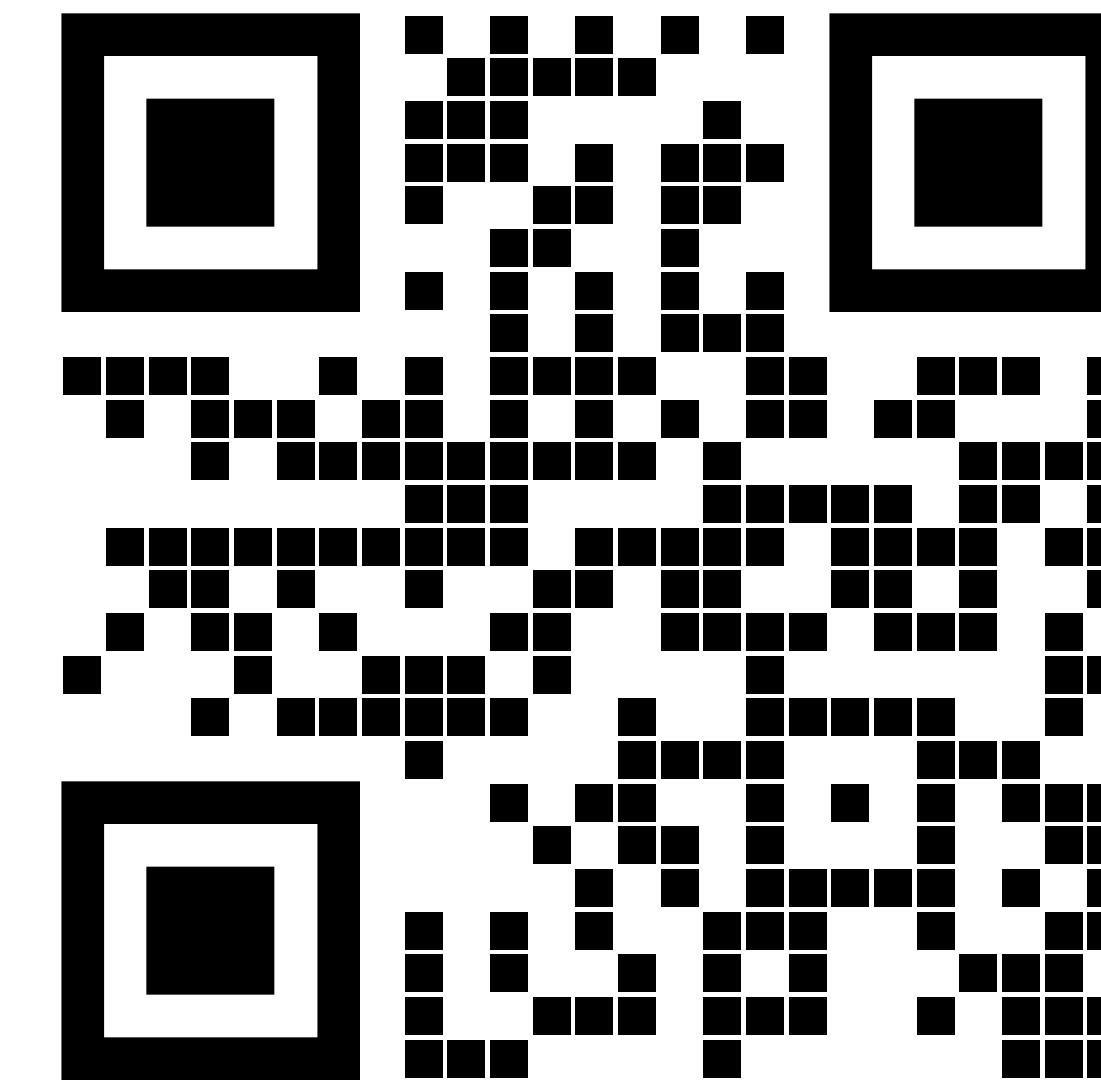
# Join the community

- Star SGLang on GitHub
- File issues and bug reports as you build
- Open your first PR! Good first issues are tagged.
- Join the Slack, follow on Twitter, and keep an eye out online + in-person meetups

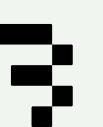


B S N B N B  
B S T B S B S  
S T S T S T  
T N T N S T N  
T N T N T N  
N B N B N  
N B N B N B  
B N B N B N B  
B S B S B S  
B S S B S B  
B S S B N B  
B S S N B N B  
N B N B N B  
N B N B T N  
T N T N T N  
T N S T N S T  
S T B S T B S T  
B S N B S N B S  
B T N B T N B  
B S T N B S T N  
T N B S T N B S  
B S T N B S T N B S  
B S T N B S T N B S  
R S T N R S T

# SGLang community Slack



S T B S T B S T  
B S N B S N B S  
B T N B T N B  
B S T N B S T N  
T N B S T N B S  
B S T N B S T N B S  
B S T N B S T N B S  
B S T N B S T



# Codebase

N B S T N B S T N  
N B S T N B S T N  
N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
T N S T N S T N  
S T N S T N S T N  
N B N B N B  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S T S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
B S T N B S T N B S T N  
S T N T N B  
T N B T N B  
T N B T N B  
T N B T N B  
T N B T N B  
T N B S T N B S T N  
T N B S T N B S T N

B S N B N B  
B S T B S B S  
S T S T S T  
T N T N S T N  
T N T N T N  
N B N B N B  
N B N B N B  
B N B N B N B  
B S B S B S  
B S B S B S  
B S B S B N B  
B S N B N B  
N B N B N B  
N B N B T N  
T N T N T N  
T N S T N S T  
S T B S T B S T  
B S N B S N B S  
B T N B T N B  
B S T N B S T N  
T N B S T N B S  
B S T N B S T N B S  
B S T N B S T N B S  
R S T N R S T

# Tour of the codebase

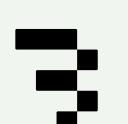
SGLang has a clear architecture and is open to contributions.

The screenshot shows the GitHub repository page for `sgl-project/sglang`. The repository is public, has 105 watchers, 1.9k forks, and 14.8k stars. It features 29 branches and 89 tags. The main branch is active. The repository description states: "SGLang is a fast serving framework for large language models and vision language models." It includes links to `docs.sglang.ai/` and a list of popular models: cuda, inference, pytorch, transformer, moe, llama, vlm, llm, llm-serving, llava, deepseek-llm, deepseek, llama3, llama3-1, deepseek-v3, deepseek-r1, deepseek-r1-zero, qwen3, llama4. The repository has 3,525 commits, with the most recent being a pull request by `fzyzcyj` to speed up rebalancing when using non-static dispatch algorithms (#6812). Other recent commits include updates to .devcontainer, .github, and various benchmarks and Dockerfiles. The releases section shows a latest release at version 0.4.6 from April 27, and there are 22 more releases listed. The packages section is currently empty.



# Tour of the codebase

- 1. SRT (SGLang Runtime):** A high-performance serving system for model inference
- 2. Frontend Language:** A domain-specific language for programming LLM applications
- 3. SGL Kernel:** Optimized CUDA/HIP operations for accelerating model inference



B S N B N B  
B S T B S B S  
S T S T S T  
T N T N S T N  
T N T N T N  
N B N B N B  
N B N B N B  
B N B N B N B  
B S B S B S  
B S B S B S  
B S B S B S  
B S B S B S  
B S N B N B  
N B N B N B  
N B N B N B  
T N T N T N  
T N S T N S T  
S T B S T B S T  
B S N B S N B S  
B T N B T N B  
B S T N B S T N  
T N B S T N B S  
B S T N B S T N B S  
B S T N B S T N B S  
R S T N R S T

# Tour of the codebase

<https://deepwiki.com/sgl-project/sglang>

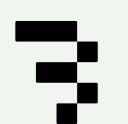
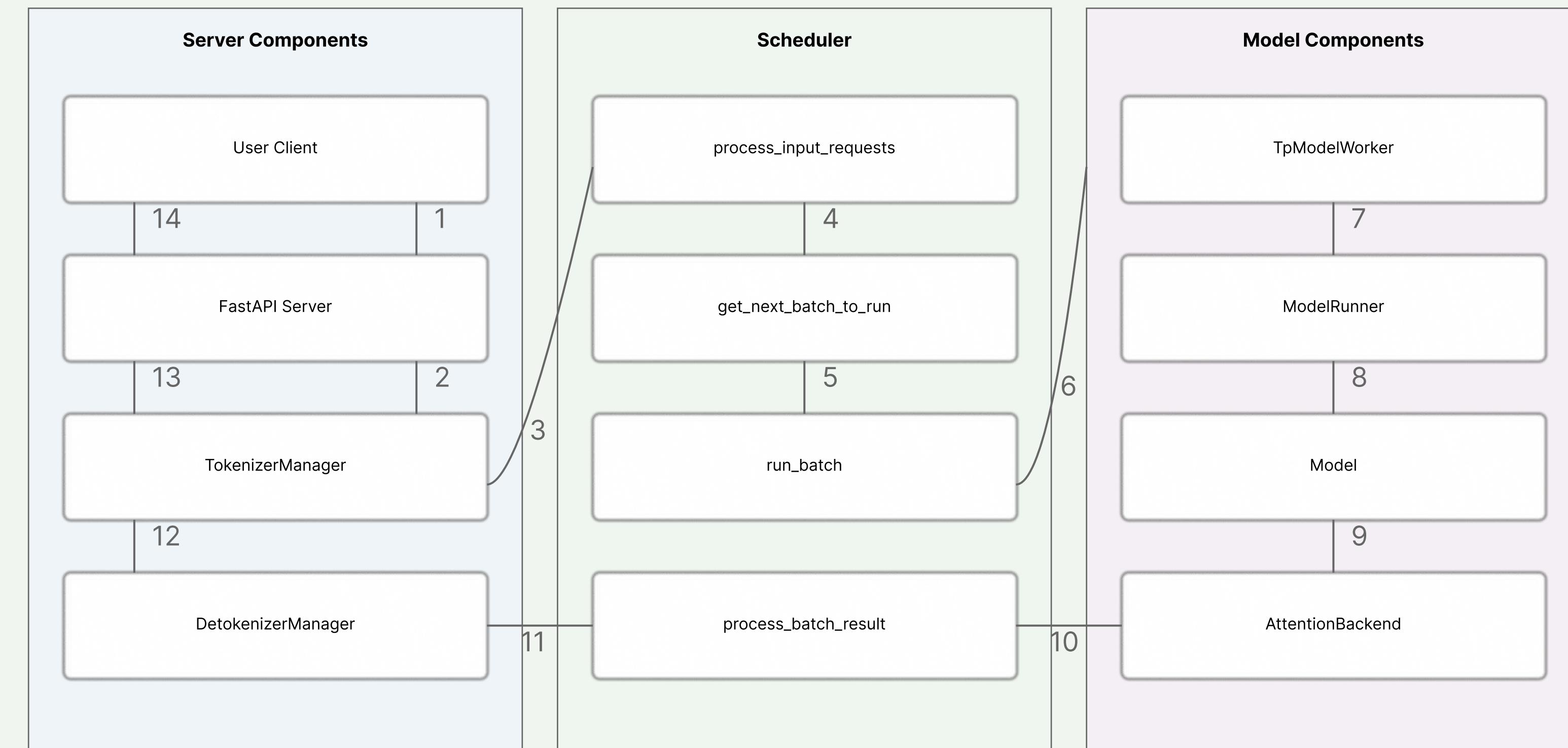
The screenshot shows a DeepWiki page for the repository `sgl-project/sglang`. The page has a dark-themed header with the URL `deepwiki.com/sgl-project/sglang`. The main content area is titled "Overview". It includes a sidebar with a list of topics such as System Architecture, Memory Management, SGLang Runtime (SRT), Server Components, Attention Mechanisms, Mixture of Experts & Quantization, Model Execution Pipeline, SGLang Frontend Language, Interpreter & Program Execution, Supported Models, Language Models, Multimodal Models, MoE Models, Optimization Techniques, SG Kernel, Model-Specific Optimizations, AMD Support, API Interfaces, Native API, OpenAI-compatible API, Development & Testing, and Testing Infrastructure. The main content area also features sections for "Purpose and Scope" and "Core Components", along with a diagram illustrating the system architecture. A sidebar on the right lists "On this page" topics like Overview, Purpose and Scope, Core Components, SRT (SGLang Runtime), Frontend Language, SG Kernel, System Architecture, Hardware Support, Model Support, API Interfaces, Version and Installation, Community and Adoption, Memory Management System, and Inference Flow. At the bottom, there is a "Sources" section with links to `README.md` and `python/pyproject.toml`, and a "Deep Research" toggle.

S T B S T B S T  
B S N B S N B S  
B T N B T N B  
B S T N B S T N  
T N B S T N B S  
B S T N B S T N B S  
B S T N B S T



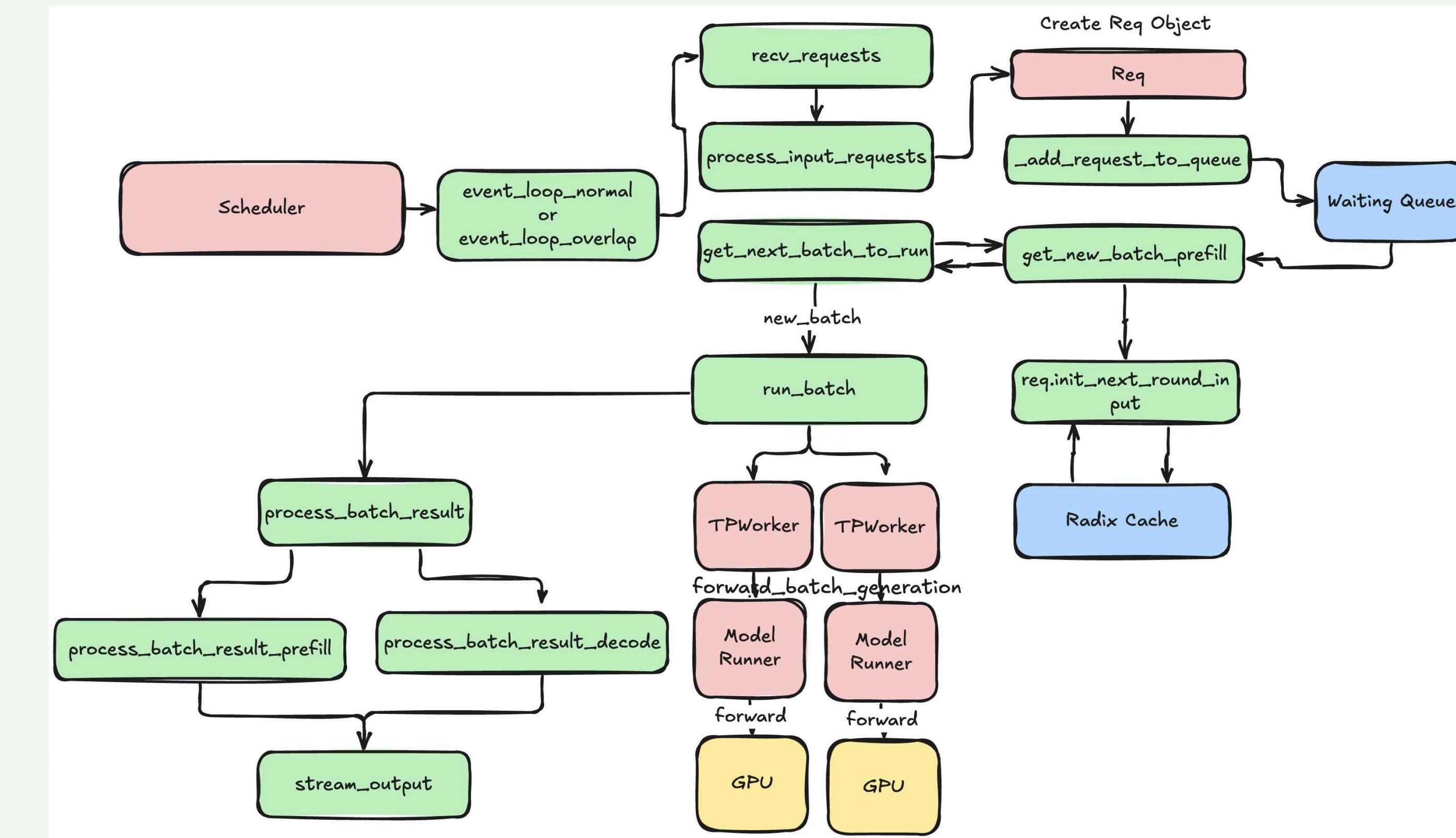
# Request flow process

Image from <https://github.com/zhaochenyang20/Awesome-ML-SYS-Tutorial/>



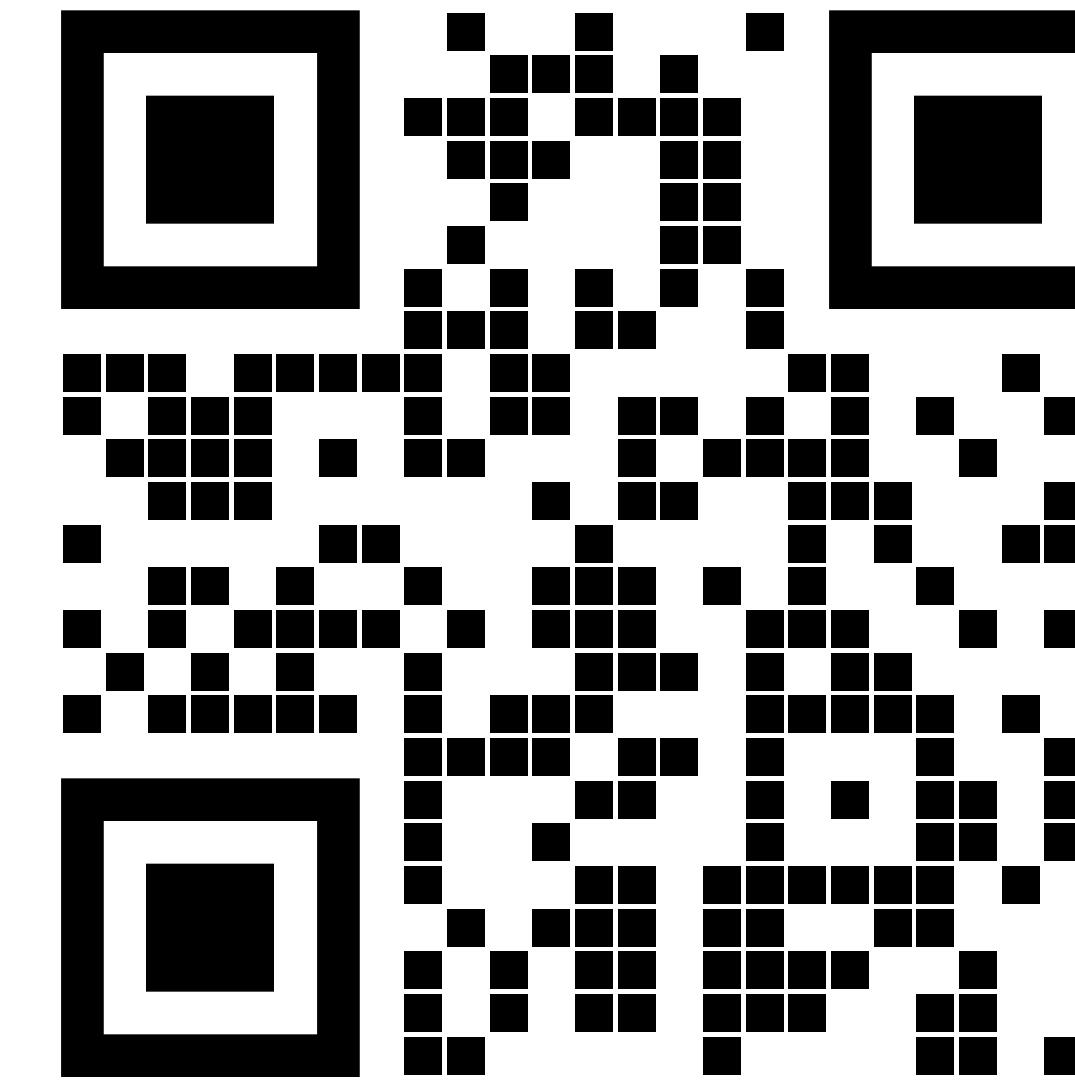
# Scheduler

Image from <https://github.com/zhaochenyang20/Awesome-ML-SYS-Tutorial/>



# Wrapping up

# Join us Wednesday evening



S T	B S T	B S T
S	N B S	N B S
	T N B	T N B
B S T N	B S T N	
T N B S	T N B S	
S T N	N B S T N	
S T N B S T N B S		
S T N B S T		

# Thank you



- [x.com/basetenco](https://x.com/basetenco)
- [linkedin.com/company/baseten](https://linkedin.com/company/baseten)



- [x.com/zhyncs42](https://x.com/zhyncs42)
- [linkedin.com/in/zhyncs](https://linkedin.com/in/zhyncs)



- [x.com/philip\\_kiely](https://x.com/philip_kiely)
- [linkedin.com/in/philipkiely](https://linkedin.com/in/philipkiely)