

```
tree 2.1.3+pkg-3852 from 林博仁 Buo-ren Lin (brlin) installed
cc@test:~/SWGENT-Bench$ tree
```

```
.
├── README.md
├── data
│   ├── cached_repo
│   ├── hooked_issue
│   │   ├── openai-codex-20251013
│   │   │   └── issue.json
│   ├── hooked_repo
│   │   ├── agent_repo.json
│   │   ├── github_archive_repo_2025-10-12-1h.json
│   │   └── github_repo_2025-10-13.json
│   └── ignore
│       ├── command.md
│       ├── github_archive_api.md
│       ├── repo.md
│       ├── setup.md
│       └── requirements.txt
└── src
```

hooked issues

hooked repos

for human to read

unified llm interface

```
├── forge
│   ├── pycache_
│   │   └── api.cpython-38.pyc
│   └── api.py
└── issue-hook
    ├── README.md
    ├── agent_issue.md
    ├── example.py
    └── issue_crawler.py
```

agent issue definition

```
├── repo-hook
│   ├── agent_repo.md
│   └── github
│       ├── README.md
│       ├── __init__.py
│       ├── pycache_
│       │   ├── awesome_search.cpython-38.pyc
│       │   └── repo_extractor.cpython-38.pyc
│       ├── awesome_search.py
│       ├── example.sh
│       ├── main.py
│       ├── repo_extractor.py
│       └── requirements.txt
│   ├── github_archive
│   │   ├── README.md
│   │   ├── __init__.py
│   │   ├── pycache_
│   │   │   ├── agent_filter.cpython-38.pyc
│   │   │   ├── config.cpython-38.pyc
│   │   │   ├── github_api.cpython-38.pyc
│   │   │   ├── github_fetcher.cpython-38.pyc
│   │   │   └── main.cpython-38.pyc
│   │   ├── agent_filter.py
│   │   ├── config.py
│   │   ├── github_api.py
│   │   ├── github_fetcher.py
│   │   ├── main.py
│   │   ├── requirements.txt
│   │   └── test_basic.py
│   └── repo_merge
│       ├── README.md
│       ├── __init__.py
│       ├── example.sh
│       └── main.py
└── 17 directories, 44 files
```

agent repo definition

candidate
repo/issue

repo/issue.md

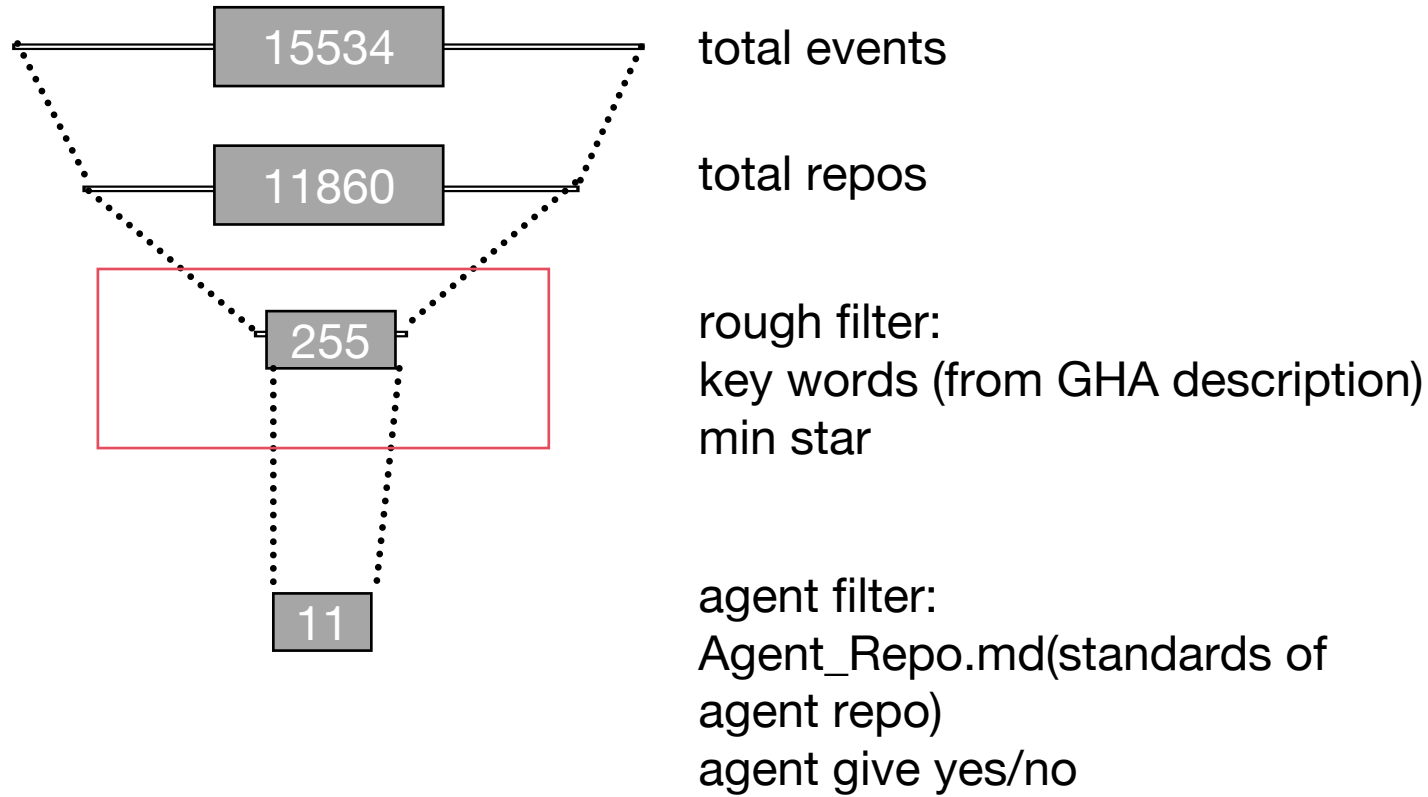
llm:
yes/no

yes

agent
repo/issue

1. Find Agent Repo

Initial Strategy



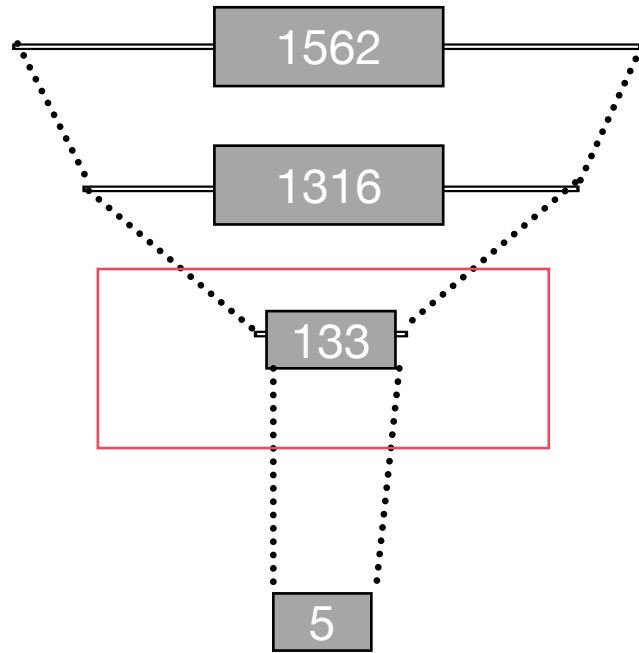
```
{  
  "date": "2025-10-12",  
  "total_events": 15534,  
  "total_repos": 11860,  
  "agent_repos_count": 11,  
  "used_ai_filter": true,  
  "min_stars": 10,  
  "agent_repos": [  
    {  
      "name": "agent-repo",  
      "description": "Agent repository",  
      "stars": 10,  
      "language": "python",  
      "created_at": "2025-10-12",  
      "updated_at": "2025-10-12",  
      "owner": "agent-repo",  
      "private": false,  
      "fork": false,  
      "has_readme": true,  
      "has_github_actions": true,  
      "has_agent_repo_md": true,  
      "agent_give_yes_no": true  
    }  
  ]  
}
```

only yield 11 repo from the whole day -> too strict

most filtering comes from “including key words in repo name” ,because the info from GHA is too little -> take readme into account as well

1. Find Agent Repo

Optimized Strategy



total events

total repos

rough filter:
key words (from GHA description)
min star and repo readme

agent filter:
Agent_Repo.md(standards of
agent repo)
agent give yes/no

yield 5 repo from single hour -> moderate ,but can only update every hour due to api limit

```
=====
Summary:
- Total events: 1562
- Active repositories: 1316
- Agent repositories: 5
=====

Sample of Agent repositories:
- block/goose (20271 stars)
- openai/openai-agents-js (1681 stars)
- awslabs/threat-designer (152 stars)
- TauricResearch/TradingAgents (22714 stars)
- envy-ai/ai_rpg (27 stars)
```

goose

a local, extensible, open source AI agent that automates engineering tasks

License Apache 2.0 Join Us 670 online build passing

OpenAI Agents SDK (JavaScript/TypeScript)

npm package 0.1.9 CI passing

The OpenAI Agents SDK is a lightweight yet powerful framework for building multi-agent workflows in JavaScript/TypeScript. It is provider-agnostic, supporting OpenAI APIs and more.



Threat Designer: AI-powered threat modeling for secure system design

Check the blogpost: Accelerate threat modeling with generative AI for an in-depth overview of the solution.



Join GitHub Community TauricResearch

Deutsch | Español | français | 日本語 | 한국어 | Português | Русский | 中文

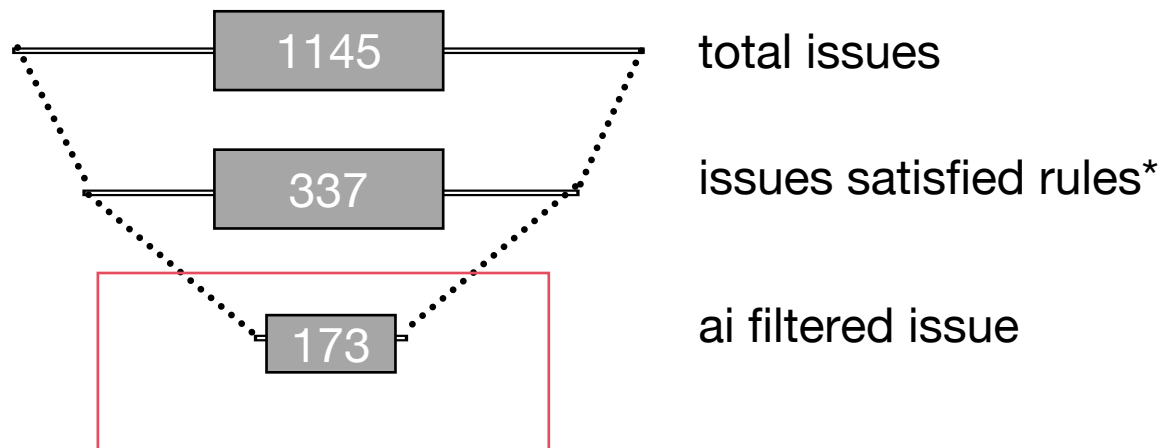
TradingAgents: Multi-Agents LLM Financial Trading Framework

AI RPG Server

AI RPG Server turns an OpenAI-compatible language model into a solo tabletop game master. The Node.js application keeps track of players, locations, regions, and items; renders structured prompts with Nunjucks; and optionally drives a ComfyUI image pipeline so every scene can ship with fresh artwork. A lightweight browser UI and JSON API sit on top, letting you explore the world, tweak settings, and review logs without leaving the app.

②. Find Agent Issue

eg Openai/Codex



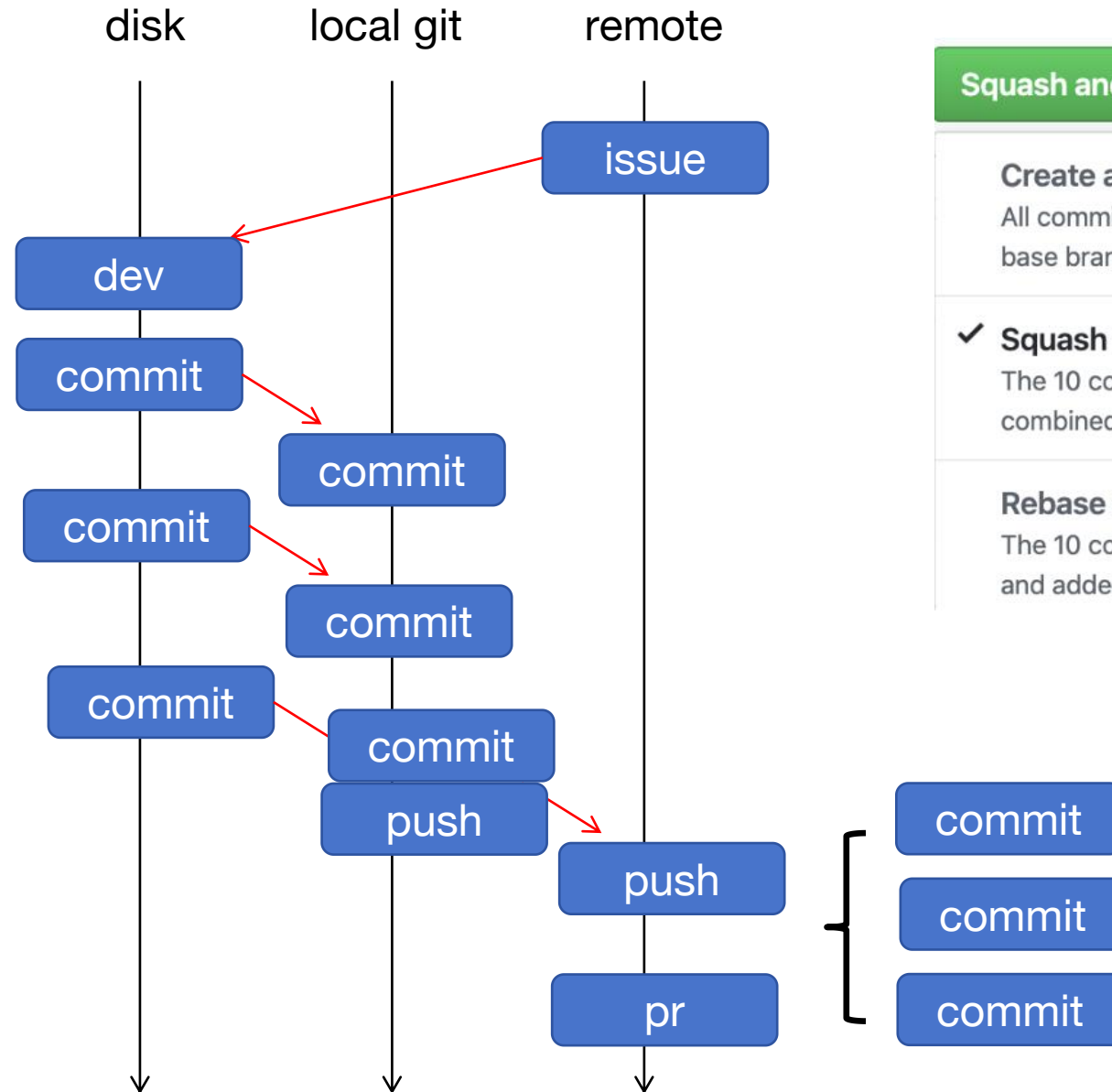
```
=====
AI judgment complete!
Total: 337 issues | Agent issues: 173
=====
```

```
Results saved to: data/hooked_issue/openai-codex-20251013/issue.json
```

rules:

#	Criterion	Check Method	Purpose
1	Issue is closed	<code>state == "closed"</code>	Only resolved issues
2	Has text description	<code>body.strip() != ""</code>	Meaningful content
3	Linked to PR(s)	Multiple detection methods	Associated with code changes
4	PR merged to main	<code>merged == True && base_branch in ['main', 'master']</code>	Changes in production

Issue-Pr Key



Squash and merge

You can also [open this in](#)

Create a merge commit

All commits from this branch will be added to the base branch via a merge commit.

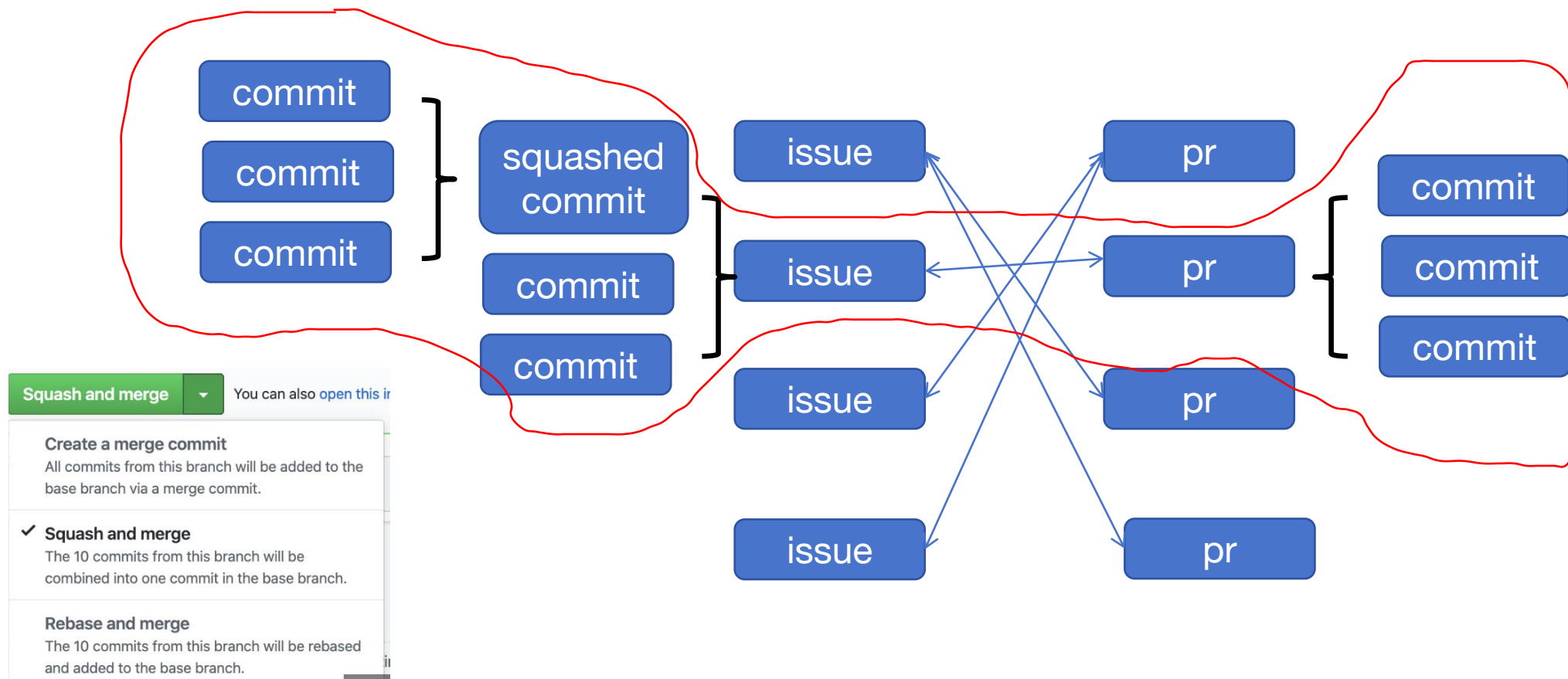
✓ Squash and merge

The 10 commits from this branch will be combined into one commit in the base branch.

Rebase and merge

The 10 commits from this branch will be rebased and added to the base branch.

github



“logically” one issue: n issue and m pr with their commit

to simplify: 1 issue related with 1 pr as a key, all their commits and patch to be value