

# Accelerating Software Development: The LLM (R)evolution

Emery Berger (he/him)

Professor, University of Massachusetts Amherst  
*Amazon Scholar, Amazon Web Services (Annapurna ML)*

*Speaking as a UMass professor, giving my personal views*

joint work with Stephen Freund, Eunice Jun, Kyla Levin, Juan Altmayer Pizzorno, Brooke Simon, Sam Stern, Nicolas van Kempen

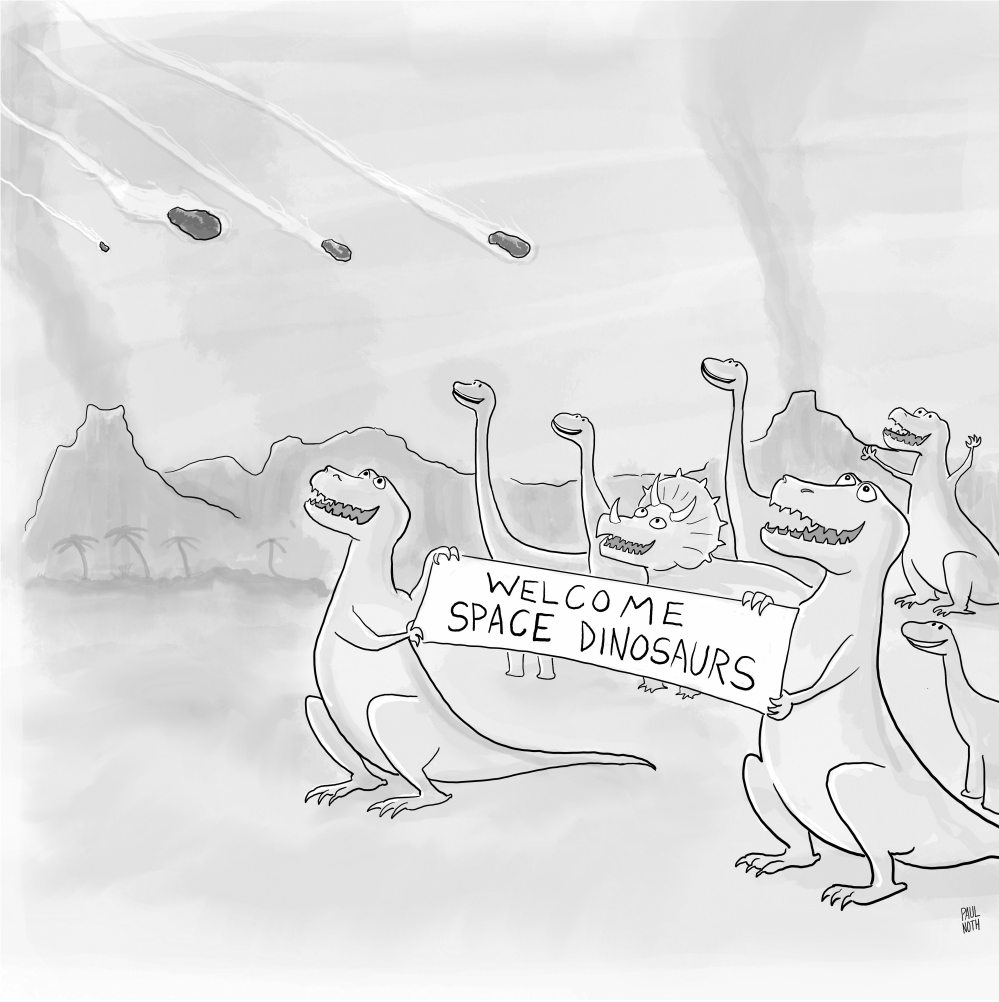


# **Cretaceous–Paleogene (K-Pg) extinction event**

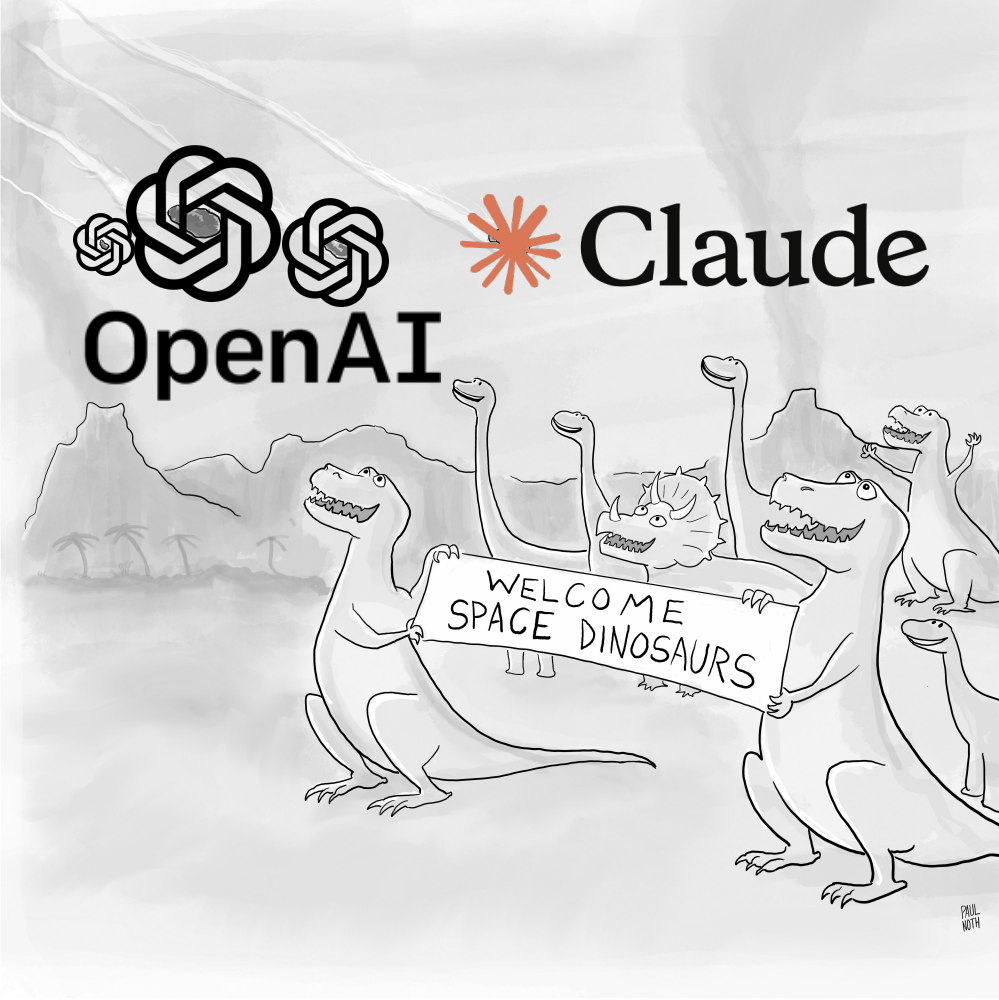
# Cretaceous–Paleogene (K-Pg) extinction event



# Cretaceous–Paleogene (K-Pg) extinction event

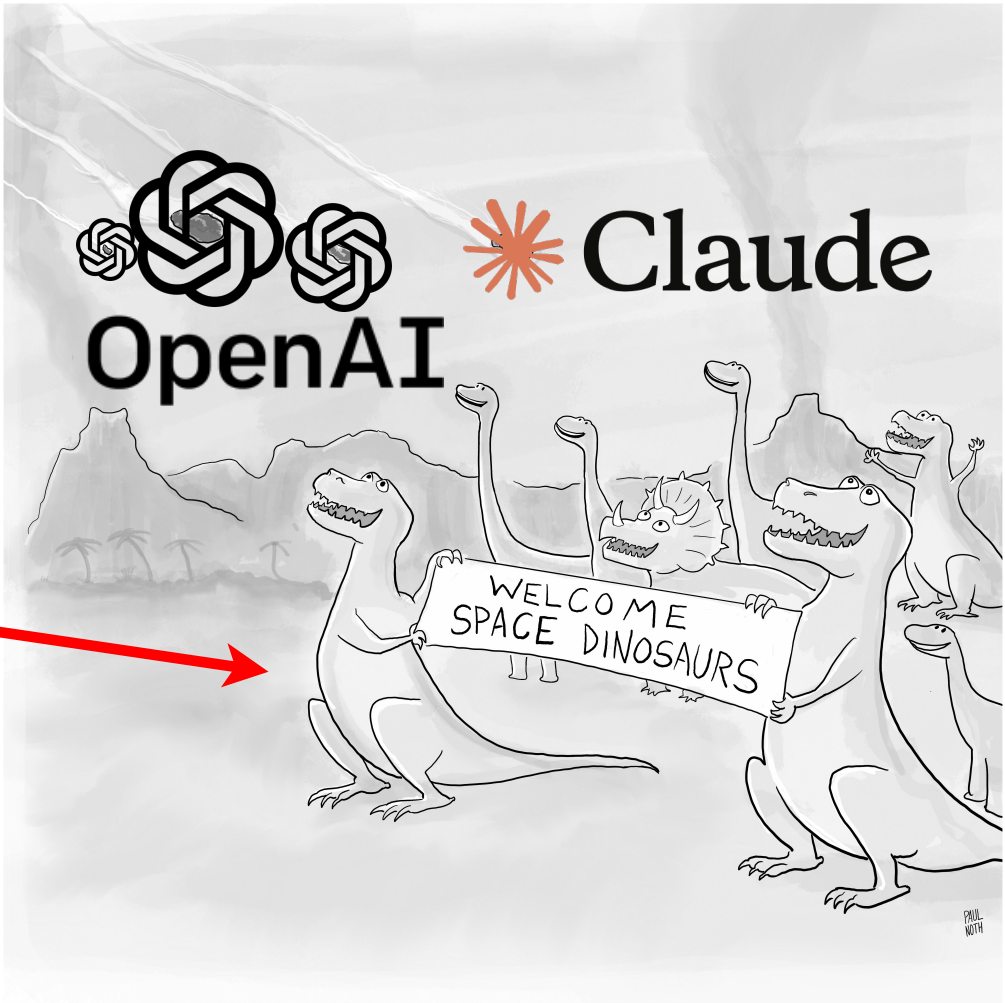


# Cretaceous–Paleogene (K-Pg) extinction event

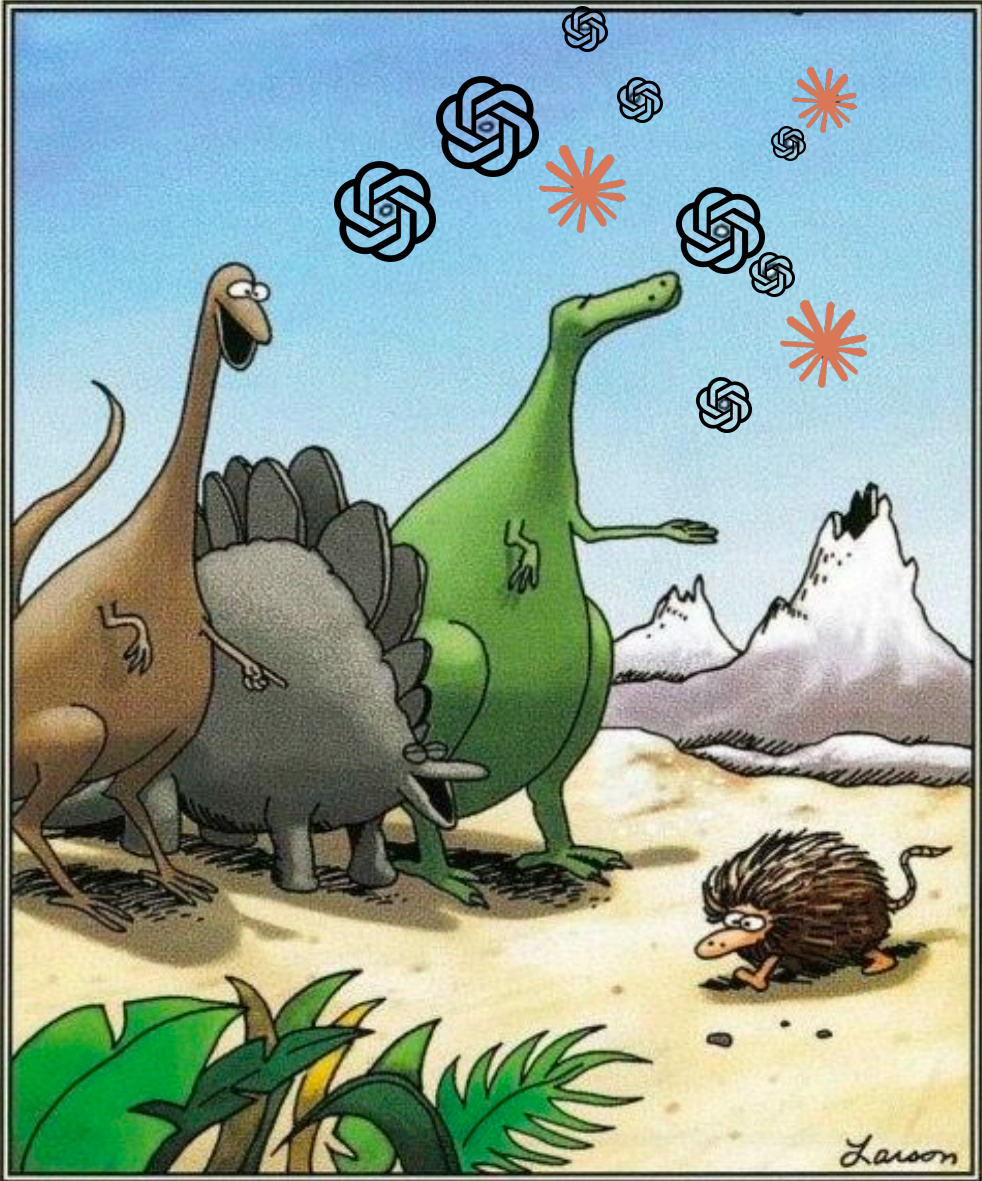


# Cretaceous–Paleogene (K-Pg) extinction event

traditional  
software  
development  
tools



# Survival of the Fittest



# Evolving New Breed of Developer Tools



# Evolving New Breed of Developer Tools



evolve

# Evolving New Breed of Developer Tools



evolve

*Close the loop  
between what the  
tool provides and  
what devs actually  
want*

# Evolving New Breed of Developer Tools



evolve

*Close the loop  
between what the  
tool provides and  
what devs actually  
want*

**"tool++"**

# Evolving New Breed of Developer Tools



evolve



exploit niche

*Close the loop  
between what the  
tool provides and  
what devs actually  
want*

**"tool++"**

# Evolving New Breed of Developer Tools



evolve

*Close the loop  
between what the  
tool provides and  
what devs actually  
want*

**"tool++"**



exploit niche

*Leverage  
information the  
tool can access to  
guide the AI*

# Evolving New Breed of Developer Tools



evolve

*Close the loop  
between what the  
tool provides and  
what devs actually  
want*

***"tool++"***



exploit niche

*Leverage  
information the  
tool can access to  
guide the AI*



ensure fitness

# Evolving New Breed of Developer Tools



evolve

*Close the loop  
between what the  
tool provides and  
what devs actually  
want*

**"tool++"**



exploit niche

*Leverage  
information the  
tool can access to  
guide the AI*



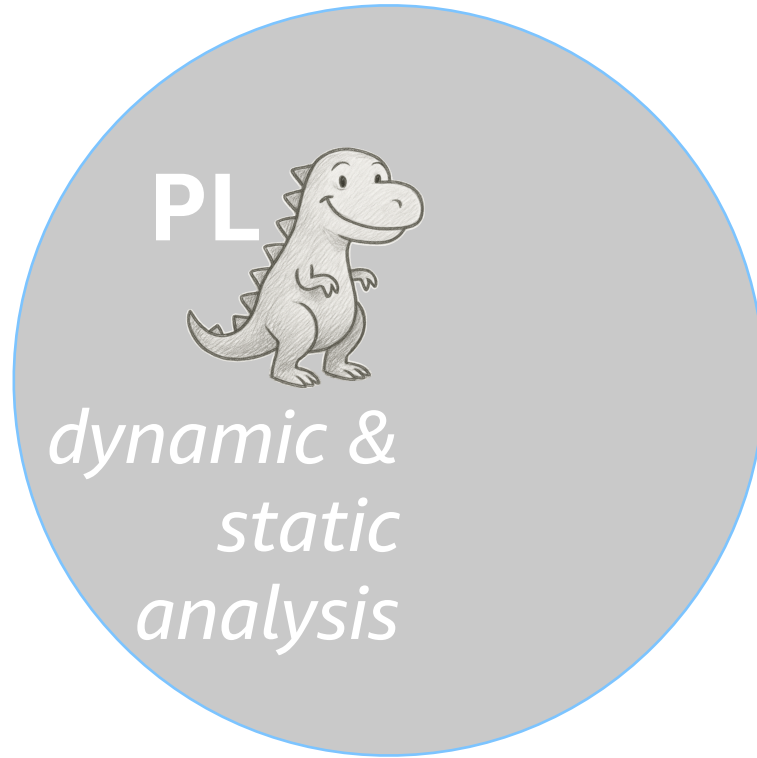
ensure fitness

*Check the AI's  
responses to  
iterate, improve,  
and validate*

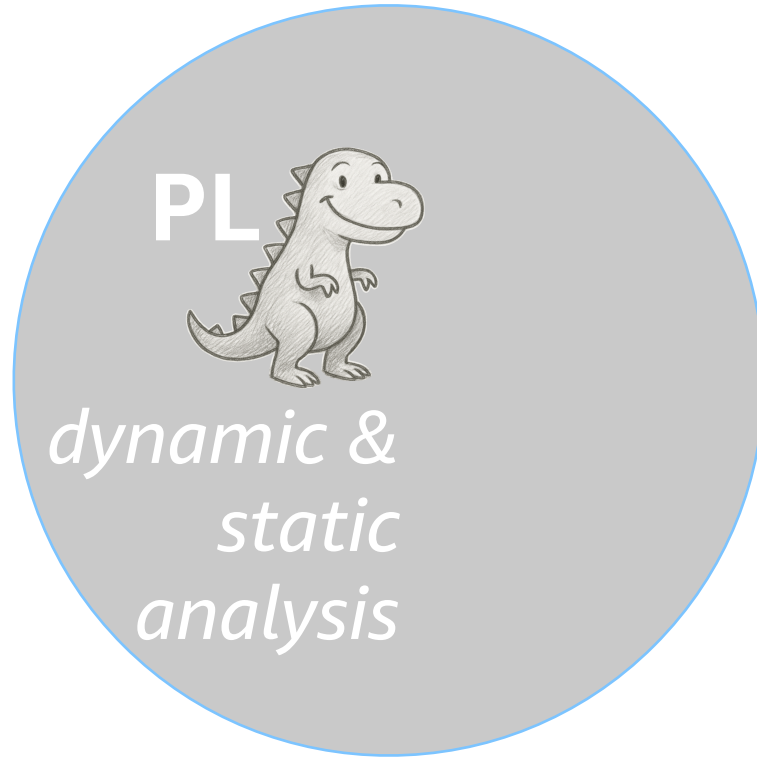
# Adaptation Strategy: Best of Both Worlds



# Adaptation Strategy: Best of Both Worlds

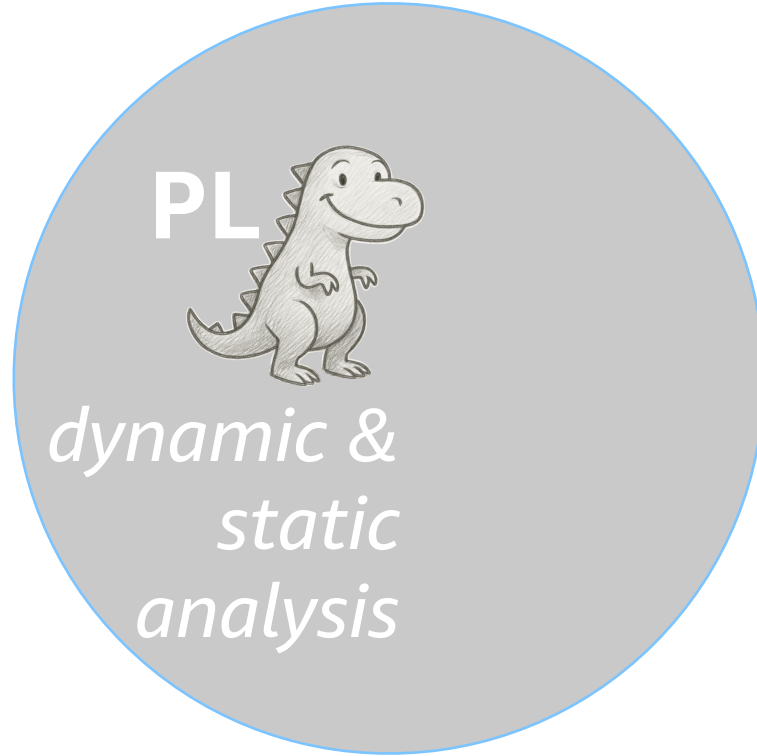


# Adaptation Strategy: Best of Both Worlds



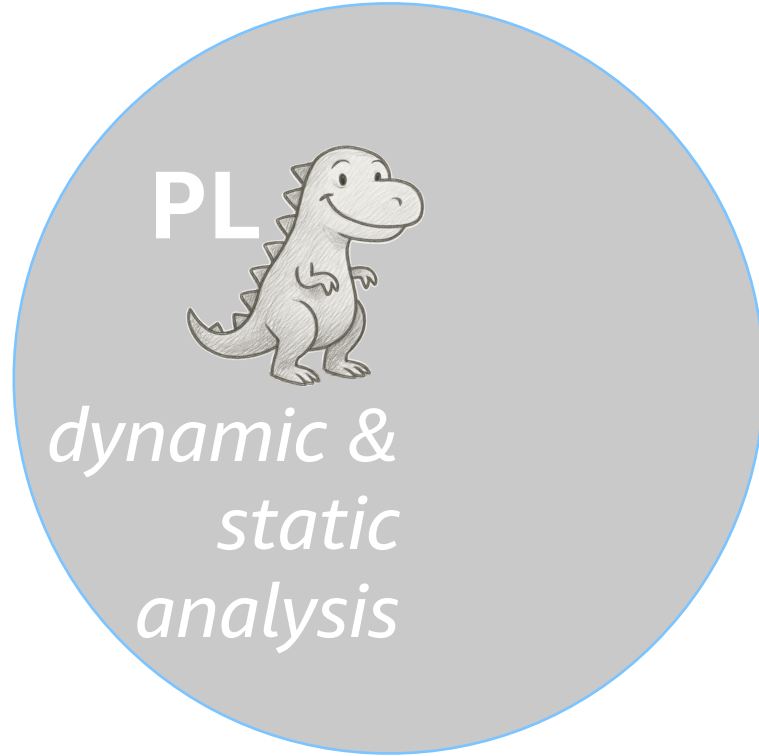
*deterministic*

# Adaptation Strategy: Best of Both Worlds



*deterministic*  
*sound/precise*  
*formal reasoning*

# Adaptation Strategy: Best of Both Worlds



*deterministic*  
*sound/precise*  
*formal reasoning*  
*ground truth*

# Adaptation Strategy: Best of Both Worlds



PL



*dynamic &  
static  
analysis*

*deterministic  
sound/precise  
formal reasoning  
ground truth*


LLMs



# Adaptation Strategy: Best of Both Worlds




PL



*dynamic & static analysis*

*deterministic  
sound/precise  
formal reasoning  
ground truth*

LLMs




*stochastic*

# Adaptation Strategy: Best of Both Worlds



PL



*dynamic & static analysis*

*deterministic  
sound/precise  
formal reasoning  
ground truth*

LLMs




*stochastic  
real-world  
understanding*

# Adaptation Strategy: Best of Both Worlds



PL



*dynamic & static analysis*

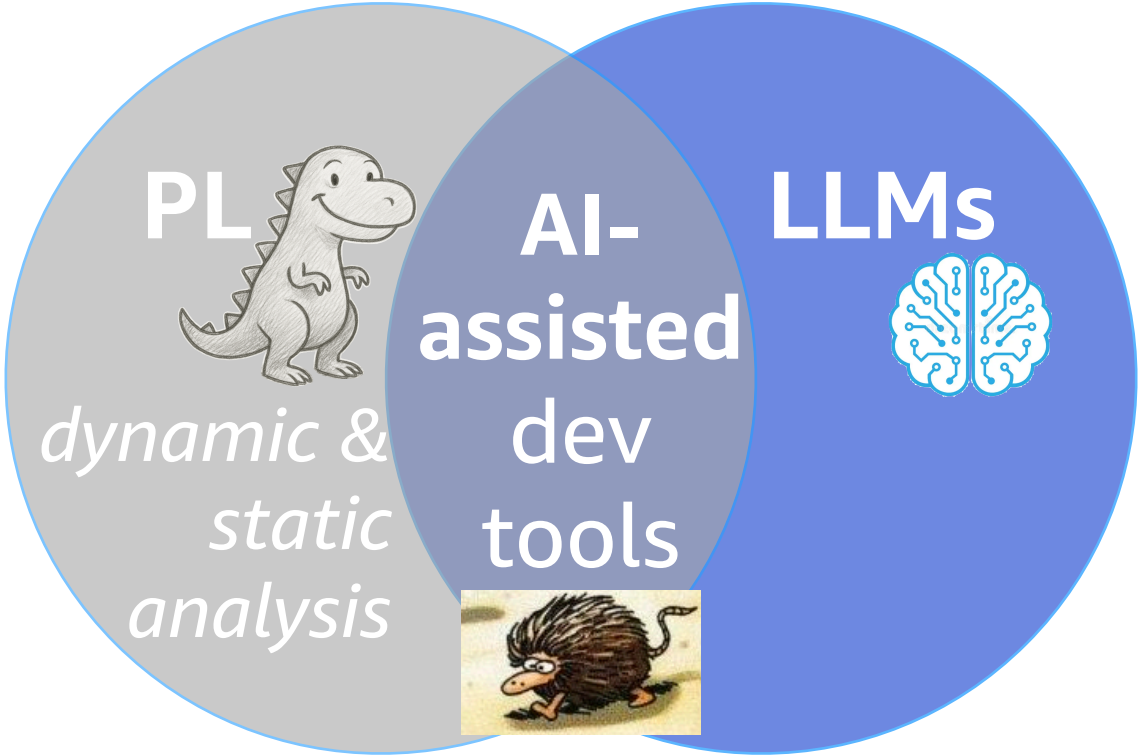
*deterministic  
sound/precise  
formal reasoning  
ground truth*

LLMs



*stochastic  
real-world  
understanding  
"heuristic"*

# Adaptation Strategy: Best of Both Worlds



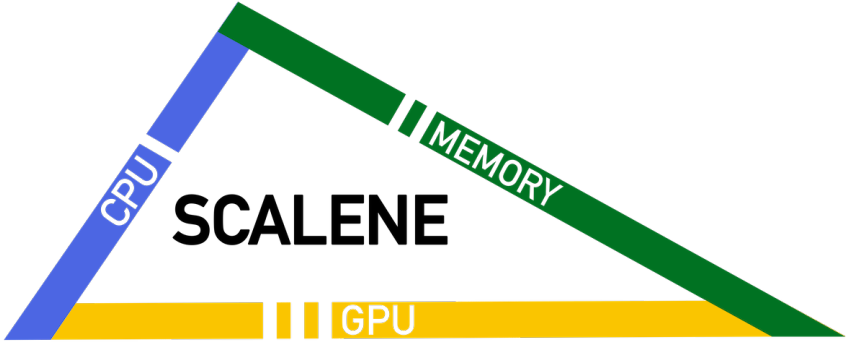
*deterministic  
sound/precise  
formal reasoning  
ground truth*

*stochastic  
real-world  
understanding  
"heuristic"*

# Evolving New Breed of Developer Tools



# Evolving New Breed of Developer Tools



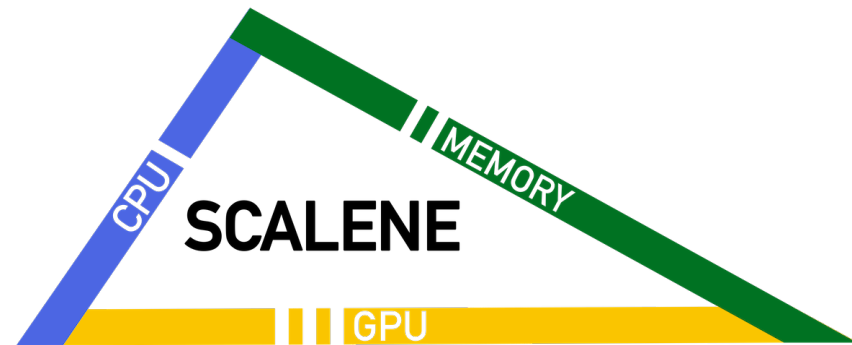
profiler++

# Evolving New Breed of Developer Tools



CoverUp

coverage analyzer++



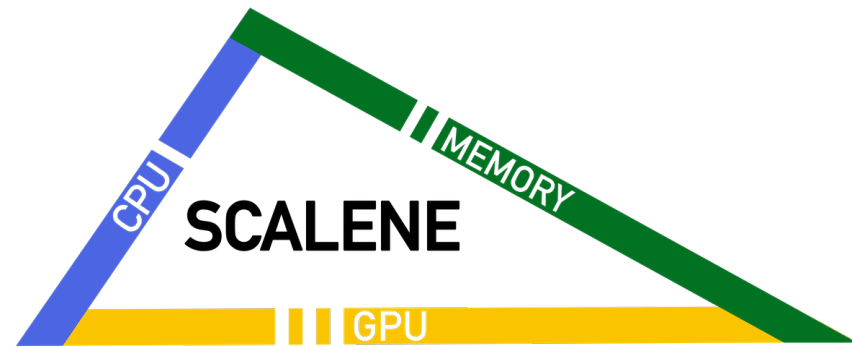
profiler++

# Evolving New Breed of Developer Tools

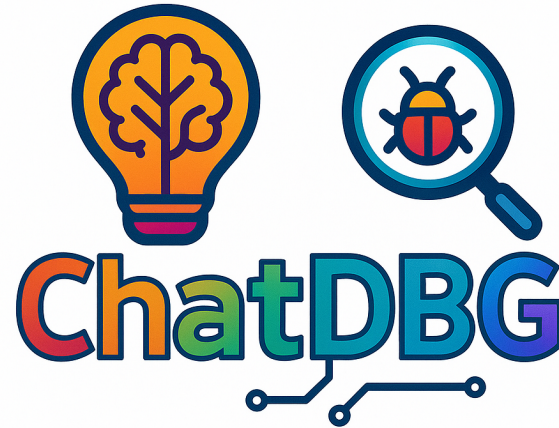


CoverUp

coverage analyzer++



profiler++



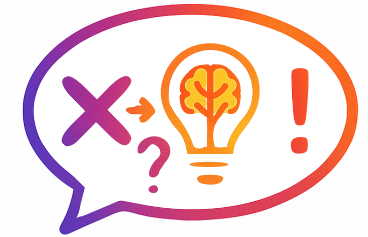
debugger++

# Evolving New Breed of Developer Tools



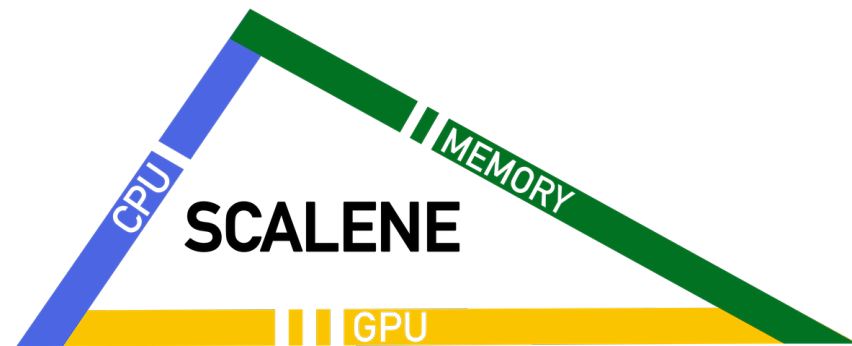
CoverUp

coverage analyzer++

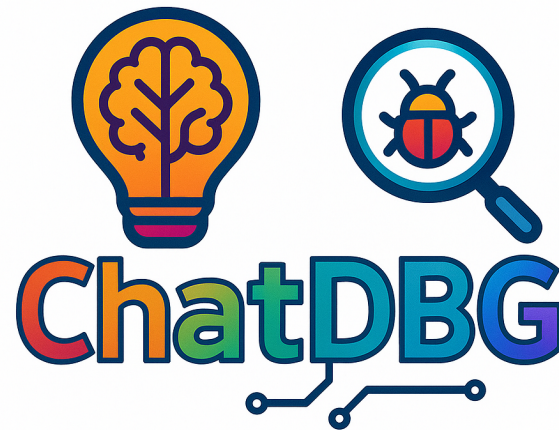


CWWhy

compiler++



profiler++



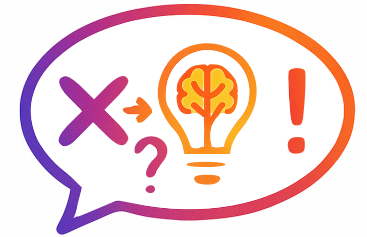
debugger++

# Evolving New Breed of Developer Tools



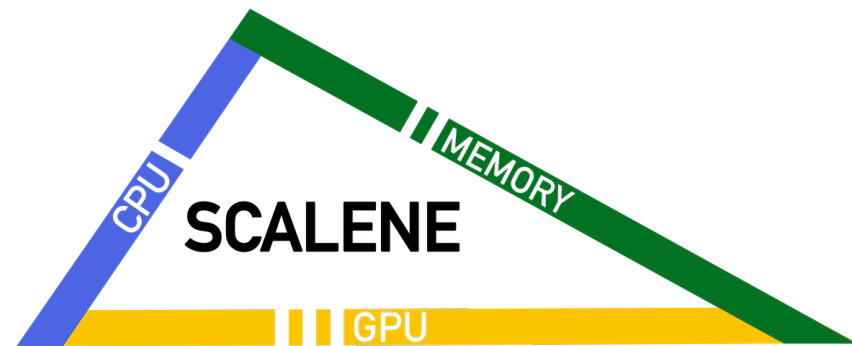
CoverUp

coverage analyzer++

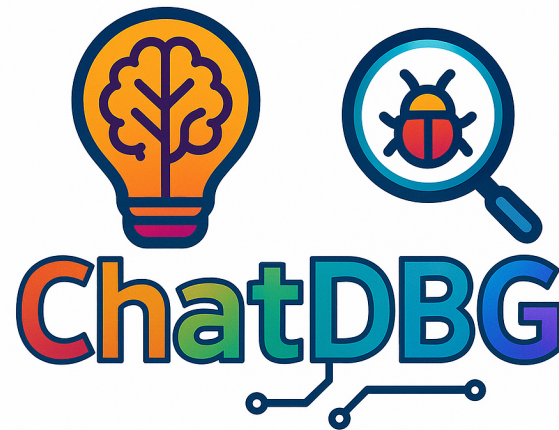


CWWhy

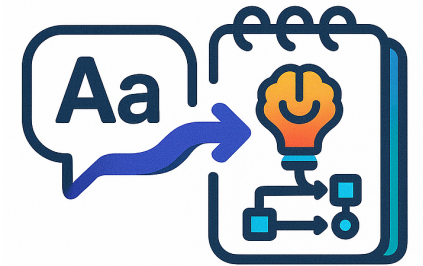
compiler++



profiler++



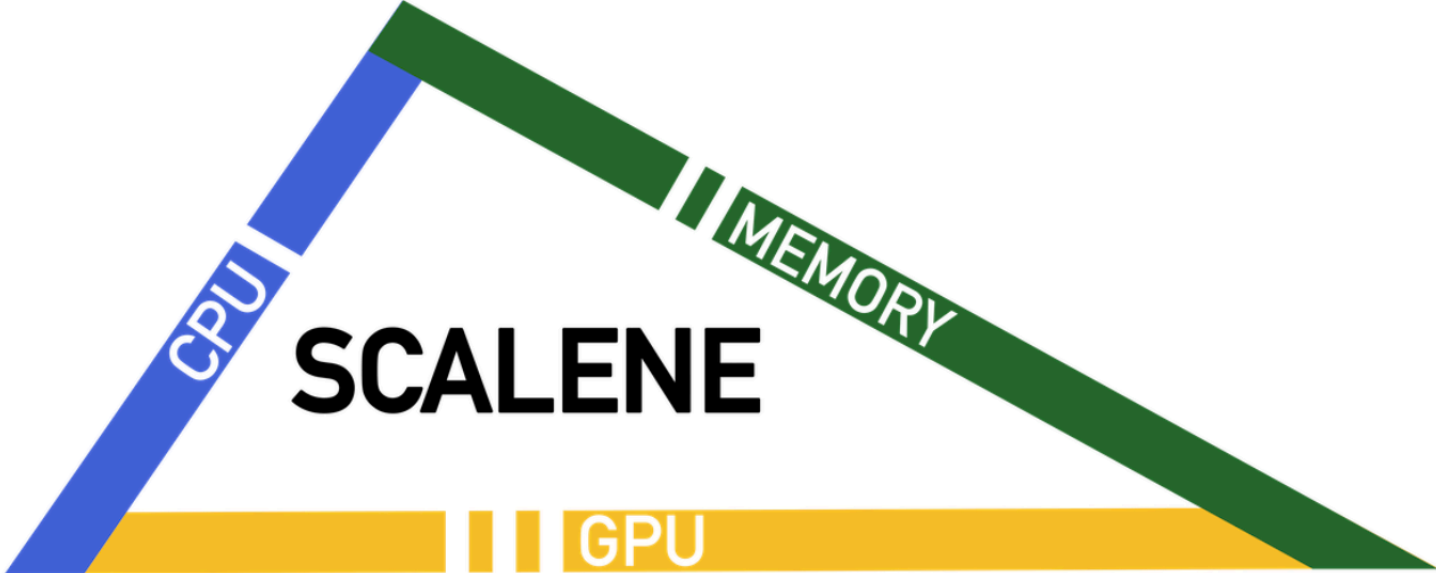
debugger++



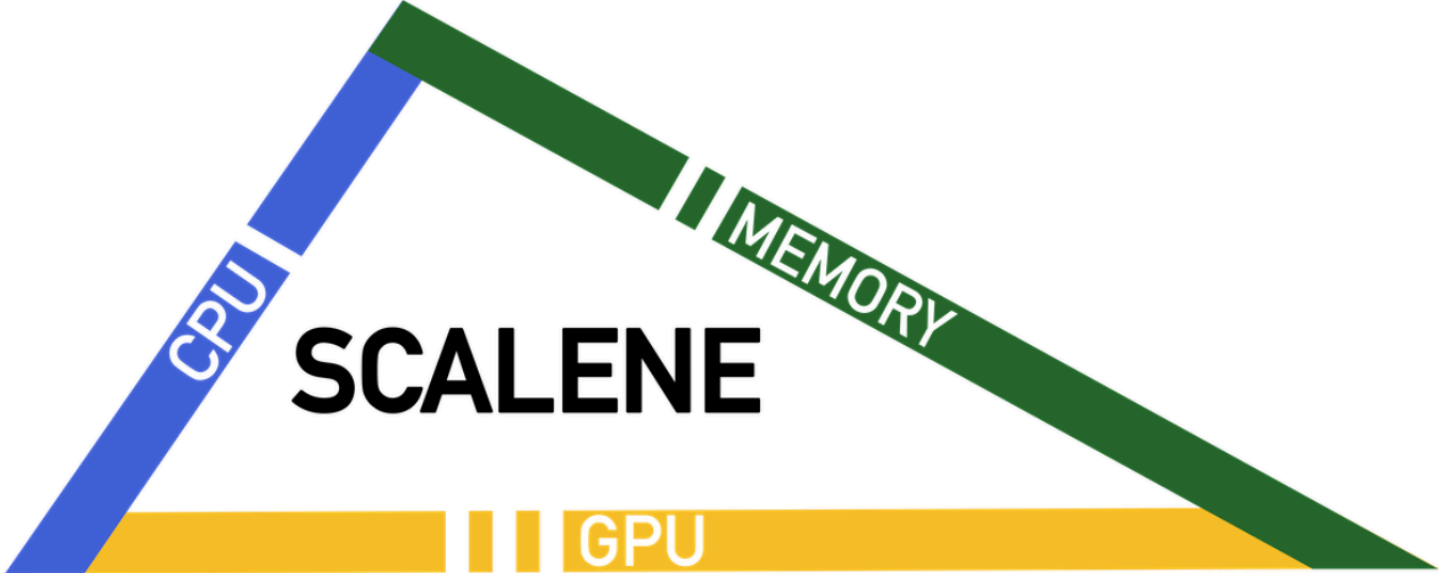
Flowco

notebook++

# Scalene: Evolving a New Profiler



# Scalene: Evolving a New Profiler



pypi v1.5.51 conda | conda-forge v1.5.41 downloads 3M conda|downloads 366k downloads/month 115k

python 3.8 | 3.9 | 3.10 | 3.11 | 3.12 | 3.13 version v0.0.10 license Apache-2.0 Stars 13k

OSDI<sup>23</sup> ATTEND PROGRAM PARTICIPATE SPONSORS ABOUT

## Triangulating Python Performance Issues with SCALENE

Authors:  
Emery D. Berger, Sam Stern, and Juan Altmayer Pizzorno, *University of Massachusetts Amherst*

*Awarded Best Paper!*

*logo by Sophia Berger*

# Scalene: Evolving a New Profiler



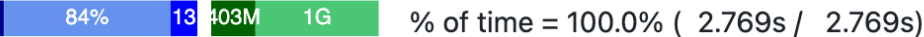
▶ AI optimization options

Time: Python | native | system    Memory: Python | native    Memory timeline: (max: 2G, growth: 0.2%)



hover over bars to see breakdowns; click on COLUMN HEADERS to sort.

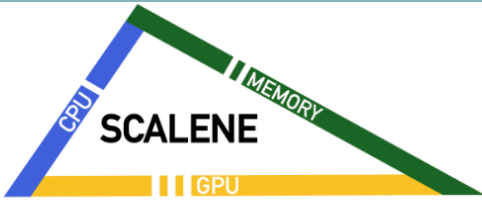
show all | hide all | only display profiled lines



▼ /Users/emerydb/git/scalene/test/issues/test-issue31.py

<u>TIME</u>	<u>MEMORY</u> peak	<u>MEMORY</u> average	<u>MEMORY</u> timeline	<u>MEMORY</u> activity	<u>COPY</u>	<u>LINE PROFILE</u> (click to reset order) /Users/emerydb/git/scalene/test/issues/test-issue31.py
42%					87	1 ⚡ import numpy as np
						3 ⚡ def main1():
21%	403M					5 ⚡ x = np.array(range(10**7))
	1G				297	6 ⚡ y = np.array(np.random.uniform(0, 100, size=10**8))

# Scalene: Evolving a New Profiler



Time: Python | native | system    Memory: Python | native    Memory timeline: (max: 2G, growth: 0.2%)

84%    13    403M    1G

hover over bars to see breakdowns; click on COLUMN HEADERS to sort.

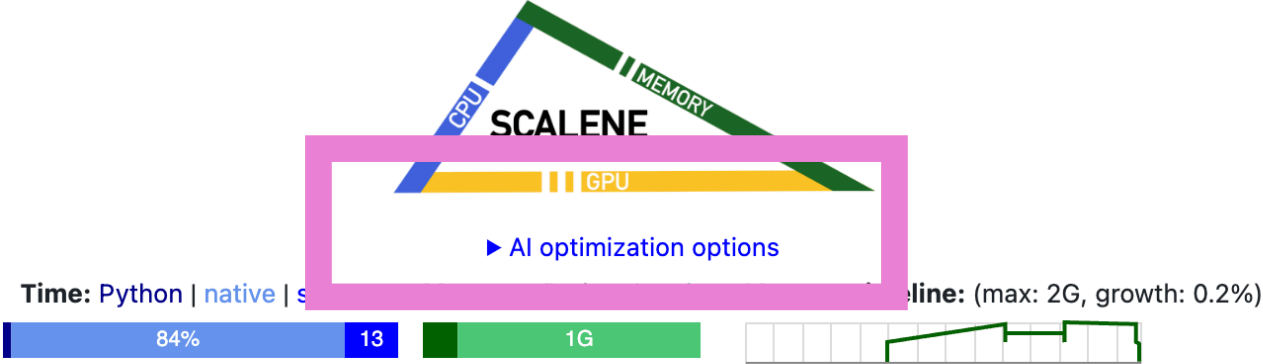
show all | hide all | only display profiled lines

84%    13    403M    1G    % of time = 100.0% ( 2.769s / 2.769s)

▼ /Users/emerydb/git/scalene/test/issues/test-issue31.py

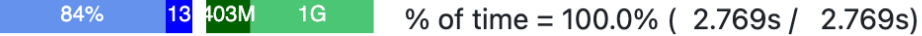
<u>TIME</u>	<u>MEMORY</u> peak	<u>MEMORY</u> average	<u>MEMORY</u> timeline	<u>MEMORY</u> activity	<u>COPY</u>	<u>LINE PROFILE</u> (click to reset order)
42%					87	1 ⚡ import numpy as np
						3 ⚡ def main1():
21%	403M					5 ⚡ x = np.array(range(10**7))
	1G				297	6 ⚡ y = np.array(np.random.uniform(0, 100, size=10**8))

# Scalene: Evolving a New Profiler



hover over bars to see breakdowns; click on COLUMN HEADERS to sort.

show all | hide all | only display profiled lines



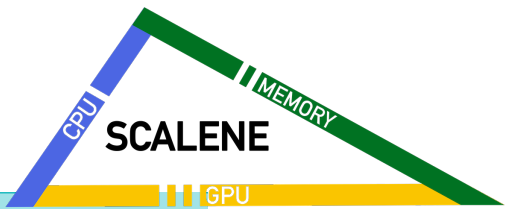
% of time = 100.0% ( 2.769s / 2.769s)

▼ /Users/emerydb/git/scalene/test/issues/test-issue31.py

<u>TIME</u>	<u>MEMORY</u> peak	<u>MEMORY</u> average	<u>MEMORY</u> timeline	<u>MEMORY</u> activity	<u>COPY</u>	<u>LINE PROFILE</u> (click to reset order)
42%					83	1 ⚡ import numpy as np
						3 ⚡ def main1():
21%	403M			▶		5 ⚡ x = np.array(range(10**7))
	1G			▶	29	6 ⚡ y = np.array(np.random.uniform(0, 100, size=10**8))



# Scalene: Evolving a New Profiler



← → ↻ localhost:11235 ☆ 🔒 1 ABP 📄 | G

hover over bars to see breakdowns; click on COLUMN HEADERS to sort.

show all | hide all | only display profiled lines

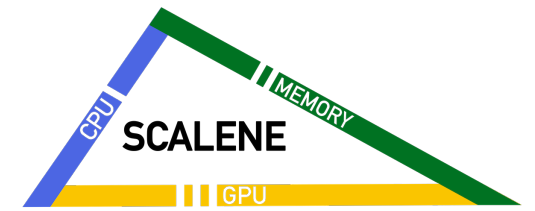
77% 21% 422M 1G % of time = 100.0% ( 2.792s / 2.792s)

▼ /Users/emerydb/git/scalene/test/issues/test-issue31.py

<u>TIME</u>	<u>MEMORY</u>	<u>MEMORY</u>	<u>MEMORY</u>	<u>MEMORY</u>	<u>COPY</u>	<u>LINE PROFILE</u> (click to reset order)
	peak	average	timeline	activity		/Users/emerydb/git/scalene/test/issues/test-issue31.py
36% 14						81 1 ⚡ import numpy as np * ⚡ def main1():

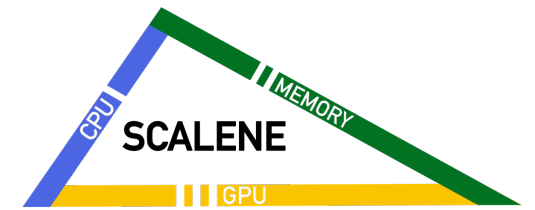
```
# Proposed optimization:  
# The original version of the code utilizes the "range" function to generate a list from 0 to 10  
# function to convert it into an array. However, you can use the Numpy "arange" function directl  
# with the same range. This speeds up the generation process.  
# Also, in the Numpy "random.uniform" function, the parameter 'size' is set to 10**8, which is s  
3 # the size of the array 'x'. This might be a mistake, but since the task says not to change the  
# was intentional and left it as it is.  
# Using such functions (that uses native Numpy functions) increases the speed because these func  
# layer underneath, which is faster than Python's interpreted layer.  
import numpy as np  
def main1():  
    # After optimization  
    x = np.arange(10**7) # replaced range() solution with np.arange() to leverage numpy's speed  
    y = np.random.uniform(0, 100, size=10**8) # no optimization needed here.
```

# Scalene: Evolving a New Profiler



```
# Proposed optimization:
# The original version of the code utilizes the "range" function to generate a list from 0 to 10**7 and the Numpy "array"
# function to convert it into an array. However, you can use the Numpy "arange" function directly to generate an array
# with the same range. This speeds up the generation process.
# Also, in the Numpy "random.uniform" function, the parameter 'size' is set to 10**8, which is significantly larger than
# the size of the array 'x'. This might be a mistake, but since the task says not to change the output, I assumed this
# was intentional and left it as it is.
# Using such functions (that uses native Numpy functions) increases the speed because these functions use a compiled
# layer underneath, which is faster than Python's interpreted layer.
import numpy as np
def main1():
    # After optimization
    x = np.arange(10**7) # replaced range() solution with np.arange() to leverage numpy's speed.
    y = np.random.uniform(0, 100, size=10**8) # no optimization needed here.
```

# Scalene: Evolving a New Profiler



90x speedup

[#58 \(comment\)](#) presents the following code:

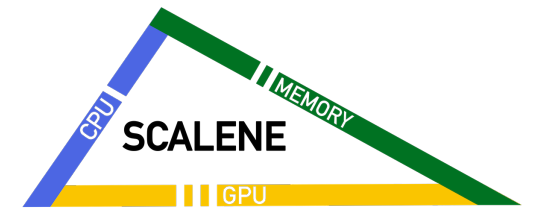
```
for i in range(n_features):
    for n in range(n_samples):
        subgrad[i] += (- y[n] * X[n][i]) if y[n] * (np.dot(X[n], w) + b) < 1 else 0
        subgrad[i] += self.lambda1 * (-1 if w[i] < 0 else 1) + 2 * self.lambda2 * w[i]
```

Scalene proposes the following optimization:

```
# Vectorized operations to replace for loops
subgrad[:-1] = np.sum(-y[:, None] * X * (y * (X.dot(w) + b) < 1)[: , None], axis=0)
subgrad[:-1] += self.lambda1 * np.sign(w) + 2 * self.lambda2 * w
subgrad[-1] = np.sum(-y * (y * (X.dot(w) + b) < 1))
```

Scalene's proposed optimization accelerates the original code by at least 90x (89 seconds to 1 second, when running 500 iterations), and takes full advantage of multiple cores.

# Scalene: Evolving a New Profiler

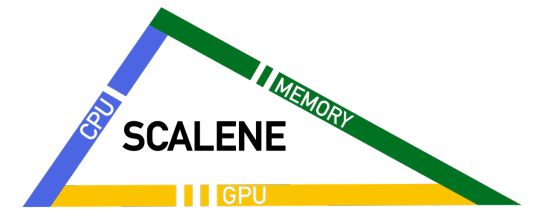


evolve

profiler that suggests optimizations

"closes loop" between tool and human's goal

# Scalene: Evolving a New Profiler



evolve

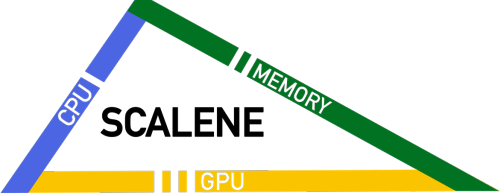
profiler that suggests optimizations

"closes loop" between tool and human's goal

**past profilers:**  
here's where your code  
could be more efficient  
maybe, you figure it out



# Scalene: Evolving a New Profiler

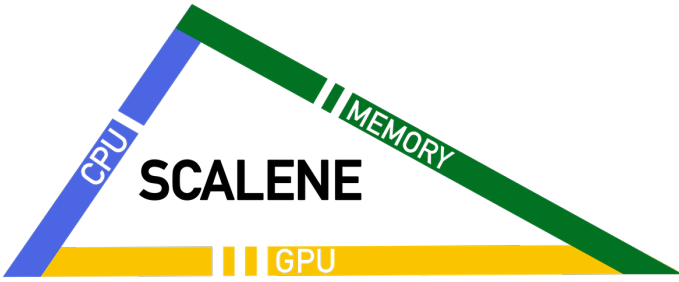


evolve

profiler that suggests optimizations

"closes loop" between tool and human's goal

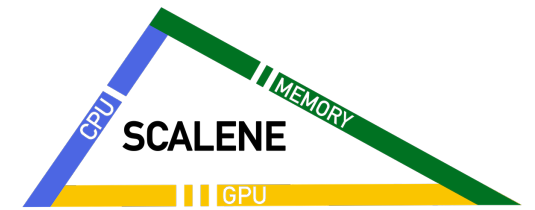
past profilers:  
here's where your code  
could be more efficient  
maybe, you figure it out



I figured out where your  
code could be more efficient  
so you can go home early



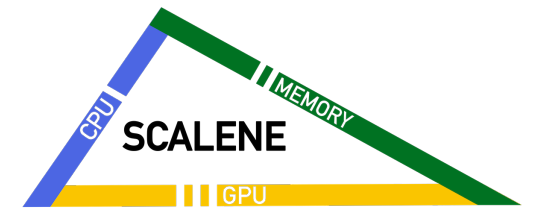
# Scalene: Evolving a New Profiler



exploit niche

profiler knows *where* code is inefficient

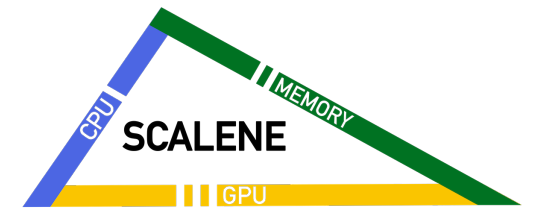
# Scalene: Evolving a New Profiler



exploit niche

profiler knows *where* code is inefficient  
profiler knows *why* code is inefficient

# Scalene: Evolving a New Profiler



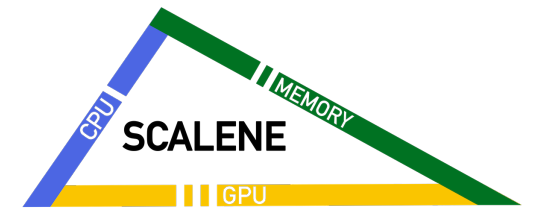
exploit niche

profiler knows *where* code is inefficient

profiler knows *why* code is inefficient

lots of time in interpreter  $\Rightarrow$  use native libraries

# Scalene: Evolving a New Profiler



exploit niche

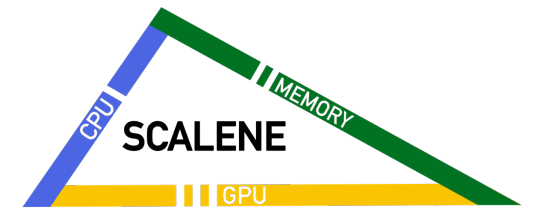
profiler knows *where* code is inefficient

profiler knows *why* code is inefficient

lots of time in interpreter  $\Rightarrow$  use native libraries

low core utilization  $\Rightarrow$  vectorize / MT / MP

# Scalene: Evolving a New Profiler



exploit niche

profiler knows *where* code is inefficient

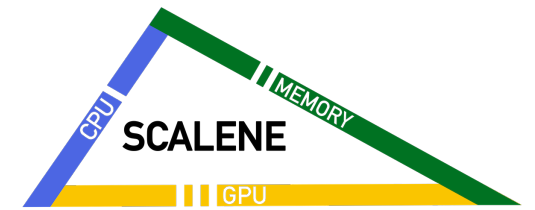
profiler knows *why* code is inefficient

lots of time in interpreter  $\Rightarrow$  use native libraries

low core utilization  $\Rightarrow$  vectorize / MT / MP

no usage of GPU  $\Rightarrow$  use GPU-optimized libraries

# Scalene: Evolving a New Profiler

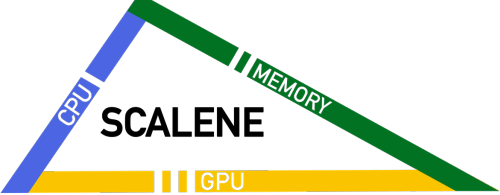


ensure fitness

use original code as oracle

correctness: compare to original via testing

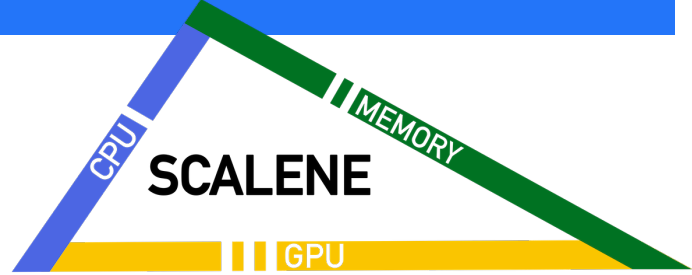
# Scalene: Evolving a New Profiler



ensure fitness

use original code as oracle

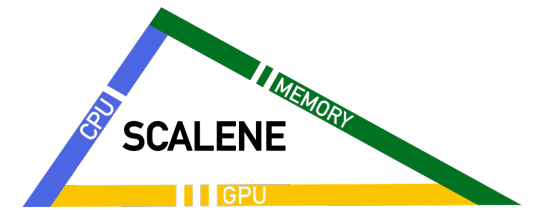
correctness: compare to original via testing



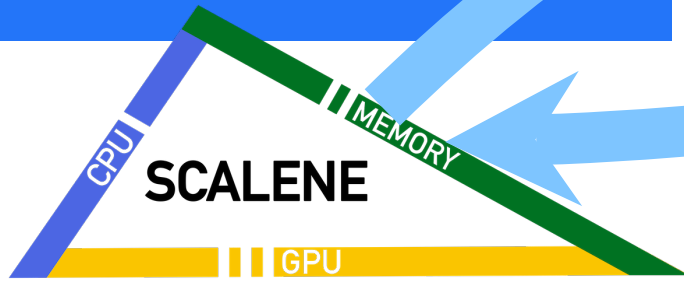
I figured out where your code could be more efficient so you can go home early



# Scalene: Evolving a New Profiler



ensure fitness



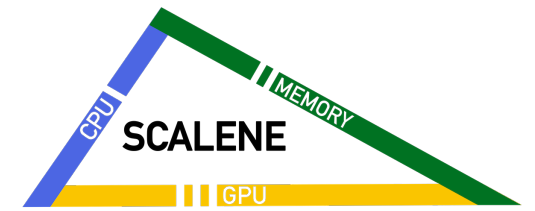
I figured out where your code could be more efficient so you can go home early

use original code as oracle

correctness: compare to original via testing



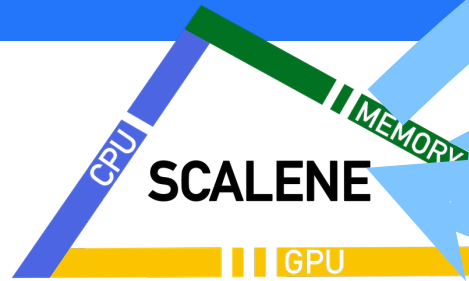
# Scalene: Evolving a New Profiler



ensure fitness

use original code as oracle

correctness: compare to original via testing

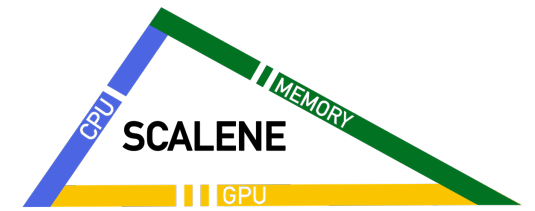


I figured out where your code could be more efficient so you can go home early



perf: compare to original via execution

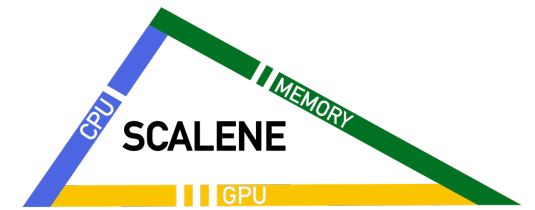
# Scalene: Evolving a New Profiler



evolve

profiler *that*  
*suggests code*  
*optimizations*

# Scalene: Evolving a New Profiler



evolve

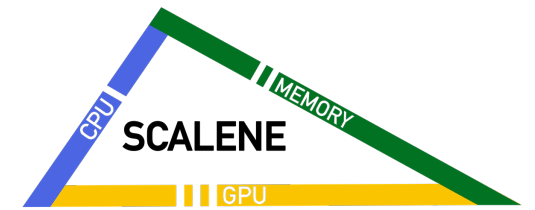
profiler *that suggests code optimizations*



exploit niche

static + dynamic information:  
*code + detailed performance stats*

# Scalene: Evolving a New Profiler



evolve

profiler *that suggests code optimizations*



exploit niche

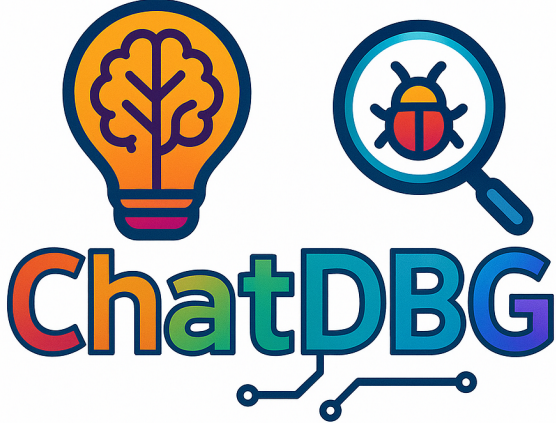
static + dynamic information:  
*code + detailed performance stats*



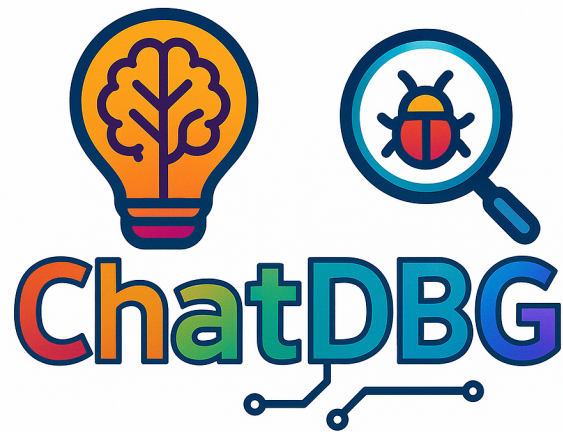
ensure fitness

use to verify success / retry / refine prompts:  
*correctness/perf*

# ChatDBG: Evolving a New Debugger



# ChatDBG: Evolving a New Debugger



by [Emery Berger](#), [Stephen Freund](#), [Kyla Levin](#), [Nicolas van Kempen](#) (ordered alphabetically)

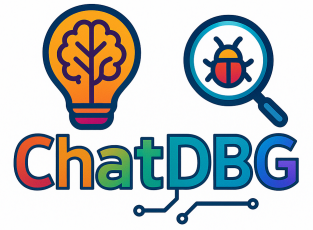
pypi v1.0.0 downloads 83k downloads/month 3k

ChatDBG is an AI-based debugging assistant for C/C++/Python/Rust code that integrates large language models into a standard debugger ( `pdb` , `lldb` , `gdb` ) to help debug your code. With ChatDBG, you can engage in a dialog with your debugger, asking open-ended questions about your program, like `why is x null?` . ChatDBG will *take the wheel* and steer the debugger to answer your queries. ChatDBG can provide error diagnoses and suggest fixes.

As far as we are aware, ChatDBG is the *first* debugger to automatically perform root cause analysis and to provide suggested fixes.

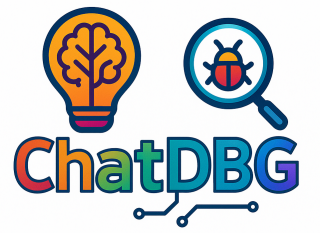
**[FSE 2025]**

# Existing Debuggers



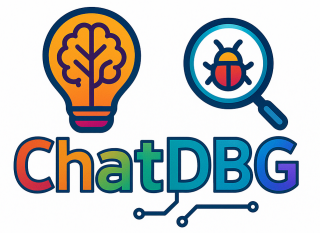
The screenshot shows the VS Code IDE interface with a Python debugger. The top bar indicates 'RUN AND DEBUG' with 'No Configurations' selected. The left sidebar contains the 'VARIABLES' panel, which is expanded to show 'Locals'. Under 'Locals', the variable 'observed\_marbles' is expanded to show its value as an array of 30 'R' characters. Below it, 'special variables' are listed, including '[0:30]', 'dtype', 'max', 'min', 'shape', and 'size'. The 'Globals' section is also visible, showing 'special variables' like '\_\_builtins\_\_'. The main editor area displays the Python code for 'bootstrap.py', with a breakpoint set at line 11. The code includes imports from 'datascience' and 'ds101', a 'make\_marble\_sample' function, a 'proportion\_blue' function, and a 'resampled\_stats' function. The 'resampled\_stats' function is highlighted in yellow. Below the code, the 'TERMINAL' panel shows the command prompt for 'bash ubuntu' and 'Python De...'. The bottom status bar indicates 'Paused on function breakpoint'.


# Existing Debuggers



```
TERMINAL python - ubuntu ⚠ + ▾ 🗑 ☰ ...
○ vscode → /workspaces/ubuntu $ python -m pdb bootstrap.py
> /workspaces/ubuntu/bootstrap.py(1)<module>()
-> from datascience import *
(Pdb) █
```

# Why Even Debug?



←  r/learnprogramming · 1 yr. ago  
Anto1674



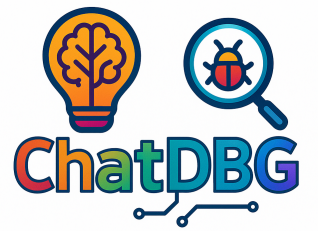
## Is the debugger really useful?

I've been developing games in Unity since about 2-3 years, and programming in general since about 5 years, and I never, and I say NEVER, used a debugger.

I find them pretty useless, it lets you walk step by step the code, I honestly have no clue how to even use them, and I survived years by not using it so I guess it's not really that useful?

I mean, OFCOURSE I use print/debug.log to show where the code reaches and see the values of variables, so my question is, what does the debugger really do? is it really faster than just doing this? I've tried sometime to use it but I just have no clue how to, and tutorial haven't been any useful.

# Why Even Debug?




←  r/learnprogramming · 1 yr. ago  
Anto1674



## Is the debugger really useful?

I've been developing games in Unity since about 2-3 years, and programming in general since about 5 years, and I never, and I say NEVER, used a debugger.

I find them pretty useless, it lets you walk step by step the code, I honestly have no clue how to even use them, and I survived years by not using it so I guess it's not really that useful?

←  r/programming · 16 yr. ago  
irishgeek




## What do you guys think of debuggers? Essential tool? Or crutch?

My boss just implied that he might force me to use a debugger one of these days. Apparently, I take too much time debugging (his convoluted) code. I'm against the idea; not because I dislike debuggers, but I hate the idea of not being able to debug without one. I am will to use one if ever the bug turns out to be complex, but if I didn't rack my brains first, I think it's more of a destructive habit than anything.

# Why Even Debug?




←  r/cscareerquestions · 2 yr. ago  
dmachop

## ← Team does not use debugger

New Grad

I've been dev... never, and I s...  
I find them p... survived year...  
I have little experience as a developer; prior work experience was on business analysis. I joined a new team where they develop backend stuff about python. The thing is the developers use VSCode running on windows but the source is on linux. As a newb, I found it really hard navigating the code - no ctrl+click or hovering over for documentation or using debugger. The linux machine is sandboxed and does not have any external connections.

I brought this up with my team and they suggested **- why not try using print statements?** already tried that but with

←  r/programming · 16 yr. ago  
irishgeek  
I mean, GPCG/IRSF I just print debug log to show where the code reaches and see the values of variables, so my question is, what does the debugger really do? is it really faster than just doing this? I've tried sometime to use it but I still have no clue how to, and to be honest, it hasn't been any useful.

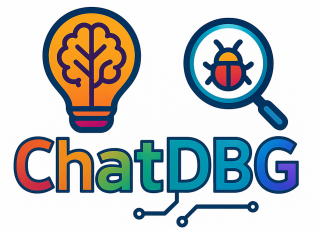
## What do you guys think of debuggers? Essential tool? Or crutch?

My boss just implied that he might force me to use a debugger one of these days. Apparently, I take too much time debugging (his convoluted) code. I'm against the idea; not because I dislike debuggers, but I hate the idea of not being able to debug without one. I am will to use one if ever the bug turns out to be complex, but if I didn't rack my brains first, I think it's more of a destructive habit than anything.





# ChatDBG Demo



```
○ vscode → /workspaces/ubuntu : chatdbg continue bootstrap.py
```

```
Traceback (most recent call last):
```

```
File "/home/vscode/.local/lib/python3.10/site-packages/ipdb/__main__.py", line 323, in main
```

```
    pdb._runscript(mainpyfile)
```

```
File "/usr/lib/python3.10/pdb.py", line 1583, in _runscript
```

```
    self.run(statement)
```

```
File "/workspaces/ubuntu/bootstrap.py", line 19, in <module>
```

```
    stats = resampled_stats(observed_marbles, 5)
```

```
File "/workspaces/ubuntu/bootstrap.py", line 15, in resampled_stats
```

```
    assert len(stats) == num_trials
```

```
AssertionError
```

```
Uncaught exception. Entering post mortem debugging
```

```
> /workspaces/ubuntu/bootstrap.py(15)resampled_stats()
```

```
10
```

```
11 def resampled_stats(observed_marbles, num_trials):
```

```
12     stats = bootstrap_statistic(observed_marbles,
```

```
13         proportion_blue,
```

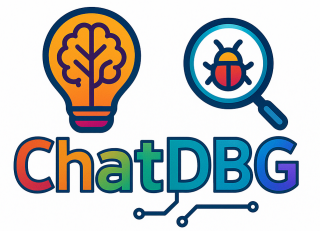
```
14         num_trials)
```

```
----> 15     assert len(stats) == num_trials
```

```
16     return stats
```

```
(ChatDBG) why
```

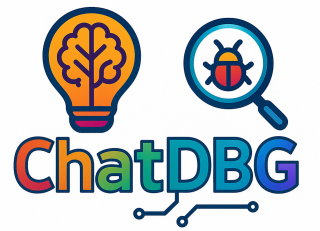
# ChatDBG Demo



```
○ vscode → /workspaces/ubuntu $ chatdbg continue bootstrap.py
Traceback (most recent call last):
  File "/home/vscode/.local/lib/python3.10/site-packages/ipdb/__main__.py", line 323, in main
    pdb._runscript(mainpyfile)
  File "/usr/lib/python3.10/pdb.py", line 1583, in _runscript
    self.run(statement)
  File "/workspaces/ubuntu/bootstrap.py", line 19, in <module>
    stats = resampled_stats(observed_marbles, 5)
  File "/workspaces/ubuntu/bootstrap.py", line 15, in resampled_stats
    assert len(stats) == num_trials
AssertionError
Uncaught exception: Entering post-mortem debugging
> /workspaces/ubuntu/bootstrap.py(15)resampled_stats()
10
11 def resampled_stats(observed_marbles, num_trials):
12     stats = bootstrap_statistic(observed_marbles,
13                                 proportion_blue,
14                                 num_trials)
---> 15     assert len(stats) == num_trials
16     return stats

(ChatDBG) why
```

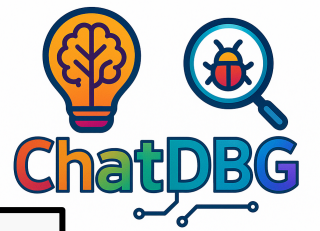
# ChatDBG Demo



```
○ vscode → /workspaces/ubuntu $ chatdbg continue bootstrap.py
Traceback (most recent call last):
  File "/home/vscode/.local/lib/python3.10/site-packages/ipdb/__main__.py", line 323, in main
    pdb._runscript(mainpyfile)
  File "/usr/lib/python3.10/pdb.py", line 1583, in _runscript
    self.run(statement)
  File "/workspaces/ubuntu/bootstrap.py", line 19, in <module>
    stats = resampled_stats(observed_marbles, 5)
  File "/workspaces/ubuntu/bootstrap.py", line 15, in resampled_stats
    assert len(stats) == num_trials
AssertionError
Uncaught exception: Entering post-mortem debugging
> /workspaces/ubuntu/bootstrap.py(15)resampled_stats()
10
11 def resampled_stats(observed_marbles, num_trials):
12     stats = bootstrap_statistic(observed_marbles,
13                                 proportion_blue,
14                                 num_trials)
---> 15     assert len(stats) == num_trials
16     return stats
```

(ChatDBG) why

# ChatDBG Demo



(ChatDBG) why

(ChatDBG) info bootstrap\_statistic

Source from file /workspaces/ubuntu/ds101.py:

```
6     def bootstrap_statistic(observed_sample, compute_statistic, num_trials):
7         if not isinstance(observed_sample, np.ndarray):
8             raise ValueError('The first parameter to bootstrap_statistic must be' + str(type(observed_sample).__name__))
9         for i in np.arange(0, num_trials):
10            simulated_resample = np.random.choice(observed_sample, len(observed_sample))
```

(ChatDBG) up

```
> /workspaces/ubuntu/bootstrap.py(19)<module>()
   18 observed_marbles = make_marble_sample()
---> 19 stats = resampled_stats(observed_marbles, 5)
   20
   21 assert np.isclose(np.mean(stats), 0.7)
```

(ChatDBG) info proportion\_blue

Source from file /workspaces/ubuntu/bootstrap.py:

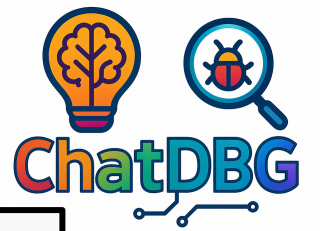
```
8     def proportion_blue(sample):
9         return sample
```

The root cause of the error is due to the use of `bootstrap_statistic` in the `resampled_stats` function. Let's break down the discovered information:

## 1 Function Signature Misalignment:

- The `bootstrap_statistic` function takes three arguments:
  - `observed_sample`: the initial sample as an array.

# ChatDBG Demo



(ChatDBG) why

(ChatDBG) info bootstrap\_statistic

Source from file /workspaces/ubuntu/ds101.py:

```
6     def bootstrap_statistic(observed_sample, compute_statistic, num_trials):
7         if not isinstance(observed_sample, np.ndarray):
8             raise ValueError('The first parameter to bootstrap_statistic must be' + str(type(observed_sample).__name__))
9         for i in np.arange(0, num_trials):
10            simulated_resample = np.random.choice(observed_sample, len(observed_sample))
```

(ChatDBG) up

```
> /workspaces/ubuntu/bootstrap.py(19)<module>()
   18 observed_marbles = make_marble_sample()
---> 19 stats = resampled_stats(observed_marbles, 5)
   20
   21 assert np.isclose(np.mean(stats), 0.7)
```

(ChatDBG) info proportion\_blue

Source from file /workspaces/ubuntu/bootstrap.py:

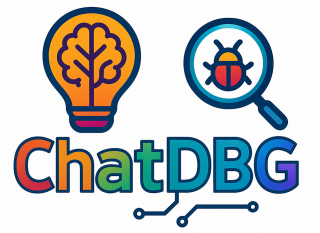
```
8     def proportion_blue(sample):
9         return sample
```

The root cause of the error is due to the use of `bootstrap_statistic` in the `resampled_stats` function. Let's break down the discovered information:

## 1 Function Signature Misalignment:

- The `bootstrap_statistic` function takes three arguments:
  - `observed_sample`: the initial sample as an array.

# ChatDBG Demo



## Recommendation

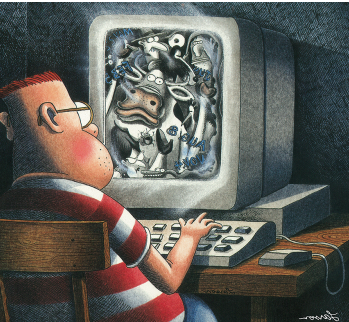
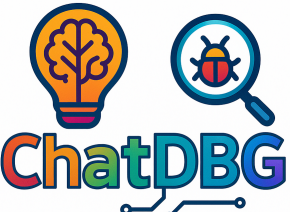
To resolve the error, redefine the `proportion_blue` function to correctly compute and return a statistic from the given sample. For example:

```
def proportion_blue(sample):  
    # Assuming 'B' represents blue in the sample  
    return np.sum(sample == 'B') / len(sample)
```

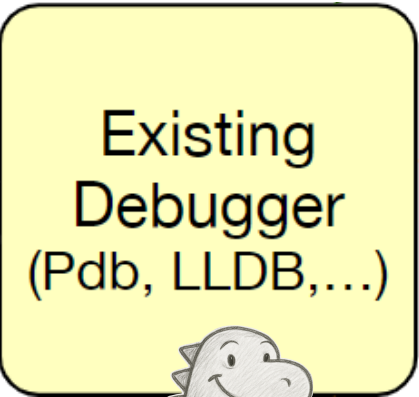
This definition calculates the proportion of blue marbles in the sample, which should align with the expected functionality for computing statistics.

[Cost: ~\$0.02 USD]  
(ChatDBG)

# ChatDBG Overview



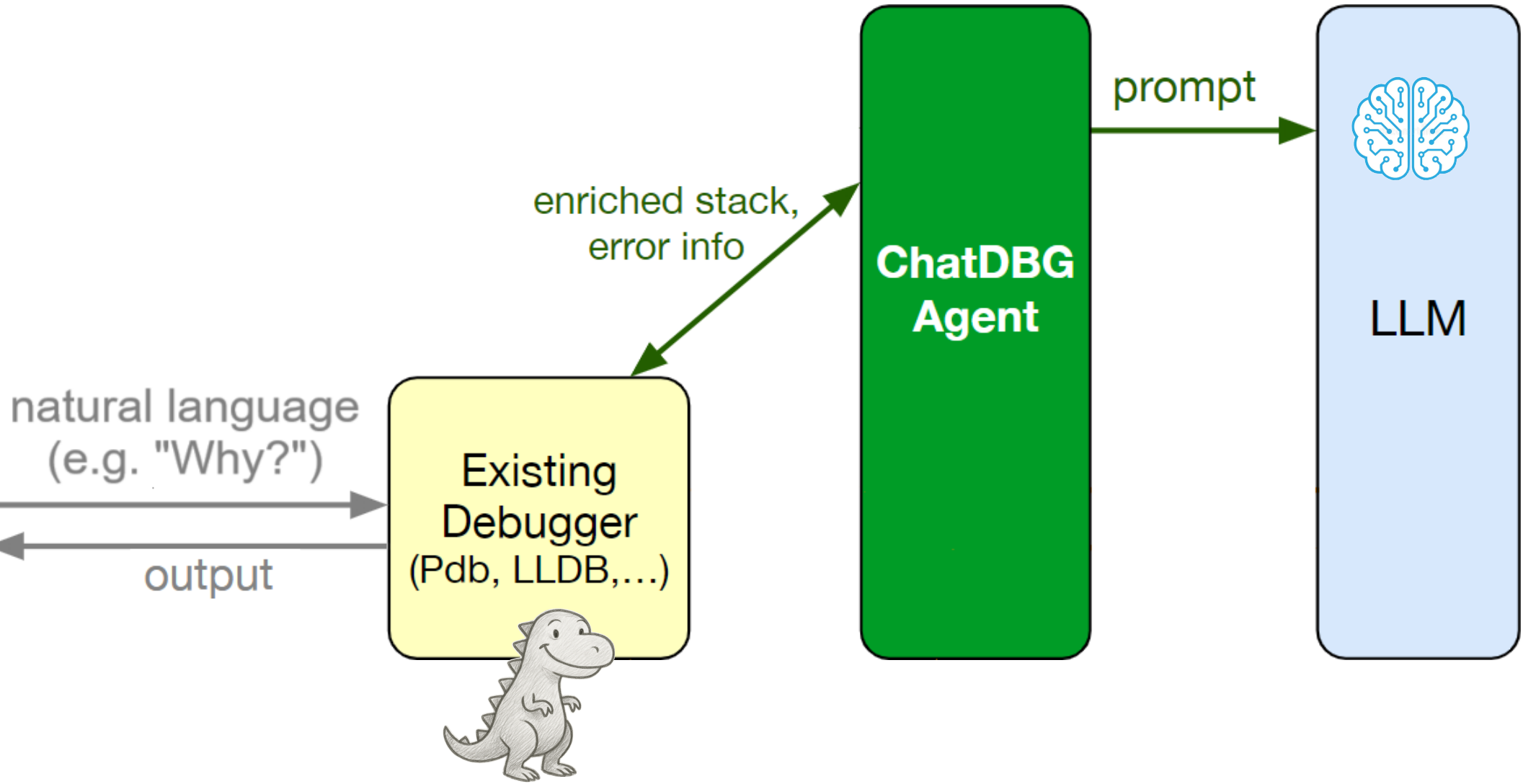
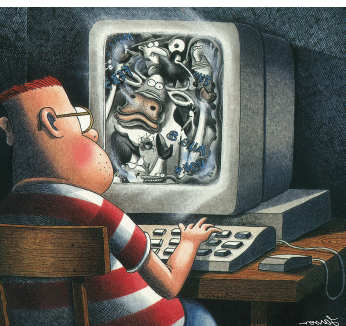
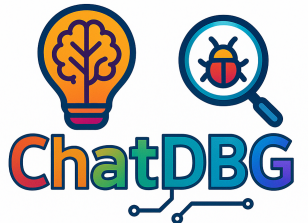
natural language  
(e.g. "Why?")



output



# ChatDBG Overview



# ChatDBG Overview: Enriched Stack

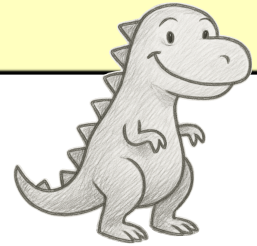
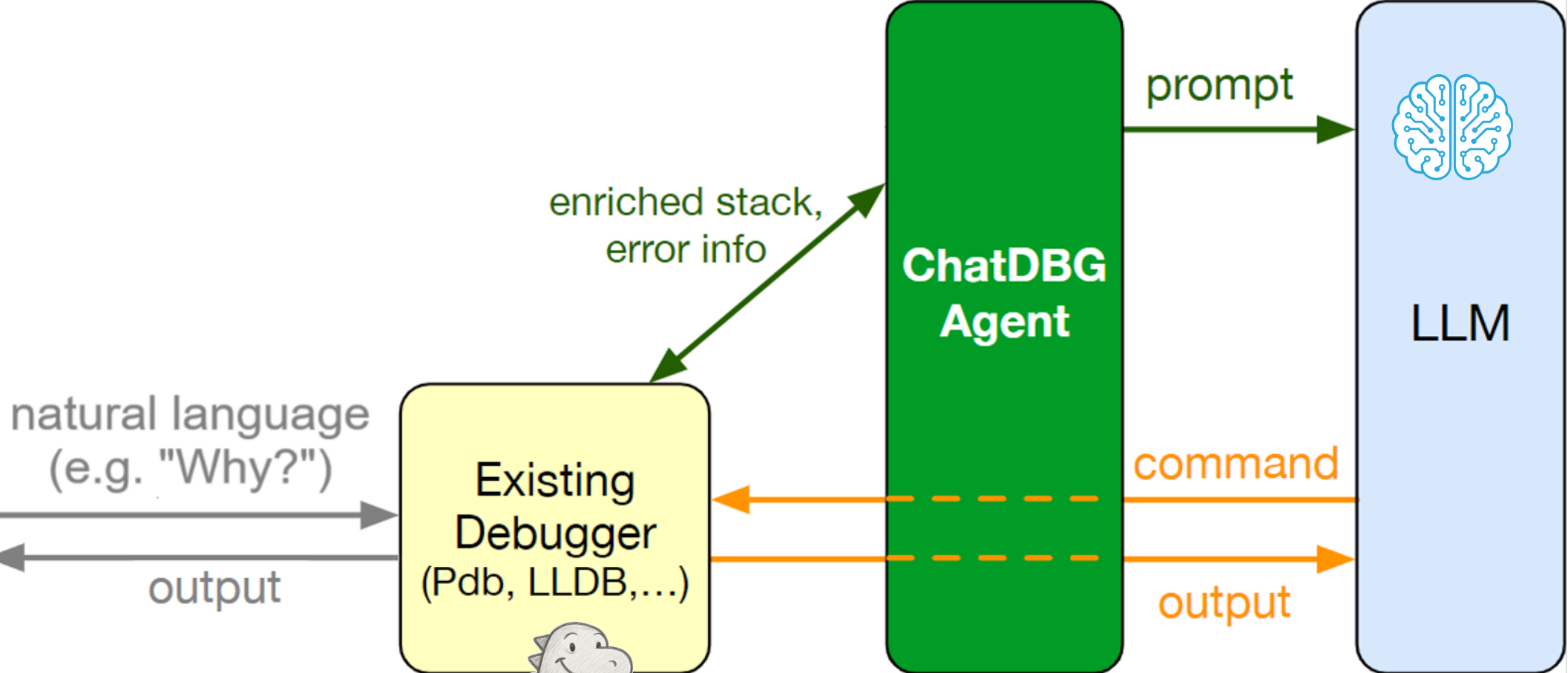
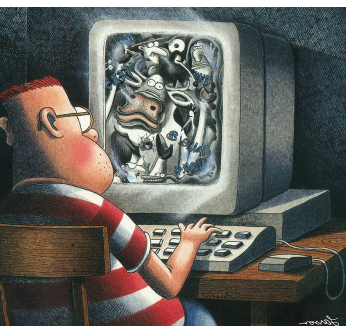
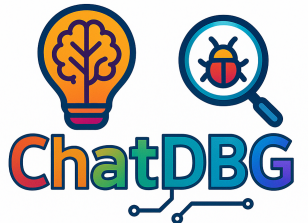
Skip library code

Increased context

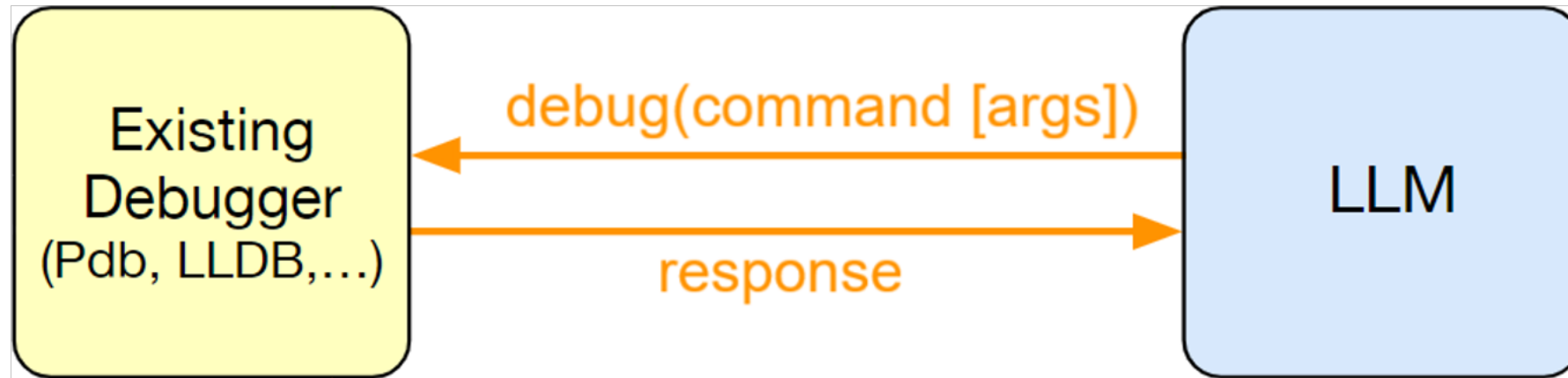
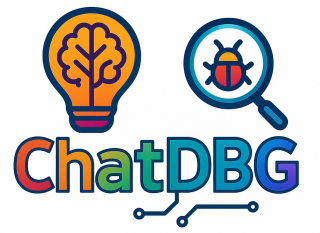
Global and local variables

```
[... skipping 4 hidden frame(s)]  
/workspaces/ubuntu/ChatDBG/samples/python/bootstrap.py(19)<module>()  
12     stats = bootstrap_statistic(observed_marbles,  
13                                 proportion_blue,  
14                                 num_trials)  
15     assert len(stats) == num_trials  
16     return stats  
17  
18 observed_marbles = make_marble_sample()  
> 19 stats = resampled_stats(observed_marbles, 5)  
20  
21 assert np.isclose(np.mean(stats), 0.7)  
  
Global variables:  
  observed_marbles: ndarray = array(['R', 'R', 'R', ..., 'B', 'B', 'B'], dtype='<U1')  
  
> /workspaces/ubuntu/ChatDBG/samples/python/bootstrap.py(15)resampled_stats()  
10  
11 def resampled_stats(observed_marbles, num_trials):  
12     stats = bootstrap_statistic(observed_marbles,  
13                                 proportion_blue,  
14                                 num_trials)  
---> 15     assert len(stats) == num_trials  
16     return stats  
17  
18 observed_marbles = make_marble_sample()  
19 stats = resampled_stats(observed_marbles, 5)  
  
Variables in this frame:  
  num_trials: int = 5  
  observed_marbles: ndarray = array(['R', 'R', 'R', ..., 'B', 'B', 'B'], dtype='<U1')  
  stats: ndarray = array(['B', 'B', 'R', ..., 'B', 'R', 'B'], dtype='<U32')
```

# ChatDBG Overview: LLM Drives Debugging

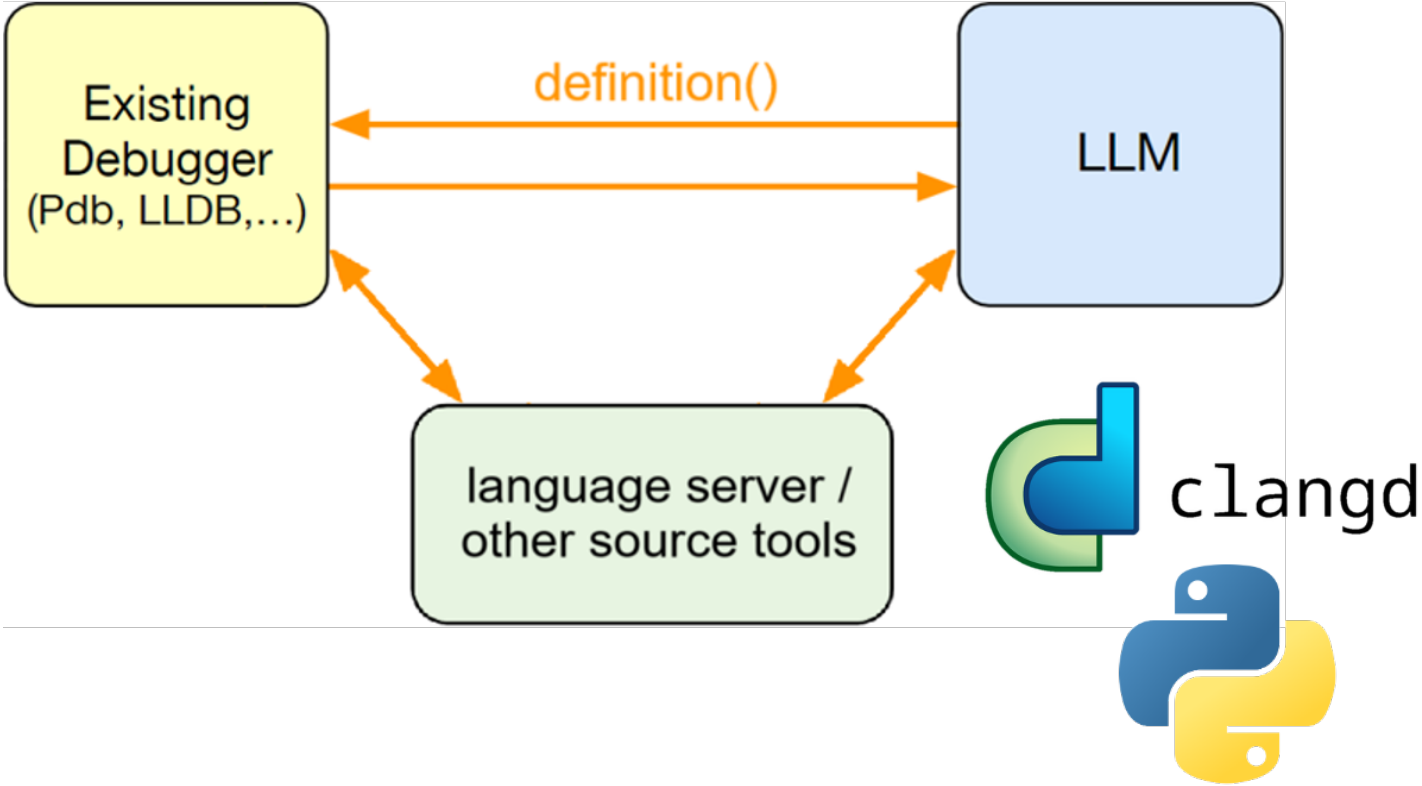
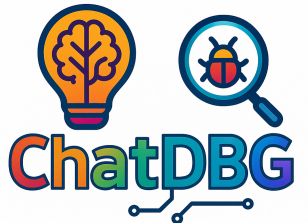


# ChatDBG: Driving the Debugger

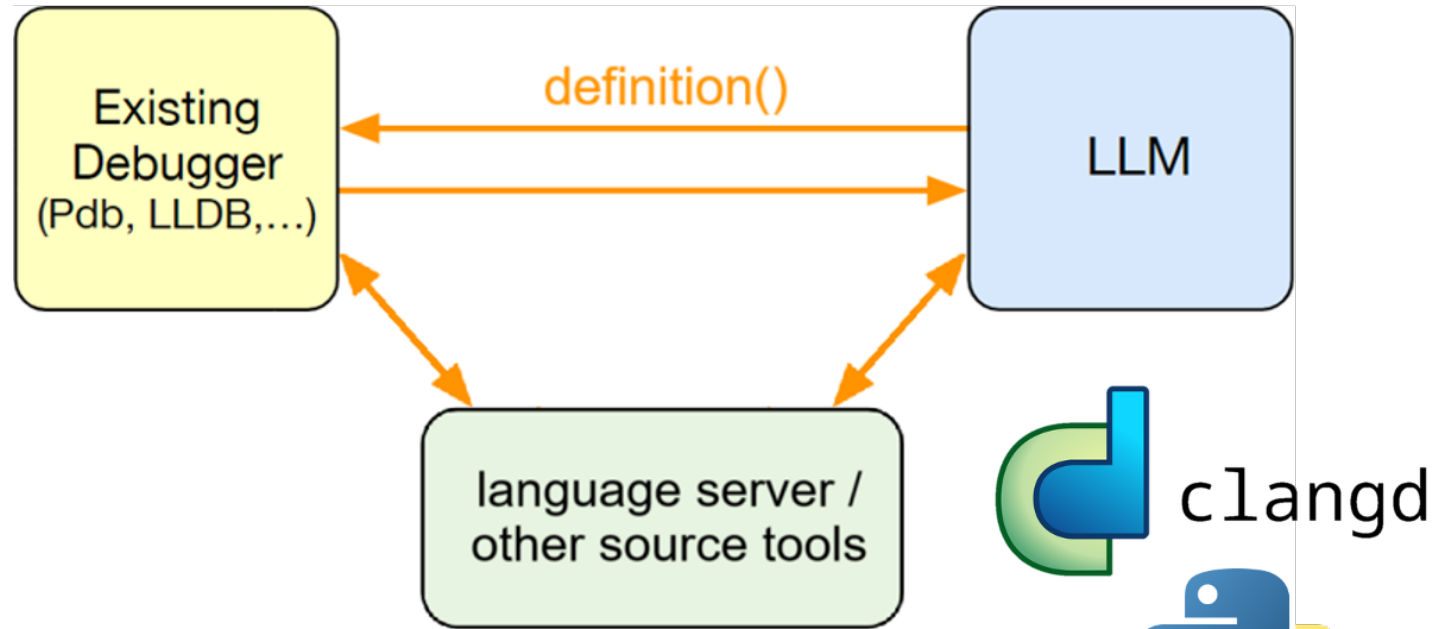
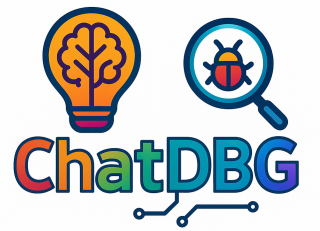


```
(ChatDBG) print len(stats)
150
```

# ChatDBG: Extended Functionality



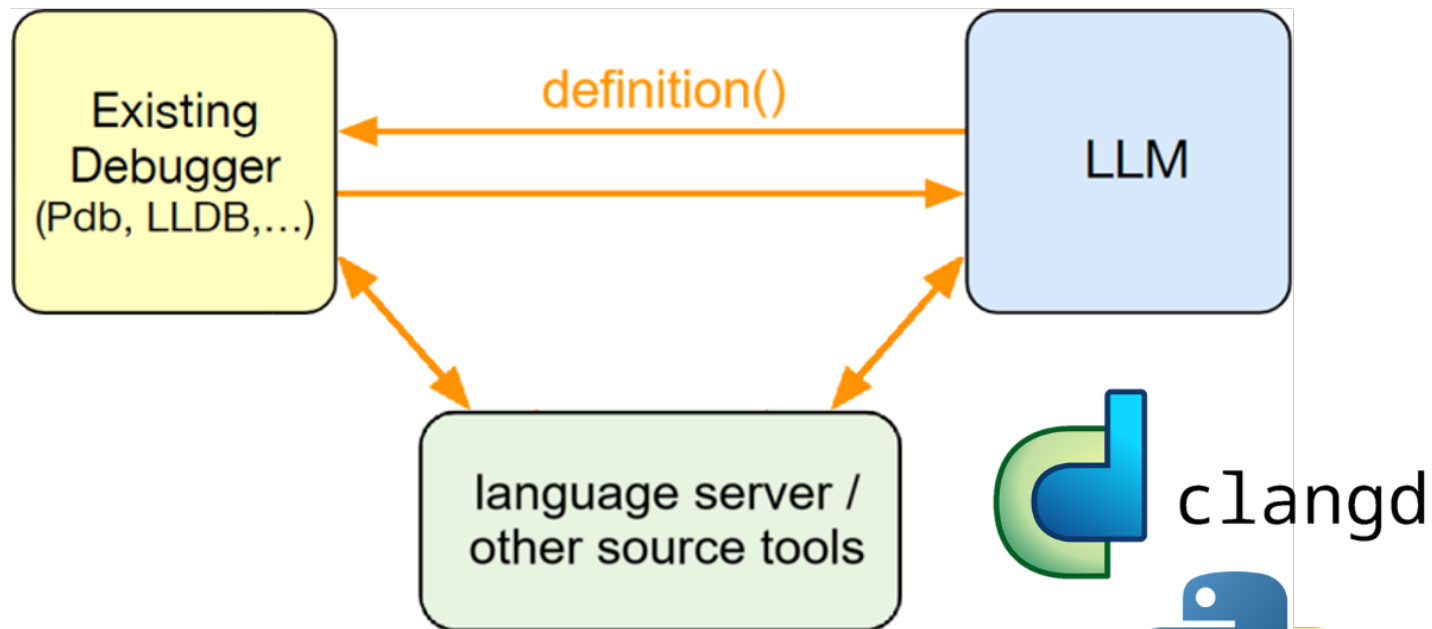
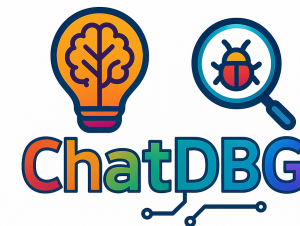
# ChatDBG: Extended Functionality



`code(filename, lineno)`: source code surrounding location

`definition(symbol, filename, lineno)`: declaration for first occurrence of symbol

# ChatDBG: Extended Functionality



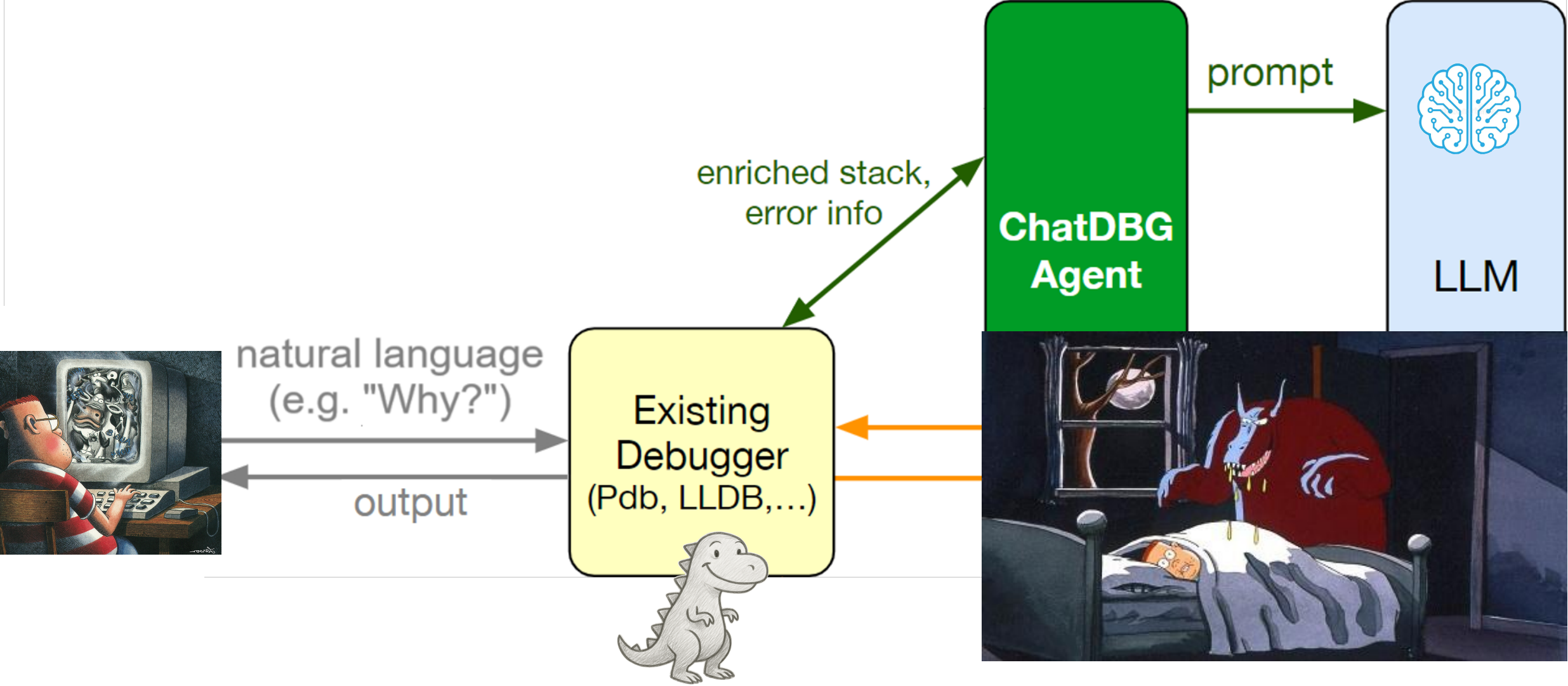
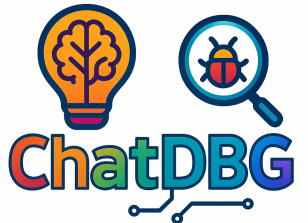
`code(filename, lineno)`: source code surrounding location  
`definition(symbol, filename, lineno)`: declaration for first occurrence of symbol



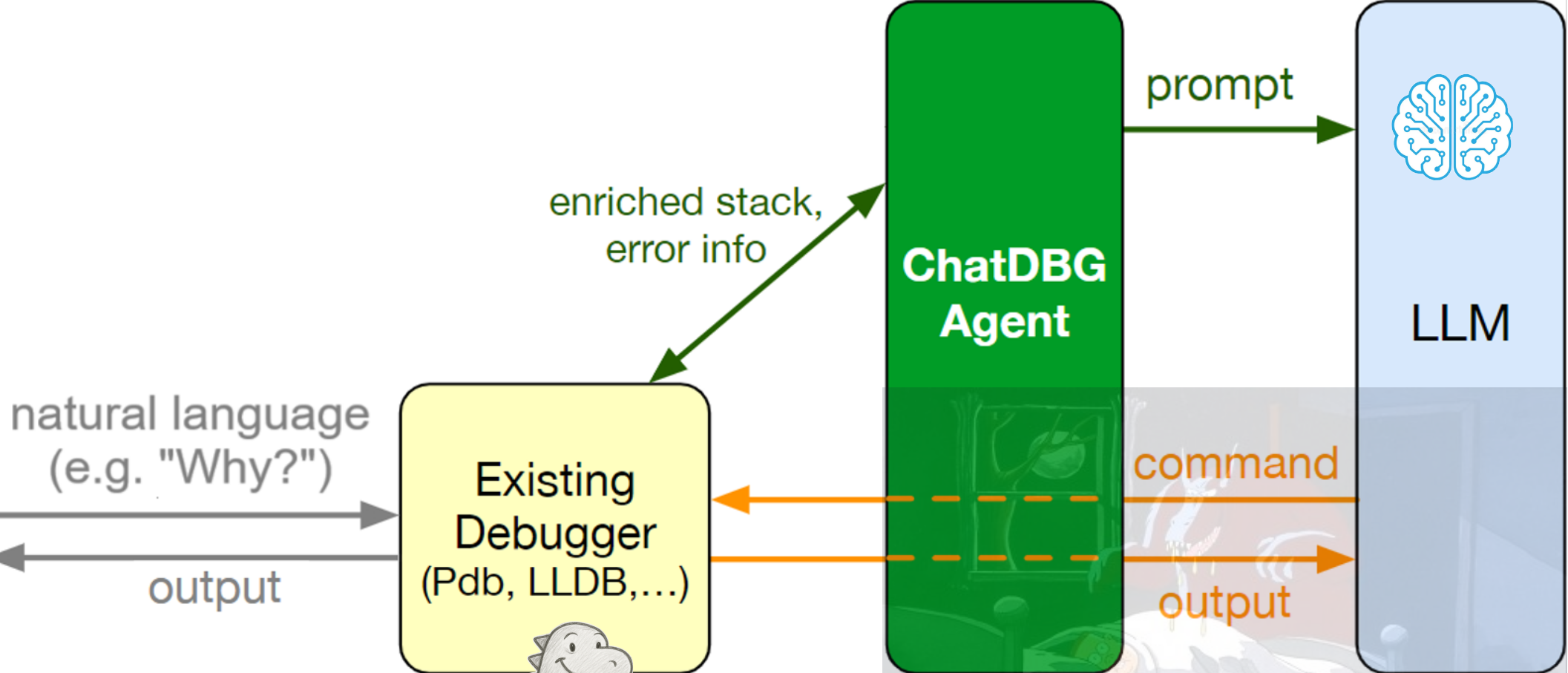
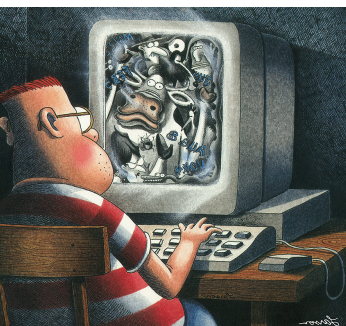
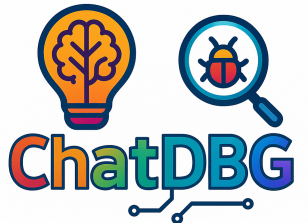
`info(symbol)`: source code/docstring for symbol (function, method, field, class...)  
`slice(symbol)`: source code in *backwards slice* of symbol



# ChatDBG: Security



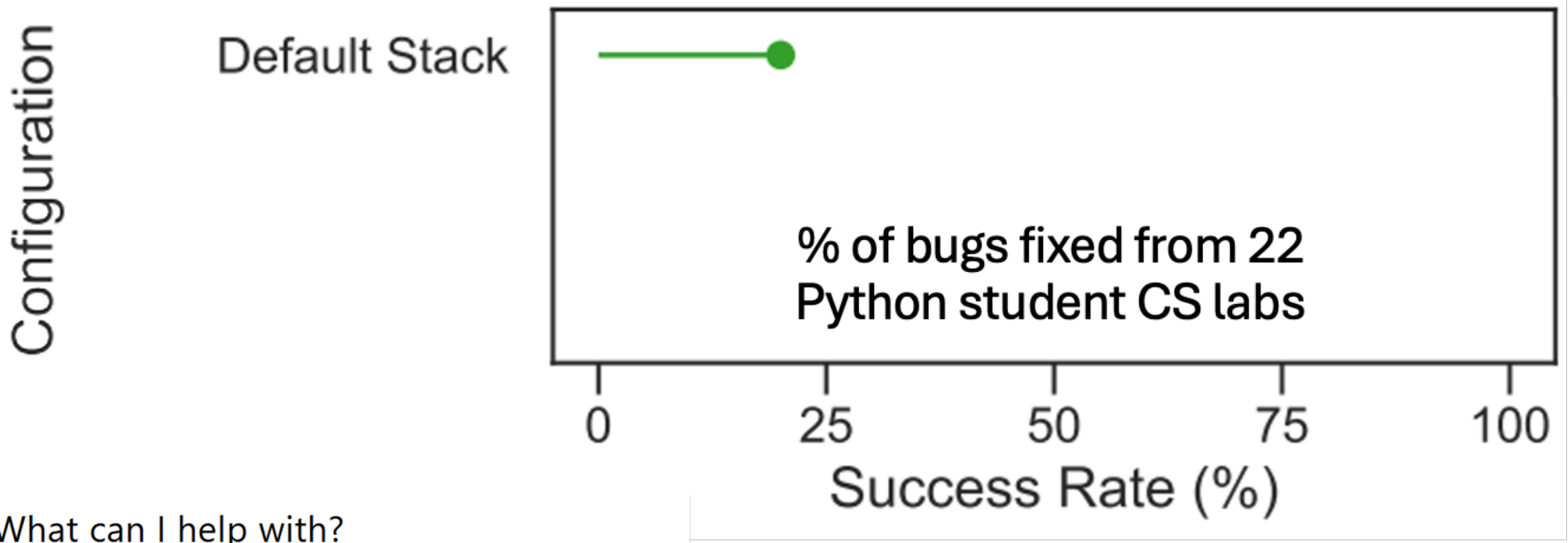
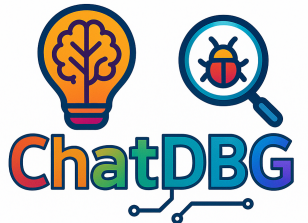
# ChatDBG: Security



Whitelisted commands + sandbox



# ChatDBG: Evolving a New Debugger

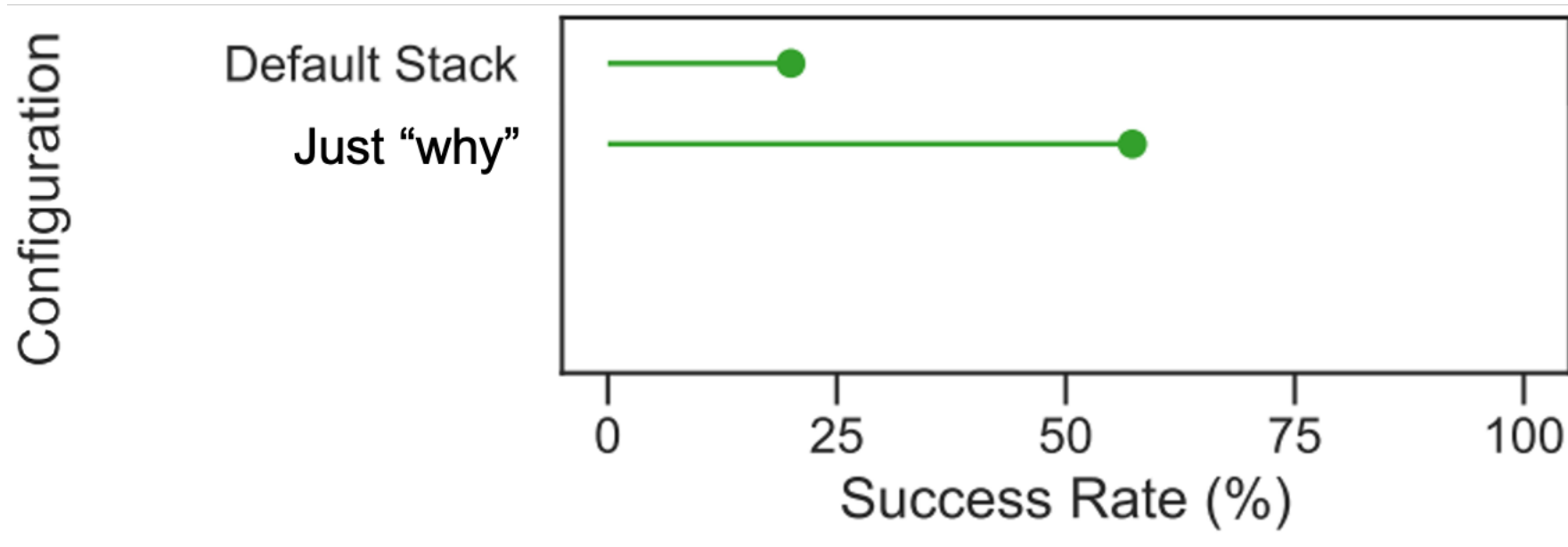
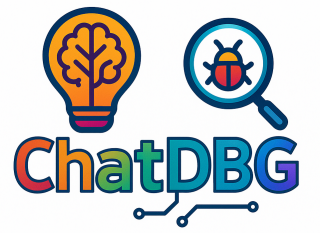


What can I help with?

```
Traceback (most recent call last):
  File "/workspaces/ubuntu/ChatDBG/samples/python/bootstrap.py", line 19, in <module>
    stats = resampled_stats(observed_marbles, 5)
  File "/workspaces/ubuntu/ChatDBG/samples/python/bootstrap.py", line 15, in resampled_stats
    assert len(stats) == num_trials
AssertionError
```

+ Search Reason Deep research Create image ...

# ChatDBG: Evolving a New Debugger

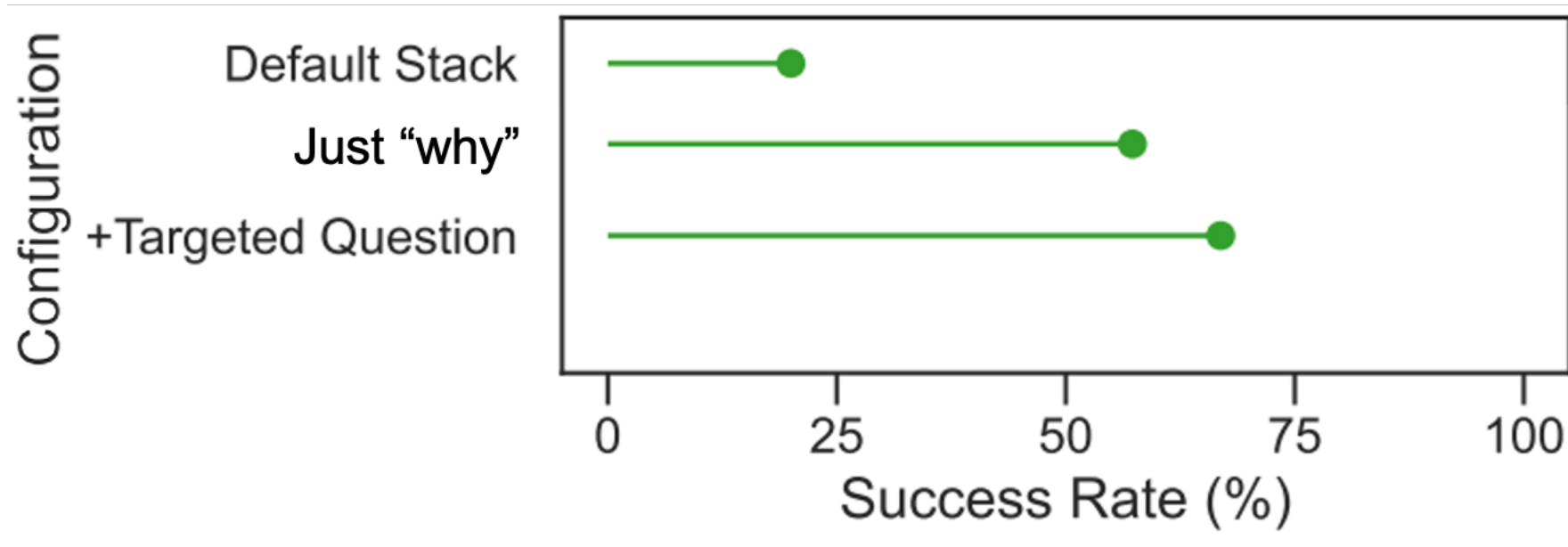
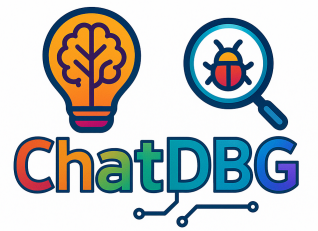


```
(ChatDBG) info proportion_blue
```

```
Source from file /workspaces/ubuntu/ChatDBG/samples/python/bootstrap.py:
```

```
8     def proportion_blue(sample):  
9         return sample
```

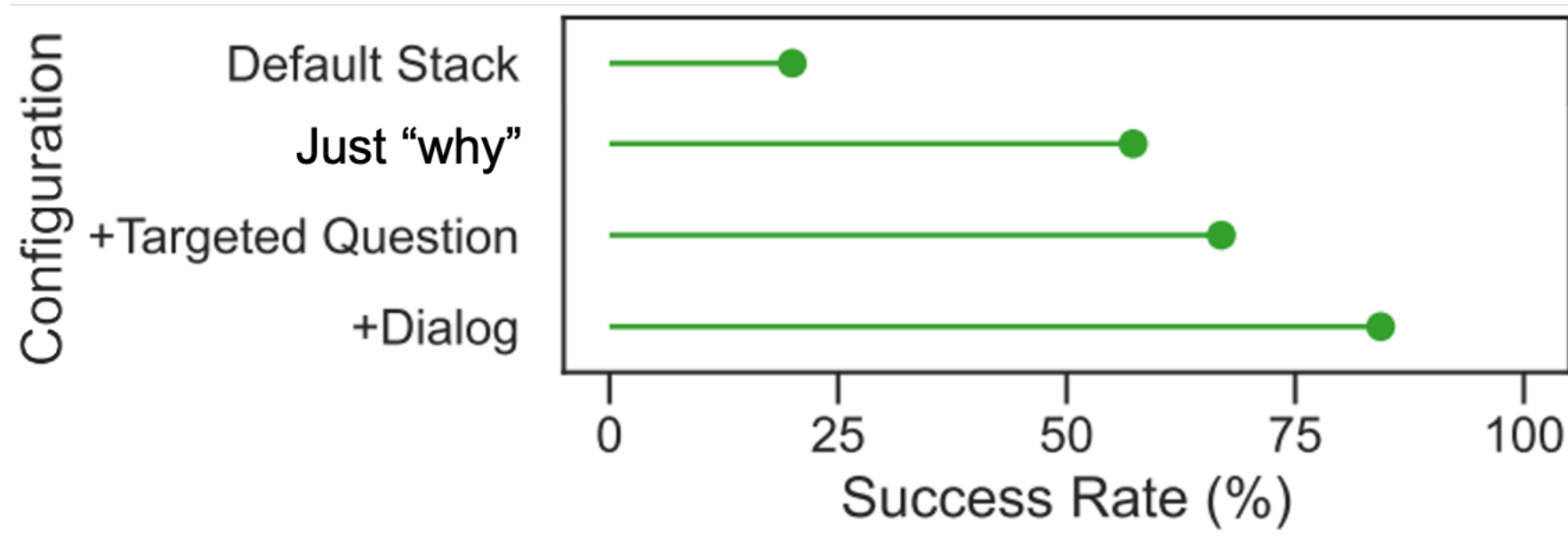
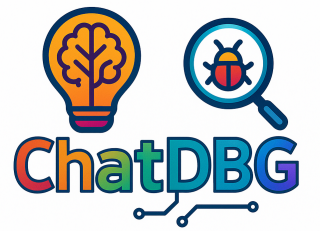
# ChatDBG: Evolving a New Debugger



```
(ChatDBG) why is the array length 5 when it should be 4?
```

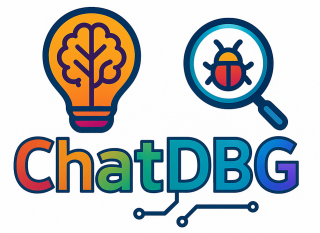
```
(ChatDBG) why is x a nullptr when it should point to an integer?
```

# ChatDBG: Evolving a New Debugger



```
(ChatDBG) Continue to explain your reasoning and give me a fix to make it work as I describe.
```

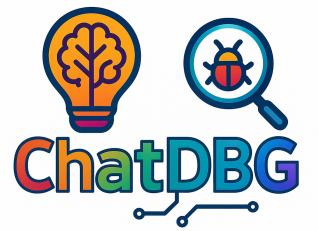
# ChatDBG: Evolving a New Debugger



evolve

debugger *that*  
*performs root*  
*cause analysis*  
*& proposes*  
*fixes*

# ChatDBG: Evolving a New Debugger



evolve

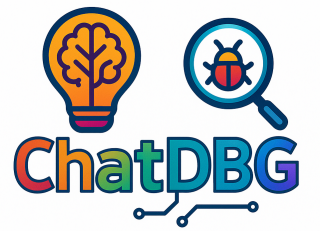
debugger *that performs root cause analysis & proposes fixes*



exploit niche

source + *dynamic state (values & types...)* + *world knowledge*

# ChatDBG: Evolving a New Debugger



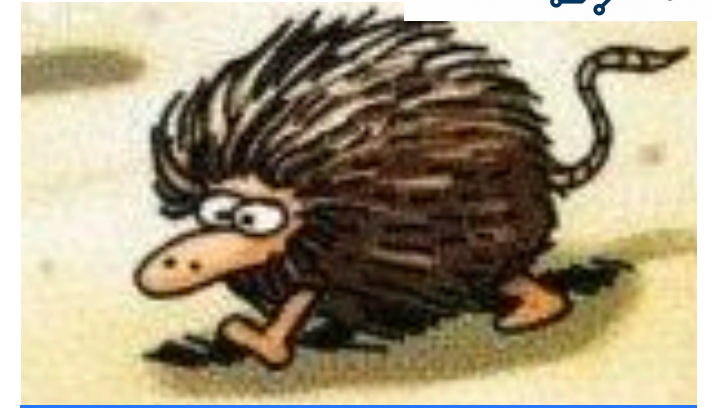
evolve

debugger *that performs root cause analysis & proposes fixes*



exploit niche

source + *dynamic state (values & types...)* + *world knowledge*



ensure fitness

patch code, restart, verify fix prevents crash / exception...

# CWhy: Evolving a New Compiler



# CWhy: Evolving a New Compiler



by [Nicolas van Kempen](#) and [Emery Berger](#).

pypi v0.5 downloads 30k

("See why")

Explains and suggests fixes for compiler error messages for a wide range of programming languages, including C, C++, C#, Go, Java, LaTeX, PHP, Python, Ruby, Rust, Swift, and TypeScript.

# Why CWhy?



r/cpp\_questions · 2 yr. ago  
gnanaprakash2918

## Interpreting errors in C++

SOLVED

I am very new to C++ and I find it extremely hard to interpret the errors thrown by the compiler ?

How do I learn to interpret C++ errors , It shows pages of errors which I dont think is human readable .

I dont understand a single word in it.

# Why CWhy?



r/cpp\_questions · 2 yr. ago  
gnanaprakash2918

## Interpreting errors in C++

SOLVED

I am very new to C++ and I find it extremely hard to interpret the errors thrown by the compiler ?

How do I learn to interpret C++ errors , It shows pages of errors which I dont think is human readable .

I dont understand a single word in it.

# Why CWhy?



r/cpp\_questions · 2 yr. ago  
gnanaprakash2918

## Interpreting errors in C++

SOLVED

I am very new to C++ and I find it extremely hard to interpret the errors thrown by the compiler ?

How do I learn to interpret C++ errors , It shows pages of errors which I dont think is human readable .

I dont understand a single word in it.

```
% clang++ --std=c++20 -c missing-hash.cpp
```

# Why CWhy?

In file included from missing-hash.cpp:1:

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/functional:548:

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/\_\_functional/function.h:24:

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/\_\_memory/builtin\_new\_allocator.h:13:

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/\_\_memory/unique\_ptr.h:21:

**/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/\_\_memory/compressed\_pair.h:77:52: error: call to implicitly-deleted default constructor of 'std::hash<std::pair<int, int>>'**

```
77 |     _LIBCPP_HIDE_FROM_ABI _LIBCPP_CONSTEXPR explicit  
__compressed_pair_elem(__default_init_tag) {}
```

**/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/\_\_memory/compressed\_pair.h:119:42: note: in instantiation of member function 'std::\_\_compressed\_pair\_elem<std::hash<std::pair<int, int>>, 1>::\_\_compressed\_pair\_elem'** requested here

```
119 |         : _Base1(std::forward<_U1>(__t1)), _Base2(std::forward<_U2>(__t2)) {}
```

**/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/\_\_hash\_table:1025:7: note: in instantiation of function template specialization 'std::\_\_compressed\_pair<unsigned long, std::hash<std::pair<int, int>>>::\_\_compressed\_pair<int, std::default\_init\_tag>'** requested here

```
1025 |     : __compressed_pair<unsigned long, std::hash<std::pair<int, int>>>::__compressed_pair<int, std::default_init_tag>
```

```
>() {}
```

```
Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/unordered_set:623:25: note: in instantiation of function template specialization
```

# Why CWhy?

```
1025 |         unordered_set<pair<int, int>, __default_init_tag(), __ps_1.01, __default_init_tag(), 1>  
      |         ^
```

```
/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/
```

```
include/c++/v1/unordered_set:623:25: note: in instantiation of member function
```

```
'std::__hash_table<std::pair<int, int>, std::hash<std::pair<int, int>>,  
std::equal_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::__hash_table' requested  
here
```

```
623 |     _LIBCPP_HIDE_FROM_ABI unordered_set(  
_NOEXCEPT_(is_nothrow_default_constructible<__table>::value) {}
```

```
missing-hash.cpp:13:45: note: in instantiation of member function  
'std::unordered_set<std::pair<int, int>>::unordered_set' requested here
```

```
13 |     std::unordered_set<std::pair<int, int>> visited;  
    |     ^
```

```
/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/
```

```
include/c++/v1/__functional/hash.h:505:36: note: default constructor of 'hash<std::pair<int, int>>'  
is implicitly deleted because base class '__enum_hash<pair<int, int>>' has a deleted default  
constructor
```

```
505 | struct _LIBCPP_TEMPLATE_VIS hash : public __enum_hash<_Tp> {};
```

```
/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/
```

```
include/c++/v1/__functional/hash.h:499:3: note: '__enum_hash' has been explicitly marked deleted  
here
```

```
499 |     __enum_hash() = delete;
```



```
'std::unordered_set<std::pair<int, int>>::~~unordered_set' requested here
```

# Why CWhy?

It

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/functional:552:

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/unordered\_map:591:

In file included from /Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/unordered\_map:591:

**/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/unordered\_map:1806:18: error: type 'hasher' (aka 'std::hash<std::pair<int, int>>') does not provide a call operator**

```
1806 |     __h->__hash_ = hash_function()(__h->__get_value());  
      |                   ^NNNNNNNNNNNNNNNNNN
```

**/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/unordered\_map:1557:30: note: in instantiation of function template specialization 'std::\_\_hash\_table<std::pair<int, int>, std::hash<std::pair<int, int>>, std::equal\_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::\_\_construct\_node<const std::pair<int, int> &>' requested here**

```
1557 |     __node_holder __h = __construct_node(std::forward<_Args>(__args)...);  
      |                   ^
```

**/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/include/c++/v1/unordered\_map:814:12: note: in instantiation of function template specialization 'std::\_\_hash\_table<std::pair<int, int>, std::hash<std::pair<int, int>>, std::equal\_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::\_\_emplace\_unique\_impl<const std::pair<int, int> &>' requested here**

```
814 |     return __emplace_unique_impl(std::forward<_Pp>(__x));  
      |           ^
```

# Why CWhy?

does not provide a call operator

```

1385 |     __nd->__hash_ = hash_function()(__nd->__get_value());
      |                ^NNNNNNNNNNNNNNNNNN

```

/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/

include/c++/v1/\_\_hash\_table:1558:30: note: in instantiation of member function 'std::\_\_hash\_table<std::pair<int, int>, std::hash<std::pair<int, int>>, std::equal\_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::\_\_node\_insert\_unique' requested here

```

1558 |     pair<iterator, bool> __r = __node_insert_unique(__h.get());
      |                ^

```

/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/

include/c++/v1/\_\_hash\_table:814:12: note: in instantiation of function template specialization 'std::\_\_hash\_table<std::pair<int, int>, std::hash<std::pair<int, int>>, std::equal\_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::\_\_emplace\_unique\_impl<const std::pair<int, int> &>' requested here

```

814 |     return __emplace_unique_impl(std::forward<_Pp>(__x));
      |     ^

```

/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/usr/

include/c++/v1/\_\_hash\_table:797:12: note: in instantiation of function template specialization 'std::\_\_hash\_table<std::pair<int, int>, std::hash<std::pair<int, int>>, std::equal\_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::\_\_emplace\_unique\_extract\_key<const std::pair<int, int> &>' requested here

```

797 |     return __emplace_unique_extract_key(std::forward<_Pp>(__x), __can_extract_key<_Pp, key_type>());
      |     ^

```

```
797 |         return __emplace_unique_extract_key(std::forward<_Pp>(__x), __can_extract_key<_Pp,
```

# Why CWhy?

```
//  
include/c++/v1/unordered_set:750:21: note: in instantiation of function template specialization  
'std::__hash_table<std::pair<int, int>, std::hash<std::pair<int, int>>,  
std::equal_to<std::pair<int, int>>, std::allocator<std::pair<int, int>>>::__emplace_unique<const  
std::pair<int, int> &>' requested here
```

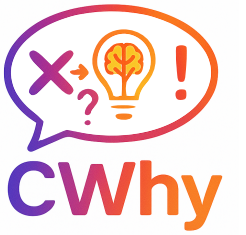
```
750 |         return __table_.__emplace_unique(std::forward<_Args>(__args)...);  
      |         ^
```

```
missing-hash.cpp:20:44: note: in instantiation of function template specialization  
'std::unordered_set<std::pair<int, int>>::emplace<const std::pair<int, int> &>' requested here
```

```
20 |         const auto [_, inserted] = visited.emplace(n->position);  
      |         ^
```

4 errors generated.

# CWhy Demo



The problem is that you are attempting to use `std::unordered_set` with `std::pair<int, int>` as the key type. However, the standard library does not provide a hash function specialization for `std::pair` out of the box, so the default constructor of the unordered set is deleted.

To resolve this, you'll need to provide a custom hash function for `std::pair<int, int>`. Here's an example of how you can define one:

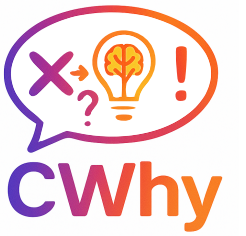
```
```cpp
struct PairHash {
    template <typename T1, typename T2>
    std::size_t operator()(const std::pair<T1, T2>& pair) const {
        std::hash<T1> hash1;
        std::hash<T2> hash2;
        return hash1(pair.first) ^ (hash2(pair.second) << 1);
    }
};
```
```

Then, when instantiating the `std::unordered_set`, you can specify the custom hash function:

```
```cpp
std::unordered_set<std::pair<int, int>, PairHash> visited;
```
```

With this change, the code should now compile and work as expected.

# CWhy Demo



The problem is that you are attempting to use `std::unordered_set` with `std::pair<int, int>` as the key type. However, the standard library does not provide a hash function specialization for `std::pair` out of the box, so the default constructor of the unordered set is deleted.

To resolve this, you'll need to provide a custom hash function for `std::pair<int, int>`. Here's an example of how you can define one:

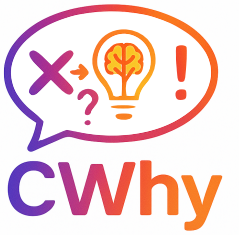
```
```cpp
struct PairHash {
    template <typename T1, typename T2>
    std::size_t operator()(const std::pair<T1, T2>& pair) const {
        std::hash<T1> hash1;
        std::hash<T2> hash2;
        return hash1(pair.first) ^ (hash2(pair.second) << 1);
    }
};
```
```

Then, when instantiating the `std::unordered_set`, you can specify the custom hash function:

```
```cpp
std::unordered_set<std::pair<int, int>, PairHash> visited;
```
```

With this change, the code should now compile and work as expected.

# CWhy Demo



The problem is that you are attempting to use `std::unordered_set` with `std::pair<int, int>` as the key type. However, the standard library does not provide a hash function specialization for `std::pair` out of the box, so the default constructor of the unordered set is deleted.

To resolve this, you'll need to provide a custom hash function for `std::pair<int, int>`. Here's an example of how you can define one:

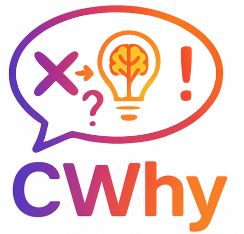
```
```cpp
struct PairHash {
    template <typename T1, typename T2>
    std::size_t operator()(const std::pair<T1, T2>& pair) const {
        std::hash<T1> hash1;
        std::hash<T2> hash2;
        return hash1(pair.first) ^ (hash2(pair.second) << 1);
    }
};
```
```

Then, when instantiating the `std::unordered_set`, you can specify the custom hash function:

```
```cpp
std::unordered_set<std::pair<int, int>, PairHash> visited;
```
```

With this change, the code should now compile and work as expected.

# CWhy Demo



The problem is that you are attempting to use `std::unordered_set` with `std::pair<int, int>` as the key type. However, the standard library does not provide a hash function specialization for `std::pair` out of the box, so the default constructor of the unordered set is deleted.

To resolve this, you'll need to provide a custom hash function for `std::pair<int, int>`. Here's an example of how you can define one:

```
```cpp
struct PairHash {
    template <typename T1, typename T2>
    std::size_t operator()(const std::pair<T1, T2>& pair) const {
        std::hash<T1> hash1;
        std::hash<T2> hash2;
        return hash1(pair.first) ^ (hash2(pair.second) << 1);
    }
};
```
```

Then, when instantiating the `std::unordered_set`, you can specify the custom hash function:

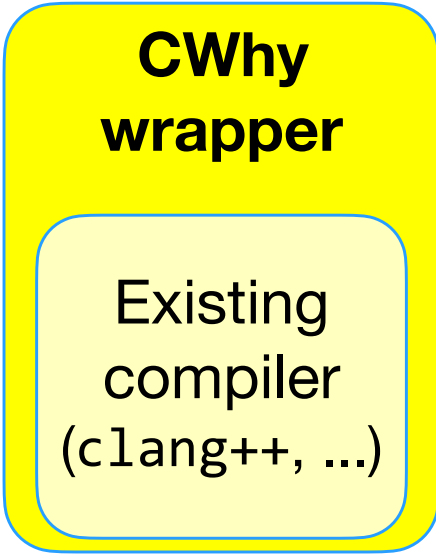
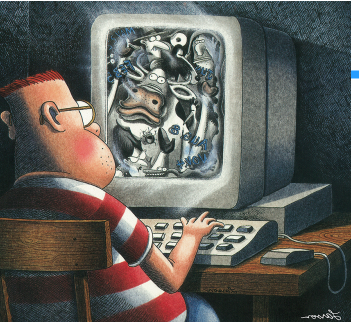
```
```cpp
std::unordered_set<std::pair<int, int>, PairHash> visited;
```
```

With this change, the code should now compile and work as expected.

# CWhy: Evolving a New Compiler



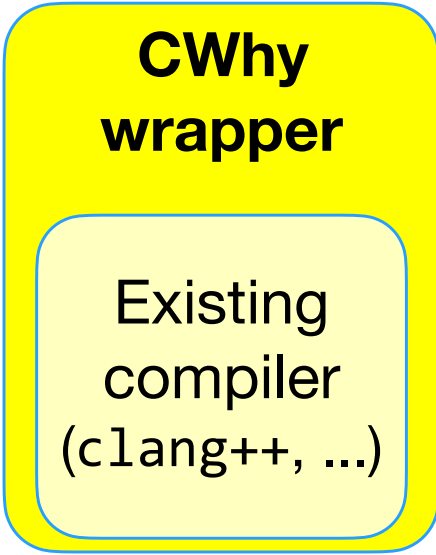
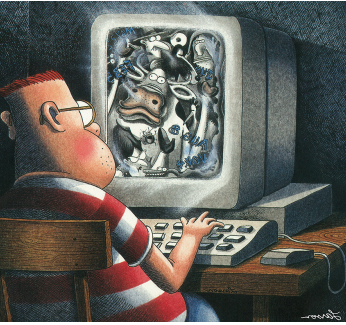
invokes cwhy:  
`% cwhy --- ....`



# CWhy: Evolving a New Compiler



invokes cwhy:  
`% cwhy --- ....`



command line,  
code context,  
error message  
(custom per language)



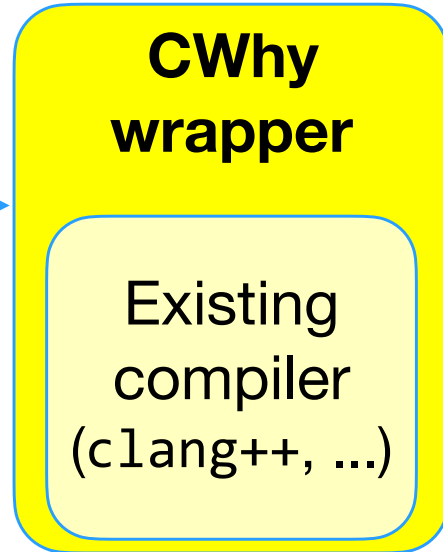
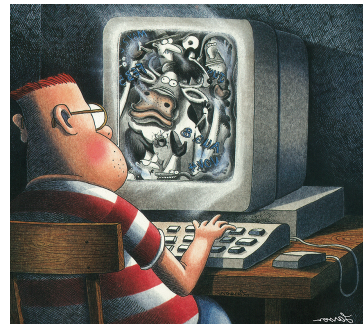
prompt



# CWhy: Evolving a New Compiler



invokes cwhy:  
% **cwhy** --- .....



command line,  
code context,  
error message  
(custom per  
language)



prompt

function  
calls

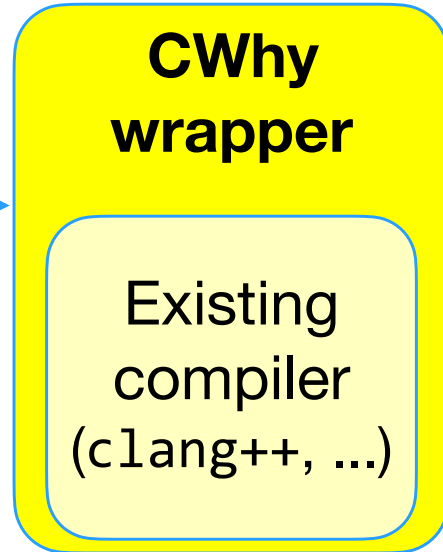
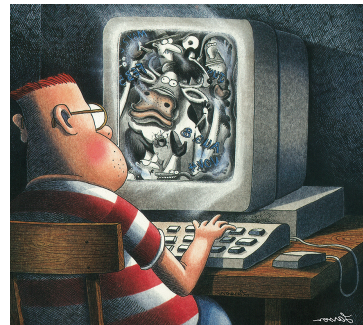


```
get_compile_or_run_command()  
get_code_surrounding(filename, lineno)  
list_directory(directory)
```

# CWhy: Evolving a New Compiler



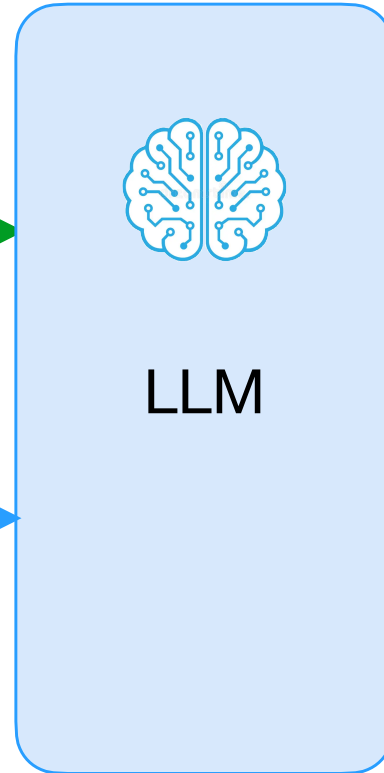
invokes cwhy:  
% **cwhy** --- .....



command line,  
code context,  
error message  
(custom per  
language)



prompt



function  
calls

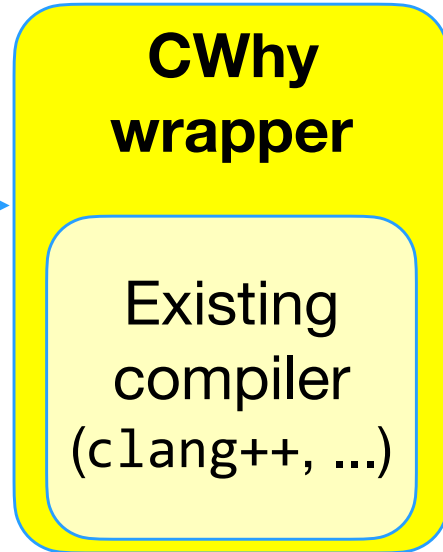
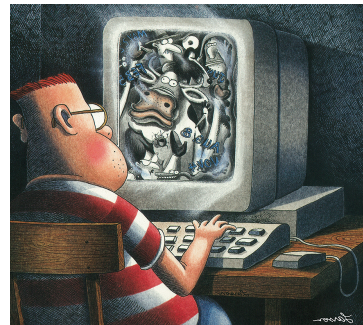


```
get_compile_or_run_command()  
get_code_surrounding(filename, lineno)  
list_directory(directory)  
find_definition(filename, lineno, symbol)
```

# CWhy: Evolving a New Compiler



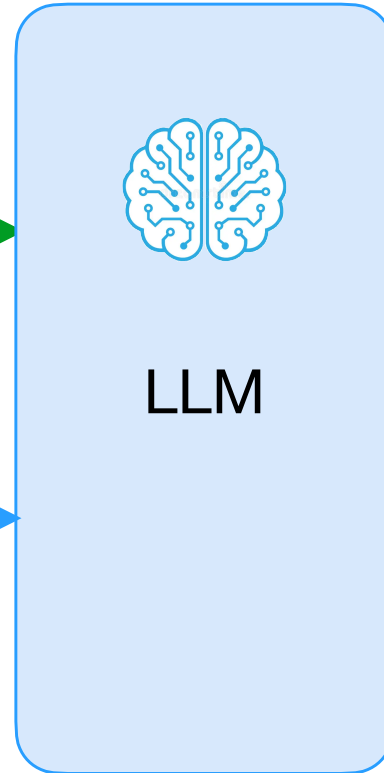
invokes cwhy:  
% **cwhy** --- .....



command line,  
code context,  
error message  
(custom per  
language)



prompt

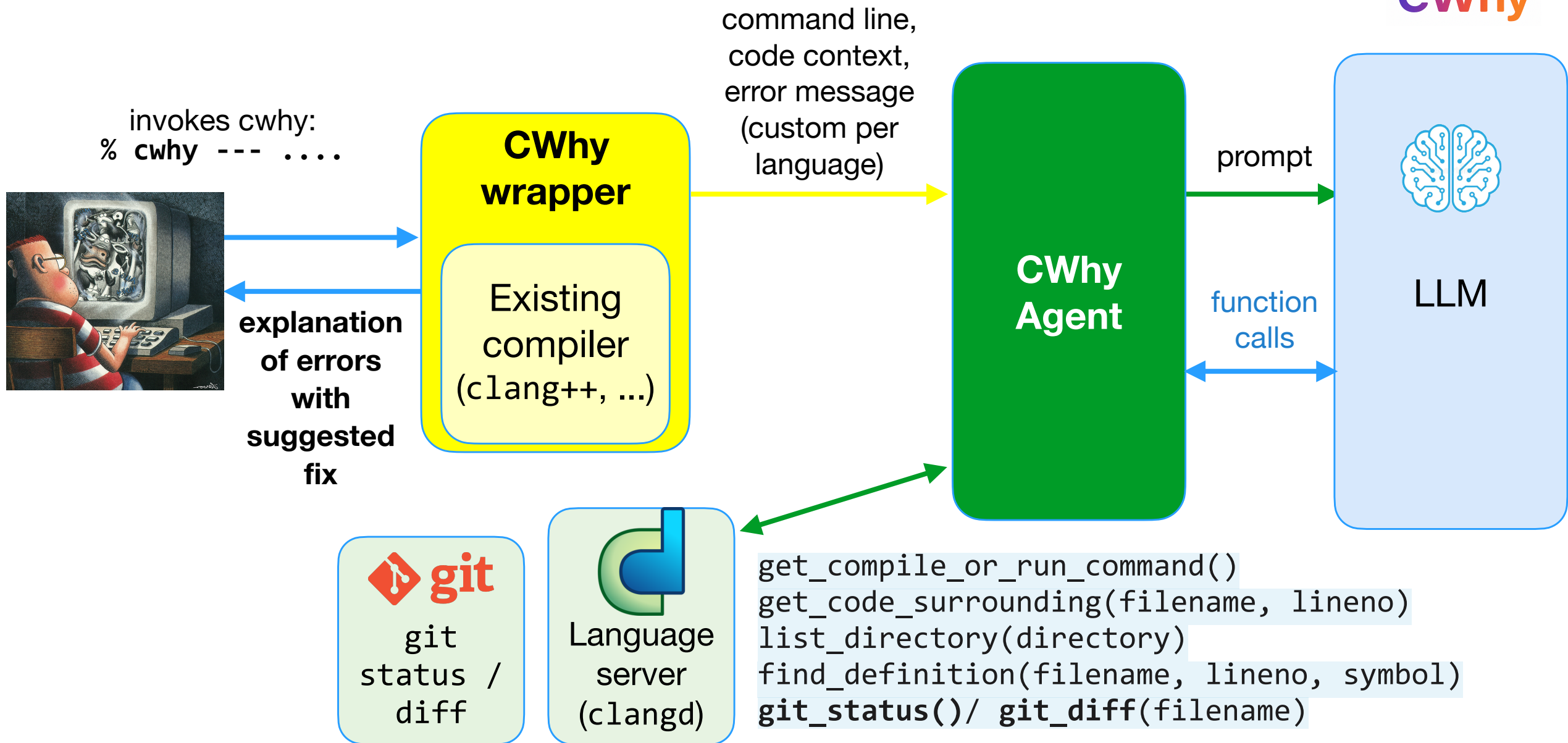


function  
calls



```
get_compile_or_run_command()  
get_code_surrounding(filename, lineno)  
list_directory(directory)  
find_definition(filename, lineno, symbol)  
git_status()/ git_diff(filename)
```

# CWhy: Evolving a New Compiler





30 MDT

[Expressive Compile-time Parsers](#)

[Give Feedback](#)

[Alon Wolf](#)

rsday, Octob

30 MDT

Add to My Schedule

[Link](#)

**Expressive Compile-time Parsers**

Modern C++ has many language features and that can be leveraged to design an expressive API or a domain-specific language. However, these features are limited by the syntax rules of C++. To overcome this limitation, we will discuss the use of compile-time parsers to write more expressive code with zero overhead.

We will analyze open-source compile-time parsing libraries from C++11 to C++23 and compare their APIs in terms of flexibility and expressiveness. Additionally, we will delve into the techniques used in the implementation of these libraries to handle compile-time parsing under the constraints of compile-time execution.

Finally, we will highlight how these libraries can be used to

**Speakers**



**Alon Wolf**

Software Engineer, Medtronic



```
C++ source #1 X
A v + v 🔍 🐞
C++
1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string
5     using namespace ctre::literals;
6     if (auto m = ctre::match<"^[a--z]+([0-9]+)$">(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
12
```



<https://gcc.godbolt.org/z/a5Kavxjab>



```
C++ source #1 x
A ▾ 🔒 + ▾ 🐞 🐞
C++ ▾
1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string
5     using namespace ctre::literals;
6     if (auto m = ctre::match<"^[-z]+([0-9]+)$">(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
12
```



<https://gcc.godbolt.org/z/a5Kavxjab>

```
x86-64 gcc (trunk) (Editor #1) x
x86-64 gcc (trunk) ▾
-std=c++2a -O3 -g0
A ▾ ⚙ ▾ 🔍 🛠 + ▾
1 <Compilation failed:
2
3 # For more informati
Output (0/102) x86-64 gcc
(trunk) i - cached 📄
Compiler License
```

```
Output of x86-64 gcc (trunk) (Compiler #1) x
A ▾ ☐ Wrap lines ☰ Select all
In file included from /opt/compiler-explorer/libs/ctre/main/include/ctre/literals.hpp:7,
    from /opt/compiler-explorer/libs/ctre/main/include/ctre.hpp:4,
    from <source>:1:
/opt/compiler-explorer/libs/ctre/main/include/ctre/wrapper.hpp: In instantiation of 'struct ctri
/opt/compiler-explorer/libs/ctre/main/include/ctre/wrapper.hpp:306:100: required from 'const
<source>:6:24: required from here
/opt/compiler-explorer/libs/ctre/main/include/ctre/wrapper.hpp:299:42: error: invalid use of ir
299 |         static_assert(result::is_correct && problem_at_position<n>(), "Regular Express:
|
/opt/compiler-explorer/libs/ctre/main/include/ctre/wrapper.hpp:277:26: note: declaration of 's'
277 | template <size_t> struct problem_at_position; // do not define!
|
<source>: In function 'std::optional<std::basic_string_view<char> > extract_number(std::string
<source>:7:24: error: no matching function for call to 'ctre::regex_results<const char*>::get<
7 |         return m.get<1>().to_view();
|
In file included from /opt/compiler-explorer/libs/ctre/main/include/ctre/evaluation.hpp:9,
    from /opt/compiler-explorer/libs/ctre/main/include/ctre/literals.hpp:6:
/opt/compiler-explorer/libs/ctre/main/include/ctre/return_type.hpp:305:136: note: candidate: '1
305 |         template <size_t Id, typename = std::enable_if_t<decltype(_captures)::template
|
/opt/compiler-explorer/libs/ctre/main/include/ctre/return_type.hpp:305:136: note: template a
In file included from /opt/compiler-explorer/gcc-trunk-20231002/include/c++/14.0.0/bits/stl_pa
    from /opt/compiler-explorer/gcc-trunk-20231002/include/c++/14.0.0/utility:69,
    from /opt/compiler-explorer/libs/ctre/main/include/ctll/fixed_string.hpp:4,
    from /opt/compiler-explorer/libs/ctre/main/include/ctll/parser.hpp:4,
    from /opt/compiler-explorer/libs/ctre/main/include/ctll.hpp:4,
    from /opt/compiler-explorer/libs/ctre/main/include/ctre/literals.hpp:4:
/opt/compiler-explorer/gcc-trunk-20231002/include/c++/14.0.0/type_traits: In substitution of '1
/opt/compiler-explorer/libs/ctre/main/include/ctre/return_type.hpp:305:23: required from here
/opt/compiler-explorer/gcc-trunk-20231002/include/c++/14.0.0/type_traits:2623:11: error: no typ
2623 |         using enable_if_t = typename enable_if<Cond, _Tp>::type;
|
/opt/compiler-explorer/libs/ctre/main/include/ctre/return_type.hpp:308:142: note: candidate: '1
308 |         template <typename Name, typename = std::enable_if_t<decltype(_captures)::temp
|
/opt/compiler-explorer/libs/ctre/main/include/ctre/return_type.hpp:308:142: note: template a
<source>:7:24: error: type/value mismatch at argument 1 in template parameter list for 'templ
7 |         return m.get<1>().to_view();
|
<source>:7:24: note: expected a type, got '1'
/opt/compiler-explorer/libs/ctre/main/include/ctre/return_type.hpp:312:152: note: candidate: '1
312 |         template <ctll::fixed_string Name, typename = std::enable_if_t<decltype(_captu
|
```



```

1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string s) {
5     using namespace ctre::literals;
6     if (auto m = ctre::match<"^[a--z]+([0-9]+)$">(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
12

```

```

=====
CWhy
=====
Your regular expression `^[a--z]+([0-9]+)$` is invalid.

Your intention here appears to be to match any string which starts
with one or more lowercase letters and ends with one or more digits.
However, your character class range `a--z` seems to be incorrectly
typed. In regular expressions, a single `--` is used to represent a
range, so this should be `a-z` instead:

So the corrected regular expression should look like this:
```cpp
"^[a-z]+([0-9]+)$"
```

The corrected code should then be:

```cpp
1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string_view s) noexcept {
5     using namespace ctre::literals;
6     if (auto m = ctre::match<"^[a-z]+([0-9]+)$">(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
```

When your original code tried to compile, the compiler tried to
compile an instantiation of `problem_at_position<4>`, in other words
it was trying to tell you there was a problem at the 4th character in
your regular expression, which is where your `--` is located. Once you
fix your regular expression to `^[a-z]+([0-9]+)$` the code should
compile without issues.

This kind of messages can help you debug your regular expressions in
the future by helping you locate problematic parts of your regular
expression. So when you see `problem_at_position<n>`, you can look at
the nth character in your regular expression for potential issues.

(Total cost: approximately $0.06 USD.)
=====

```

```

os/ctre/main/include/ctre/literals.hpp:7,
os/ctre/main/include/ctre.hpp:4,
ctre/wrapper.hpp: In instantiation of 'struct ctre::
ctre/wrapper.hpp:306:100: required from 'const
ctre/wrapper.hpp:299:42: error: invalid use of in
t && problem_at_position<n>()', "Regular Express:
ctre/wrapper.hpp:277:26: note: declaration of 'st
ition; // do not define!
_string_view<char> > extract_number(std::string
call to 'ctre::regex_results<const char*>::get<
os/ctre/main/include/ctre/evaluation.hpp:9,
os/ctre/main/include/ctre/literals.hpp:6:
ctre/return_type.hpp:305:136: note: candidate: '
std::enable_if_t<decltype(_captures)::template
ctre/return_type.hpp:305:136: note: template ar
c-trunk-20231002/include/c++/14.0.0/bits/stl_pa
c-trunk-20231002/include/c++/14.0.0/utility:69,
os/ctre/main/include/ctll/fixed_string.hpp:4,
os/ctre/main/include/ctll/parser.hpp:4,
os/ctre/main/include/ctll.hpp:4,
os/ctre/main/include/ctre/literals.hpp:4:
de/c++/14.0.0/type_traits: In substitution of '
ctre/return_type.hpp:305:23: required from here
de/c++/14.0.0/type_traits:2623:11: error: no ty
if<_Cond, _Tp>::type;
ctre/return_type.hpp:308:142: note: candidate: '
ne = std::enable_if_t<decltype(_captures)::temp
ctre/return_type.hpp:308:142: note: template ar
argument 1 in template parameter list for 'templat
ctre/return_type.hpp:312:152: note: candidate: '
ne, typename = std::enable_if_t<decltype(_captur

```



```

1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string s) noexcept {
5     using namespace ctre::literals;
6     if (auto m = ctre::match<"^[a--z]+([0-9]+)$">(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
12

```

```

=====
CWhy
=====
Your regular expression "^[a--z]+([0-9]+)$" is invalid.

Your intention here appears to be to match any string which starts
with one or more lowercase letters and ends with one or more digits.
However, your character class range `a--z` seems to be incorrectly
typed. In regular expressions, a single `-` is used to represent a
range, so this should be `a-z` instead:

So the corrected regular expression should look like this:

```cpp
"^[a-z]+([0-9]+)$"
```

```

```

The corrected code should then be:

```cpp
1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string_view s) noexcept {
5     using namespace ctre::literals;
6     if (auto m = ctre::match<"^[a-z]+([0-9]+)$">(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
```

```

When your original code tried to compile, the compiler tried to compile an instantiation of ``problem_at_position<4>``, in other words it was trying to tell you there was a problem at the 4th character in your regular expression, which is where your ``--`` is located. Once you fix your regular expression to `"^[a-z]+([0-9]+)$"` the code should compile without issues.

This kind of messages can help you debug your regular expressions in the future by helping you locate problematic parts of your regular expression. So when you see ``problem_at_position<n>``, you can look at the nth character in your regular expression for potential issues.

**(Total cost: approximately \$0.06 USD.)**

```

.../ctre/main/include/ctre/literals.hpp:7,
.../ctre/main/include/ctre.hpp:4,
.../ctre/wrapper.hpp: In instantiation of 'struct ctre::
.../ctre/wrapper.hpp:306:100: required from 'conste
.../ctre/wrapper.hpp:299:42: error: invalid use of ir
... t && problem_at_position<n>()', "Regular Express:
...
.../ctre/wrapper.hpp:277:26: note: declaration of 'st
... ition; // do not define!
...
..._string_view<char> > extract_number(std::string
... call to 'ctre::regex_results<const char*>::get<
...
.../ctre/main/include/ctre/evaluation.hpp:9,
.../ctre/main/include/ctre/literals.hpp:6:
.../ctre/return_type.hpp:305:136: note: candidate: '
... std::enable_if_t<decltype(_captures)::template
...
.../ctre/return_type.hpp:305:136: note: template ar
... c-trunk-20231002/include/c++/14.0.0/bits/stl_pa
... c-trunk-20231002/include/c++/14.0.0/utility:69,
.../ctre/main/include/ctll/fixed_string.hpp:4,
.../ctre/main/include/ctll/parser.hpp:4,
.../ctre/main/include/ctll.hpp:4,
.../ctre/main/include/ctre/literals.hpp:4:
.../ctre/c++/14.0.0/type_traits: In substitution of '
.../ctre/return_type.hpp:305:23: required from here
.../ctre/c++/14.0.0/type_traits:2623:11: error: no ty
... if<_Cond, _Tp>::type;
.../ctre/return_type.hpp:308:142: note: candidate: '
... ne = std::enable_if_t<decltype(_captures)::temp
...
.../ctre/return_type.hpp:308:142: note: template ar
... gument 1 in template parameter list for 'templat
...
.../ctre/return_type.hpp:312:152: note: candidate: '
... ne, typename = std::enable_if_t<decltype(_captur

```







```
C++ source #1 x86-64 gcc (trunk) (Editor #1) Output of x86-64 gcc (trunk) (Compiler #1)
A C++ x86-64 gcc (trunk) Wrap lines Select all
1 #include <ctre.hpp>
2 #include <optional>
3
4 std::optional<std::string_view> extract_number(std::string s) {
5     using namespace ctre::literals;
6     if (auto m = ctre::match<`^[a--z]+([0-9]+)$`>(s)) {
7         return m.get<1>().to_view();
8     } else {
9         return std::nullopt;
10    }
11 }
12
```

```
=====  
CWhy  
=====  
Your regular expression `^[a--z]+([0-9]+)$` is invalid.  
Your intention here appears to be to match any string which starts  
with one or more lowercase letters and ends with one or more digits.  
However, your character class range `a--z` seems to be incorrectly  
typed. In regular expressions, a single `--` is used to represent a  
range, so this should be `a-z` instead:  
So the corrected regular expression should look like this:
```

```
os/ctre/main/include/ctre/literals.hpp:7,  
os/ctre/main/include/ctre.hpp:4,  
ctre/wrapper.hpp: In instantiation of 'struct ctri  
ctre/wrapper.hpp:306:100: required from 'const  
ctre/wrapper.hpp:299:42: error: invalid use of in  
t && problem_at_position<n>()', "Regular Express:  
ctre/wrapper.hpp:277:26: note: declaration of 'st  
ition: // do not define!  
extract_number(std::string  
_results<const char*>::get<  
/ctre/evaluation.hpp:9,  
/ctre/literals.hpp:6:  
305:136: note: candidate: 't  
ctype(_captures)::template  
305:136: note: template ar  
lude/c++/14.0.0/bits/stl_pa  
lude/c++/14.0.0/utility:69,  
/ctll/fixed_string.hpp:4,  
/ctll/parser.hpp:4,  
/ctll.hpp:4,  
/ctre/literals.hpp:4:  
raits: In substitution of 't  
tre/return_type.hpp:305:23: required from here  
le/c++/14.0.0/type_traits:2623:11: error: no typ  
if<_Cond, _Tp>::type;  
tre/return_type.hpp:308:142: note: candidate: 't  
ne = std::enable_if_t<decltype(_captures)::temp  
tre/return_type.hpp:308:142: note: template ar  
argument 1 in template parameter list for 'templat  
tre/return_type.hpp:312:152: note: candidate: 't  
ne, typename = std::enable_if_t<decltype(_captu
```

 **Alon Wolf** Wed, Oct 4, 8:13 PM (11 hours ago) 

to me ▾

Hi,  
Yes this is correct.  
It provided an amazing answer, way better than I expected.

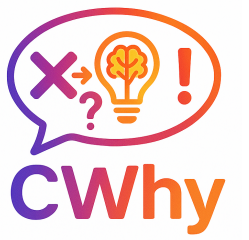
When your original code tried to compile, the compiler tried to compile an instantiation of `problem\_at\_position<4>`, in other words it was trying to tell you there was a problem at the 4th character in your regular expression, which is where your `--` is located. Once you fix your regular expression to `^[a-z]+([0-9]+)\$` the code should compile without issues.

This kind of messages can help you debug your regular expressions in the future by helping you locate problematic parts of your regular expression. So when you see `problem\_at\_position<n>`, you can look at the nth character in your regular expression for potential issues.

(Total cost: approximately \$0.06 USD.)



# CWhy: Evolving a New Compiler



evolve

compiler *that*  
*explains error*  
*messages &*  
*fixes them*

# CWhy: Evolving a New Compiler



evolve

compiler *that explains error messages & fixes them*



exploit niche

source +  
*language server*  
(values & types...)  
+ git + *domain knowledge*

# CWhy: Evolving a New Compiler



evolve

compiler *that explains error messages & fixes them*



exploit niche

source +  
*language server*  
(values & types...)  
+ git + *domain knowledge*



ensure fitness

patch code, re-compile, verify  
fix eliminates  
compiler error

# CoverUp: Evolving a New Coverage Tool



# CoverUp: Evolving a New Coverage Tool



by [Juan Altmayer Pizzorno](#) and [Emery Berger](#) at UMass Amherst's [PLASMA lab](#).

pypi v0.6.1

python 3.10 | 3.11 | 3.12 | 3.13

DOI 10.5281/zenodo.15187805

downloads 14k

downloads/month 1k

## About CoverUp

---

CoverUp automatically generates tests that ensure that more of your code is tested (that is, it increases its [code coverage](#)). CoverUp can also create a test suite from scratch if you don't yet have one. The new tests are based on your code, making them useful for [regression testing](#).

**[FSE 2025]**

# CoverUp: Why Coverage Analysis?



```
def foo(x):  
    y = x - 10  
    if y < 0:  
        y = 0  
    return y
```

```
def test_foo():  
    assert 0 == foo(10)  
    assert 10 == foo(20)  
    assert 99990 == foo(100000)
```

```
$ python3 -m coverage run -m pytest x.py  
-----  
platform darwin -- Python 3.9.12, pytest-6.2.5, py-cov-2.11.1  
rootdir: /Users/juan/project/slipcover  
plugins: hypothesis-6.39.3, mock-3.7.0  
collected 1 item  
  
x.py .  
  
-----  
  
$ python3 -m coverage html  
Wrote HTML report to htmlcov/index.html
```

Coverage for x.py: 89%

9 statements 8 run 1 missing 0 excluded

```
1 def foo(x):  
2     y = x - 10  
3     if y < 0:  
4         y = 0  
5     return y  
6
```

Tests ideally "cover" (run) as many lines as possible

# CoverUp: Why Coverage Analysis?



```
def foo(x):  
    y = x - 10  
    if y < 0:  
        y = 0  
    return y  
  
def test_foo_negative():  
    assert 0 == foo(0)  
    assert 0 == foo(-1)
```

```
$ python3 -m coverage run --branch -m pytest x.py  
===== test session starts =====  
platform darwin -- Python 3.9.12, pytest-7.1.2, pluggy-1.0.0  
rootdir: /Users/juan/project/slipcover  
plugins: hypothesis-6.39.3, mock-3.7.0, repeat-0.9.1, doctestplus-0.12.0, arraydiff-0.5.0  
collected 1 item  
  
x.py . [100%]  
  
===== 1 passed in 0.01s =====
```

Improving / more tests can "increase coverage"



# CoverUp: Why Coverage Analysis?



```
def foo(x):  
    y = x - 10  
    if y < 0:  
        y = 0  
    return y
```

```
def test_foo_negative():  
    assert 0 == foo(0)  
    assert 0 == foo(-1)
```

```
$ python3 -m coverage run  
-----  
platform darwin -- Python  
rootdir: /Users/juan/proj  
plugins: hypothesis-6.39.  
collected 1 item  
  
x.py .  
-----  
  
$ python3 -m coverage htm  
Wrote HTML report to htm
```

Coverage for **x.py**: 90%

8 statements   8 run   0 missing   0 excluded   1 partial

```
1 def foo(x):  
2     y = x - 10  
3     if y < 0: 3 →/5  
4         y = 0  
5     return y  
6
```

...but not necessarily all *branches*

# CoverUp: Why Coverage Analysis?



Cog coverage: 38.58%

Files

Functions

Classes


coverage.py v7.8.0, created at 2025-03-30 15:44 -0400

| File ▲                    | statements  | missing    | excluded | branches   | partial   | coverage      |
|---------------------------|-------------|------------|----------|------------|-----------|---------------|
| cogapp/__init__.py        | 1           | 0          | 0        | 0          | 0         | 100.00%       |
| cogapp/__main__.py        | 3           | 3          | 0        | 0          | 0         | 0.00%         |
| cogapp/cogapp.py          | 483         | 228        | 1        | 190        | 27        | 46.66%        |
| cogapp/makefiles.py       | 22          | 18         | 0        | 14         | 0         | 11.11%        |
| cogapp/test_cogapp.py     | 854         | 598        | 2        | 20         | 1         | 29.63%        |
| cogapp/test_makefiles.py  | 68          | 51         | 0        | 6          | 0         | 22.97%        |
| cogapp/test_whiteutils.py | 68          | 50         | 0        | 0          | 0         | 26.47%        |
| cogapp/utils.py           | 37          | 8          | 0        | 6          | 2         | 76.74%        |
| cogapp/whiteutils.py      | 44          | 5          | 0        | 32         | 4         | 88.16%        |
| <b>Total</b>              | <b>1580</b> | <b>961</b> | <b>3</b> | <b>268</b> | <b>34</b> | <b>38.58%</b> |

**Real test suites often fall far short of 100% line / branch coverage...why?**

# CoverUp: Testing Code is Hard



←  r/csharp · 3 yr. ago  
Malory-in-midnight



## Testing code is hard

Discussion

Generally writing tests feel like a barrier to start when I write code, specially when I'm going into someone else's to fix a bug where running the application is easier than writing a test. Spending time to set up a test through mocking and making sure all the dependencies are right really take me out of the flow, but once it's set up, it's easy to get going. It's just that first hurdle.

Do y'all run into issues with testing? How do you avoid them?

↑ 19 ↓


💬 35



↪ Share

# CoverUp: Testing Code is Hard



←  r/csharp · 3 yr. ago  
Malory-in-midnight

## Testing code is hard


Discussion

Generally writing tests feel like a barrier someone else's to fix a bug where run set up a test through mocking and ma flow, but once it's set up, it's easy to g Do y'all run into issues with testing? H

↑ 19 ↓

💬 35



←  r/cscareerquestions · 7 yr. ago  
[deleted]

## Why is writing SW tests so hard?

Seriously, it feels like writing code that tests the actual code is the tough part of this job. And with everyone's favorite hobby being Leetcode Medium, this seems to be a chronically underrated / underappreciated part of SW development.

Can anyone else actually attest to being in a team with a strong testing culture? What differences did you notice between teams with a weak testing culture vs one with a strong culture? More velocity/ agility?

I have half a feeling that engineers don't write tests to protect their jobs lol.

↑ 7 ↓

💬 13



↪ Share



# CoverUp: Testing Code is Hard



←  **r/csharp** · 3 yr. ago  
Malory-in-midnight

## Testing code is hard

Discussion

Generally writing tests feel like a barrier to start when I write code, specially when I'm going into someone else's to fix a bug where running the code is a pain. Spending time to set up a test through mocking and making sure all the dependencies are right really take me out of the flow, but once it's set up, it's easy to get going. Seriously, it feels like writing code that tests the actual code is the tough part of this job. And with everyone's favorite hobby being Leetcode Medium, this seems to be a chronically underrated / underappreciated part of SW development.

↑ 19 ↓

## Testing is hard

Help

I am jun dev, I write unit tests and for basic scenarios there is not many problems. But when it comes to test more complex objects or objects with complex properties my brain stops working. Many online tutorials uses moq but it is said it was mining emails from codes, so I stop using. How can I improve my testing, any recommendations?

↑ 7 ↓

13



Share

←  **r/Frontend** · 4 yr. ago  
Chris\_Cross\_Crash

## Writing tests is killing my creativity and motivation

←  **r/cscareerquestions** · 7 yr. ago  
[deleted]

## Why is writing SW tests so hard?

Do y'all run into issues with testing? How do you avoid them?

←  **r/csharp** · 1 yr. ago  
KriptoVolkano

Can anyone else actually attest to being in a team with a strong testing culture? What differences did you notice between teams with a weak testing culture vs one with a strong culture? More velocity/agility?

I have half a feeling that engineers don't write tests to protect their jobs lol.

# CoverUp: Testing Code is Hard



←  **r/csharp** · 3 yr. ago  
Malory-in-midnight

## Testing code is hard

Discussion

Generally writing tests feel like a k someone else's to fix a bug where set up a test through mocking and flow, but once it's set up, it's easy

Do y'all run into issues with testing

↑ 19 ↓

←  **r/csha**  
Kripto


## Testing is h

Help

I am jun dev, I writ test more complex tutorials uses moc testing, any recor

←  **r/ Frontend** · 4 yr. ago  
Chris\_Cross\_Crash

## Writing tests is killing my creativity and motivation

←  **r/ ExperiencedDevs** · 5 yr. ago  
anuaps Senior Software Engineer

## Writing unit tests is sucking the joy out of programming

How do you make it bearable? Writing TDD doesn't work for me as I keep changing structure of classes till I finish implementation.

When I finally finish my code implementation and start writing unit tests, I feel extremely bored and I have to force myself to finish it as soon as possible.

I wound not want to skip writing tests and I know the importance of one. Just want to hear others experience of writing unit test. And how do you make it not boring?

↑ 27 ↓

🗨 47



➦ Share

# CoverUp: Evolving a New Coverage Tool



evolve

coverage  
analyzer *that*  
*generates tests*  
*that increase*  
coverage

```
import pytest
from scalene.scalene_analysis import ScaleneAnalysis
from unittest.mock import patch

@pytest.fixture
def cleanup_imports():
    # Fixture to clean up sys.modules after the test
    import sys
    before = set(sys.modules.keys())
    yield
    after = set(sys.modules.keys())
    for extra in after - before:
        del sys.modules[extra]

def test_get_native_imported_modules(cleanup_imports):
    # Mock the is_native method to control which modules are considered native
    with patch.object(ScaleneAnalysis, 'is_native', return_value=True):
        source_code = """
import math
import os
from sys import path
"""
        expected_imports = ['import math', 'import os', 'from sys import path']
        actual_imports = ScaleneAnalysis.get_native_imported_modules(source_code)
        assert set(actual_imports) == set(expected_imports), "The list of native imports does not match"

    with patch.object(ScaleneAnalysis, 'is_native', return_value=False):
        source_code = """
import math
import os
from sys import path
"""
```

# CoverUp: Evolving a New Coverage Tool



You are an expert Python test-driven developer.

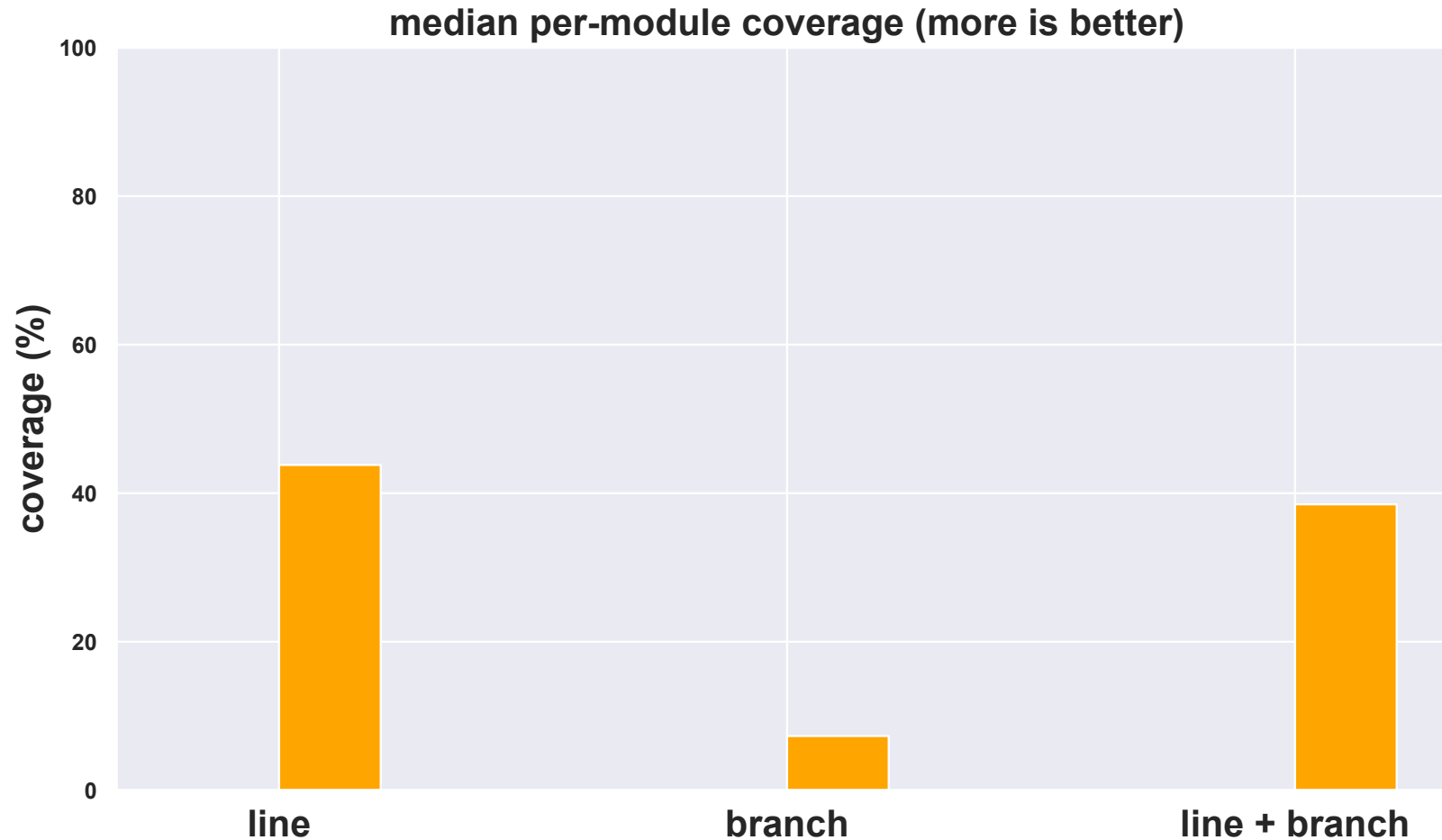
The code below, extracted from tornado/  
simple\_httpclient.py, does not achieve full coverage.

Create new pytest test functions that execute all lines  
and branches, always making sure that each test is correct  
and indeed improves coverage.

[...]

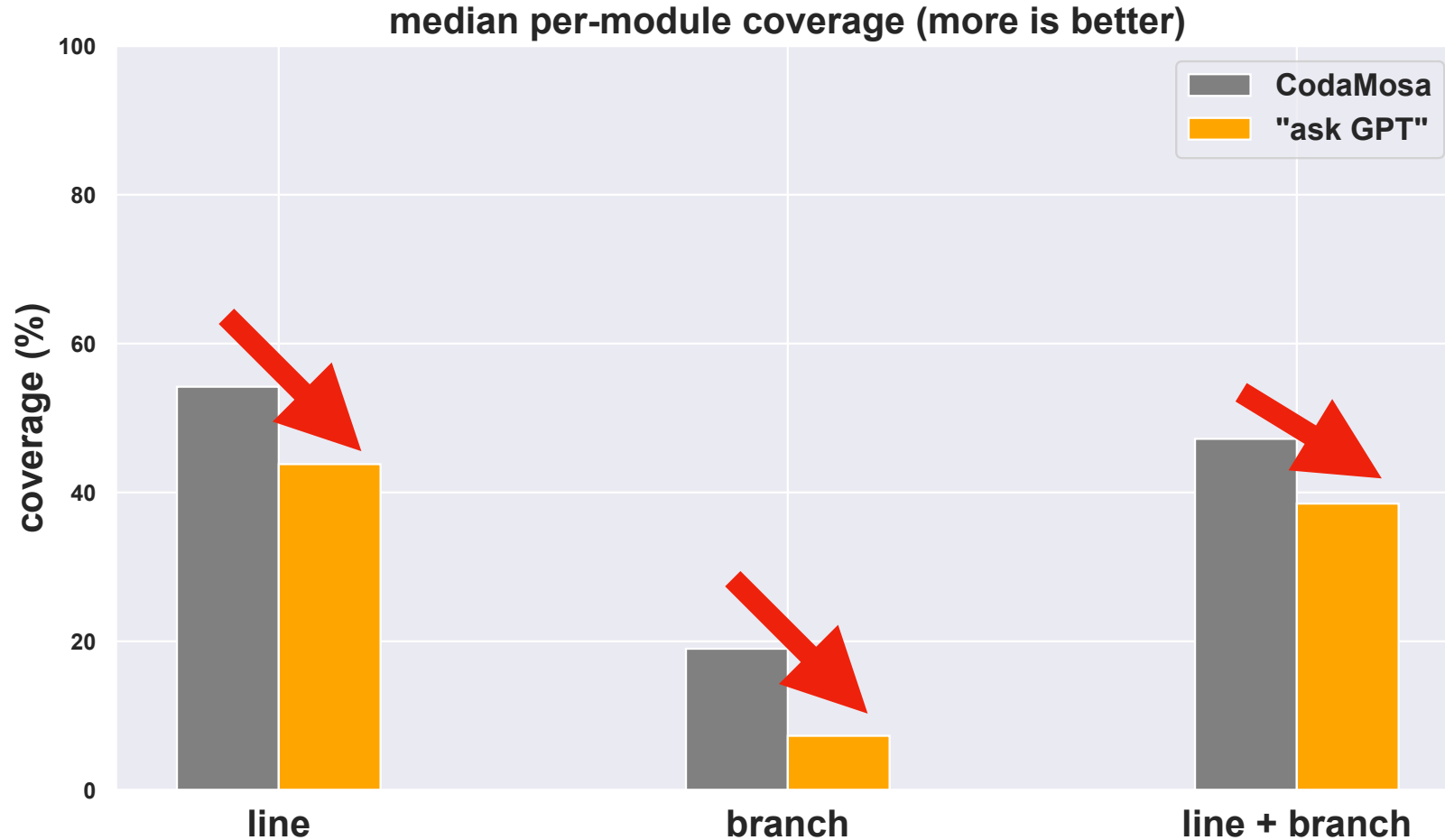
**prompting alone ("ask GPT") does *something*...**

# CoverUp: Evolving a New Coverage Tool



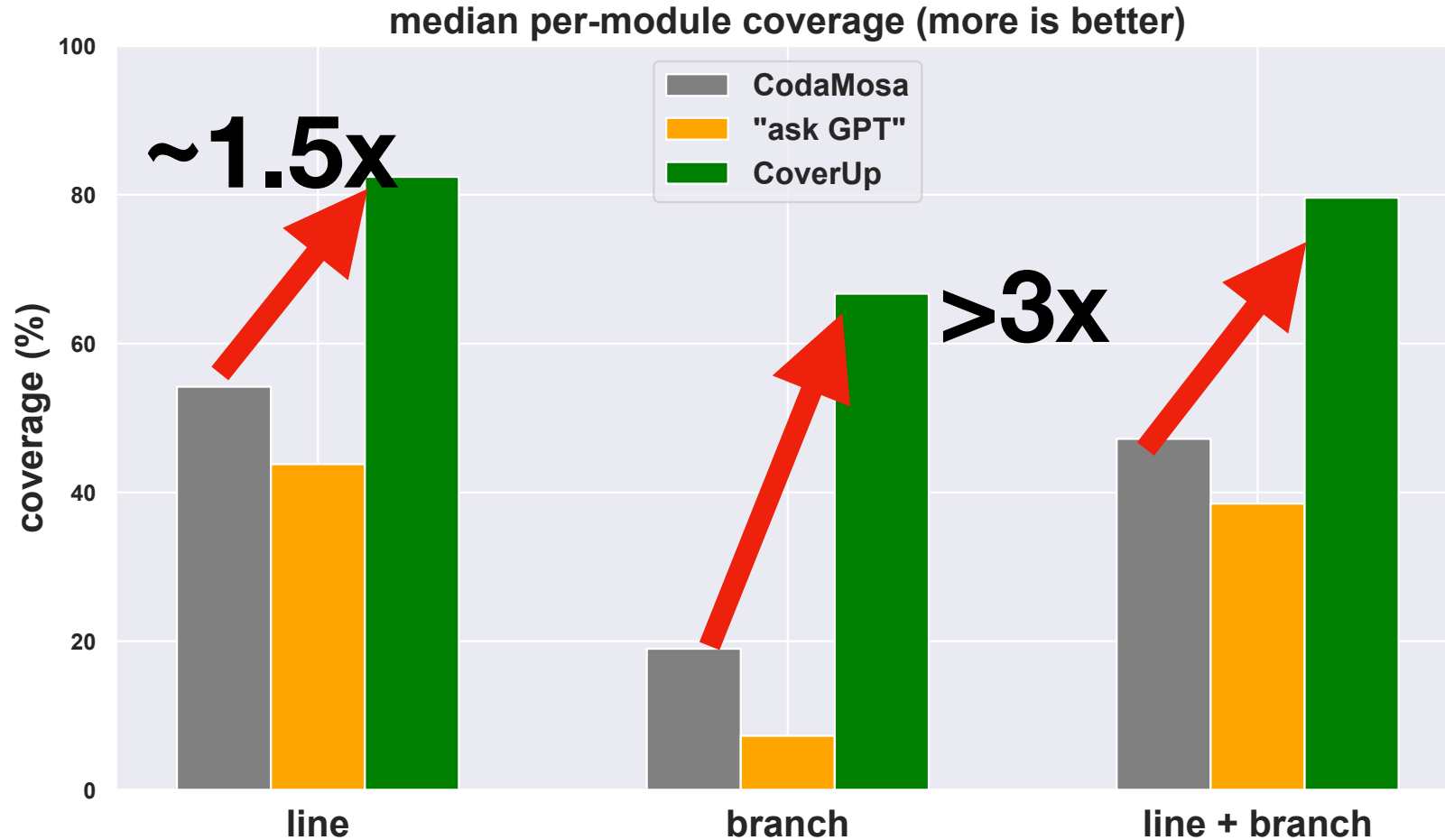
**prompting alone ("ask GPT") does *something*...**

# CoverUp: Evolving a New Coverage Tool



but worse than previous state-of-the-art

# CoverUp: Evolving a New Coverage Tool



CoverUp generated-tests: *much higher coverage*

# CoverUp: Need to Prevent Broken Tests



```
import pytest
from code.funcs import func
from code.models import A
```

LLM assumes **incorrect path**,  
leading to **error**

```
def test_func_x_greater_than_5():
    a = A(x=6)
    result = func(a)
    assert result is True
```

```
def test_func_x_less_than_2():
    a = A(x=1)
    result = func(a)
    assert result is True
```

LLM **incorrectly assumes**  
**x can be set directly**,  
leading to **errors**

```
def test_func_x_between_2_and_5():
    a = A(x=3)
    result = func(a)
    assert result is False
```

# CoverUp: Need to Prevent Broken Tests



```
import pytest
from code.funcs import func
```

```
class A:
    def __init__(self, x):
        self.x = x
```

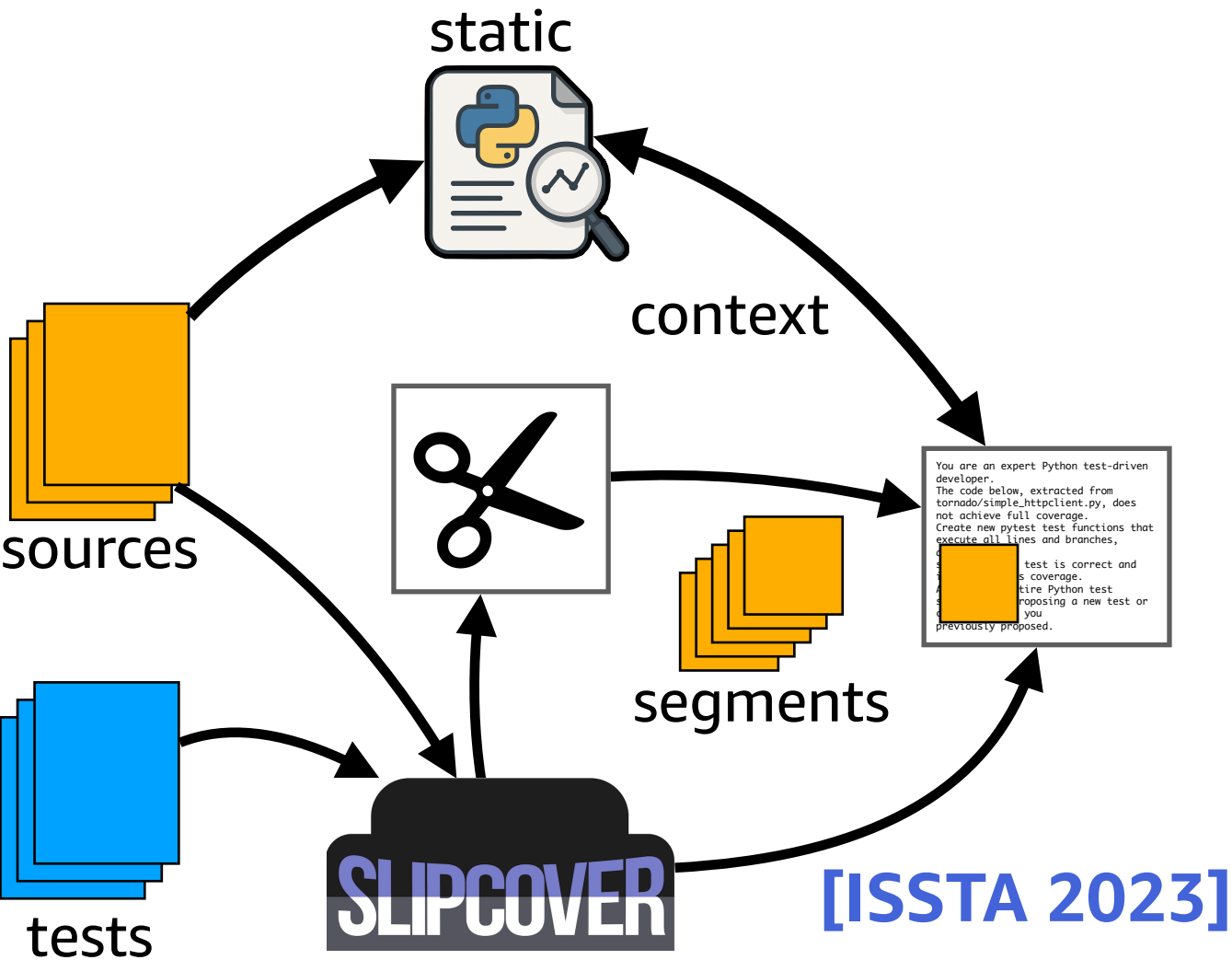
```
def test_func_greater_than_5():
    a = A(6)
    result = func(a)
    assert result is True
```

```
def test_func_less_than_2():
    a = A(1)
    result = func(a)
    assert result is True
```

```
def test_func_between_2_and_5():
    a = A(3)
    result = func(a)
    assert result is False
```

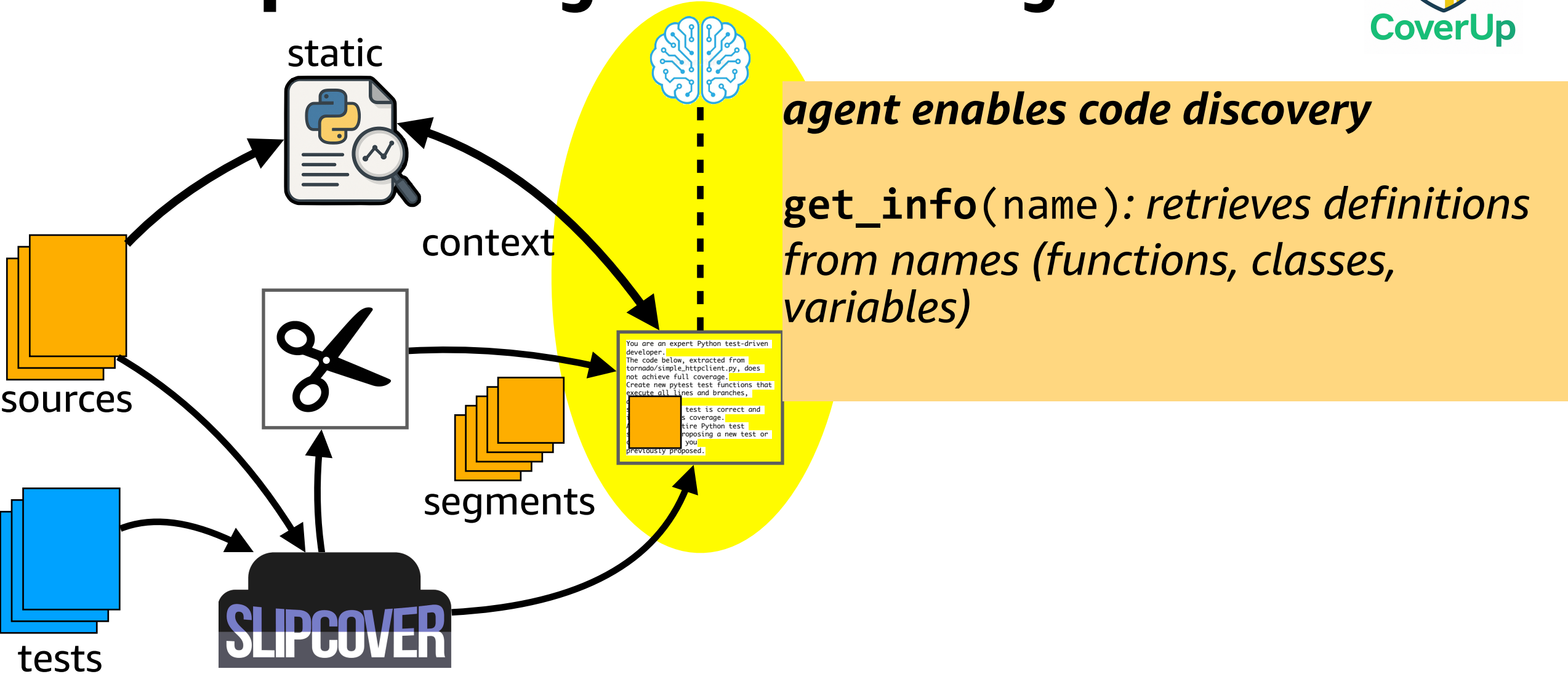
LLM redefines the class under test, producing **useless tests**

# CoverUp: Evolving a New Coverage Tool



*coverage analysis: identifies uncovered segments of code*

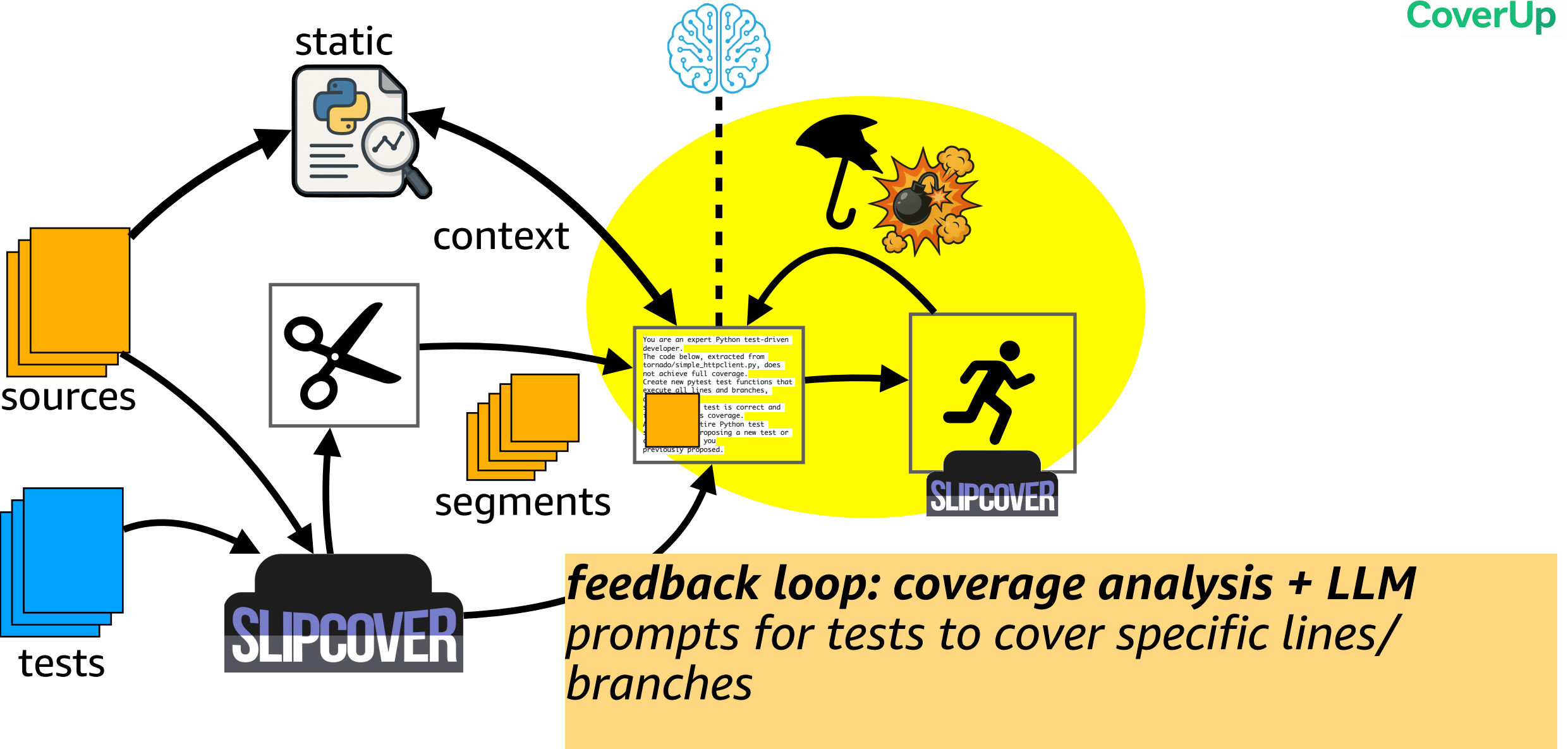
# CoverUp: Evolving a New Coverage Tool



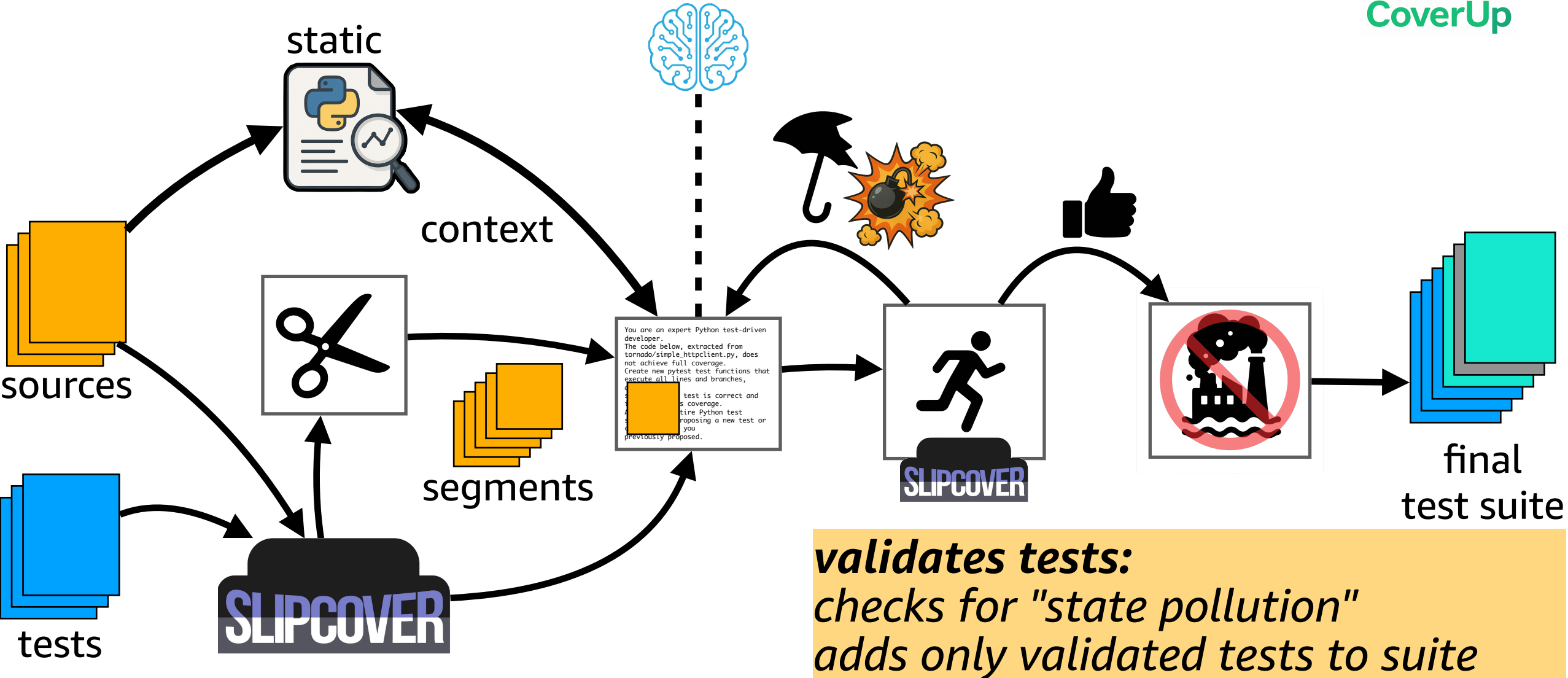
*agent enables code discovery*

*get\_info(name): retrieves definitions from names (functions, classes, variables)*

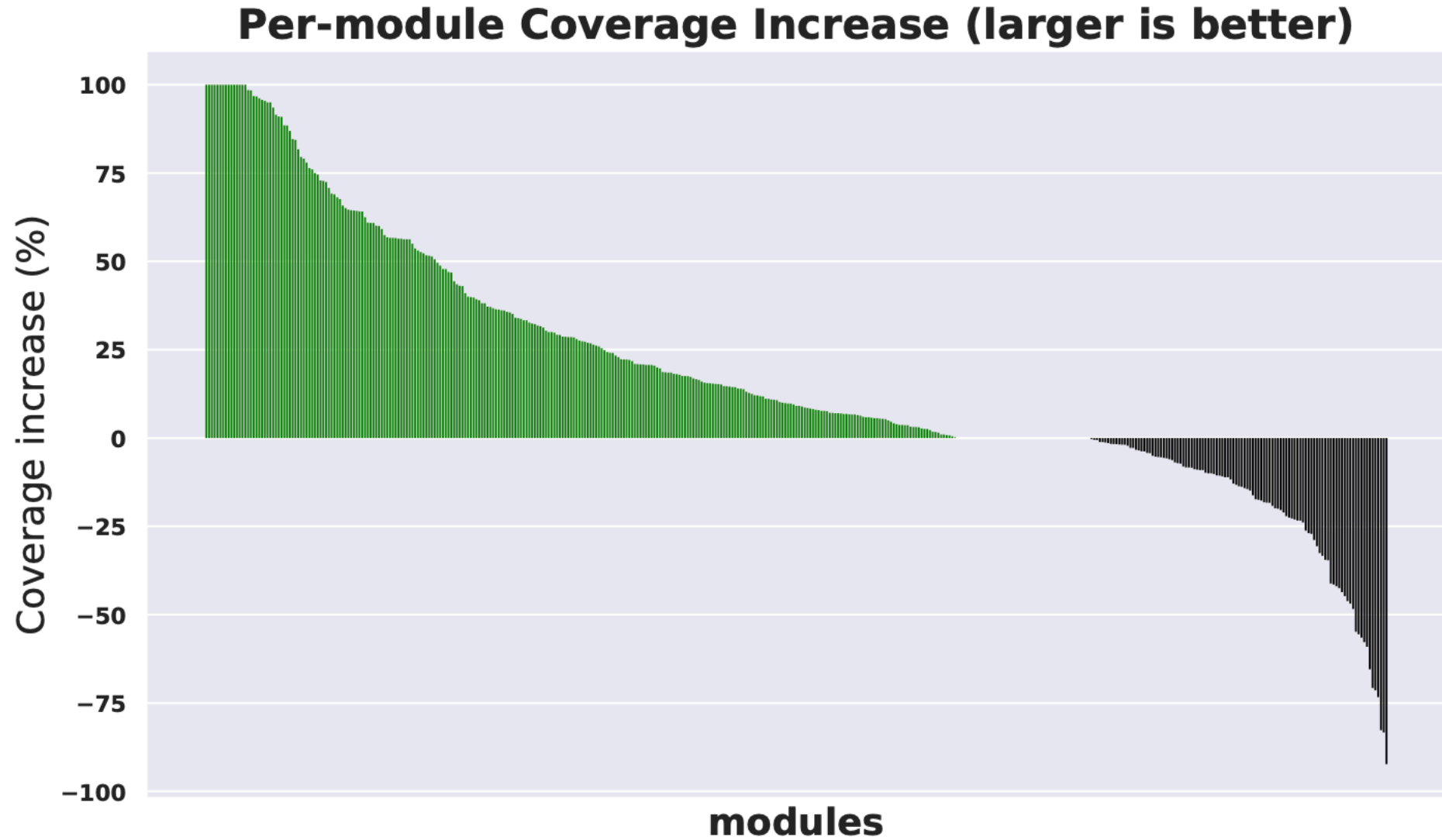
# CoverUp: Evolving a New Coverage Tool



# CoverUp: Evolving a New Coverage Tool



# CoverUp: Evolving a New Coverage Tool



# CoverUp: Evolving a New Coverage Tool



evolve

coverage  
analyzer *that*  
*generates tests*  
*that increase*  
*coverage*

# CoverUp: Evolving a New Coverage Tool



evolve

coverage analyzer *that generates tests that increase coverage*



exploit niche

source + *static & dynamic analysis (esp. coverage information)*

# CoverUp: Evolving a New Coverage Tool



evolve

coverage analyzer *that generates tests that increase coverage*



exploit niche

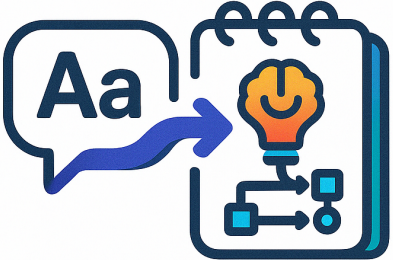
source + *static & dynamic analysis (esp. coverage information)*



ensure fitness

verify tests succeed and increase coverage

# Flowco: Rethinking Notebooks



Flowco

# Flowco: Rethinking Notebooks



Computational notebooks

👍: rapid iteration

👎: difficult to maintain

The screenshot shows a Jupyter Notebook window titled 'geyser.ipynb'. It contains three code cells and their outputs. The first cell reads a CSV file and shows the first three rows of the resulting DataFrame. The second cell creates a scatter plot of eruption duration versus waiting time. The third cell performs KMeans clustering on the data.

```
[2]: df = pd.read_csv('old_faithful.csv')
df.head(3)
```

|   | eruptions | waiting |
|---|-----------|---------|
| 0 | 3.600     | 79      |
| 1 | 1.800     | 54      |
| 2 | 3.333     | 74      |

```
[3]: plt.scatter(df['eruptions'], df['waiting'])
plt.xlabel('Eruption Duration')
plt.ylabel('Waiting Time');
```

The scatter plot shows two distinct clusters of data points. The first cluster is located at lower eruption durations (around 1.5 to 2.5) and lower waiting times (around 45 to 65). The second cluster is located at higher eruption durations (around 3.5 to 5.0) and higher waiting times (around 65 to 90).

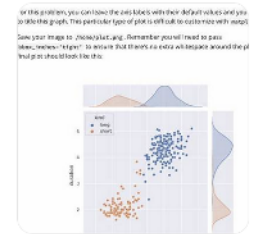
```
[7]: X = df[['eruptions', 'waiting']]
kmeans = KMeans(n_clusters=2)
df['cluster'] = kmeans.fit_predict(X)
```

```
[5]: centers = kmeans.cluster_centers_
if centers[0, 0] < centers[1, 0]:
```

# Flowco: Rethinking Notebooks

Analyze this dataset.

Here's a detailed analysis of the `geyser.csv` dataset from Seaborn (tracking Old Faithful eruptions):



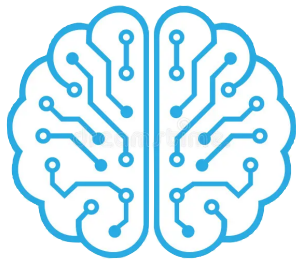
## Dataset Overview

- **Source:** Originally from the "faithful" dataset in R, included in Seaborn's example data [chegg.com +7](#) [github.com](#).
- **Structure:** 2 numerical columns — `eruptions` (eruption duration in minutes) and `waiting` (time until next eruption in minutes).

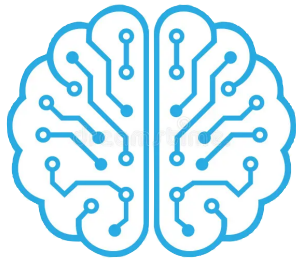
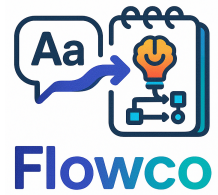
## Descriptive Statistics (approximate)

Based on similar datasets and general knowledge:

| Metric  | Eruptions (min)   | Waiting (min) |
|---------|-------------------|---------------|
| Count   | ~272 observations | ~272          |
| Mean    | ~3.50             | ~70.90        |
| Median  | ~4.00             | ~76.00        |
| Minimum | ~1.60             | ~43.00        |
| Maximum | ~5.10             | ~96.00        |



# Flowco: Rethinking Notebooks



LLMs

👍: lots of code, fast

Here's a detailed analysis of the **geyser** Old Faithful eruptions):

## 🗺️ Dataset Overview

- **Source:** Originally from the "faithfu example data" [chegg.com](#) +7 [github.com](#)
- **Structure:** 2 numerical columns — until next eruption in minutes).

## 📊 Descriptive Statistics (a

based on similar datasets and general k

metric

Count

Mean

Median

Minimum

Maximum

python

Copy Edit

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load dataset
geyser = pd.read_csv("/mnt/data/geyser.csv")

# Basic info and stats
print(geyser.info())
print(geyser.describe())
print(geyser['kind'].value_counts())

# Correlation matrix (numerical columns only)
print(geyser[['duration', 'waiting']].corr())

# Histogram of eruption durations
plt.figure(figsize=(8, 4))
sns.histplot(geyser['duration'], bins=30, kde=True)
plt.title('Distribution of Eruption Durations')
plt.xlabel('Duration (minutes)')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()

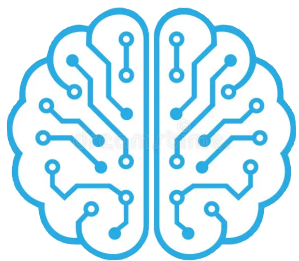
# Histogram of waiting times
plt.figure(figsize=(8, 4))
sns.histplot(geyser['waiting'], bins=30, kde=True)
plt.title('Distribution of Waiting Times')
plt.xlabel('Waiting Time (minutes)')
```

Ask anything

+ Tools



# Flowco: Rethinking Notebooks



Here's a detailed analysis of the **geyser** Old Faithful eruptions):

## 🦖 Dataset Overview

- **Source:** Originally from the "faithful" example data [chegg.com](#) +7 [github.com](#)
- **Structure:** 2 numerical columns — until next eruption in minutes).

## 📊 Descriptive Statistics (a

ised on similar datasets and general k

stribu

unt

Mean

Median

Minimum

Maximum

Outliers

python

Copy Edit

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load dataset
geyser = pd.read_csv("/mnt/data/geyser.csv")

# Basic info and stats
print(geyser.info())
print(geyser.describe())
print(geyser['kind'].value_counts())

# Correlation matrix (numerical columns only)
print(geyser[['duration', 'waiting']].corr())

# Histogram of eruption durations
plt.figure(figsize=(8, 4))
sns.histplot(geyser['duration'], bins=30, kde=True)
plt.title('Distribution of Eruption Durations')
plt.xlabel('Duration (minutes)')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()

# Histogram of waiting times
plt.figure(figsize=(8, 4))
sns.histplot(geyser['waiting'], bins=30, kde=True)
plt.title('Distribution of Waiting Times')
plt.xlabel('Waiting Time (minutes)')
```

Ask anything

+ Tools

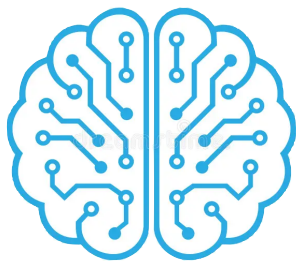


## LLMs

👍: lots of code, fast

👎: lots of code, fast

# Flowco: Rethinking Notebooks



Here's a detailed analysis of the **geyser** Old Faithful eruptions):

## Dataset Overview

- **Source:** Originally from the "faithfu example data" [chegg.com](#) +7 [github.com](#)
- **Structure:** 2 numerical columns — until next eruption in minutes).

## Descriptive Statistics (a

sed on similar datasets and general k

tric

unt

Mean

ian

num

.....mum

Out D...iti

python

Copy Edit

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load dataset
geyser = pd.read_csv("/mnt/data/geyser.csv")

# Basic info and stats
print(geyser.info())
print(geyser.describe())
print(geyser['kind'].value_counts())

# Correlation matrix (numerical columns only)
print(geyser[['duration', 'waiting']].corr())

# Histogram of eruption durations
plt.figure(figsize=(8, 4))
sns.histplot(geyser['duration'], bins=30, kde=True)
plt.title('Distribution of Eruption Durations')
plt.xlabel('Duration (minutes)')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()

# Histogram of waiting times
plt.figure(figsize=(8, 4))
sns.histplot(geyser['waiting'], bins=30, kde=True)
plt.title('Distribution of Waiting Times')
plt.xlabel('Waiting Time (minutes)')
```

Ask anything

+ Tools



## LLMs

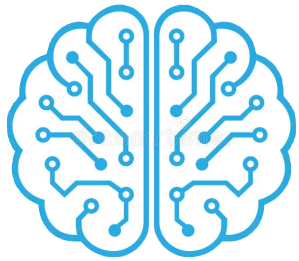
👍: lots of code, fast


👎: lots of code, fast

👎: little control

👎: difficult to verify

# Flowco: Rethinking Notebooks



 **r/SaaS** · 5 days ago  
SpacetimeSorcerer

## I am giving up

I just want to say that I am giving up on creating anything anymore. I was trying to create my little project, but every time there are more and more errors and I am sick of it. I am working on it for about 3 months, I do not have any experience with coding and was doing everything through AI (Cursor, ChatGPT etc.). But everytime I want to change a liiiiitle thing, I kill 4 days debugging other things that go south.

So I do not have any more energy in me to work on this. It is hopeless. AI is still just soooooo stupid and it will fix one thing but destroy 10 other things in your code. I am really sad, because I was enjoying it in the beginnings but now it is just pain and rage. Hat down for those people, who can create something and it is working without coding knowledge.

93   146   Share

## LLMs

👍: lots of code

👎: lots of code

👎: little control

👎: difficult to verify

ian

num

.....num

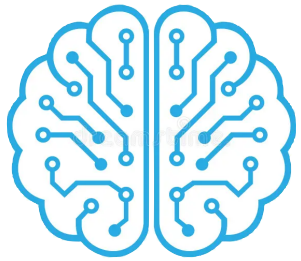
```
plt.title('Distribution of Waiting Times')
```


```
plt.xlabel('Waiting Time (minutes)')
```

Ask anything

+ Tools

# Flowco: Rethinking Notebooks







 **r/SaaS** · 5 days ago  
SpacetimeSorcerer

## I am giving up

I just want to say that I am giving up on creating anything anymore. I was trying to create my little project, but every time there are more and more errors and I am sick of it. I am working on it for about 3 months, I do not have any experience with coding and was doing everything through AI (Cursor, ChatGPT etc.). But everytime I want to change a liiiiittle thing, I kill 4 days debugging other things that go south.

**So I do not have any more energy in me to work on this. It is hopeless. AI is still just soooooo stupid and it will fix one thing but destroy 10 other things in your code. I am really sad, because I was enjoying it in the beginnings but now it is just pain and rage.**

Hat down for those people, who can create something and it is working without coding knowledge.

 93   146  Share

## LLMs

: lots of code

: lots of code

: little control

: difficult to verify

ian

num

.....num

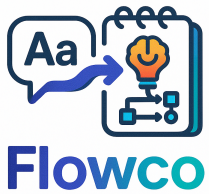
```
plt.title('Distribution of Waiting Times')
```

```
plt.xlabel('Waiting Time (minutes)')
```

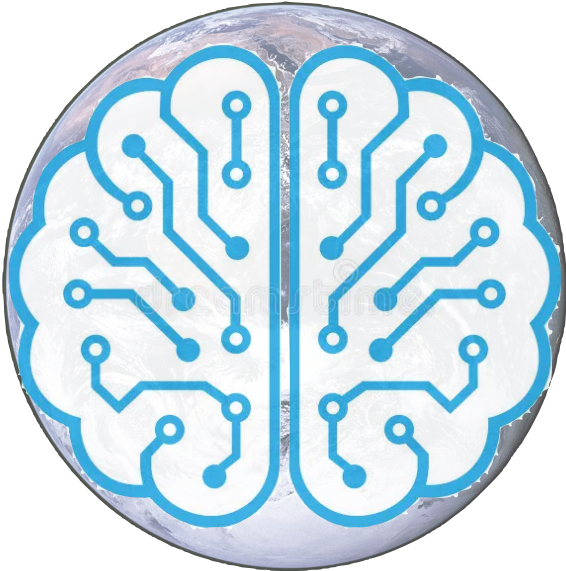
Ask anything

+ Tools

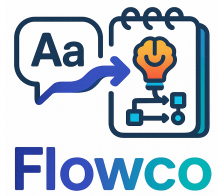
# Flowco: Rethinking Notebooks



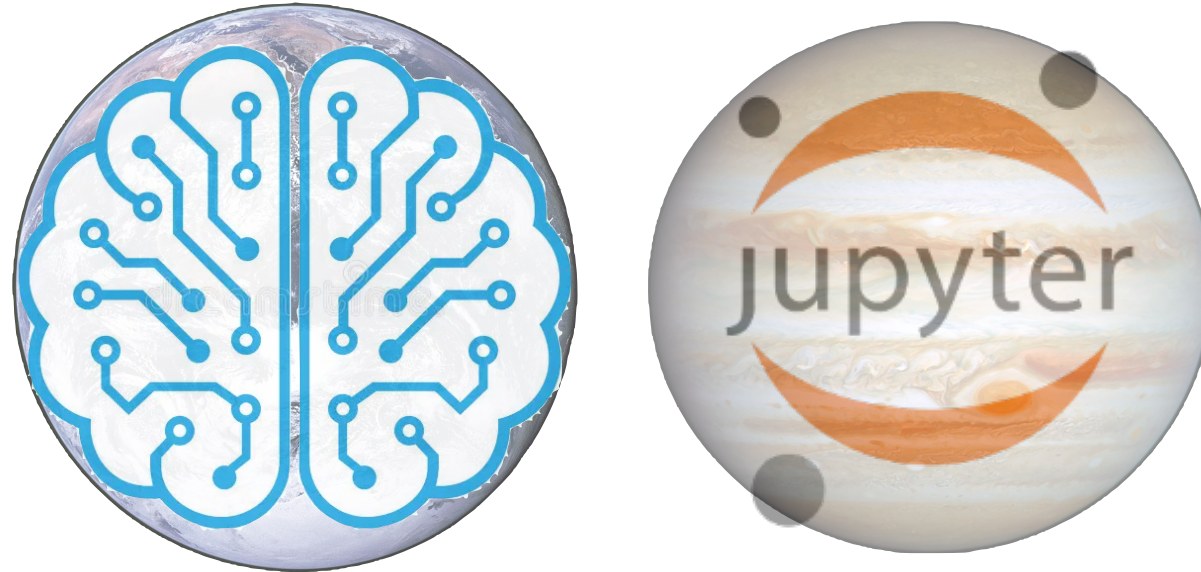
Best of both worlds?



# Flowco: Rethinking Notebooks

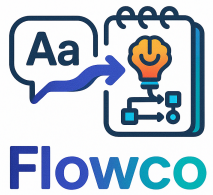


Best of both worlds?

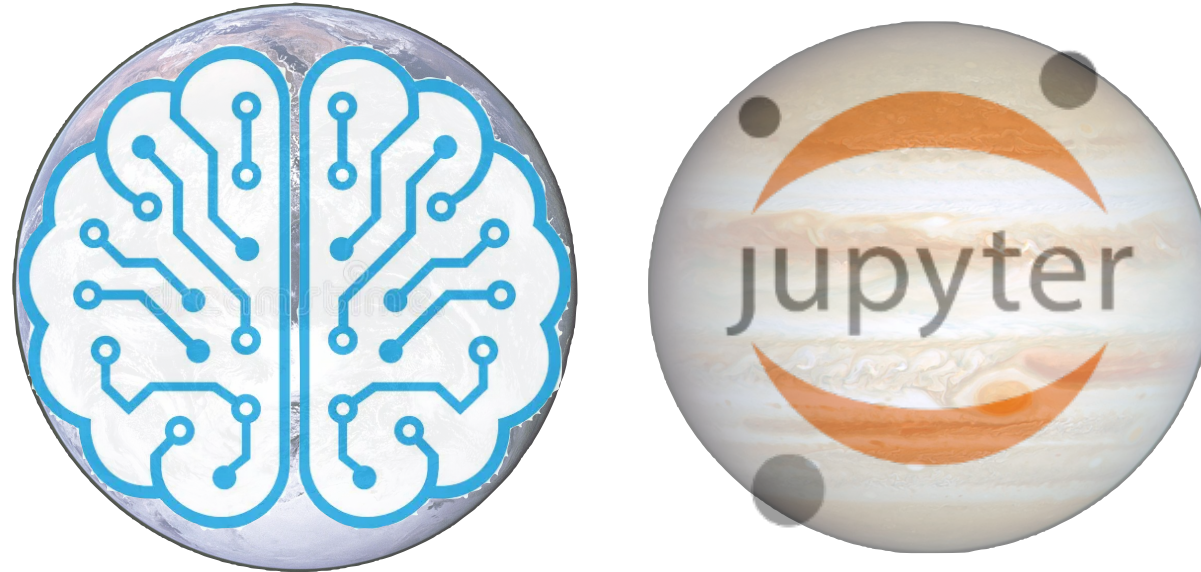


**What's needed:  
A new programming model  
+ new interaction model**

# Flowco: Rethinking Notebooks

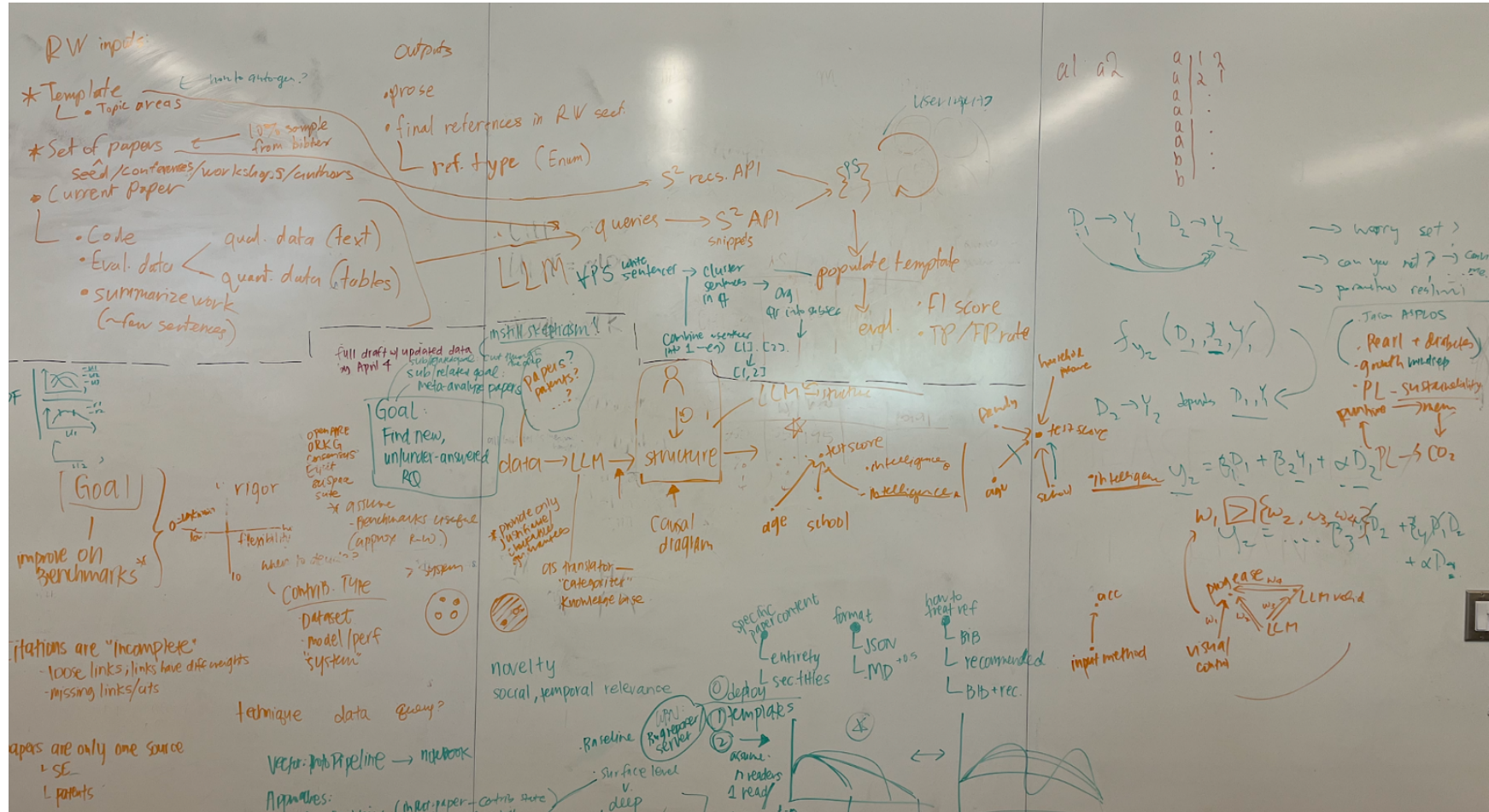
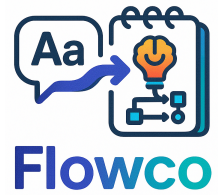


Best of both worlds?

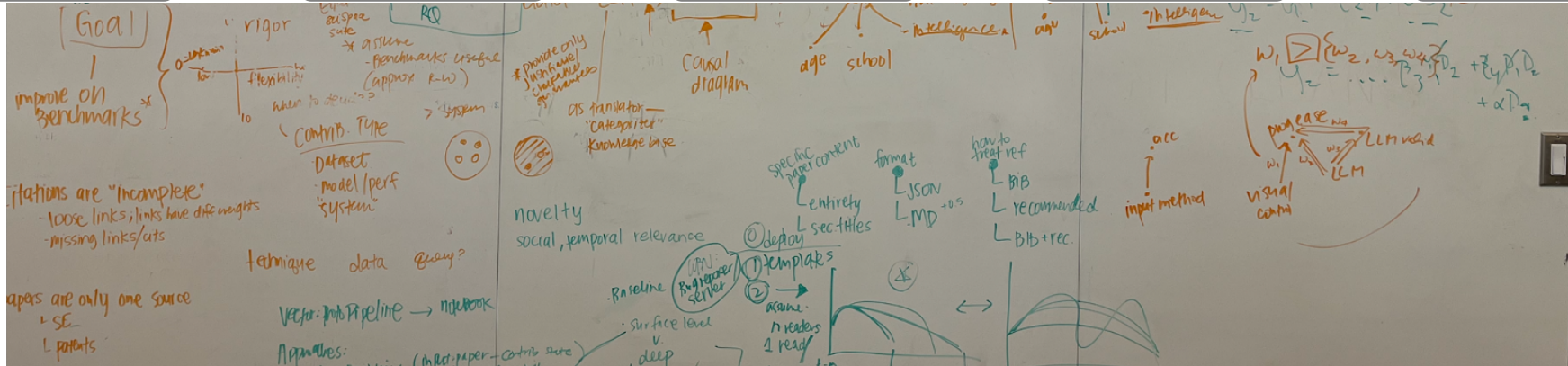
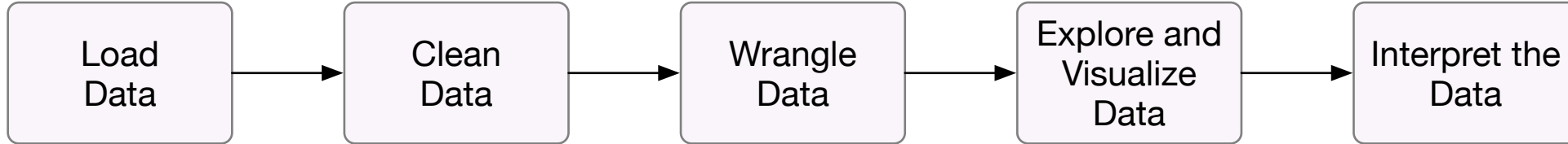
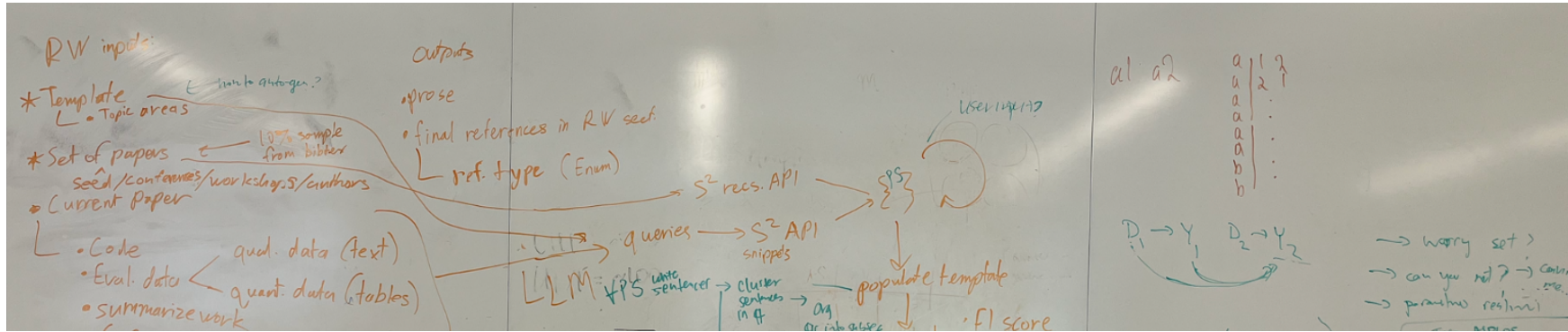


**old** What's needed:  
**A new programming model  
+ new interaction model**

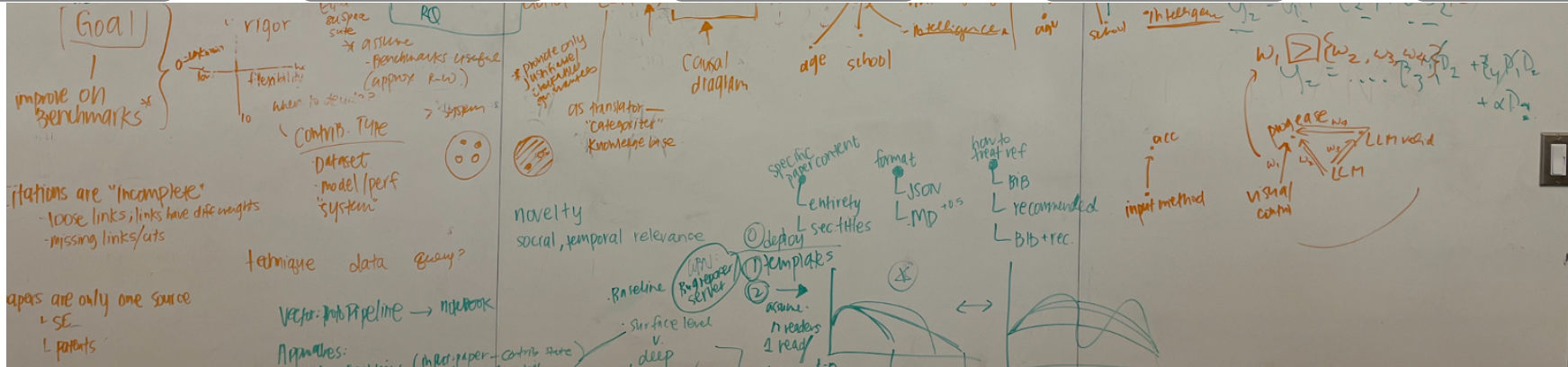
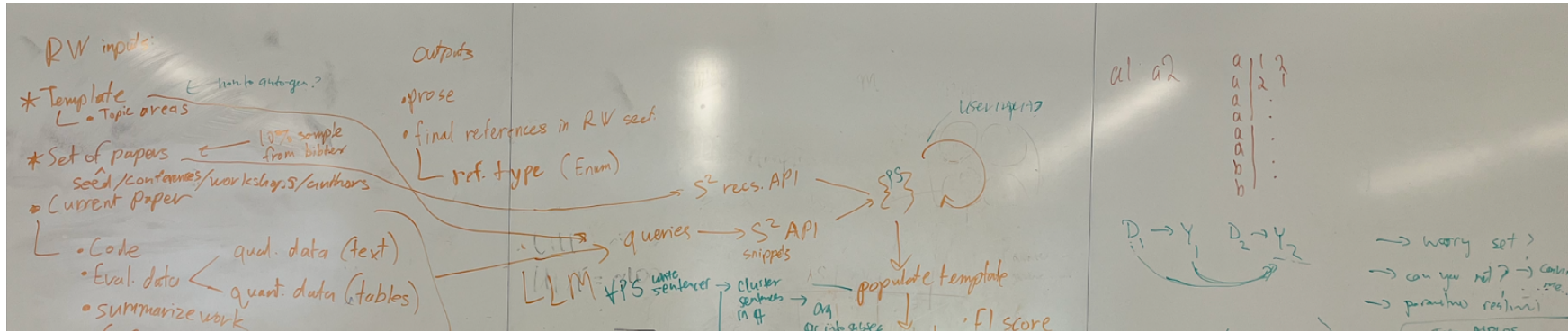
# Flowco: Rethinking Notebooks



# Flowco: Rethinking Notebooks



# Flowco: Rethinking Notebooks



Data flow graphs: control code generation + validate generated code

# Flowco: Rethinking Notebooks



**Projects**

- Edit
- Check
- Test
- Help

**geyser.flowco**

Total cost: 0.09 USD

Run

Abstraction Level

Requirements Code

Which cluster exhibits higher correlation?

Restating the Question

The question asks us to determine which of the two clusters (`short` or `long`) exhibits a higher correlation between eruption duration (`eruptions`) and waiting time (`waiting`) in the Old Faithful dataset.

Approach

To answer this question, we will:

- Use the `k_means_clusters` DataFrame to separate the data into the two clusters (`short` and `long`).
- Calculate the Pearson correlation

Ask Me Anything!

**DataFrame**

|     | eruptions | waiting |
|-----|-----------|---------|
| 0   | 3.600     | 79      |
| 1   | 1.800     | 54      |
| 2   | 3.333     | 74      |
| 3   | 2.283     | 62      |
| 4   | 4.533     | 85      |
| ... | ...       | ...     |
| 267 | 4.117     | 81      |
| 268 | 2.150     | 46      |
| 269 | 4.417     | 90      |
| 270 | 1.817     | 46      |
| 271 | 4.467     | 74      |

[272 rows x 2 columns]

**old\_faithful**  
Load the 'old\_faithful.csv' table

old\_faithful

**Duration-vs-Wait**  
Plot duration vs. wait time

old\_faithful

**K-Means-Clusters**  
Make two clusters with k-means

k\_means\_clusters

**Linear-Regression**  
Fit linear model to each cluster

linear\_regression

**Plot-Regressions**  
Plot the data and regression lines

**Old Faithful Eruption Duration vs. Waiting Time**

**Old Faithful Eruption Data with Cluster Regression Lines**

**K-Means-Clusters**

Status: run\_checked

**Output**

| eruptions | waiting | cluster |
|-----------|---------|---------|
| 3.6       | 79      | long    |
| 1.8       | 54      | short   |
| 3.333     | 74      | long    |
| 2.283     | 62      | short   |
| 4.533     | 85      | long    |

**Requirements**

- The `k_means_clusters` output contains the clustering results of the `old_faithful` dataset using k-means clustering with two clusters.
- The output is a Pandas DataFrame with the same rows as `old_faithful` and an additional column named `cluster` indicating the cluster assignment for each row.
- The `cluster` column contains string values, either 'short' or 'long', representing the two clusters.

**Output Type**

The output is a DataFrame with the original Old Faithful data and an additional column for cluster assignments labeled as 'short' or 'long'.

# Flowco: Rethinking Notebooks



Projects

- Edit
- Check
- Test
- Help

### geyser.flowco

Total cost: 0.09 USD

Run [undo] [redo] [share] [close]

Abstraction Level

Requirements Code [search] [refresh] [help]

#### Which cluster exhibits higher correlation?

Restating the Question

The question asks us to determine which of the two clusters (`short` or `long`) exhibits a higher correlation between eruption duration (`eruptions`) and waiting time (`waiting`) in the Old Faithful dataset.

Approach

To answer this question, we will:

- Use the `k_means_clusters` DataFrame to separate the data into the two clusters (`short` and `long`).
- Calculate the Pearson correlation

Ask Me Anything! [arrow]

```
graph TD; A([old_faithful  
Load the 'old_faithful.csv' table]) -- old_faithful --> B[Duration-vs-Wait  
Plot duration vs. wait time]; A -- old_faithful --> C[K-Means-Clusters  
Make two clusters with k-means]; C -- k_means_clusters --> D[Linear-Regression  
Fit linear model to each cluster]; D -- linear_regression --> E[Plot-Regressions  
Plot the data and regression lines];
```

**DataFrame**

|     | eruptions | waiting |
|-----|-----------|---------|
| 0   | 3.600     | 79      |
| 1   | 1.800     | 54      |
| 2   | 3.333     | 74      |
| 3   | 2.283     | 62      |
| 4   | 4.533     | 85      |
| ... | ...       | ...     |
| 267 | 4.117     | 81      |
| 268 | 2.150     | 46      |
| 269 | 4.417     | 90      |
| 270 | 1.817     | 46      |
| 271 | 4.467     | 74      |

[272 rows x 2 columns]

**Duration-vs-Wait**  
Plot duration vs. wait time

**K-Means-Clusters**  
Status: run\_checked

| eruptions | waiting | cluster |
|-----------|---------|---------|
| 3.6       | 79      | long    |
| 1.8       | 54      | short   |
| 3.333     | 74      | long    |
| 2.283     | 62      | short   |
| 4.533     | 85      | long    |

**Old Faithful Eruption Duration vs. Waiting Time**

**Old Faithful Eruption Data with Cluster Regression Lines**

### K-Means-Clusters

Status: run\_checked

**Output**

| eruptions | waiting | cluster |
|-----------|---------|---------|
| 3.6       | 79      | long    |
| 1.8       | 54      | short   |
| 3.333     | 74      | long    |
| 2.283     | 62      | short   |
| 4.533     | 85      | long    |

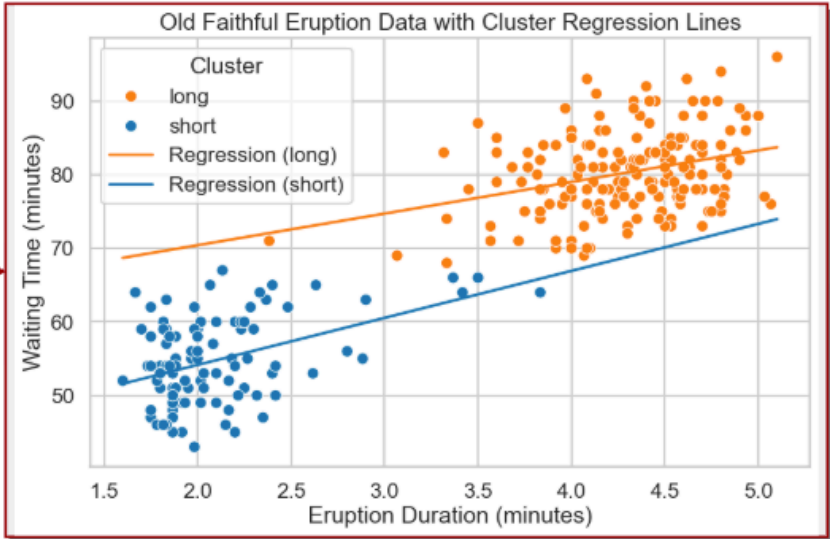
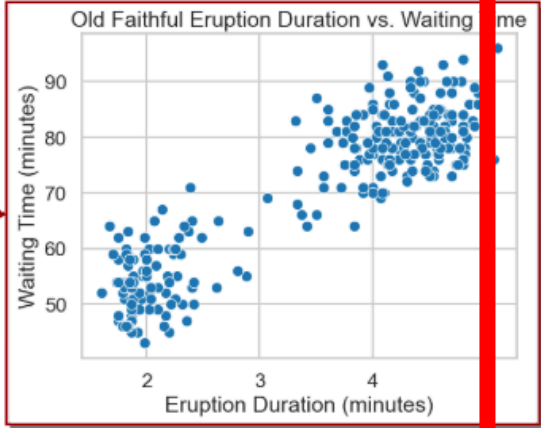
**Requirements**

- The `k_means_clusters` output contains the clustering results of the `old_faithful` dataset using k-means clustering with two clusters.
- The output is a Pandas DataFrame with the same rows as `old_faithful` and an additional column named `cluster` indicating the cluster assignment for each row.
- The `cluster` column contains string values, either 'short' or 'long', representing the two clusters.

**Output Type**

The output is a DataFrame with the original Old Faithful data and an additional column for cluster assignments labeled as 'short' or 'long'.

**Duration-vs-Wait**  
Plot duration vs. wait time



Deploy ⋮



## K-Means-Clusters

Status: run\_checked

### Output

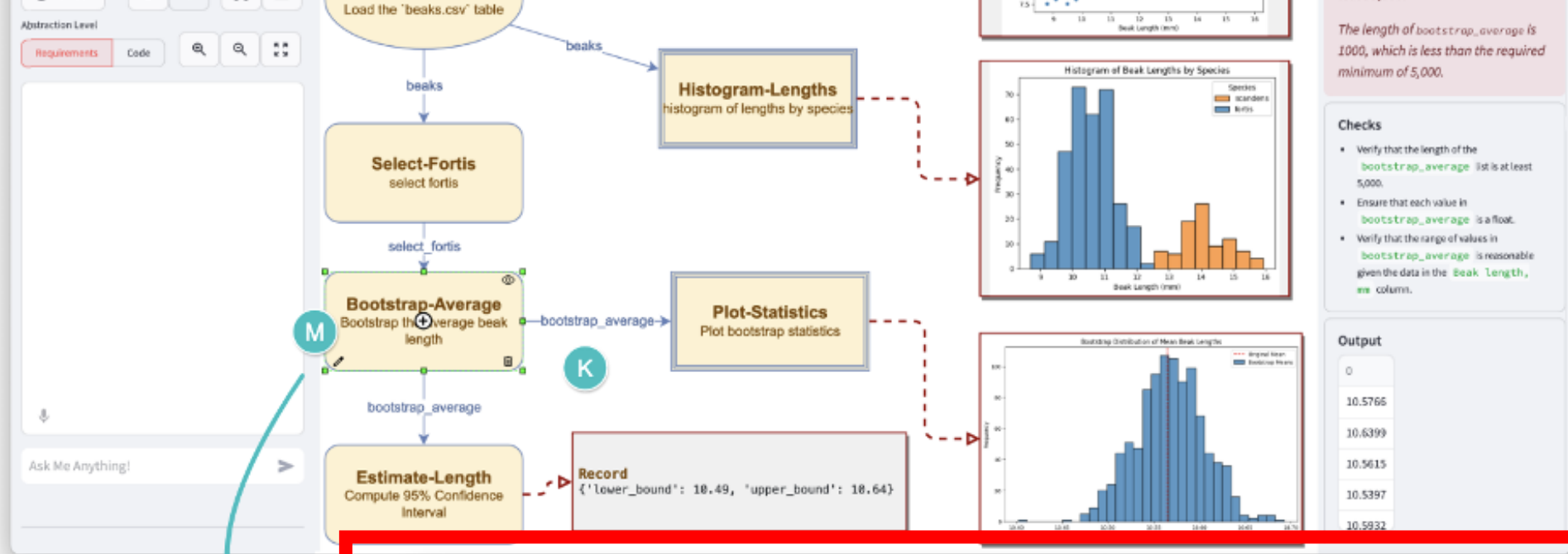
| eruptions | waiting | cluster |
|-----------|---------|---------|
| 3.6       | 79      | long    |
| 1.8       | 54      | short   |
| 3.333     | 74      | long    |
| 2.283     | 62      | short   |
| 4.533     | 85      | long    |

### Requirements

- The `k_means_clusters` output contains the clustering results of the `old_faithful` dataset using k-means clustering with two clusters.
- The output is a Pandas DataFrame with the same rows as `old_faithful` and an additional column named `cluster` indicating the cluster assignment for each row.
- The `cluster` column contains string values, either 'short' or 'long', representing the two clusters.

### Output Type

The output is a DataFrame with the original Old Faithful data and an additional column for cluster assignments labeled as 'short' or 'long'.



The length of `bootstrap_average` is 1000, which is less than the required minimum of 5,000.

- Checks**
- Verify that the length of the `bootstrap_average` list is at least 5,000.
  - Ensure that each value in `bootstrap_average` is a float.
  - Verify that the range of values in `bootstrap_average` is reasonable given the data in the `Beak length, mm` column.

**Output**

|         |
|---------|
| 0       |
| 10.5765 |
| 10.6399 |
| 10.5615 |
| 10.5397 |
| 10.5932 |

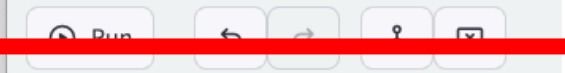
## Output Checks for Bootstrap-Average

Save

Suggest

- checks
- Verify that the length of the `bootstrap_average` list is at least 5,000. N
  - Ensure that each value in `bootstrap_average` is a float.
  - Verify that the range of values in `bootstrap_average` is reasonable given the data in the `Beak length, mm` column.

geyser.flowco  
Total cost: 0.09 USD



Abstraction Level  
Requirements Code

Which cluster exhibits higher correlation?

**Restating the Question**

The question asks us to determine which of the two clusters ( `short` or `long` ) exhibits a higher correlation between eruption duration ( `eruptions` ) and waiting time ( `waiting` ) in the Old Faithful dataset.

**Approach**

To answer this question, we will:

1. Use the `k_means_clusters` DataFrame to separate the data into the two clusters ( `short` and `long` ).
2. Calculate the Pearson correlation

Ask Me Anything!

DataFrame

|     | eruptions | waiting |
|-----|-----------|---------|
| 0   | 3.600     | 79      |
| 1   | 1.800     | 54      |
| 2   | 3.333     | 74      |
| 3   | 2.283     | 62      |
| 4   | 4.533     | 85      |
| ... | ...       | ...     |
| 267 | 4.117     | 81      |
| 268 | 2.150     | 46      |
| 269 | 4.417     | 90      |
| 270 | 1.817     | 46      |
| 271 | 4.467     | 74      |

[272 rows x 2 columns]

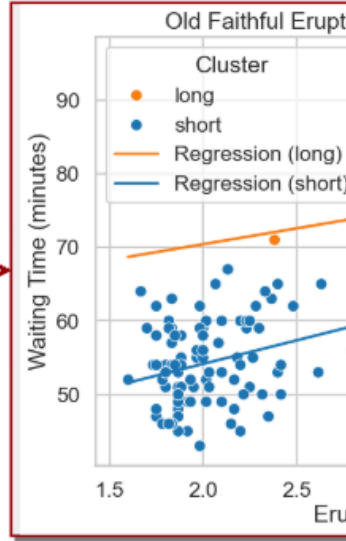
**old\_faithful**  
Load the 'old\_faithful.csv' table

**Duration-vs-Wait**  
Plot duration vs. wait time

**K-Means-Clusters**  
Make two clusters with k-means

**Linear-Regression**  
Fit linear model to each cluster

**Plot-Regressions**  
Plot the data and regression lines



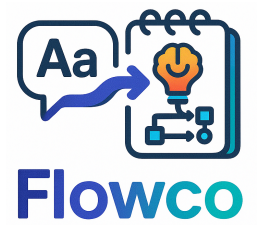
# Flowco: Evolving a New Notebook



evolve

notebook *that*  
*requires no*  
*coding*

# Flowco: Evolving a New Notebook



evolve

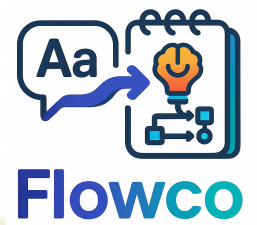
notebook *that*  
*requires no*  
*coding*



exploit niche

*dataflow match*  
*to data science*  
*workflows;*  
*enables analysis*  
*+ validation*

# Flowco: Evolving a New Notebook



evolve

notebook *that*  
*requires no*  
*coding*



exploit niche

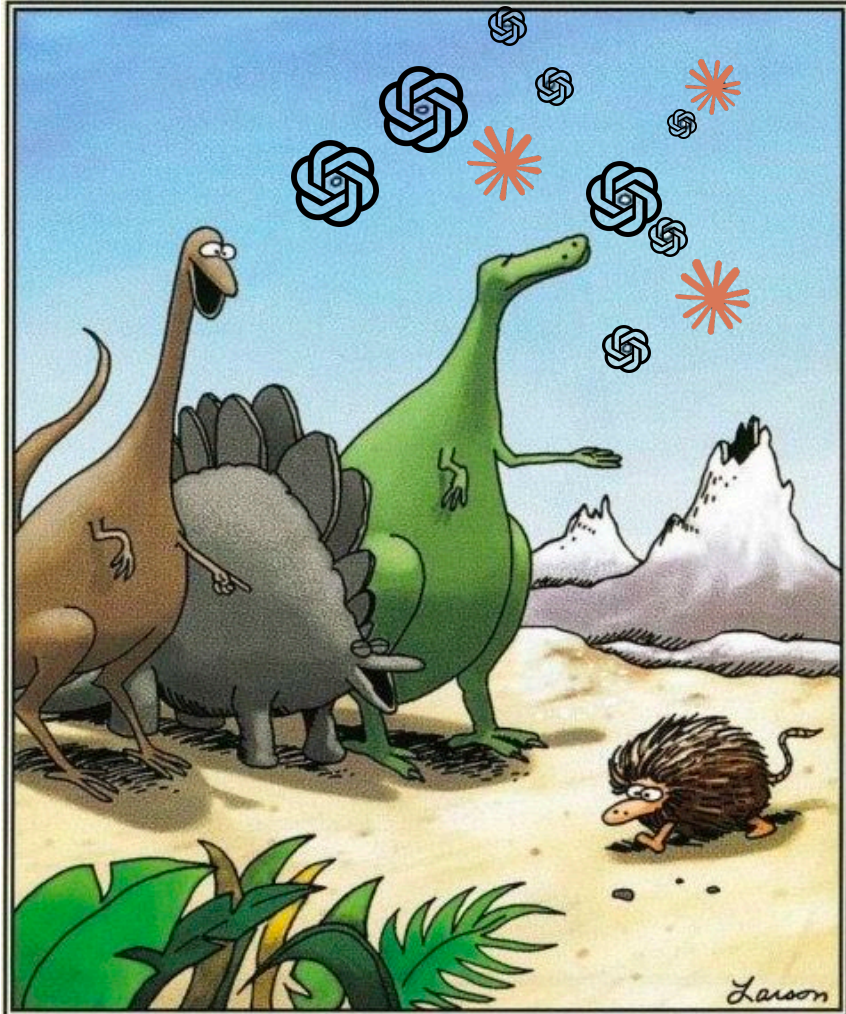
*dataflow match*  
*to data science*  
*workflows;*  
*enables analysis*  
*+ validation*



ensure fitness

checking pre-/  
postconditions,  
review of high  
level specs, visual  
analysis

# The LLM (R)evolution

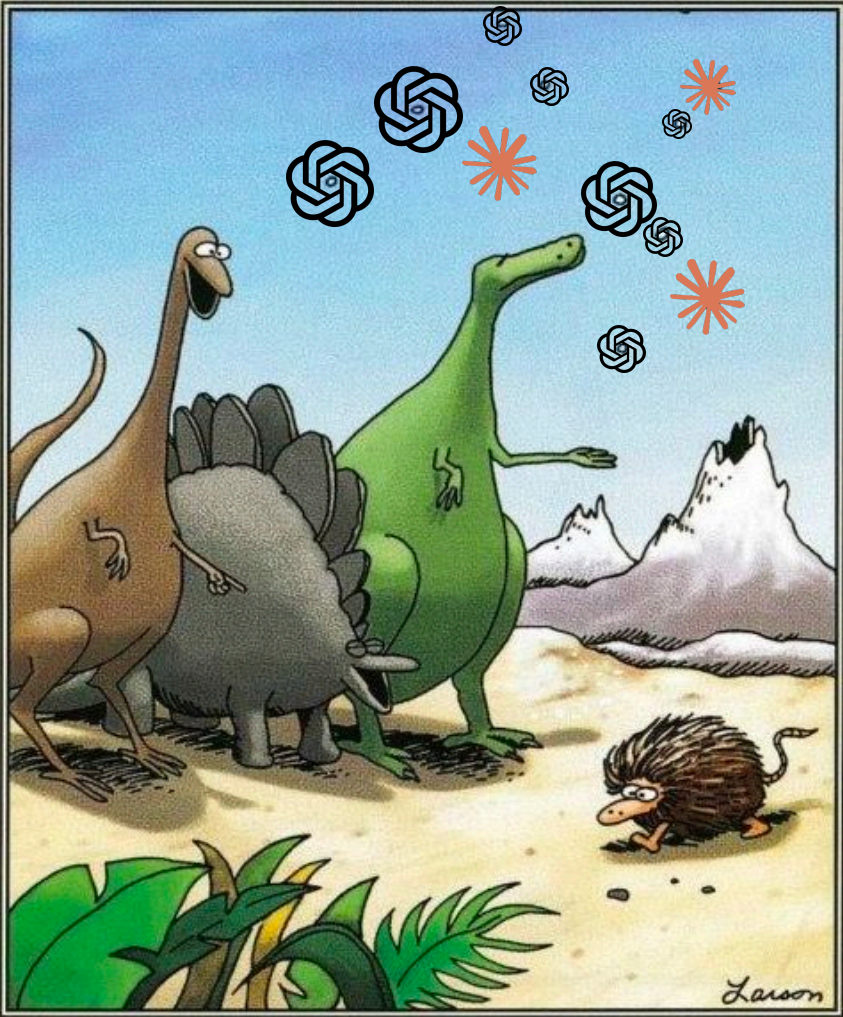


evolve

exploit niche

ensure fitness

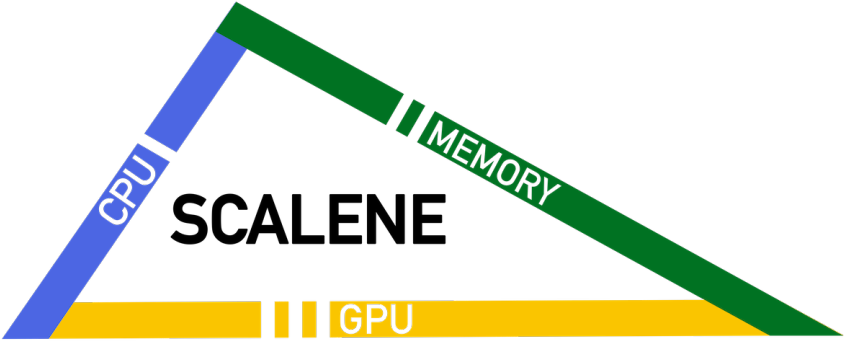
# The LLM (R)evolution



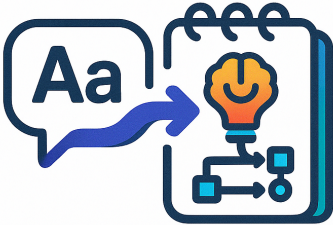
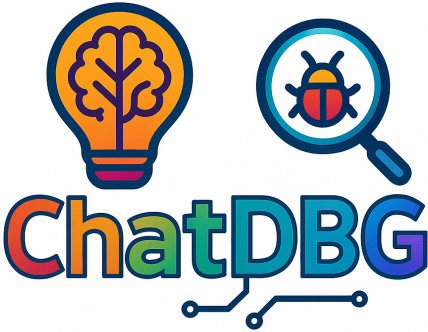
evolve

exploit niche

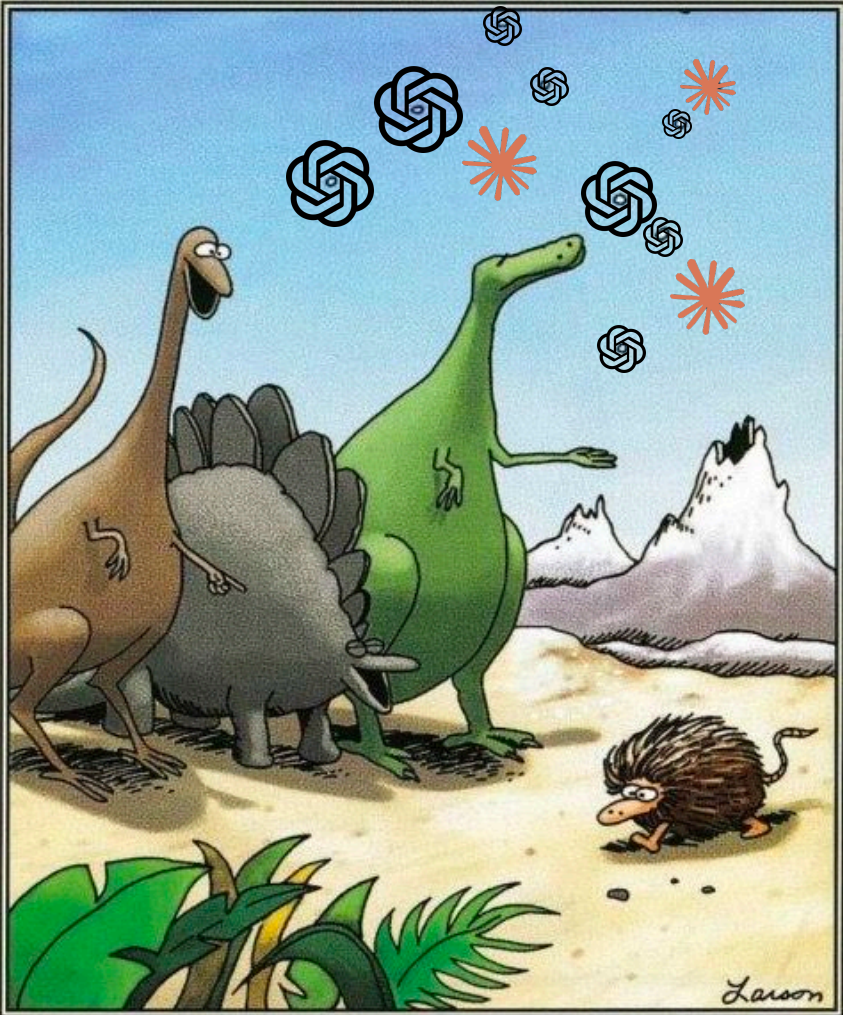
ensure fitness



CoverUp



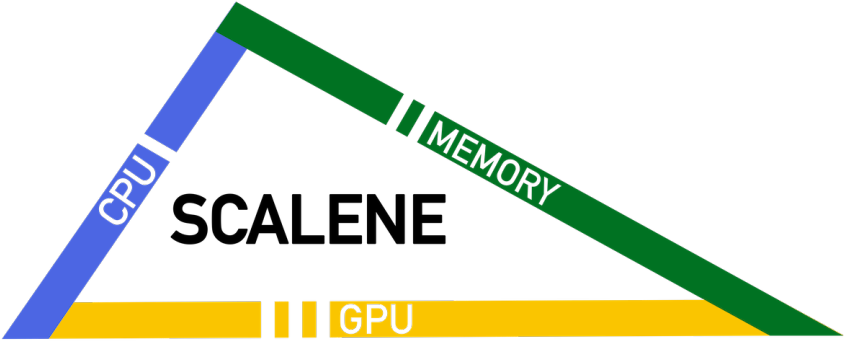
# The LLM (R)evolution



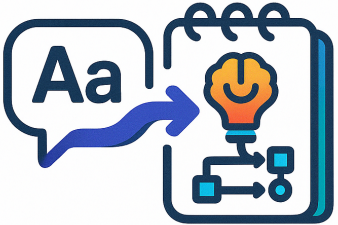
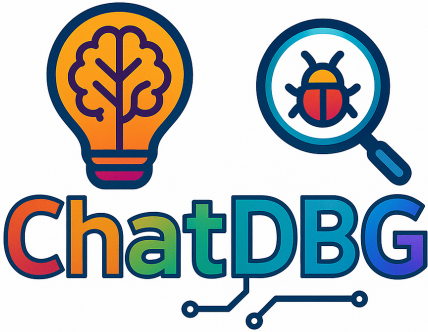
evolve

exploit niche

ensure fitness



CoverUp



<https://go-flow.co>

<https://plasma-umass.org>