



Escaping AI POC purgatory: Techniques for enterprise AI engineers

WRITER

Generative AI: Gold rush or bust?

AI's Hype Phase is Dying and Fast

The AI hype bubble is deflating. Now comes the hard part.

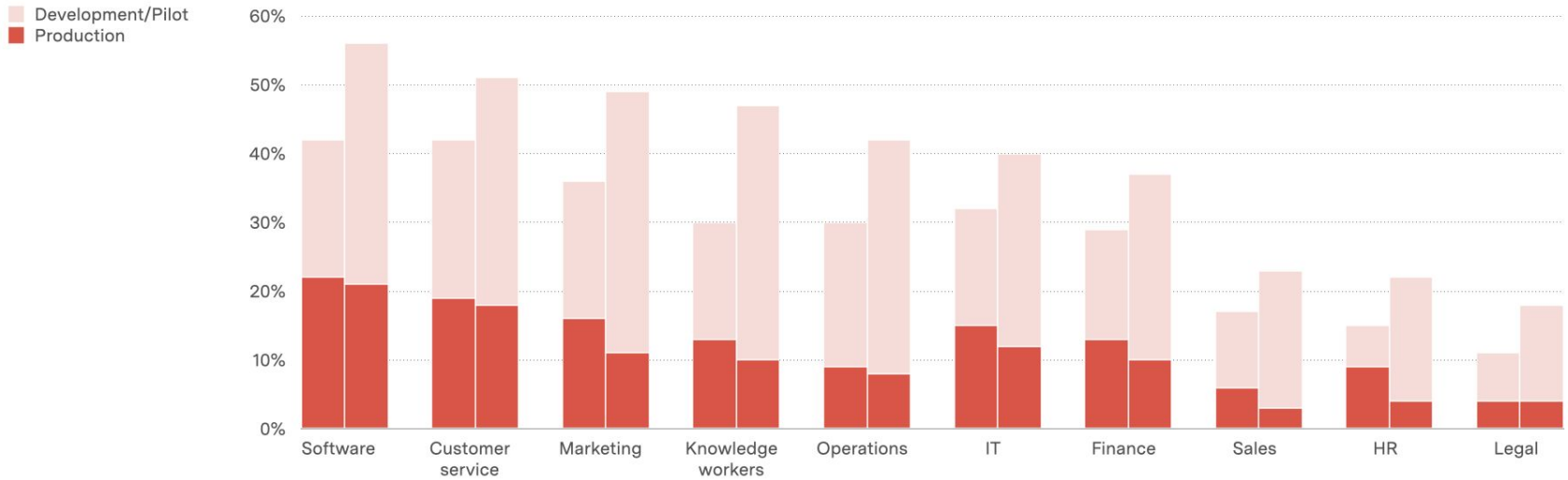
Why 85% of AI projects fail

Why we may be headed for a generative AI winter

Huge interest, limited use so far

Enterprise software takes time, and come with early disappointments

Enterprise use case adoption rates for generative AI, October 2023 & February 2024



Source: Bain

Benedict Evans — July 2024

5



AI can become a gravity well for enterprises.

Why?

Enterprise-grade
generative AI
is hard



Low accuracy



Low efficiency



Low adoption

**How do we move from
POC to production?**

**How do we achieve
escape velocity?**

Hi, I'm Sam 🖐️



Sam Julien

Director of Developer Relations at Writer
Getting Started in DevRel | egghead.io
Developer Microskills

**How do we achieve
escape velocity?**

WRITER

The only fully-integrated solution for building AI apps with faster time to value

AI digital assistants

AI workflows

AI apps



Palmyra LLMs

State-of-the-art top-benchmarked Writer-built LLMs that are compliant, efficient, and transparent.



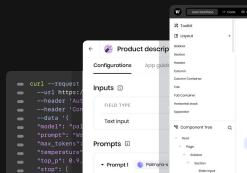
Knowledge Graph

Connects your AI apps to structured and unstructured data with graph-based RAG.



AI guardrails

Enforce regulatory, accuracy, and brand compliance across all AI apps and workflows.



AI Studio and integrations

Code (for developers) and no-code (for business users) tools for quickly building production-grade AI apps.



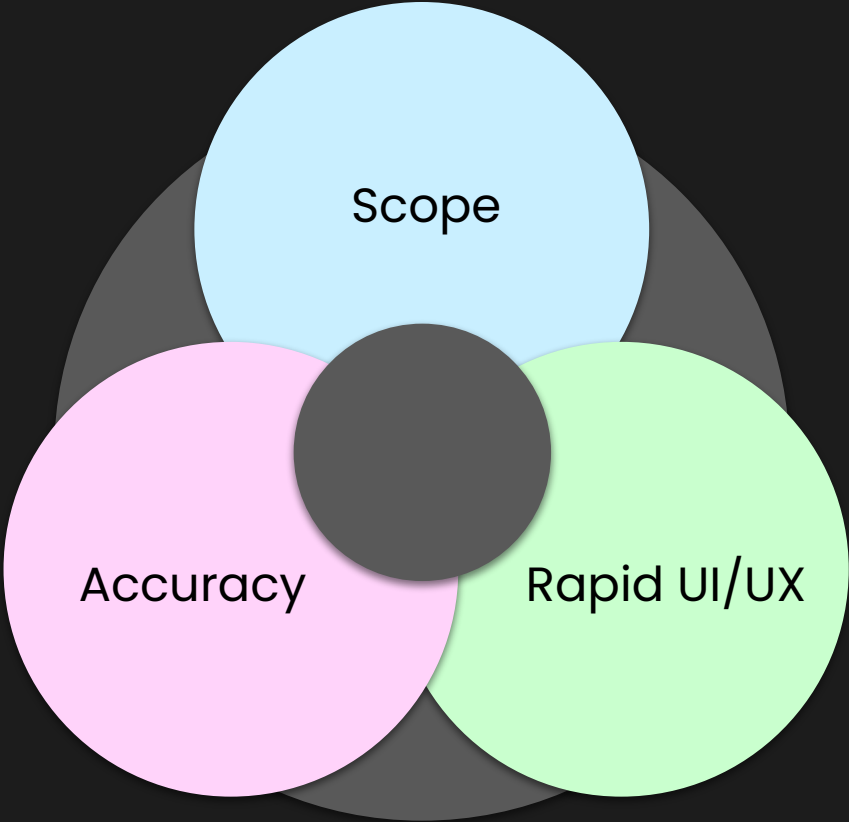
AI program management

Drive change management, workflow by workflow, to realize business value.

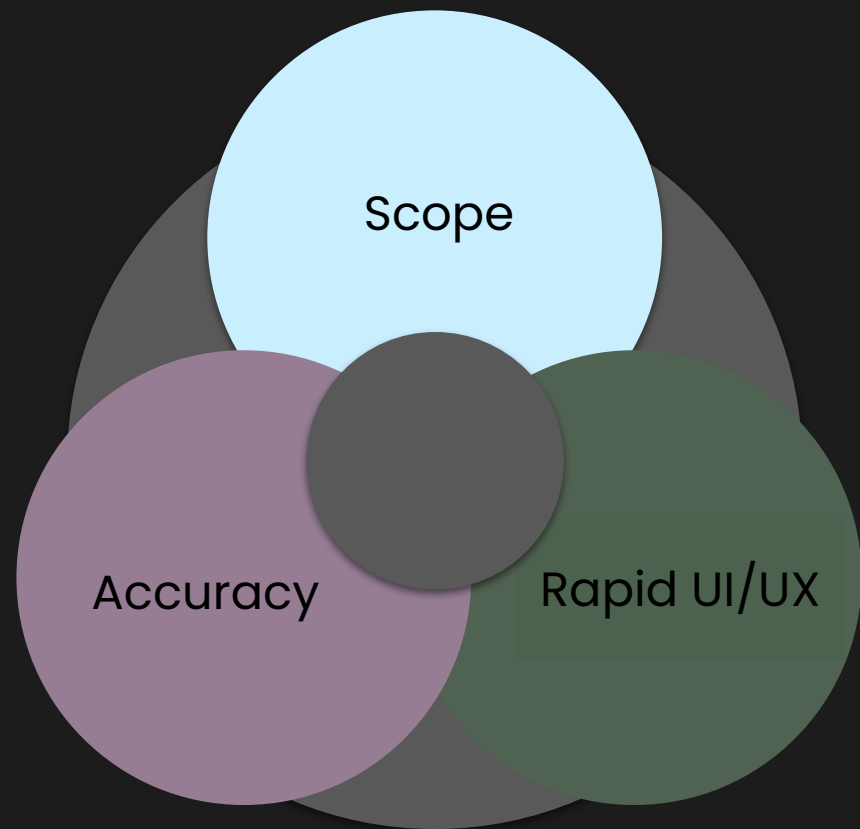
delta-v

The change in velocity needed to launch from a planet's surface; the total "effort" needed to achieve a certain change in velocity, including overcoming gravity and atmospheric drag.

Enterprise AI delta-v



Component 1: Reconciling AI engineering scope



The (current) enterprise AI paradox



AI engineering
requires lots of
infrastructure
and resources



Immediate ROI
comes from
scoped, focused
tasks at scale.

Point solutions offer speed, but come up short for enterprise scale

Narrow scope

Security risks

Management overhead

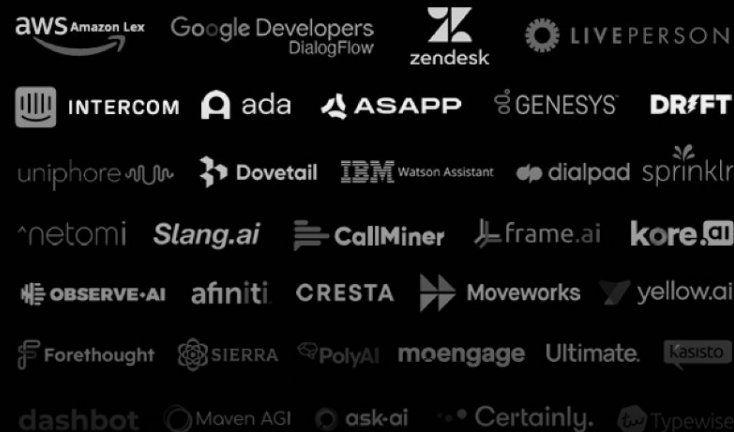
SALES



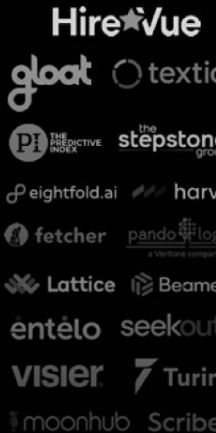
MARKETING



CUSTOMER EXPERIENCE



HUMAN CAPITAL



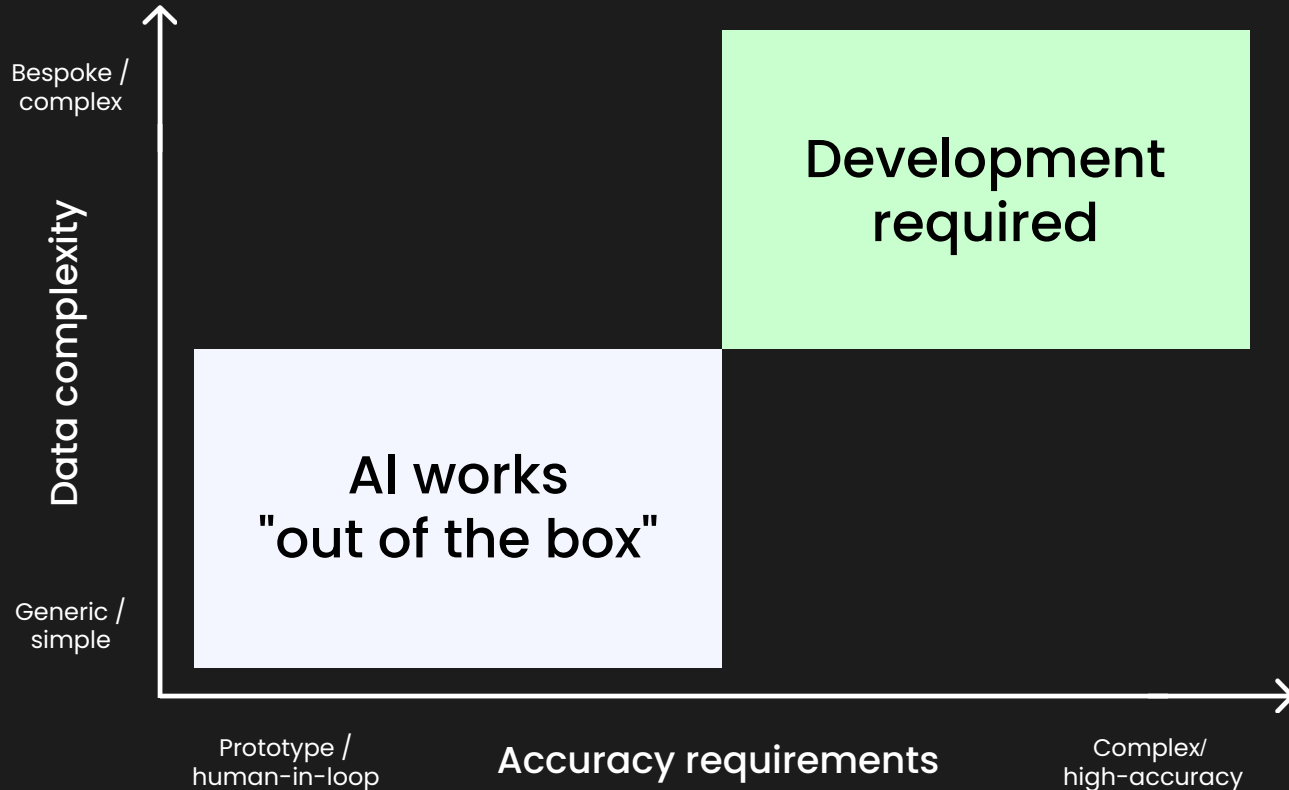
LEGAL

PARTNERSHIPS

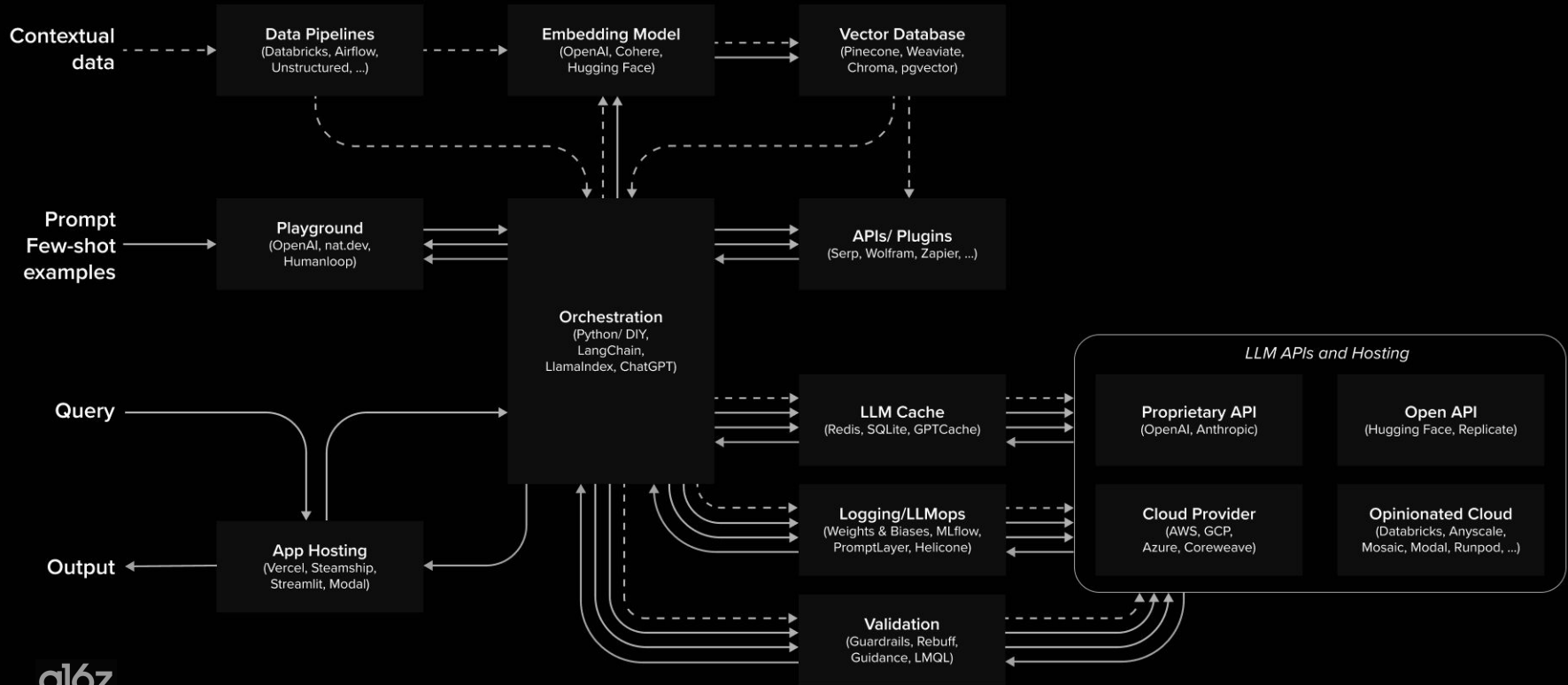
REGTECH & COMPLIANCE

FINANCE

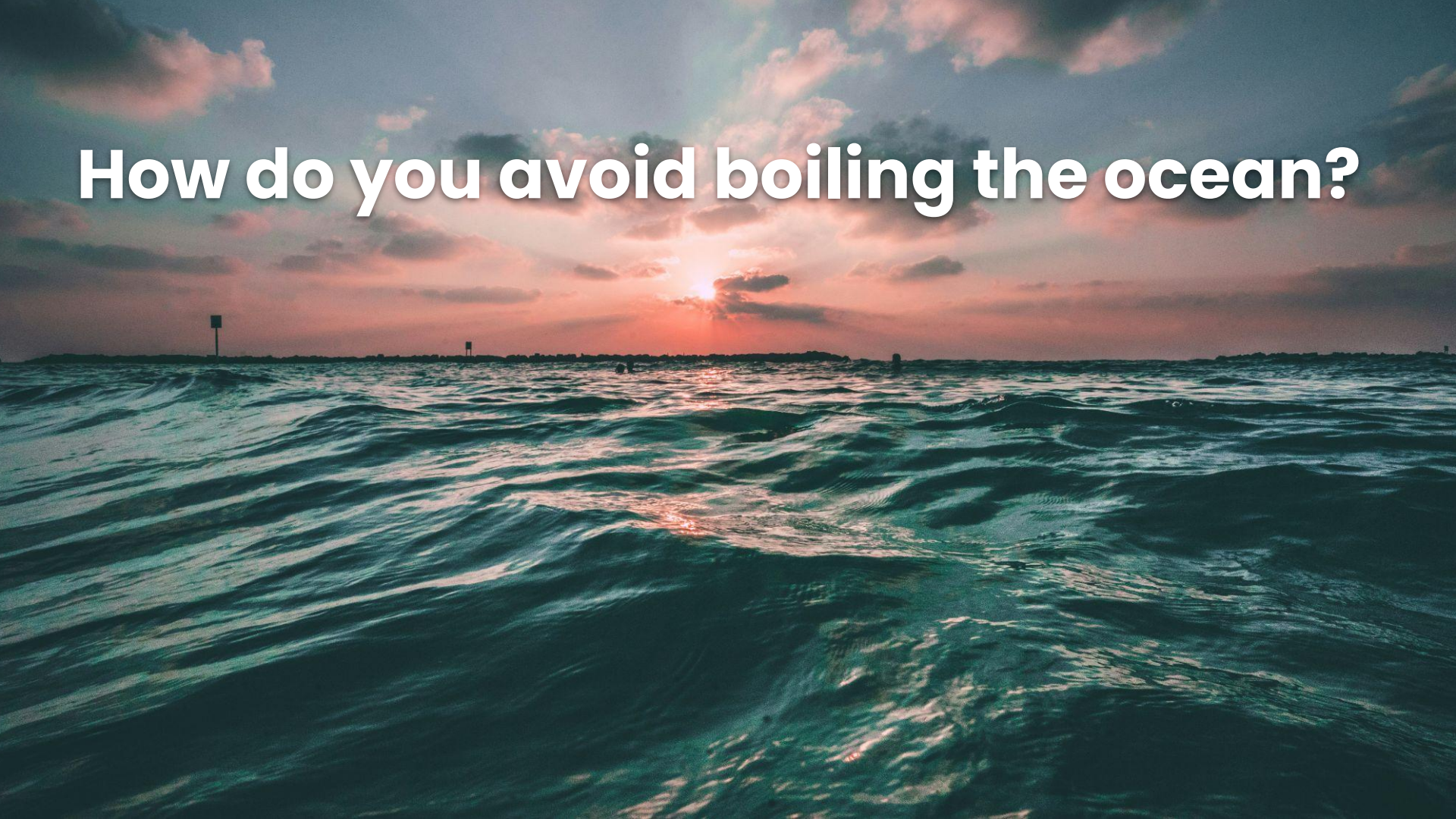
Some use cases will work "out of the box" – but the hardest & most value-aligned won't



DIY offers depth, but implementation is slow and hairy



How do you avoid boiling the ocean?



First, think microservices, not monoliths.

IT, finance, HR

Incidence reports

Training materials

Investor relations materials

Supply chain analysis

Competitor analysis

Report deep dives

Job descriptions

L&D content

Financial summaries

Product + design

Parsing user research

Push notifications

Error messages

Release notes

UX copy

Internal messages

Zero-state pages

User stories

Onboarding material

Marketing, Comms, PR

Blog posts

Landing pages

Case studies

Client stories

Messaging ideas

SEO optimization

White papers

Newsletters

Social media campaigns

Sales + support

Sales enablement assistants

RFPs

Expert agent assistants

Clinical analysis

Support call summaries

Coverage descriptions

Compliance

Client comms

Knowledgebases

Second, aim for a full-stack approach.

Here's how we approach it at Writer for customers:



Prebuilt apps

Customize, integrate,
and manage OOTB apps



AI Studio

Use a platform to build AI apps for
your business

Platform-as-a-service

LLMs

RAG

AI guardrails



swyx 🧐

@swyx

Software engineering:

- make it work
- make it right
- make it fast

AI engineering:

- make it work on 1 thing you want
- make it generally work on most user queries
- make it efficient

11:04 AM · Aug 6, 2024 · **28K** Views

📺 View post engagements



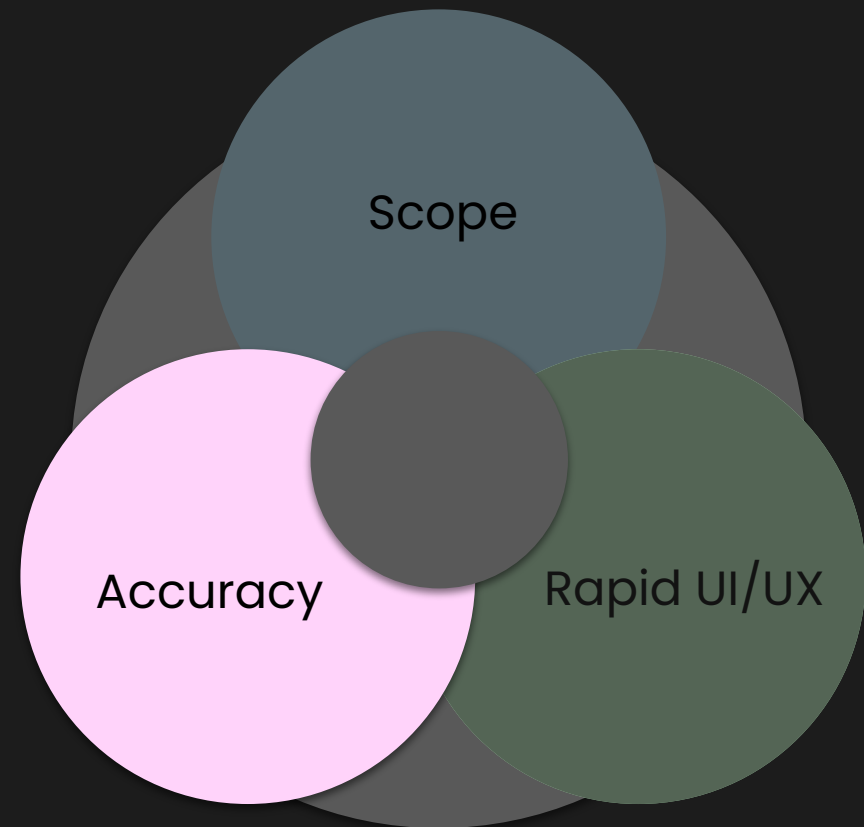
Full-stack enterprise AI

Make it work for one use case

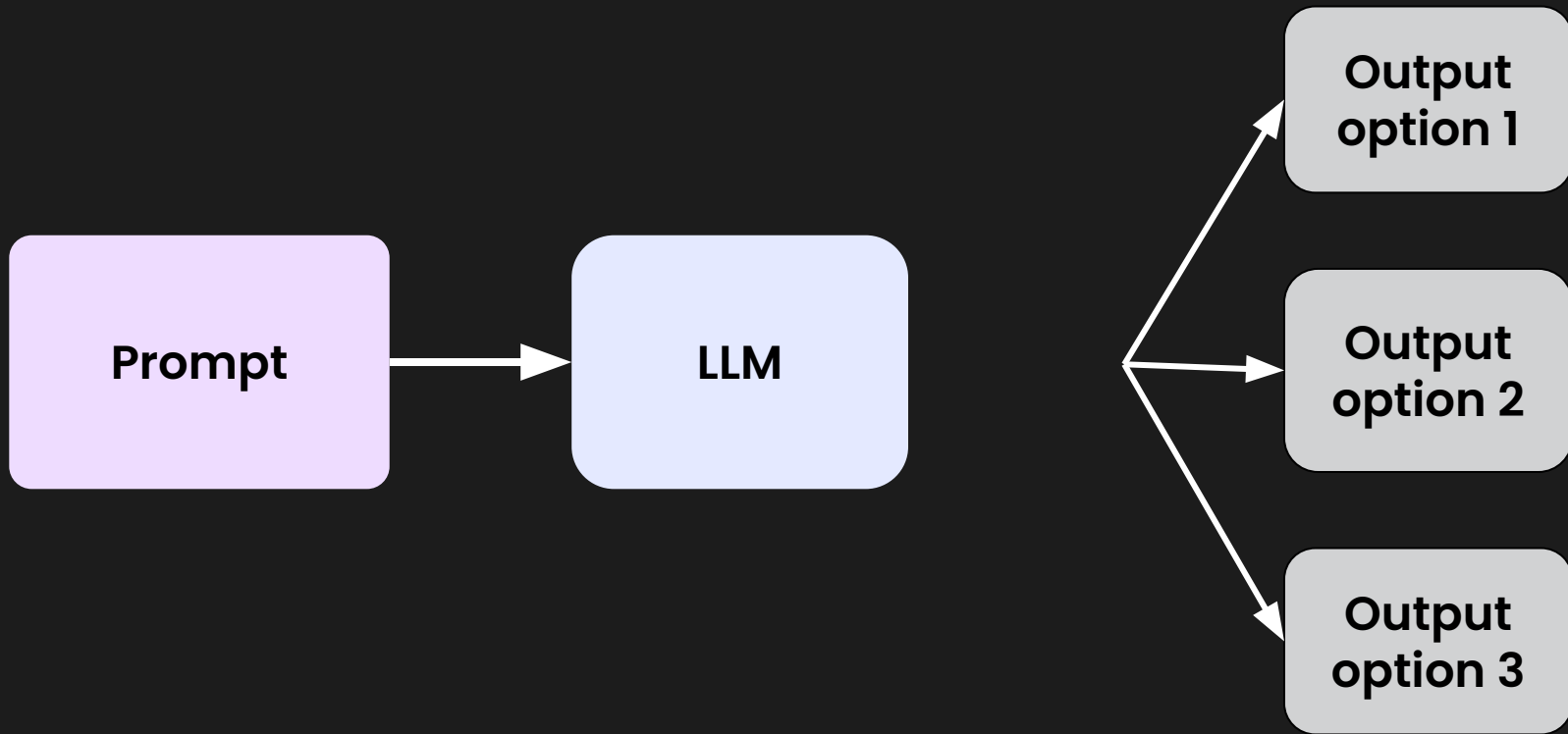
Generalize to other use cases or verticals

Make it efficient for other teams to build

Component 2: Repeatable accuracy

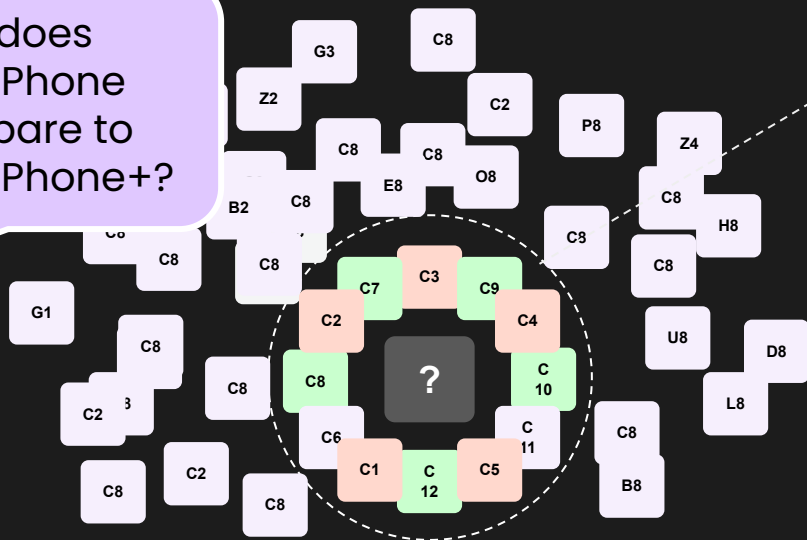


LLMs are non-deterministic



Enterprise data is dense and specialized

How does NovaPhone compare to NovaPhone+?



LLM

The NovaPhone+ is water resistance, has a 12-megapixel camera, and has a 18 hour battery.

The NovaPhone is also water resistant, has a 36-megapixel camera, and has a 12 hour battery.

Both phones both cost \$795.

Three layers to improve accuracy

Prompt/Code

RAG

LLM

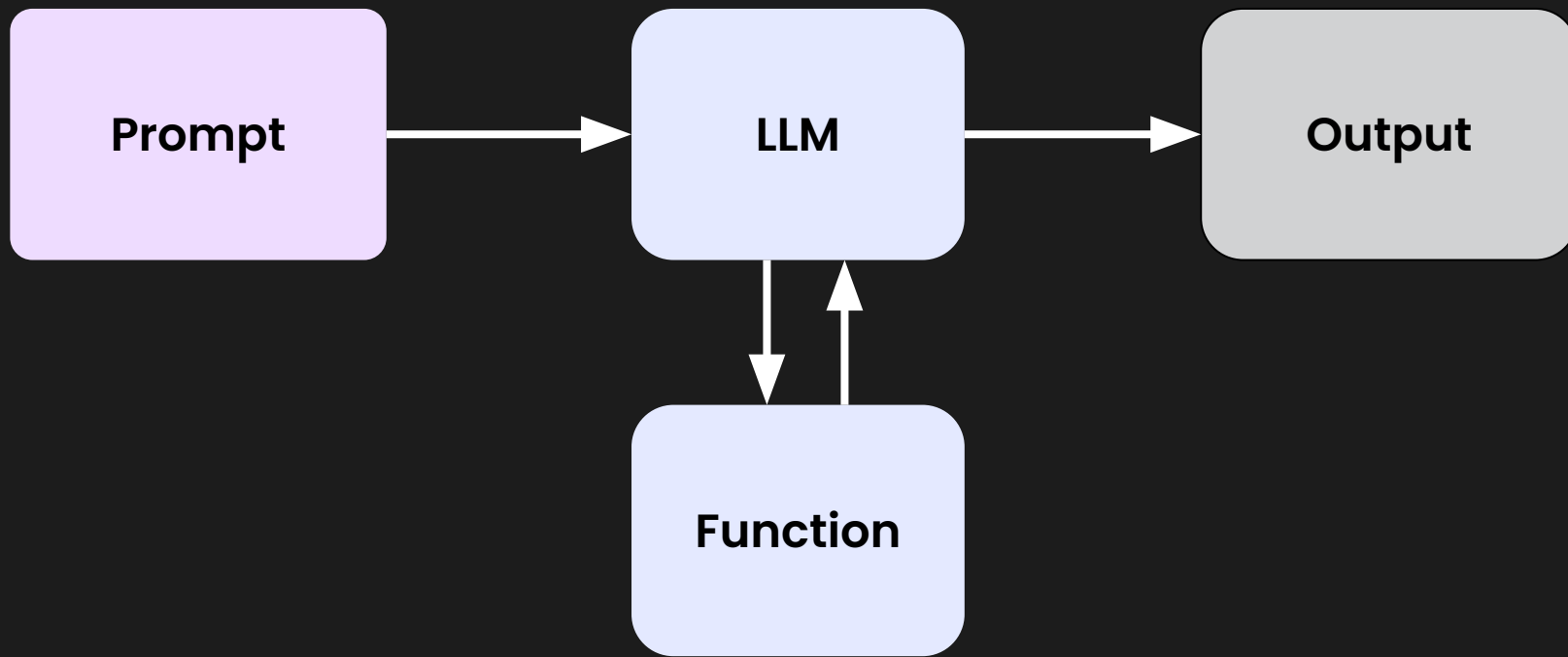
Three layers to improve accuracy

Prompt/Code

RAG

LLM

Structured output with function calling



Function calling

```
tools = [  
  {  
    "type": "function",  
    "function": {  
      "name": "get product info",  
      "description": "Get information about a  
product by its ID",  
      "parameters": {  
        "type": "object",  
        "properties": {  
          "id": {  
            "type": "number",  
            "description": "The ID of the  
product to retrieve information for",  
          }  
        },  
        "required": ["id"],  
      },  
    },  
  },  
]
```

```
messages = [{"role": "user", "content":  
"what is the name of product 1234?" }]
```

```
response = client.chat.chat.create(  
  model="your-model-here",  
  messages=messages,  
  tools=tools,  
  tool_choice="auto"  
)
```

```
response_message = response.choices[0].message  
messages.append(response_message)
```

```
print(response_message)
```

```
# The name of product 1234 is "Terra running  
shoe."
```


What We've Learned From A Year of Building with LLMs

A practical guide to building successful LLM products, covering the tactical, operational, and strategic.

AUTHORS

[Eugene Yan](#)
[Bryan Bischof](#)
[Charles Frye](#)
[Hamel Husain](#)
[Jason Liu](#)
[Shreya Shankar](#)

PUBLISHED

June 8, 2024

Also published on O'Reilly Media in three parts: [Tactical](#), [Operational](#), [Strategic \(podcast\)](#).
Also translated to [Japanese](#) (by [Kazuya Kanno](#))

It's an exciting time to build with large language models (LLMs). Over the past year, LLMs have become "good enough" for real-world applications. And they're getting better and cheaper every year. Coupled with a parade of demos on social media, there will be an [estimated \\$200B investment in AI by 2025](#). Furthermore, provider APIs have made LLMs more accessible, allowing everyone, not just ML engineers and scientists, to build intelligence into their products. Nonetheless, while the barrier to entry for building with AI has been lowered, creating products and systems that are effective—beyond a demo—remains deceptively difficult.

We've spent the past year building, and have discovered many sharp edges along the way. While we don't claim to speak for the entire industry, we'd like to share what we've learned to help you avoid our mistakes and iterate faster. These are organized into three sections:

On this page

[1 Tactical: Nuts & Bolts of Working with LLMs](#)

[1.1 Prompting](#)

[1.2 Information Retrieval / RAG](#)

[1.3 Tuning and optimizing workflows](#)

[1.4 Evaluation & Monitoring](#)

[2 Operational: Day-to-day and Org concerns](#)

[3 Strategy: Building with LLMs without Getting Out-Maneuvered](#)

[4 Enough 0 to 1 demos, it's time for 1 to N products](#)

[5 Stay In Touch](#)

[6 Acknowledgements](#)

Join 2000+ readers getting updates on how to apply LLMs effectively

Email Address

Subscribe

Three layers to improve accuracy

Prompt/Code

RAG

LLM

Combine applied research with graph-based RAG

- ✓ **Graph structure** preserves relationships between data
- ✓ **Compression techniques** help prevent loss of context
- ✓ Applied research like **Fusion-in-Decoder** minimizes hallucinations



Our approach at Writer: >86% accuracy with <3% hallucinations

GUI for data connectors & APIs

1. Specialized LLM to build graph

2. Retrieval-aware compression

3. Fusion-in-decoder

4. Transparent thought process

Palmyra LLMs

Evaluation of RAG approaches: accuracy and response time

| Pipeline | RobustQA Avg. score | Avg. response time(s) |
|--|---------------------|-----------------------|
| Azure Cognitive Search Retriever + GPT-4 + Ada | 72.36 | >1.0s |
| Canopy (Pinecone) | 59.61 | >1.0s |
| LangChain + Pinecone + OpenAI | 61.42 | <0.8s |
| LangChain + Pinecone + Cohere | 69.02 | <0.6s |
| LlamaIndex + Weaviate Vector Store - Hybrid Search | 75.89 | <1.0s |
| RAG Google Cloud Vertex AI Search + Bison | 51.08 | >0.8s |
| RAG Amazon SageMaker | 32.74 | <2.0s |
| Writer Knowledge Graph | 86.31 | <0.6s |

Writer Knowledge Graph receives the highest RobustQA score (>86), with the fastest average response time (<0.6s)

See <https://writer.com/engineering/rag-benchmark/>

Three layers to improve accuracy

Prompt/Code

RAG

LLM

Specialized, domain specific LLMs

Trained on industry-specific data

More control over sensitive information and
reduce the risk of leaks

Less computational power and data storage

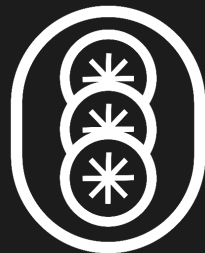
Create and deploy AI solutions much faster

Specialized, domain specific LLMs

- 40% more accurate
- 50% less time to deploy
- 35% cheaper to run



PALMYRA
Medical

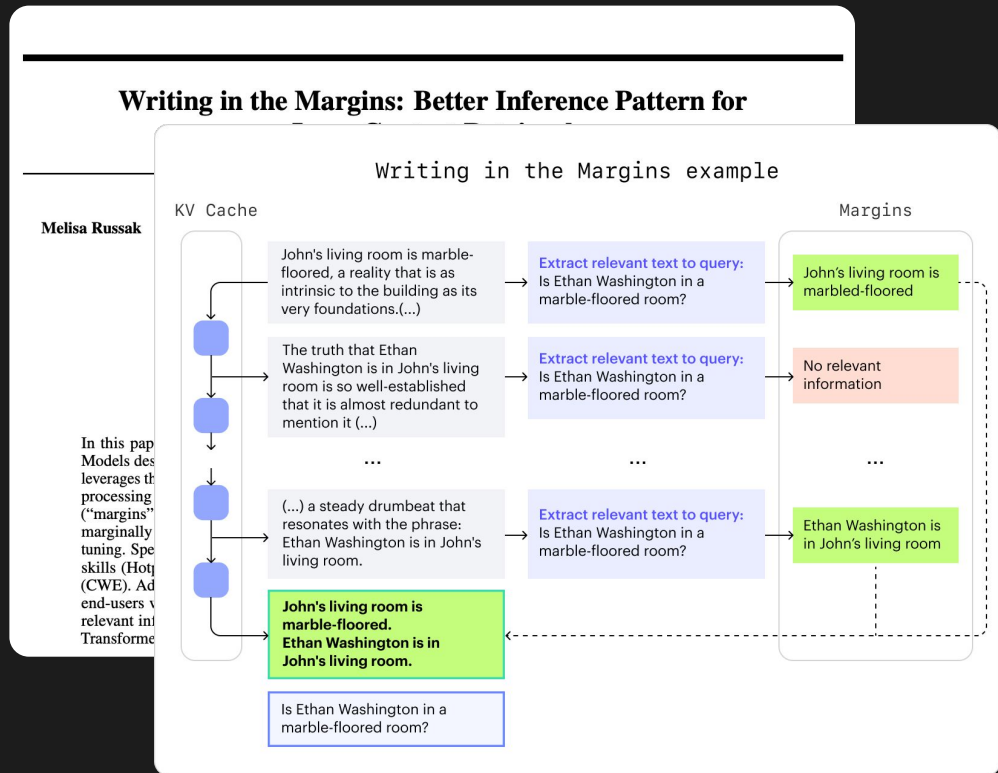


PALMYRA
Finance

Available at <https://huggingface.co/Writer>

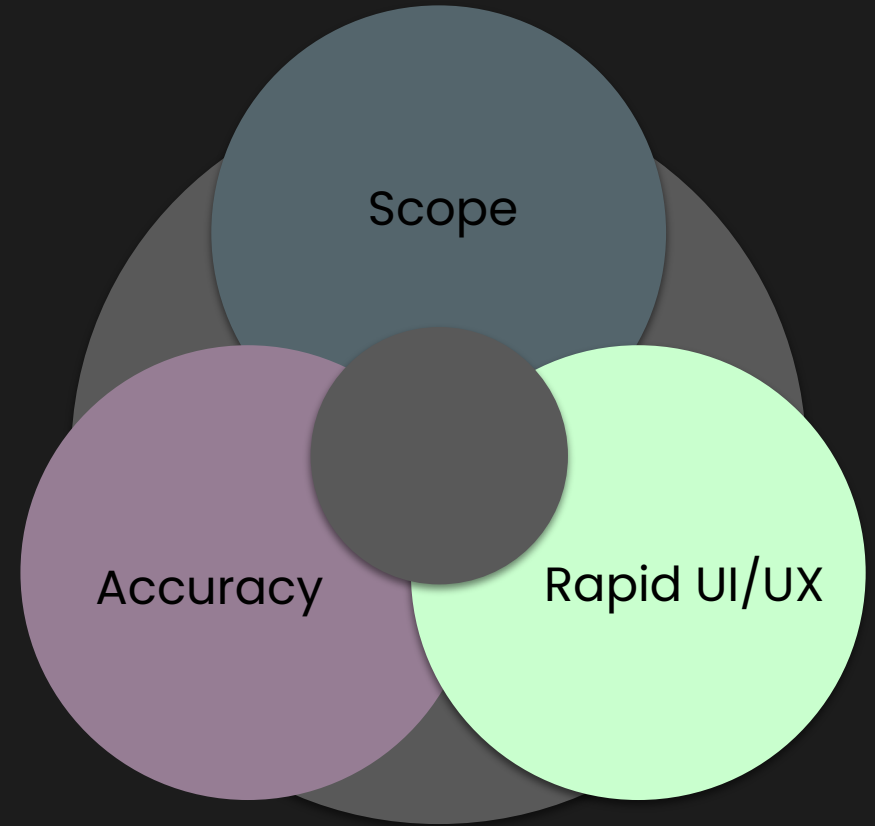
Writing in the Margins (WiM)

- Inference pattern that enhances an LLM's ability to accurately interpret long prompts
- Segments an input sequence into smaller units
- Relevant "margins" are then appended to the prompt
- Results:
 - Average 7.5% improvement in accuracy for reasoning skills
 - Over a 30.0% increase in aggregation tasks when compared to Long Context LLM and RAG approaches.

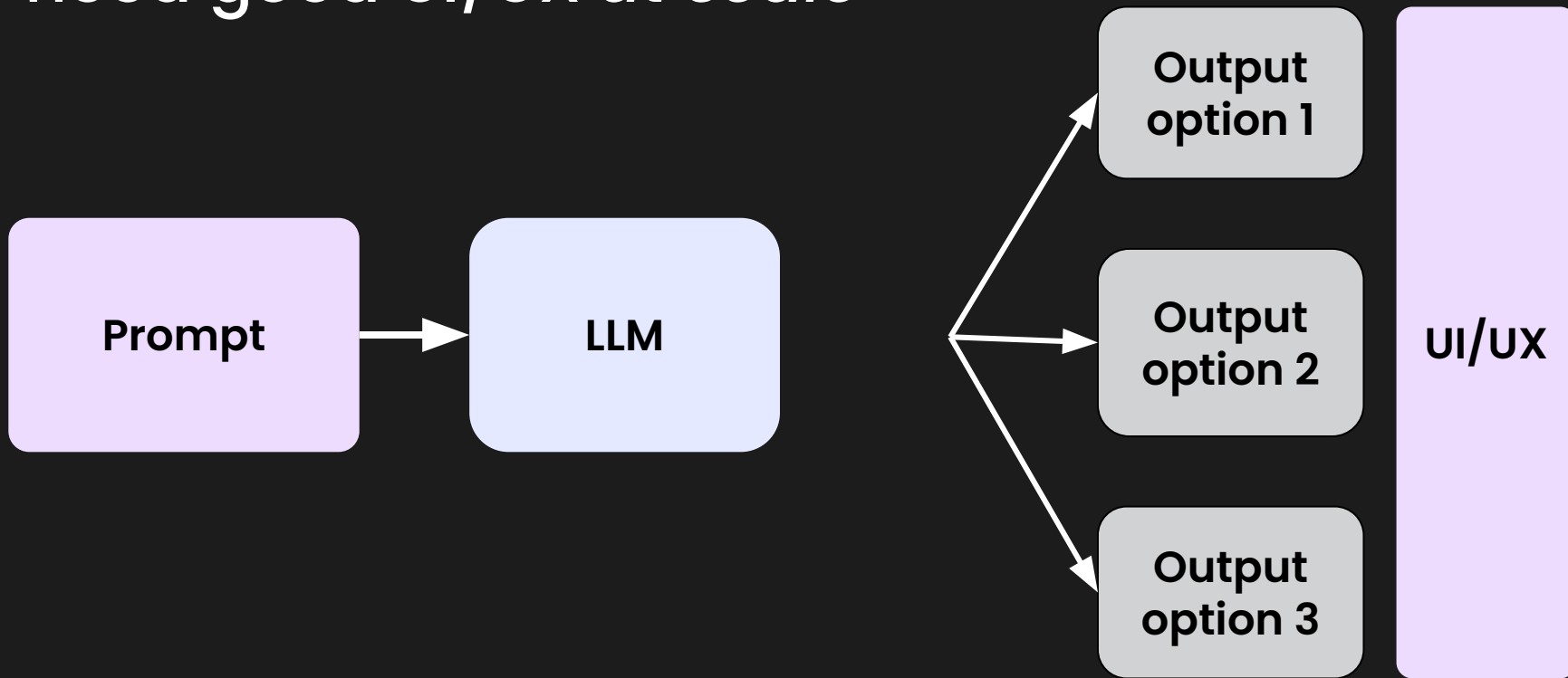


<https://arxiv.org/abs/2408.14906>

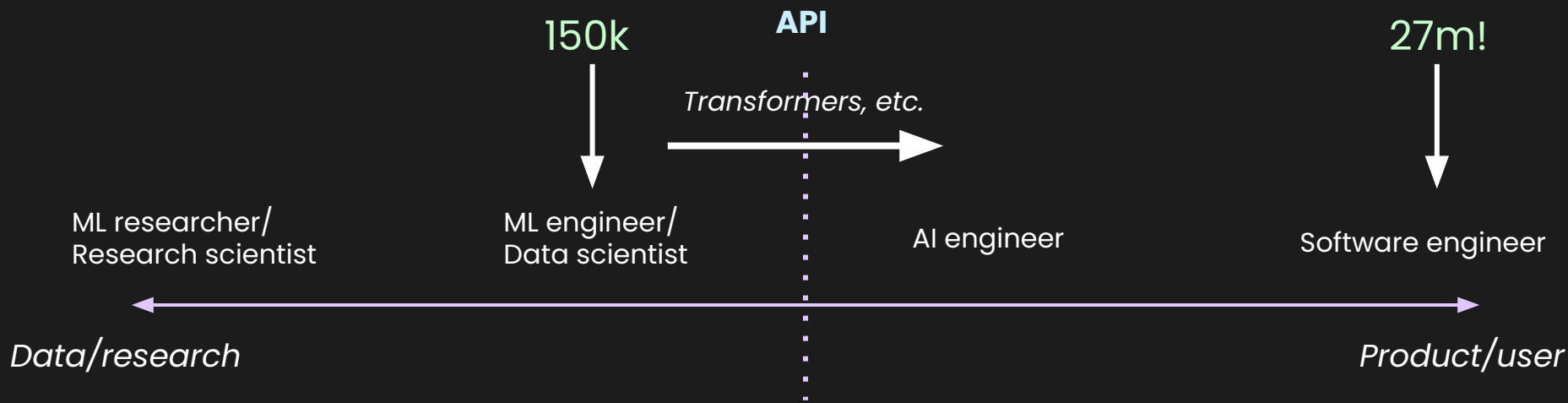
Component 3: Rapid UI/UX



LLMs are non-deterministic...but you still need good UI/UX *at scale*



Wide chasm between skill set of AI/ML engineers and web developers, but we are crossing the divide



The **shift right** caused by GenAI

<https://www.latent.space/p/ai-engineer>

Rapid UI/UX for AI apps

Tried and true technologies

State-driven UI and separation of concerns

Multimodal-first UX

New power tools

For JavaScript: v0.dev

The screenshot shows the v0.dev website interface. At the top left is the v0 logo, and at the top right is a "Sign In" button. The main heading is "What can I help you ship today?" followed by the subtext "Generate UI, ask questions, debug, execute code, and much more." Below this is a large text input field with the placeholder "Ask v0 a question...". To the left of the input field is an icon for attaching files, and to the right is a "Send" button with an upward arrow. Below the input field are three example prompts in rounded buttons: "Generate a multi-step onboarding flow", "How can I structure LLM output?", and "Calculate the factorial of a number". At the bottom of the page is a footer with links for "FAQ", "Terms", "AI Policy", "Privacy", and "Vercel".

v0

Sign In

What can I help you ship today?

Generate UI, ask questions, debug, execute code, and much more.

Ask v0 a question...

Send

Generate a multi-step onboarding flow

How can I structure LLM output?

Calculate the factorial of a number

FAQ | Terms | AI Policy | Privacy | Vercel

For Python: Writer Framework

The image displays the Writer Framework interface, which is a Python-based framework for building web applications. The interface is split into two main sections: a code editor on the right and a preview window on the left.

Code Editor: The code editor shows a Python script that initializes the Writer AI framework and implements a data update function. The code is as follows:

```
1 import writer as wf
2 import writer.ai
3 import pandas as pd
4 from prompts import stock_prompts, income_prompts, earnings_prompt
5 from stock_data import download_data, download_sp500, stock_news, _one
6 from charts import update_scatter_chart
7 from dotenv import load_dotenv
8 import os
9
10 load_dotenv()
11
12 writer.ai.init(os.getenv("WRITER_API_KEY"))
13
14 # Update all data
15 def updates(state):
16     state["message"] = "% Refreshing stock data..."
17     earnings_calls(state)
18     download_sp500(state)
19     stock_news(state)
20     download_data(state)
21     income_statement(state)
22     update_scatter_chart(state)
23     _one_day_data(state)
24     _refresh_window(state)
25
26 # Refresh the window
27 def _refresh_window(state):
28     state["show_income_metrics"]["visible"] = False
29     state["show_bar_graph"]["visible"] = False
30     state["show_analysis_text"]["visible"] = False
31     state["show_analysis_text"]["language"] = False
32     state["message"] = "Writer AI insights will be generated here"
33
```

Preview Window: The preview window shows a web application titled "Finance Research Dashboard". The dashboard has a dark theme and includes the following elements:

- Select a stock ticker:** A section with buttons for AAPL, IBM, NVDA, MSFT, and TSLA.
- Header Text:** A section for displaying header text.
- Investment research options:** A section with two buttons: "Analyze trends" and "Visualize income statement".
- Writer AI insights will be generated here:** A placeholder for AI-generated insights.
- Latest news:** A section displaying a news article titled "Is Apple Siri the Best AI Assistant in 2024?" with a date of "August 28, 2024 at 13:00" and a link to "Insider Monk".

The interface also includes a "Toolkit" on the left side with various components like "Layout", "Sidebar", "Section", "Header", "Column", "Column Container", "Tab", "Tab Container", "Horizontal Stack", and "Separator". A "Component Tree" is also visible, showing the hierarchy of the components in the preview window.

<https://github.com/writer/writer-framework>



- Toolkit
- Layout
 - Sidebar
 - Section
 - Header
 - Column
 - Column Container
 - Tab
 - Tab Container
 - Horizontal stack
- Input
 - Text input
 - Number input
 - Slider input
 - Date input
 - Checkbox input
- Other
 - Button
 - HTML element
 - Pagination
 - Repeater
 - Timer
 - Webcam Capture

Upload a CSV of websites to check

Choose file URL_content_list.csv

Check URLs

- Passive voice
- Wordiness
- Confidence
- Grammar
- Spelling
- Content safeguards
- Compliance guidelines

Website content results

| URL | Text | Description | Suggestion |
|----------------------|-------------------------------------|--|--|
| www.vistafinance.cor | To meet these challenges, invest | Pronoun: Be mindful of the 'royal we'. V | Clarify who 'we' refers to |
| www.vistafinance.cor | Sustainability is Vista Finance stc | Outcome Language: Be mindful not to | Our fiduciary approach to sustainability anc |
| www.vistafinance.cor | The MSCI ESG Quality Score of an | Performance Principle: When we are tc | The MSCI ESG Quality Score of an ETF is ca |
| www.vistafinance.cor | Committed to innovative solution | ESG Language: Use ESG when referring | Committed to innovative solutions for client |
| www.vistafinance.cor | Reduce Exposure to carbon emis | Deprecated Framework: Remove any i | Delete any usage of "Reduce Exposure" |

Text

Text



- Toolkit
- Layout
- Sidebar
- Section
- Header
- Column
- Column Container
- Tab
- Tab Container
- Horizontal stack
- Input
- Text input
- Number input
- Slider input
- Date input
- Checkbox input
- Other
- Button
- HTML element
- Pagination
- Repeater
- Timer
- Webcam Capture

Upload a CSV of websites to check

Choose file URL_content_list.csv

Check URLs

- Passive voice
- Wordiness
- Confidence
- Grammar
- Spelling
- Content safeguards
- Compliance guidelines

Website content results

| URL | Text | Description | Suggestion |
|----------------------|-------------------------------------|--|--|
| www.vistafinance.com | To meet these challenges, invest | Pronoun: Be mindful of the 'royal we'. | Clarify who 'we' refers to |
| www.vistafinance.com | Sustainability is Vista Finance stc | Outcome Language: Be mindful not to | Our fiduciary approach to sustainability anc |
| www.vistafinance.com | The MSCI ESG Quality Score of an | Performance Principle: When we are tr | The MSCI ESG Quality Score of an ETF is ca |
| www.vistafinance.com | Committed to innovative solution | ESG Language: Use ESG when referring | Committed to innovative solutions for client |
| www.vistafinance.com | Reduce Exposure to carbon emis | Deprecated Framework: Remove any i | Delete any usage of "Reduce Exposure" |

Analyze results

Analysis

Header



Properties

General

Text : Text

Analysis

Style

Primary text : Color

Default

CSS

Pick

Custom SS classes : Text

CSS classes, separated by spaces. You can define classes in custom stylesheets.

Events

+ Add custom handler

Visibility

Yes

No

Custom

Let's recap

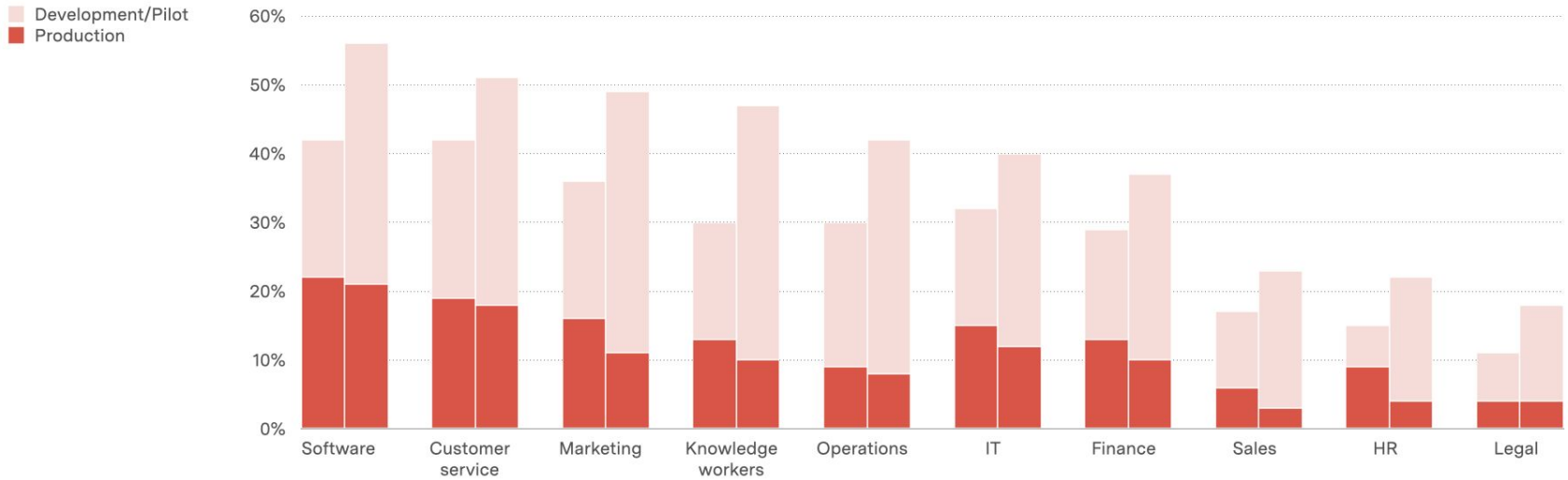


AI can become a gravity well for enterprises.

Huge interest, limited use so far

Enterprise software takes time, and come with early disappointments

Enterprise use case adoption rates for generative AI, October 2023 & February 2024

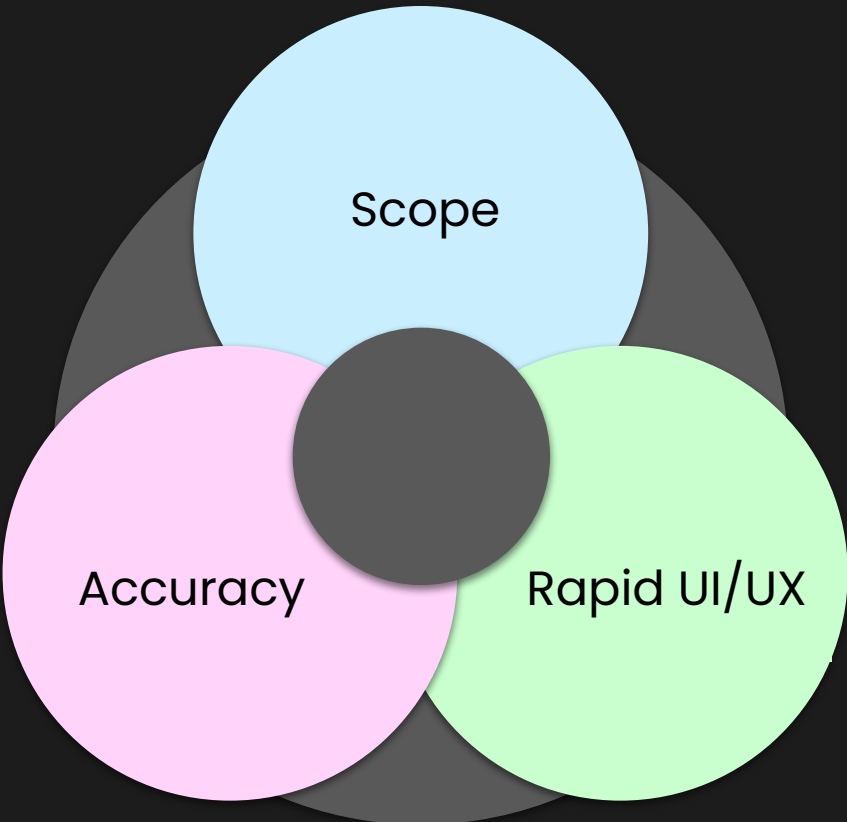


Source: Bain

Benedict Evans — July 2024

5

Enterprise AI delta-v



First, think microservices, not monoliths.

IT, finance, HR

Incidence reports

Training materials

Investor relations materials

Supply chain analysis

Competitor analysis

Report deep dives

Job descriptions

L&D content

Financial summaries

Product + design

Parsing user research

Push notifications

Error messages

Release notes

UX copy

Internal messages

Zero-state pages

User stories

Onboarding material

Marketing, Comms, PR

Blog posts

Landing pages

Case studies

Client stories

Messaging ideas

SEO optimization

White papers

Newsletters

Social media campaigns

Sales + support

Sales enablement assistants

RFPs

Expert agent assistants

Clinical analysis

Support call summaries

Coverage descriptions

Compliance

Client comms

Knowledgebases

Second, aim for a full-stack approach.

Here's how we approach it at Writer for customers:



Prebuilt apps

Customize, integrate,
and manage OOTB apps



AI Studio

Use a platform to build AI apps for
your business

Platform-as-a-service

LLMs

RAG

AI guardrails

Three layers to improve accuracy

Prompt/Code

RAG

LLM

Three layers to improve accuracy

Function calling

Graph RAG + research

Specialized LLMs, WiM

Rapid UI/UX for AI apps

Tried and true technologies

State-driven UI and separation of concerns

Multimodal-first UX

New power tools

For Python: Writer Framework

The image displays the Writer Framework interface, which is a Python-based web application framework. The interface is split into two main sections: a code editor on the right and a preview window on the left.

Code Editor: The code editor shows a Python script for a "Finance Research Dashboard". The code includes imports for the writer framework, pandas, and various data fetching and charting libraries. It defines a state object and a function to update the state with stock data and earnings calls. The code also includes a function to refresh the window and hide certain UI elements.

```
1 import writer as wf
2 import writer.ai
3 import pandas as pd
4 from prompts import stock_prompts, income_prompts, earnings_prompt
5 from stock_data import download_data, download_sp500, stock_news, _one
6 from charts import update_scatter_chart
7 from dotenv import load_dotenv
8 import os
9
10 load_dotenv()
11
12 writer.ai.init(os.getenv("WRITER_API_KEY"))
13
14 # Update all data
15 def updates(state):
16     state["message"] = "% Refreshing stock data..."
17     earnings_calls(state)
18     download_sp500(state)
19     stock_news(state)
20     download_data(state)
21     income_statement(state)
22     update_scatter_chart(state)
23     _one_day_data(state)
24     _refresh_window(state)
25
26 # Refresh the window
27 def _refresh_window(state):
28     state["show_income_metrics"]["visible"] = False
29     state["show_bar_graph"]["visible"] = False
30     state["show_analysis_text"]["visible"] = False
31     state["show_analysis_text"]["language"] = False
32     state["message"] = "Writer AI insights will be generated here"
33
```

Preview Window: The preview window shows a web application titled "Finance Research Dashboard". It features a "Select a stock ticker" section with buttons for AAPL, IBM, NVDA, MSFT, and TSLA. Below this is a "Header Text" section. The main content area is titled "Investment research options" and contains two buttons: "Analyze trends" and "Visualize income statement". A message box below the buttons says "Writer AI insights will be generated here". The bottom section is titled "Latest news" and displays a news article snippet: "Is Apple Siri the Best AI Assistant in 2024?" dated August 28, 2024, at 13:00, by Insider Monk.

<https://github.com/writer/writer-framework>

WRITER

The only fully-integrated solution for building AI apps with faster time to value

AI digital assistants

AI workflows

AI apps



Palmyra LLMs

State-of-the-art top-benchmarked Writer-built LLMs that are compliant, efficient, and transparent.



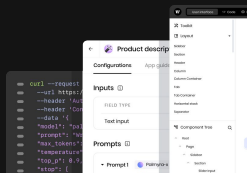
Knowledge Graph

Connects your AI apps to structured and unstructured data with graph-based RAG.



AI guardrails

Enforce regulatory, accuracy, and brand compliance across all AI apps and workflows.



AI Studio and integrations

Code (for developers) and no-code (for business users) tools for quickly building production-grade AI apps.



AI program management

Drive change management, workflow by workflow, to realize business value.

WRITER

INTUIT

L'ORÉAL

accenture

T Mobile

 Dropbox


Qualcomm

Vanguard®

covermymeds®

COMMVault 

 kenvue

 Northwestern
Mutual

 salesforce

Anaplan

 Spotify

 McAfee

 Pinterest

 sense

servicenow

VICTORIA'S SECRET

HubSpot



Thank you!
Questions? @samjulien

WRITER