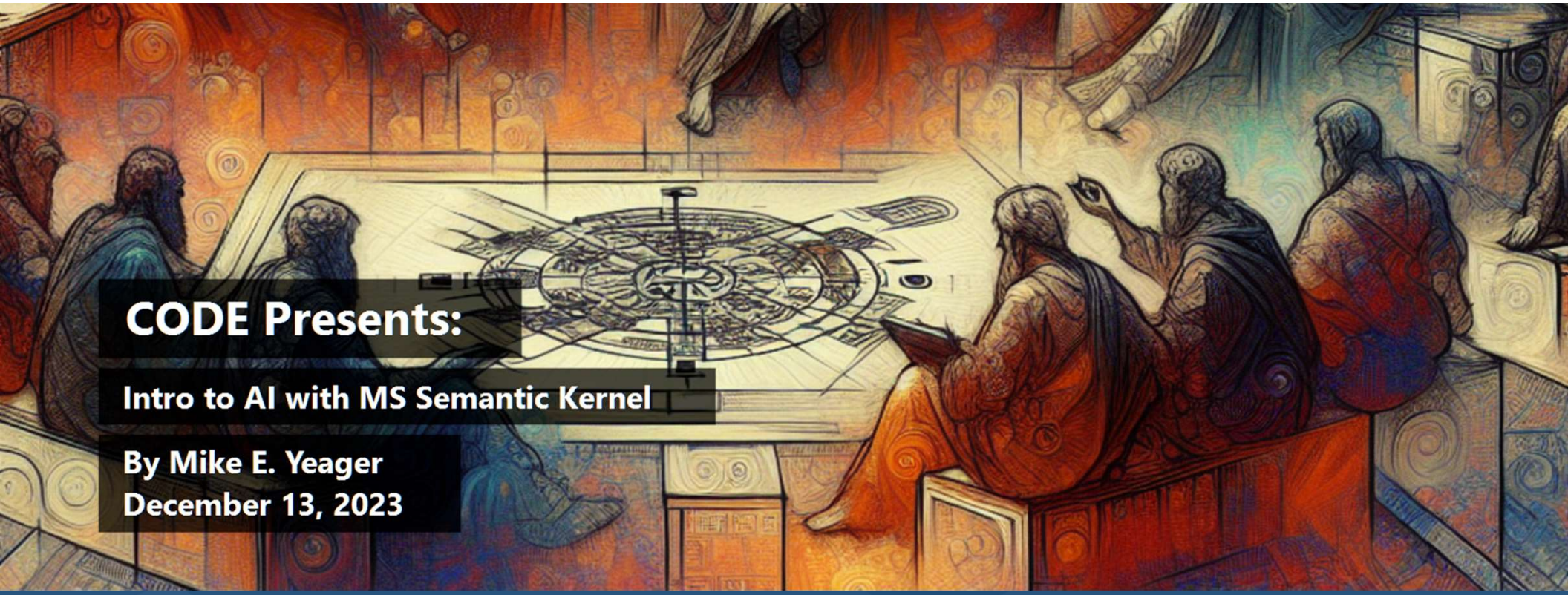


Mike Yeager
CEO of CODE Consulting

CODE CONSULTING CODE STAFFING CODE MAGAZINE CODE TRAINING

About Mike Yeager

CEO of CODE Consulting

20+ years of professional experience developing software for different sectors

Author

Speaker

myeager@codemag.com

30 Years of

"Helping People Build Better Software"

About CODE Consulting

CODE CONSULTING

Microsoft

eog resources

Gulfstream
A GENERAL DYNAMICS COMPANY

BRIDGESTONE

wisetech global

Solera

BLUE BELL ICE CREAM EST 1907

- Custom software development (web, mobile, desktop and cloud platform apps)
- Copilot development, AI, GPT, Azure OpenAI, machine learning, and more…
- Project rescue (cloud, web, mobile, windows,…)
- App modernization (primarily Visual FoxPro)
- Support & maintenance for existing applications

AI Consulting Services

CODE CONSULTING

Check out our new **Executive Briefing** offer!

We can help with your AI needs

What does AI mean for you?

“Skunk Works” Projects

codemag.com/AI

codemag.com/ExecutiveBriefing

CODE Staffing

Disrupting the world of staffing!

Giving our customers the ability to have staff on par with Silicon Valley companies ...

... and our employees a work environment in a bleeding-edge tech company with the **industry leading benefits!**

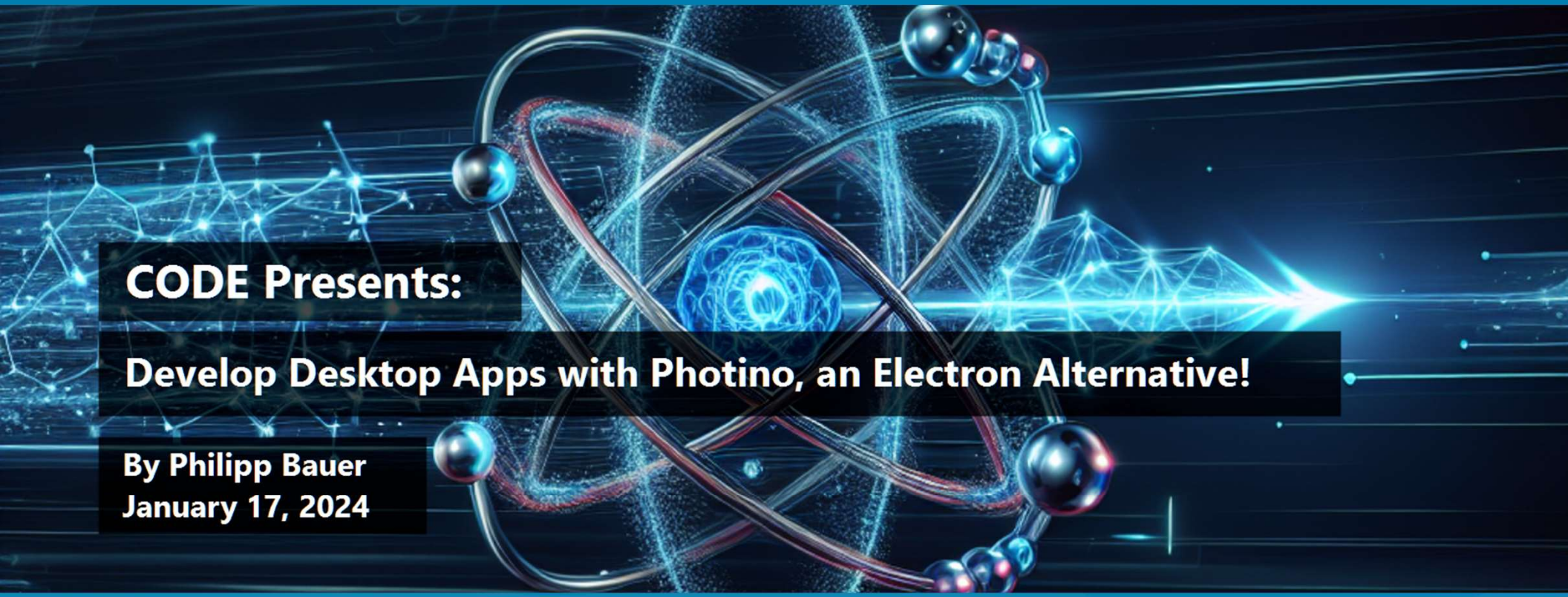
codestaffing.com

CODE STAFFING

WE NEED YOU!

codestaffing.com/careers

CODE TRAINING



CODE Presents:

Develop Desktop Apps with Photino, an Electron Alternative!

By Philipp Bauer
January 17, 2024

CODE TRAINING

CODE Training presents:

Introduction to Development with (Azure) OpenAI, GPT, DALL-E, and more...

By Markus Egger

January 30, 2024 - Online & In-Person

CODE TRAINING

State of .NET:

State of AI 2024

By Markus Egger

February 28, 2024 - Online

Free CODE Magazine Subscription

CODE MAGAZINE

The leading software development magazine, written by expert developers for developers.

Free subscription for all of you
(and your friends)

Subscribers get CODE Focus issues free of charge!

Share this link to our free subscription:
bit.ly/CP121323Subscription

Agenda

CODE PRESENTS

What is Semantic Kernel?

Why a framework?

Which frameworks do I care about?

What’s in these frameworks?

Introduction to Semantic Kernel – Coding!

What is Semantic Kernel?

CODE PRESENTS

Semantic Kernel is an open-source framework, created by Microsoft for working with LLMs.

Semantic Kernel is at the center of the copilot stack.

Why a framework?

CODE PRESENTS

To handle more complex implementations in a generic way

To abstract the underlying LLMs, APIs and tooling

Which frameworks to care about?

CODE PRESENTS

LangChain is the most popular LLM framework

- https://docs.langchain.com/docs/
- Implemented in Python and JavaScript/TypeScript
- Powerful and mature

Semantic Kernel is Microsoft's LLM framework

- https://learn.microsoft.com/en-us/semantic-kernel/
- https://github.com/microsoft/semantic-kernel
- Implemented in .NET (C#), Python and Java
- Incomplete, but moving fast

What’s in these frameworks?

CODE PRESENTS

Create a library of prompts in a generic way (templating)

Create and use large contexts (grounding, rag)

Integrate with external tools and systems (connectors/agents/plugins)

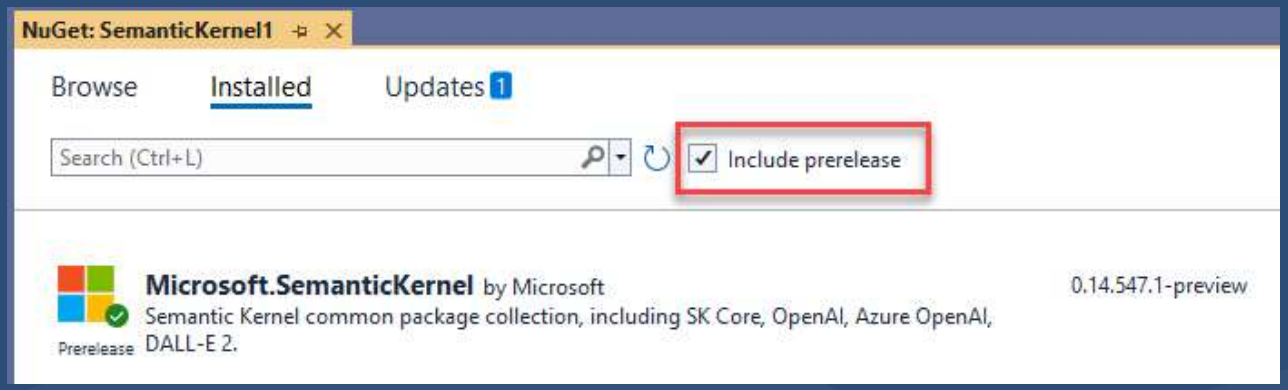
Utilities like text splitters & chunkers

Plan or “chain” prompts together to achieve a goal

Introduction to Semantic Kernel

CODE PRESENTS

It’s a NuGet package



https://github.com/microsoft/semantic-kernel

- Jupyter Notebook samples
- .NET and React! samples

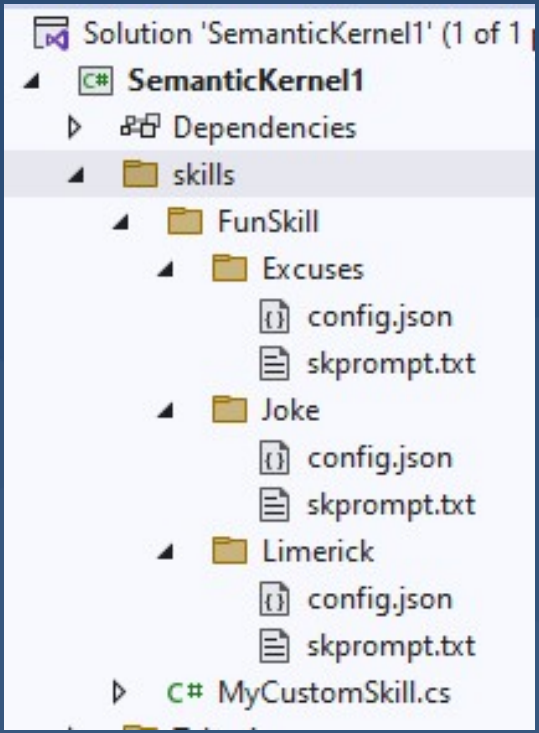
Functions ~~and Skills~~

CODE PRESENTS

A Function is a prompt, templated prompt or native code

- Semantic Function
  - Inline
  - External
- Native Function

~~A Skill is a set of Functions~~



Semantic Functions (Templated Prompts 1 of 2)

CODE PRESENTS

Text – like a mail merge document (skprompt.txt)

skprompt.txt | Program.cs

```
Generate a creative reason or excuse for the given event. Be
  creative and be funny. Let your imagination run wild.

Event:I am running late.
Excuse:I was being held ransom by giraffe gangsters.

Event:{{$input}}
```

Semantic Functions (Settings 2 of 2)

CODE PRESENTS

Settings for running the prompt (config.json)

config.json

Schema: https://json.schemastore.org/config.json

```
1  {
2    "schema": 1,
3    "description": "Turn a scenario into a creative or humorous excuse to send your boss"
4    "type": "completion",
5    "completion": {
6      "max_tokens": 60,
7      "temperature": 0.5,
8      "top_p": 0.0,
9      "presence_penalty": 0.0,
10     "frequency_penalty": 0.0
11   }
12 }
```

Native Function (C# Code)

String as input

String as output

MyCustomSkill.cs

SemanticKernel1 | SemanticKernel1.skills.MyCSharpSkill

```
using Microsoft.SemanticKernel.SkillDefinition;

namespace SemanticKernel1.skills;

0 references
public class MyCSharpSkill
{
    [SKFunction("Return the first row of a qwerty keyboard")]
    0 references
    public string Qwerty(string input)
    {
        return "qwertyuiop";
    }

    [SKFunction("Return a string that's duplicated")]
    0 references
    public string DupDup(string text)
    {
        return text + text;
    }
}
```

Pipelines (Chaining)

CODE PRESENTS

String as input

String as output

Default parameter name for templated functions: {{$input}}

“Caught in a whirlpool on the way to work my speedboat was!”

Function Examples

CODE PRESENTS

Inline Functions

Basic Templated Functions

Native Functions

Pipelines (Chaining)

Memories

CODE PRESENTS

Memories are a powerful way to provide broader context for your ask

- a. k. a. How to get around prompt size limits & ground your models

Types of Memories

- Key-Value pairs
- Local Storage
- Semantic Memories (Embeddings)

Memory Examples

CODE PRESENTS

Key-value pairs

Embeddings (vector databases)

Utilities (chunkers)

Plugins

CODE PRESENTS

3rd party functions give access to their systems & features

Added to catalogs with descriptions for you to use

E.g. Microsoft365, Microsoft Graph, Bing, Google, Slack

Plugins are just groups of native and/or sematic functions exposed by creating a JSON manifest following the OpenAI plugin specification

https://platform.openai.com/docs/plugins/getting-started

Plugins

CODE PRESENTS

OpenAI Assistants are Plugins in SK

Access Ollama (local run LLMS)

Planner

There are actually a few planners:

- ActionPlanner (single step)
- SequentialPlanner
- StepwisePlanner (incremental planning)

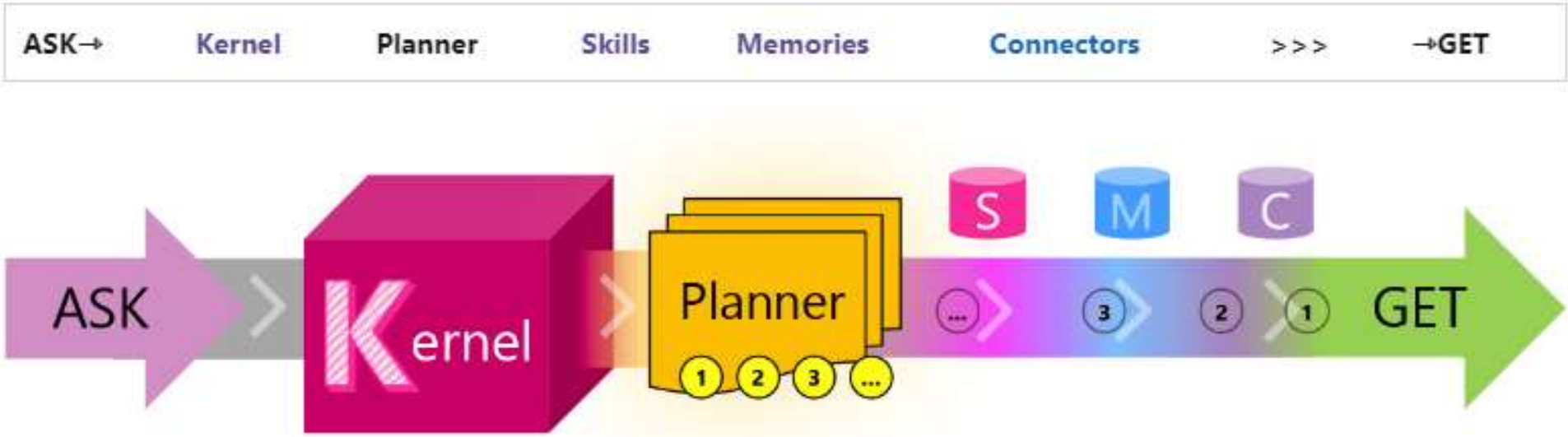
Planner

Because the planner has access to either a pre-defined library of pre-made skills and/or a dynamically defined set of skills it is able to fulfill an ASK with confidence. In addition, the planner calls upon memories to best situate the ASK's context and plugins to call APIs and to leverage other external capabilities.

Planner

The planner will operate within the skills it has available. In the event that a desired skill does not exist, the planner can suggest you create the skill. Or, depending upon the level of complexity the kernel can help you write the missing skill.

Planner



The *planner* works backwards from a goal that's provided from a user's *ASK*.

Planner Example

RECAP...

CODE PRESENTS

Create a library of prompts in a generic way (templating)

FUNCTIONS

Create and use large contexts (grounding)

MEMORY

Integrate with external tools and systems

PLUGINS

Plan or “chain” prompts together to achieve a goal

PLANNERS

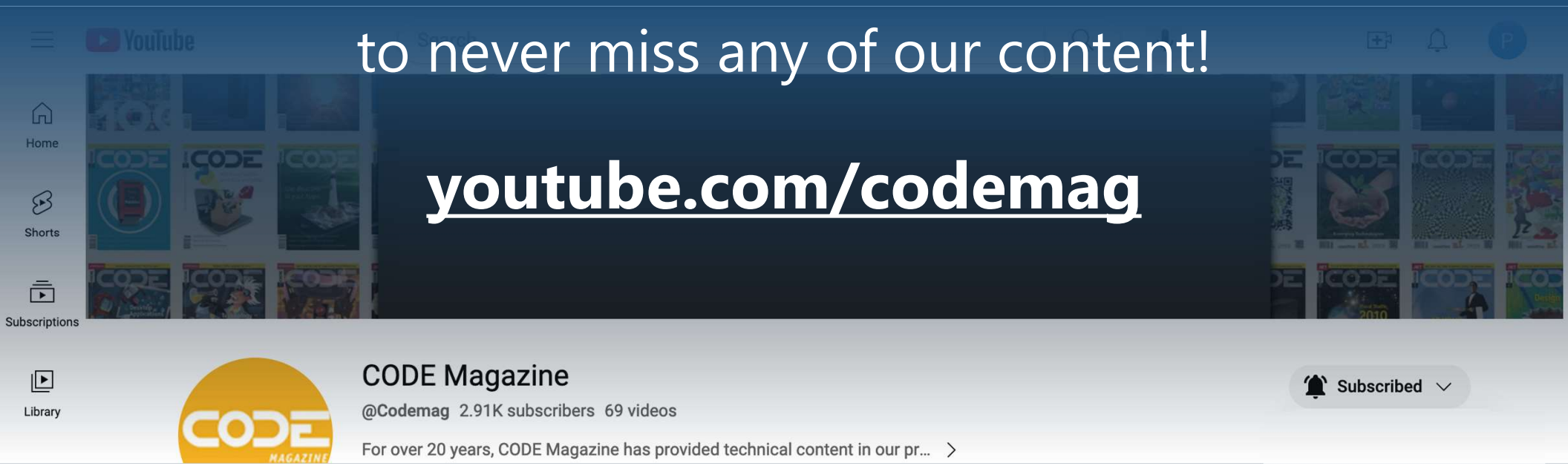
CODE PRESENTS

YouTube

Subscribe and “ring that bell”

to never miss any of our content!

youtube.com/codemag



Recordings & Slide Decks

CODE PRESENTS

State of .NET Webinar Series

codemag.com/StateOfDotNet

CODE Presents Webinar Series

codemag.com/CODEPresents

CODE PRESENTS

Q & A

Contact us with questions!

CODE/EPS Contact
codemag.com
info@codemag.com
facebook.com/codemag
twitter.com/codemagazine

Presenter Contact
myeager@codemag.com