Lessons Learned on Applied Advanced Agentic AI

Jose Luis Latorre

Lessons Learned on Applied Advanced Agentic AI

José Luis Latorre

Developer Community Lead

Swiss Life

LinkedIn Learning

MVP Microsoft Most Valuable Professional

.NET User Group Zürich

Thank you to our Sponsors !

Global AI Bootcamp 2025
Barcelona
Spain

Platinum Sponsor

algoritmia
powered by Alba

Bronze Sponsors

AvePoint
encamina
PIENSA EN COLORES
isolutions
V-Valley
enhancing your business

Collaborators

HARBOUR.SPACE
Microsoft

Organized by
sug.cat

Agenda

- Agents
- Agentic AI
- Extending Semantic Kernel
- The Commanding Pattern
- Consciousness Prompting

Agents!
Planner
tool
Persona
Single Eye
Use Tools
Use Tools
Autonomously AI Actions
Adachanot AI Technology
kuando

Usual usage vs Agentic

Usual Workflow

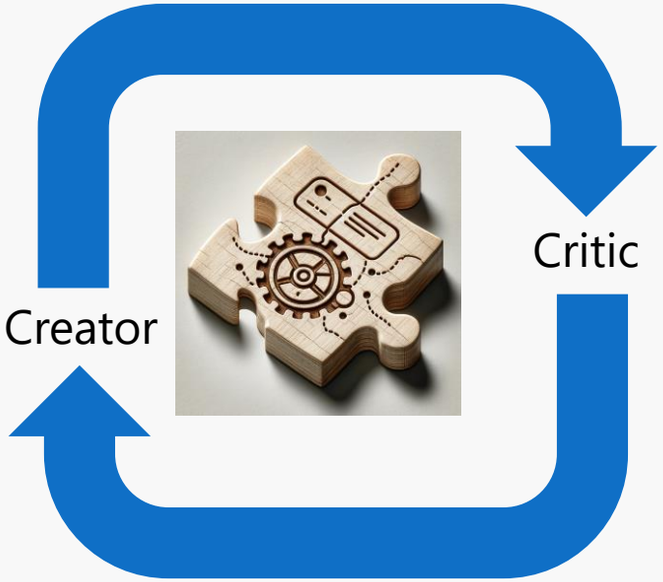
Please write me a blog post entry about .NET day for LinkedIn in one go.

Start

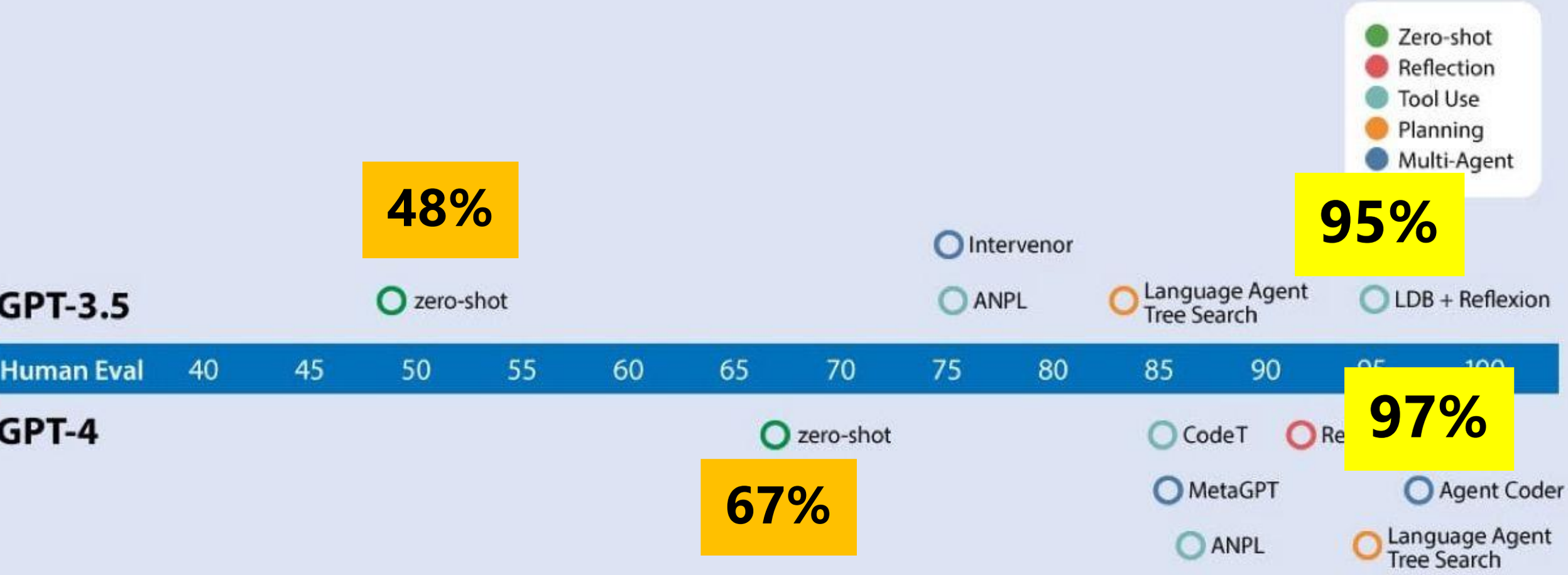
Finish

Agentic Workflow

- Please write me a blog post entry about .NET day
- Need more information.
- Write the starting draft
- Consider what areas need revision
- Review for style
- Etc…



GPT-3.5 and GPT-4 performance using zero-shot and agent workflows



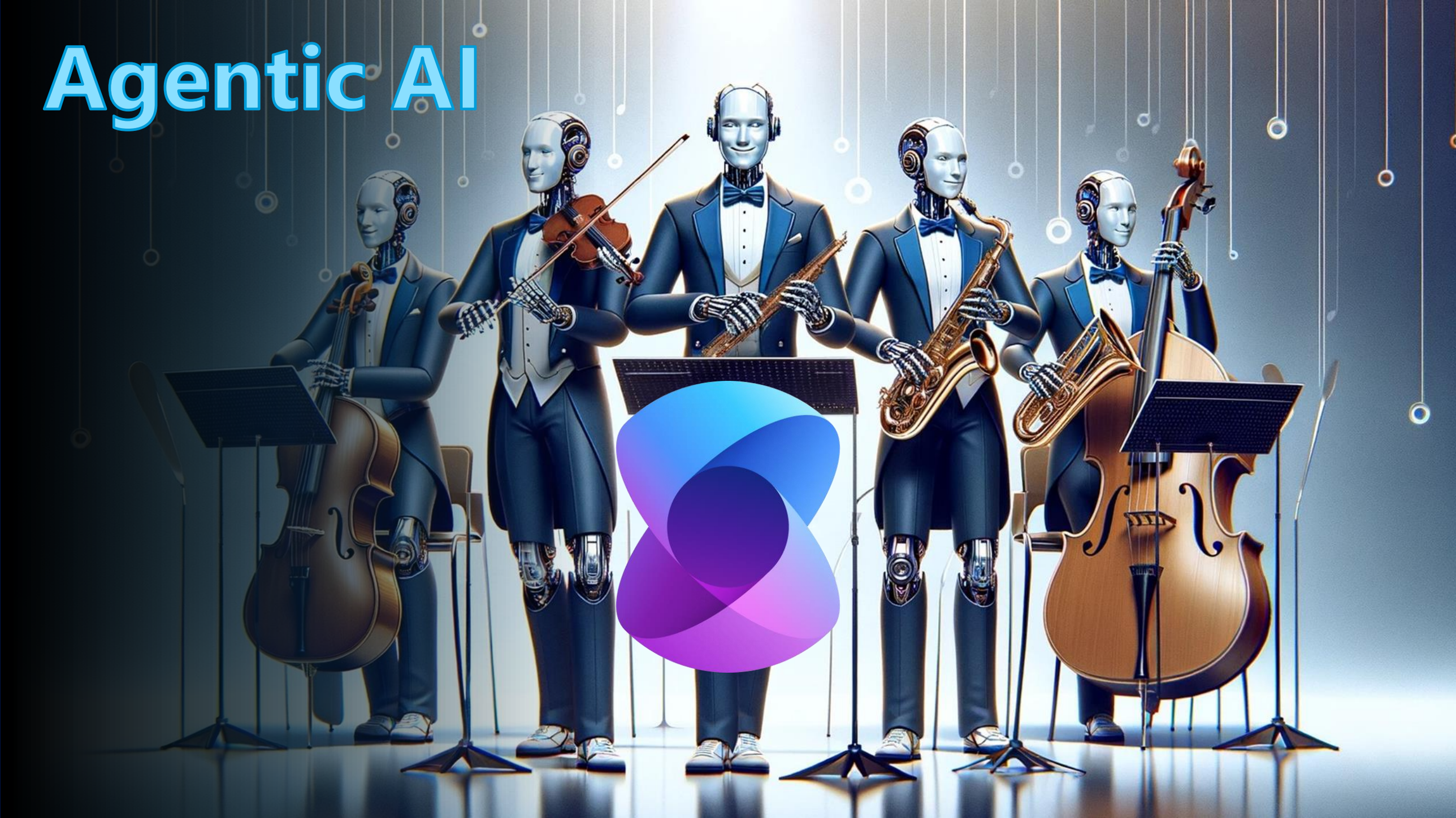
Source: Andy Ng

Performance of GPT-3.5 and GPT-4 (zero-shot) on HumanEval, along with algorithms that use agent workflows on top of GPT-3.5 or GPT-4. Thanks to Joaquin Dominguez and John Santerre for help with this analysis.

Altera.ai 1.000 Agents “group Chat”

https://youtu.be/2tbaCn0Kl90?si=ioiOSY5GlI2z5Pm5
https://arxiv.org/abs/2411.10109

Agentic AI



TO DO
DO
TO DO.
TO DO.
TO DO
TO DO
TO DO
TO DO
AI
AI
TO DO

Enter The “Group” Chat…

Ĥuṇǎŋ

Group Chat

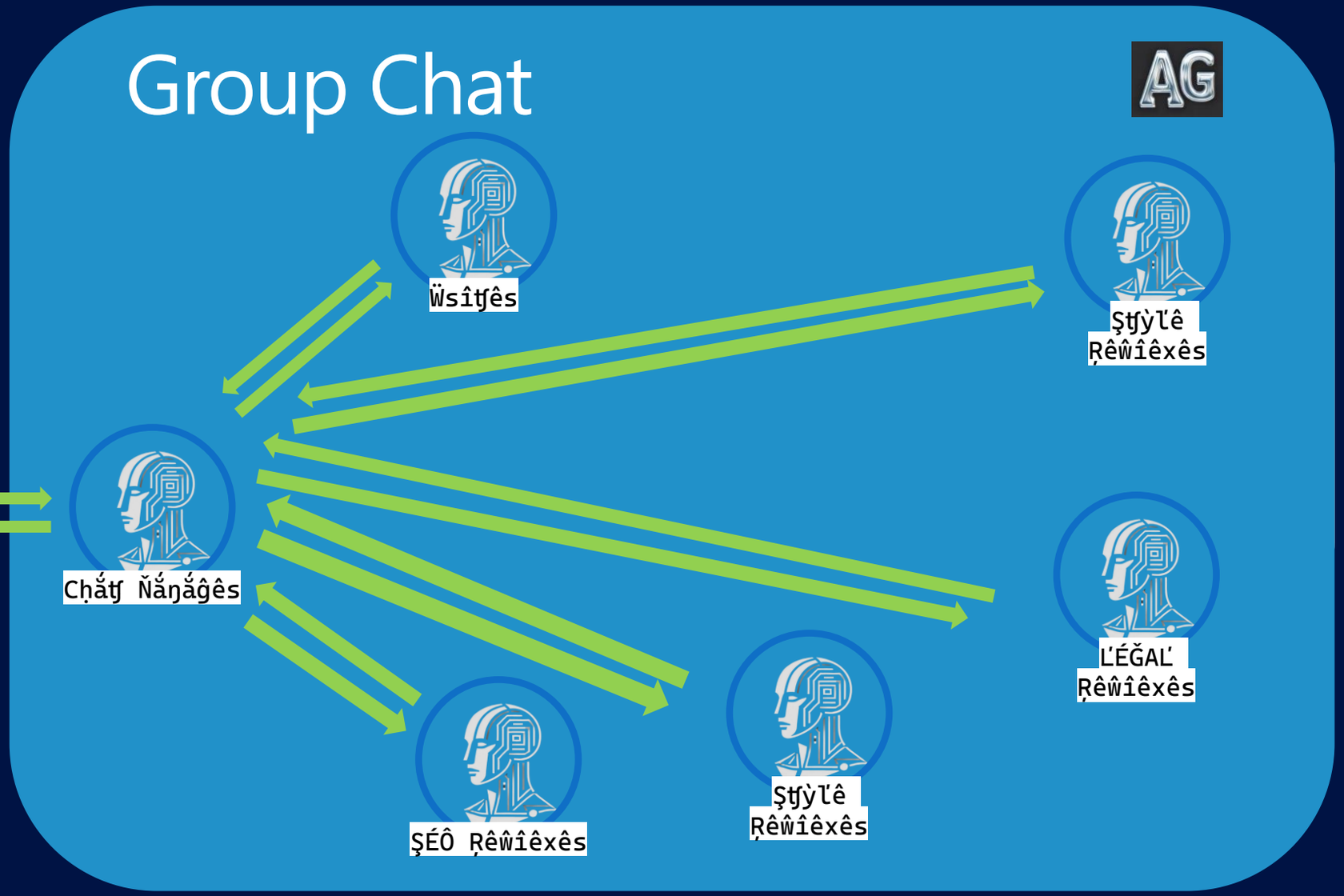
Ẅsîţês

N-times

Ş̌ţŷ̀ľê Ṛêŵîêxês

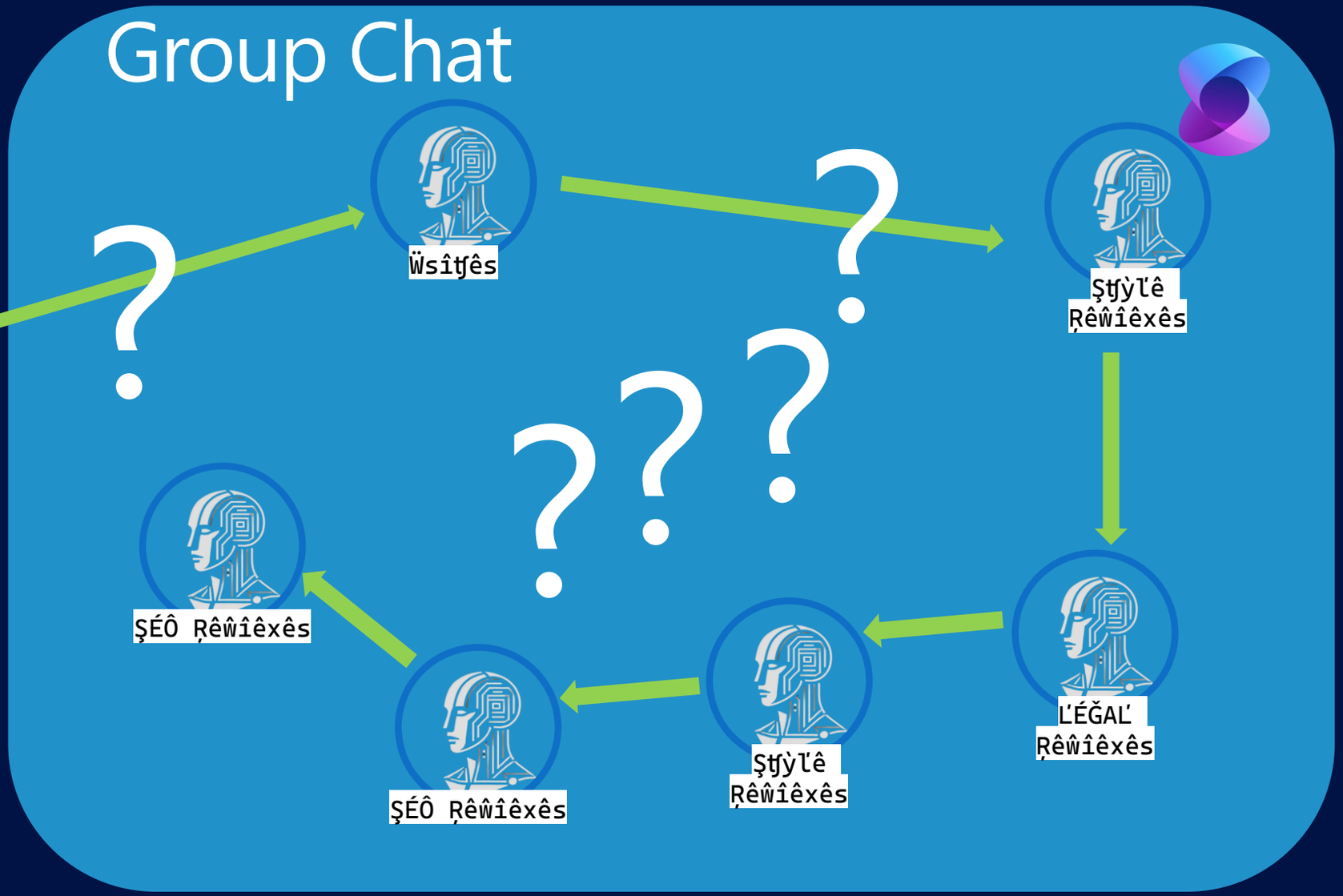
I Approve!!

The Inspiration...



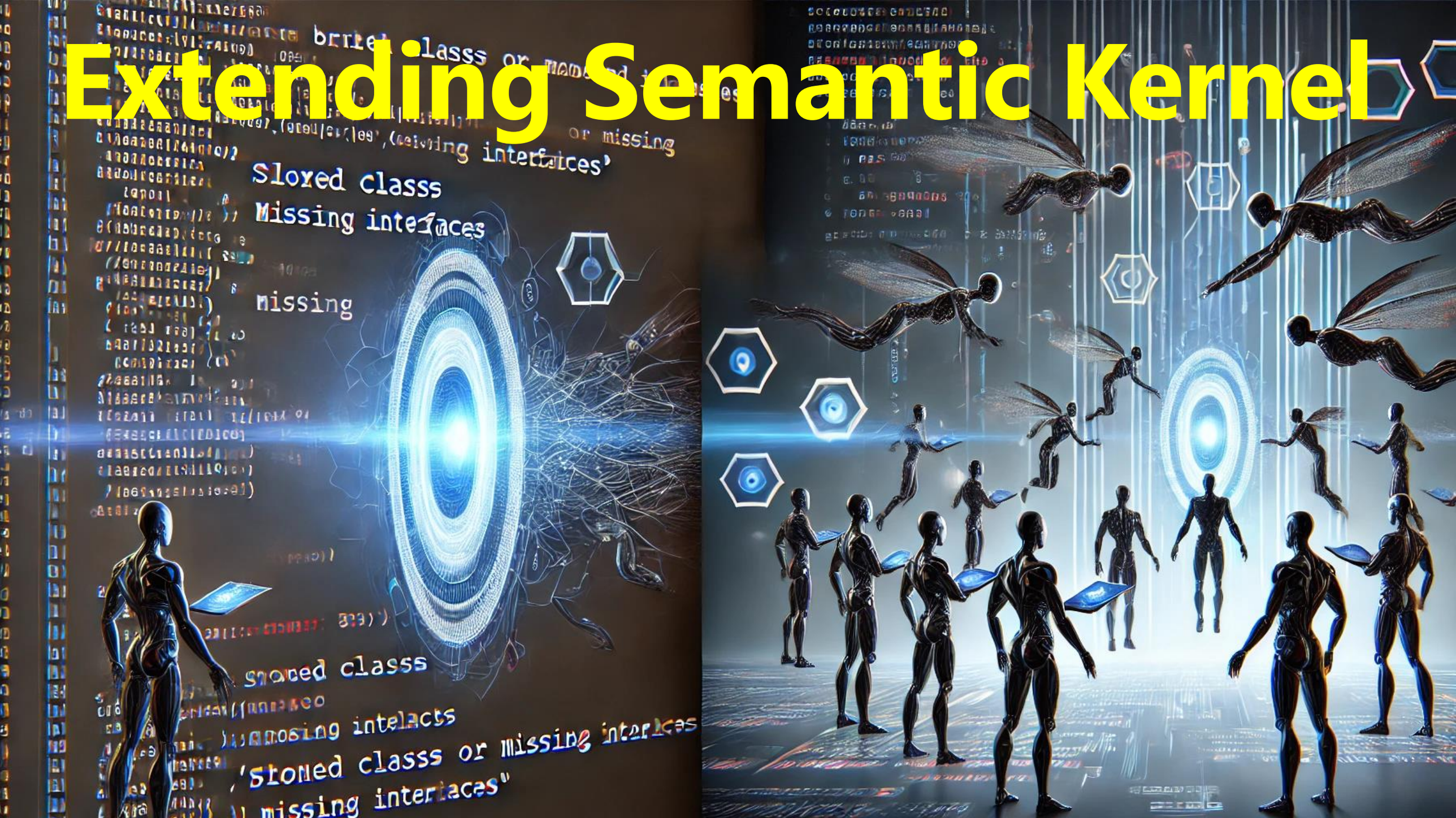
But..

Ĥụṇǎŋ


Group Chat
Ẅsîţŷês
Şţŷỳľê Ŗêŵîêxês
ĽÉĞAĽ Ŗêŵîêxês
Şţŷỳľê Ŗêŵîêxês
ŞÉÔ Ŗêŵîêxês
ŞÉÔ Ŗêŵîêxês

AutoGen's Two Agent Chat

Extending Semantic Kernel



is:issue author:joslat

Labels | Milestones | New issue

Open 8 | Closed 15

Author | Labels | Projects | Milestones | Assignees | Types | Sort

- .Net: Enhance Extensibility of Agent Classes by Unsealing and Implementing Interfaces (.NET, Agents) [.NET] [agents] [enhancement] [follow up] [wontfix] — 6
  #8719 · joslat opened on Sep 14
- New Feature: Agents story - EPIC - Enhance Base Agent & chat workflows Functionality [agents] [duplicate] — 2
  #7248 · joslat opened on Jul 12
- New Feature: Agents story - Containerization of common frameworks (or custom code) as Plugins [enhancement] — 1
  #7247 · joslat opened on Jul 12
- New Feature: Agents story - Collaboration Model with other Agentic Frameworks [agents] [enhancement] [wontfix] — 1
  #7246 · joslat opened on Jul 12
- New Feature: Agents story - Tighten collaboration with the AutoGen Team [agents] [enhancement] [wontfix] — 2
  #7245 · joslat opened on Jul 12
- New Feature: Agents story - Remove "Experimental" Tag [agents] [enhancement] [feature_graduation] [stale] — 1
  #7244 · joslat opened on Jul 12
- New Feature: Agents story - Deployable/Containerized Code Executor(s) [enhancement] — 0
  #7243 · joslat opened on Jul 12
- New Feature: Agents story - Develop "Agent" and "Chat Workflow" as Graphs [agents] [enhancement] [processes] — 1
  #7242 · joslat opened on Jul 12
- New Feature: Agents story - Enable Nested Chats [agents] [enhancement] [processes] — 1
  #7241 · joslat opened on Jul 12
- New Feature: Agents story - Support Different Chat Models [agents] [enhancement] [processes] — 0
  #7240 · joslat opened on Jul 12
- New Feature: Agents story - Summarization/processing of Agent Chat Output [agents] [enhancement] — 1
  #7239 · joslat opened on Jul 12
- New Feature: Agents story - Implement Parallel/Async Execution [agents] [enhancement] [processes] — 0
  #7238 · joslat opened on Jul 12

m?

to search

2024
2023
2022
2021
2020

Aug Sep Oct

Less More

Oct 5

joslat / SemanticKernelAgenticChatWorkflowsExtension

Type / to search

Code | Issues | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

SemanticKernelAgenticChatWorkflowsExtension Public

Pin | Unwatch 1 | Fork 0 | Starred 9

github.com/joslat/SemanticKernelAgenticChatWorkflowsExtension

joslat minor adjustments c0ffef6 · 3 weeks ago 9 Commits

AgenticChatWorkflows	minor update	3 weeks ago
AutoGen	minor adjustments	3 weeks ago
Helpers	AutoGen integration with Semantic Kernel :)	3 weeks ago
SemanticKernel	checkin of code.	last month
.gitattributes	checkin of code.	last month
.gitignore	Initial commit	last month
AgenticChatWorkflows.csproj	AutoGen integration with Semantic Kernel :)	3 weeks ago
AgenticChatWorkflows.sln	checkin of code.	last month
AgenticWorkflowModel.cs	AutoGen integration with Semantic Kernel :)	3 weeks ago
App.xaml	checkin of code.	last month
App.xaml.cs	checkin of code.	last month
ApprovalTerminationStrategy.cs	checkin of code.	last month
AssemblyInfo.cs	checkin of code.	last month
LICENSE	Initial commit	last month
LICENSE.txt	checkin of code.	last month

Playing around with extending Semantic Kernel and implementing several agentic chat workflow patterns

Readme

MIT, MIT licenses found

Activity

9 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

C# 100.0%

Suggested workflows

Based on your tech stack

Key Components

- **IAgentChat -** basic chat functionalities
- **IAgentGroupChat-** Extends IAgentChat to support group chat features, including managing multiple agents and chat history
- **AgentGroupChatExt -** A concrete implementation of IAgentGroupChat that wraps an underlying AgentGroupChat instance from the Semantic Kernel framework
- **BaseAgentGroupChat -** An abstract class implementing IAgentGroupChat, serving as a base for custom chat workflows
- **TwoAgentChat**, **SequentialAgentChat**, **SequentialTwoAgentChat**: Concrete classes extending BaseAgentGroupChat to implement specific chat workflows involving agents interacting in predefined patterns.

The Commanding Pattern

The Commanding Pattern

Cộṇṇắŋđês

Plugin: Critic

Csîţĵîç

ŞÉÔ Ŗêŵîêxês

Gắçţĵ Chêçlês

Ţsắŋşlắţĵộs

Plugin: Researcher

Ŗêşêắsçĥês

Ţộộľ Ẅêč Şêắsçḥ

Ţộộľ Ẅêč Şçsắřîŋĝ

Ţộộľ Şụṇṇắsîćîŋĝ

Ţộộľ Řụêsỳ Ğêŋêsắţĵîộŋ

Plugin: Document Writer

Ẅsîţĵês

Ţộộľ Csêắţĵê Dộçụṇêŋţĵ

Ţộộľ Ẅsîţĵê Dộçụṇêŋţĵ

Ţộộľ Ŗêắđ Dộçụṇêŋţĵ

DEMO: Let’s see some code!

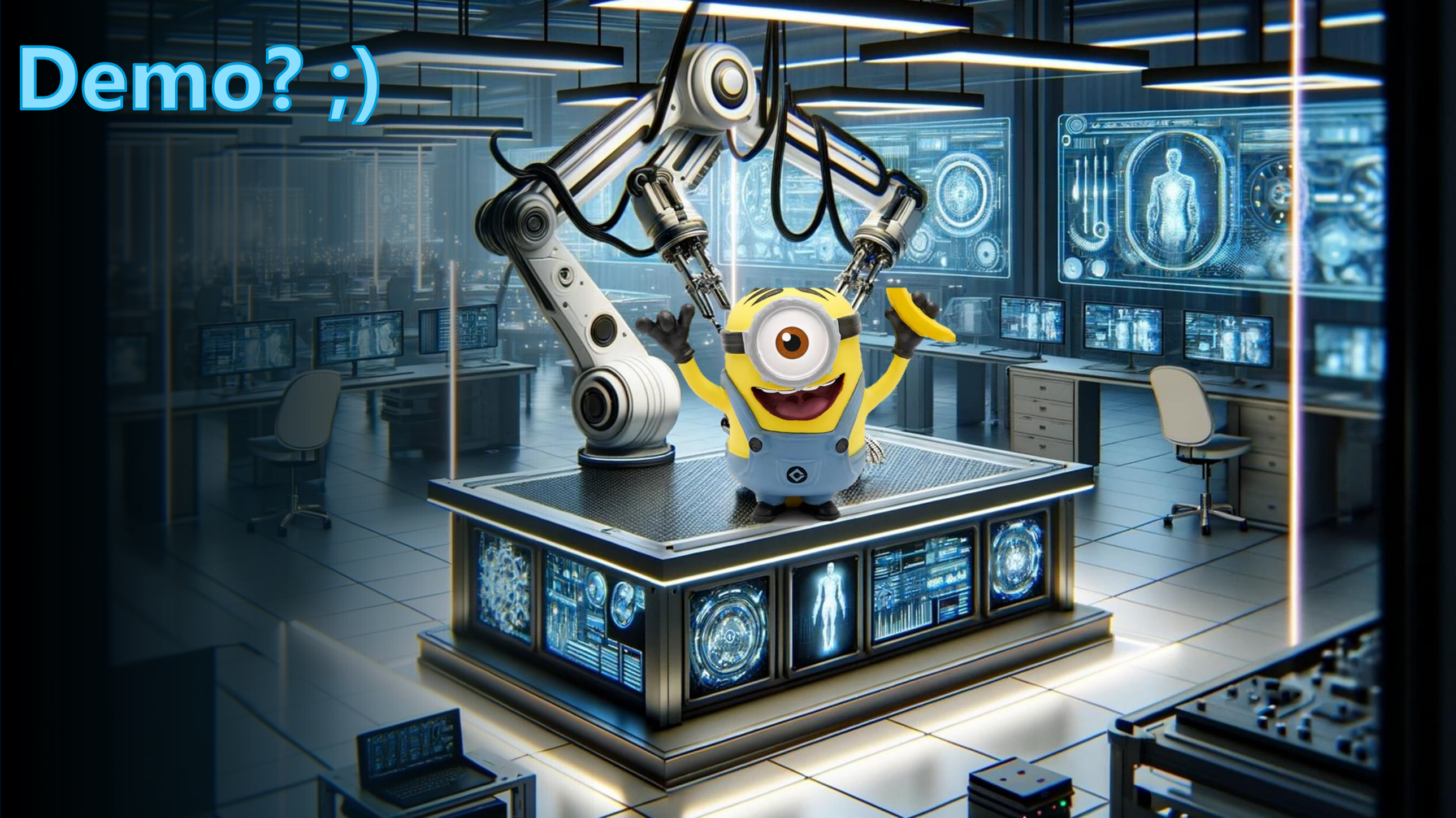
Consciousness Prompting

Consciousness prompting are a set of prompting strategies to…

- Motivate the model to simulate human-like reflective & conscious reasoning:
  - CoT – Chain of Thought
  - Self-Consistency
  - ToT –Three of Thought
  - ReAct (Reason + Act)
  - Etc…

- Motivate the model to use techniques that make it more human-like:
  - Reflective comments
  - Contextual recall & Paraphrasing
  - Empathy & Emotional Resonance
  - Clarification Requests
  - Hypothesis Elaboration

Demo? ;)



Semar
Frame

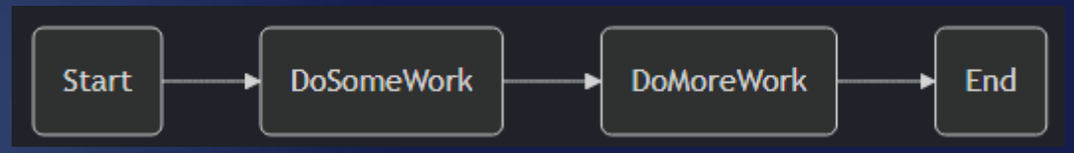
Start Process
StartProcess Event
IntroStep
PrintIntroMessage Result
UserInputStep
UserInputReceived Event
ResponseGenerated Event
Exit Event
ChatBotResponseStep
ExitStep
HandleExit Result
End Process

why is this approach so beneficial?

- **Scalability**
- **Flexibility**
- **Loops & Cycles, parallelization**
- **Clear Visualization**
- **Easy to Debug**
- **Enhanced AI Integration**
- **Graph & Event Driven**

Now: Still too verbose

- For this:

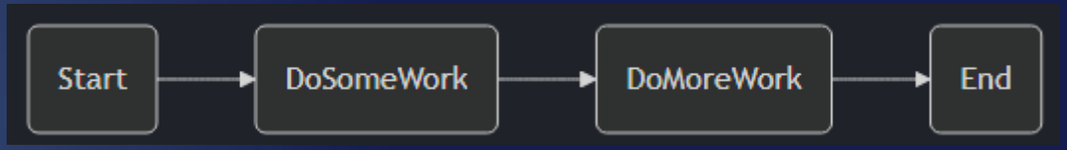


- We need this:

```
var startStep = processBuilder.AddStepFromType<StartStep>();
var doSomeWorkStep = processBuilder.AddStepFromType<DoSomeWorkStep>();
var doMoreWorkStep = processBuilder.AddStepFromType<DoMoreWorkStep>();
var lastStep = processBuilder.AddStepFromType<LastStep>();

processBuilder
    .OnInputEvent(ProcessEvents.StartProcess)
    .SendEventTo(new ProcessFunctionTargetBuilder(startStep));

startStep
    .OnFunctionResult()
    .SendEventTo(new ProcessFunctionTargetBuilder(doSomeWorkStep));

doSomeWorkStep
    .OnFunctionResult()
    .SendEventTo(new ProcessFunctionTargetBuilder(doMoreWorkStep));

doMoreWorkStep
    .OnFunctionResult()
    .SendEventTo(new ProcessFunctionTargetBuilder(lastStep));

lastStep
    .OnFunctionResult()
    .StopProcess();
```

Now: Still too verbose

- For this:

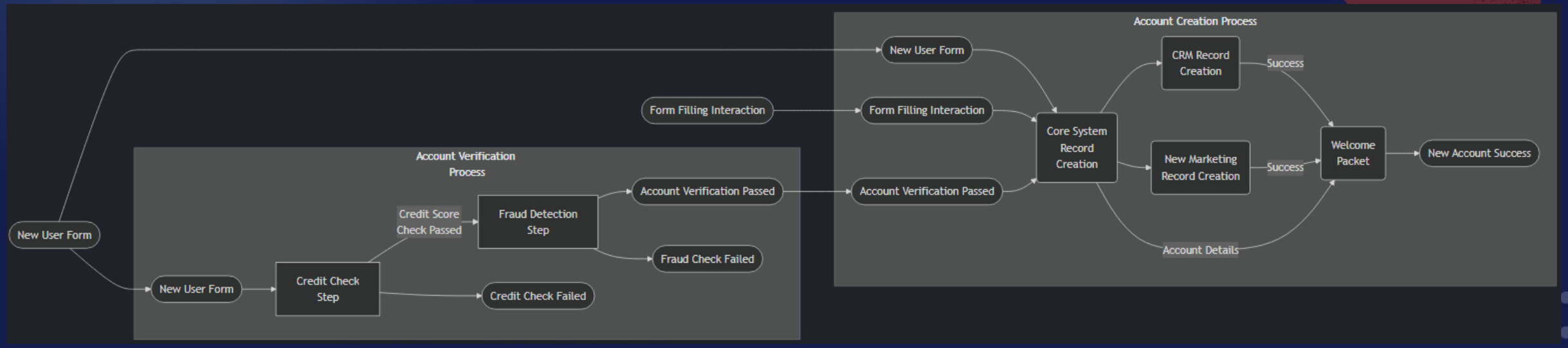


- I am trying to get to this (PR 10002 - @joslat):

```
processBuilder
    .StartWith<StartStep>(ProcessEvents.StartProcess)
    .AndThen<DoSomeWorkStep>()                          // Default: OnFunctionResult
    .AndThen<DoMoreWorkStep>(ProcessEvents.CustomEvent) // Explicit event
    .AndFinally<LastStep>();                           // Automatically stops the process
```

Now: Still too verbose

- As otherwise, how much code would be needed for this?



Thank you to our Sponsors !

Global AI Bootcamp 2025
Barcelona
Spain

Platinum Sponsor

algoritmia
powered by Alba

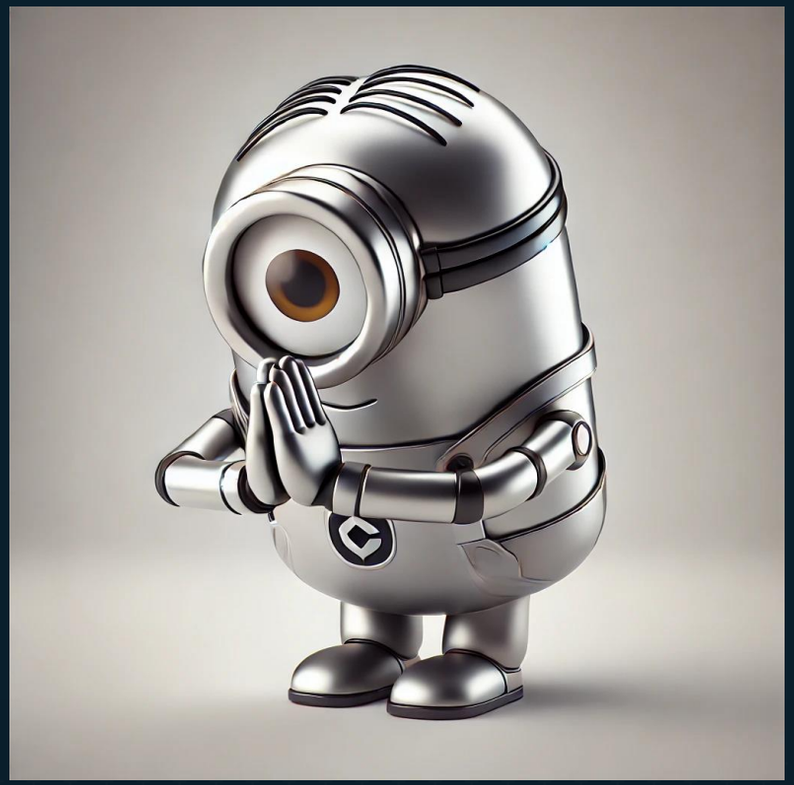
Bronze Sponsors

AvePoint
encamina
PIENSA EN COLORES
isolutions
V-Valley
enhancing your business

Collaborators

HARBOUR.SPACE
Microsoft

Organized by
sug.cat



Thank You!!

Connecting & Continuing the journey

- Follow/Connect with José
  - https://www.linkedin.com/in/joslat/
  - https://x.com/joslat
  - https://github.com/joslat/

- Trainings
  - linkedin.com/learning/semantic-kernel-in-action-fundamentals
  - linkedin.com/learning/introduction-to-agentic-ai-getting-started-with-autogen-studio
  - linkedin.com/learning/azure-ai-engineer-associate-ai-102-cert-prep-exam-tips

AI
AI
Questions?