



WELCOME!

G
L
O
B
O
T
I
C
K
E
T

Global DevOps Experience

Where DevOps meets Developer Productivity



Pedro Sousa

Microsoft Azure MVP | Tech Lead & ISO
in/pmsousa



Valtech *

Xebia GitHub Microsoft



Global DevOps Experience

The creators



Marcel de
Vries



Rene van
Osnabrugge



Rob Bos



Michiel Van
Oudheusden



Erick Segaar

Global DevOps Experience

Portugal team



Pedro
Sousa



Tiago
Costa



António
Silva



Valtech *

Global DevOps Experience

The Actors + backend team



Marcel de Vries



Sofie Wisse



Roy Cornelissen



Hidde de Smet



Geert van der Cruijzen



Jesse
Wellenberg



Rutger
Buiteman



Jesse
Houwing



Duncan
Roosma



Sander
Trijssenaar



Jasper Gilhuis



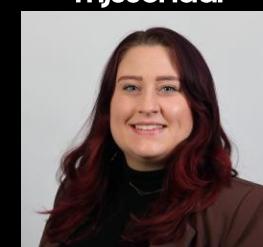
Kees Verhaar



Tijmen van
der Kamp



Brian Randell



Kim
Ellermann



ROBERT GREEN
CEO



Keynote

Mario Rodriguez

SVP GitHub

@mariorod



MARIO RODRIGUEZ

GITHUB

@mariorod

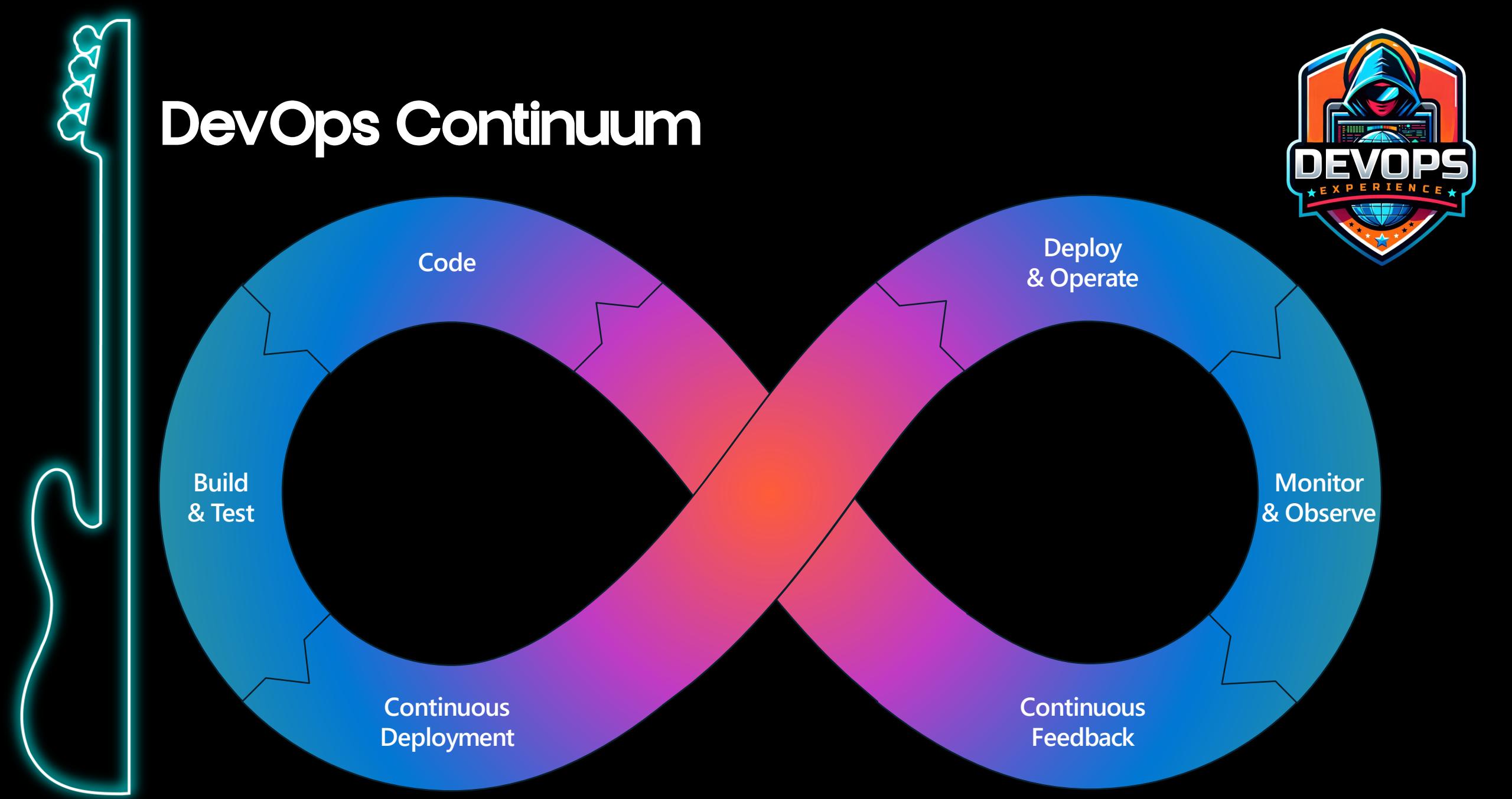


DevOps

DevOps is the **union of people, process, and products** to enable **continuous delivery of value** to our **end users**

-- Donovan Brown

Retired (previously a legend in the DevOps space)





DevOps over the years

- Product companies started naming their products DevOps
 - E.g. Azure DevOps
- Companies did not entirely make their promise to move to a DevOps way of work and started naming Operations people, DevOps engineers
- True DevOps is hard because it needs a change in:
 - Organization
 - Mindset
 - Leadership



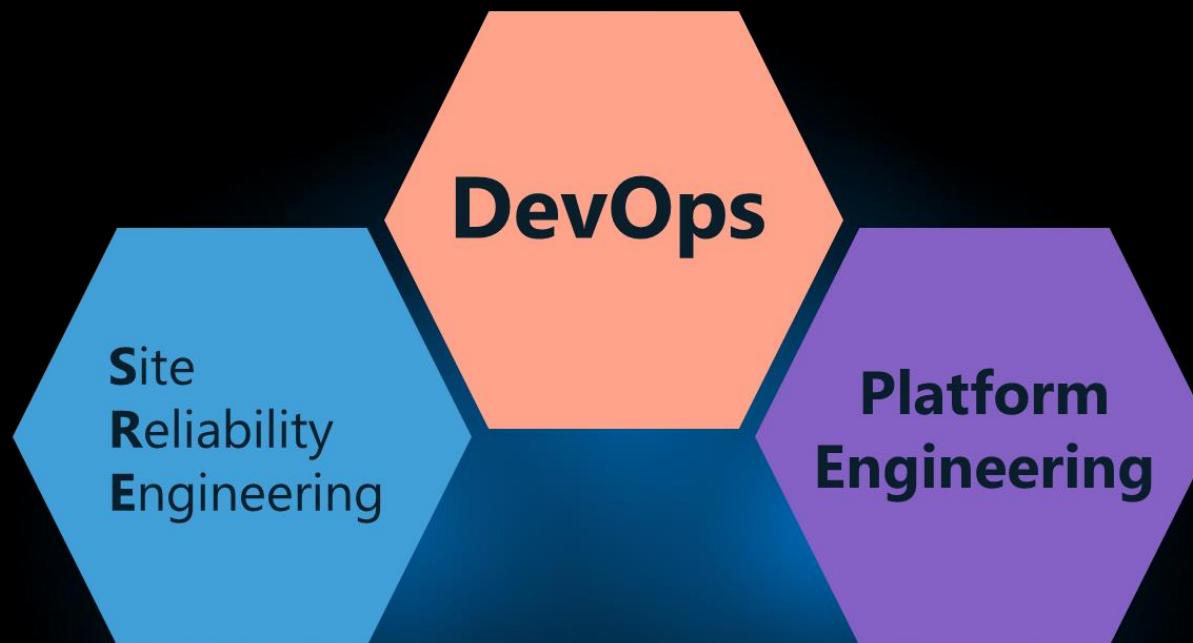
DevOps is Dead!

- People proclaim DevOps is dead and pitch the next big thing
- Site Reliability Engineering
 - First tossed by Google when sharing how they ensured the stability of their systems and having specialists that focus on the reliability engineering aspect of systems in production
- Platform engineering
 - The approach to creating systems that make it easier and more manageable to deliver quality products to production every day



The Big Picture

- DevOps is still what it is and provides and works for organizations that were able to make the change
- SRE and PFE are part of DevOps and improves the way DevOps teams deliver software to production





Developer Productivity

The **SPACE** of Developer Productivity

—Nicole Forsgren

S - **Satisfaction** and well-being

P - **Performance**

A - **Activity**

C - **Communication** and collaboration

E - **Efficiency** and flow

TLDR;

The **SPACE** framework provides a way to logically and systematically think about productivity in a much bigger space and to carefully choose balanced metrics linked to goals—and how they may be limited if used alone or in the wrong context.



Developer Experience (DevEx)

DevEx refers to the systems, technology, process, and culture that influence the effectiveness of software development. It looks at all components of a developer's ecosystem—from environment to workflows to tools—and asks how they are contributing to developer productivity, satisfaction, and operational impact.

[Developer experience: What is it and why should you care? - The GitHub Blog](#)



Global DevOps Experience

The event where you learn more about Platform Engineering with GitHub,
Deploying software to Azure and infuse your applications with AI



Welcome to GloboTicket

How can Developer Experience and a DevOps way of work
Help GloboTicket achieve their goals?

GLOBOTICKET



ROBERT GREEN
CEO



EMILY CHASE
COO



JORDAN SPARKS
PO



MORGAN BLAIR
CISO



ALEX FLETCHER
LEAD DEV





Selling Tickets Online

- The company has just been acquired
- Capital injection of X Million
- Want to become the #1 in ticket sales
- New CEO Appointed to make it all work



GLOBOTICKET



Microsoft | Learn Documentation Training Credentials Q&A Code Samples Assessments S

Azure Product documentation ▾ Architecture ▾ Learn Azure ▾ Develop ▾ Troubleshooting Resources ▾

Platform engineering guide

Learn how platform engineering teams can use building blocks from Microsoft and other vendors to create deeply personalized, optimized, and secure developer experiences.

 OVERVIEW
What is platform engineering?

 CONCEPT
Platform engineering principles



Principles

- ☰ Prioritize customers
- ☰ Adopt a product mindset
- ☰ Empower with self-service
- ☰ Improve discovery and eliminate waste

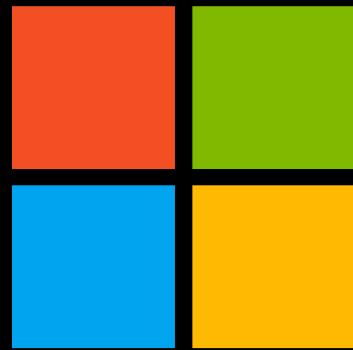
Journey

- ☒ Build the team
- ☒ Define the problem space
- ☒ Plan and prioritize
- ☒ Apply software engineering systems
- ☒ Refine application platform

Platform engineering guidance on Microsoft Learn

aka.ms/plat-eng-learn

aka.ms/plat-eng-signup



MICROSOFT'S VISION

Set up a **complete engineering system** in seconds, contribute from any device, build for any platform, collaborate with anyone with confidence, and scale without limit

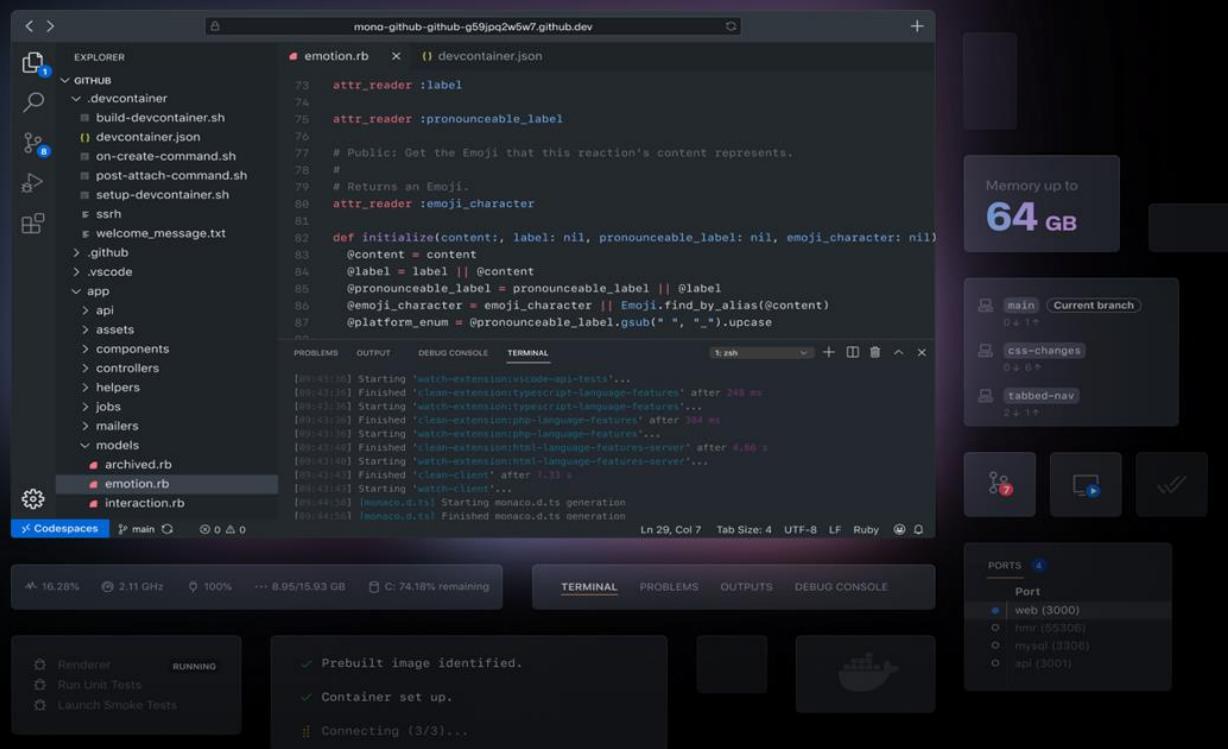
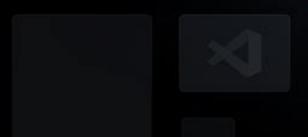
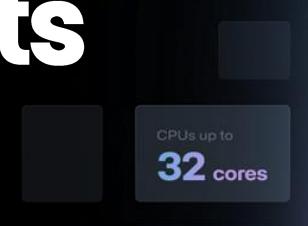


Enable Developer Onboarding

Blazing fast cloud developer environments

Full compute backing, up to 32 cores and 64 GB RAM

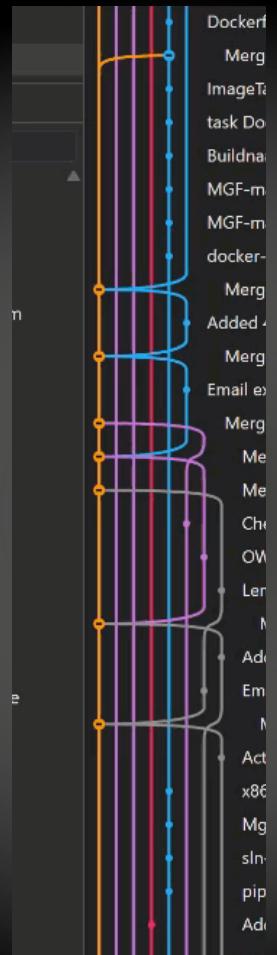
Reducing time to set up development environments from hours per week to seconds





Ensure Quality Delivery

- Setup your DevOps platform so it helps you deliver quality software
- Branching strategy: GitHubFlow
- Use pull requests to ensure code reviews
 - Also improves compliance for your organization
- Branch Protection rules
- Action workflows to validate various steps in the process
 - Continuous Integration
 - Continuous Delivery
 - Continuous Security Checks





Improve Code Understanding



GitHub Copilot

Your AI pair programmer

Offers code suggestions,
contextualized to your project

- ✓ Convert comments to code
- ✓ Autofill for repetitive code
- ✓ Autosuggest tests
- ✓ Show alternatives



The screenshot shows a code editor interface with several tabs at the top: sentiment.ts, write_sql.go, parse_expenses.py, and addresses.rb. The main pane displays a snippet of TypeScript code:

```
1 #!/usr/bin/env ts-node
2
3 import { fetch } from "fetch-h2";
4
5 // Determine whether the sentiment of text is positive
6 // Use a web service
7 async function isPositive(text: string): Promise<boolean> {
8   const response = await fetch(`http://text-processing.com/api/sentiment/`, {
9     method: "POST",
10    body: `text=${text}`,
11    headers: {
12      "Content-Type": "application/x-www-form-urlencoded",
13    },
14  });
15  const json = await response.json();
16  return json.label === "pos";
17}
```

A blue callout box labeled "Copilot" is positioned at the bottom right of the code editor.



Prevent Security Incidents

GitHub Advanced Security

-  Dependency Analysis
-  Code Scanning
-  Secret Scanning

Code scanning alerts / #1

Arbitrary file access during archive extraction ("Zip Slip")

 Open in main on Aug 4

```
code/src/Attendee/Attendee.cs:12 ↗
 9     {
10         public void WriteToDirectory(ZipArchiveEntry entry, string destDirectory)
11         {
12             string destFileName = Path.Combine(destDirectory, entry.FullName);
13             entry.ExtractToFile(destFileName);
14         }
15     }
```

Unsanitized archive entry, which may contain '...', is used in a file system operation.

CodeQL Show paths

Tool	Rule ID	Query
CodeQL	cs/zipslip	View source

Extracting files from a malicious zip file, or similar type of archive, is at risk of directory traversal attacks if filenames from the archive are not properly validated.

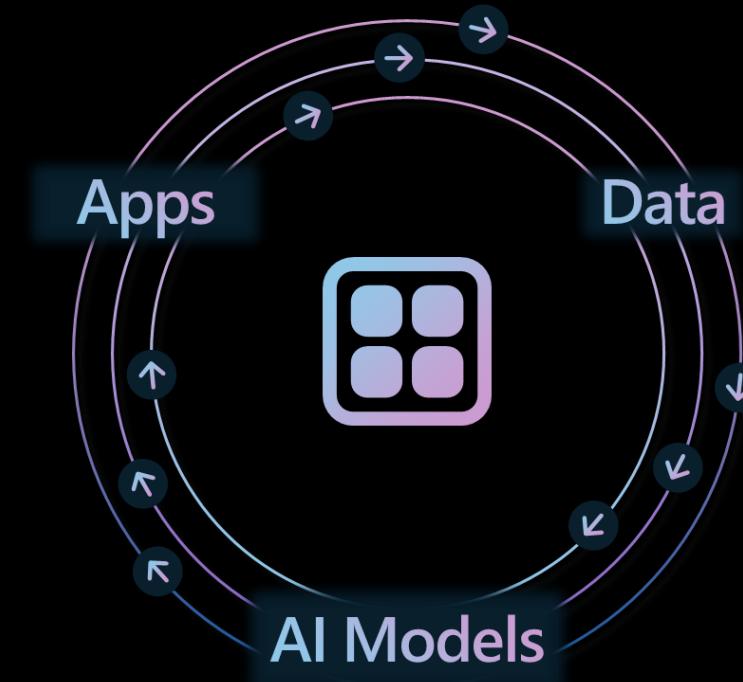
Zip archives contain archive entries representing each file in the archive. These entries include a file path for the entry, but these file paths are not restricted and may contain unexpected special elements such as the directory traversal element (...). If these file paths are used to create a filesystem path, then a file operation may happen in an unexpected location. This can result in sensitive information being revealed or deleted, or an attacker being able to influence behavior by modifying unexpected files.

For example, if a zip file contains a file entry ..\sneaky-file, and the zip file is extracted to the directory c:\output, then naively combining the paths would result in an output file path of c:\output..\sneaky-file, which would cause the file to be written to c:\sneaky-file.



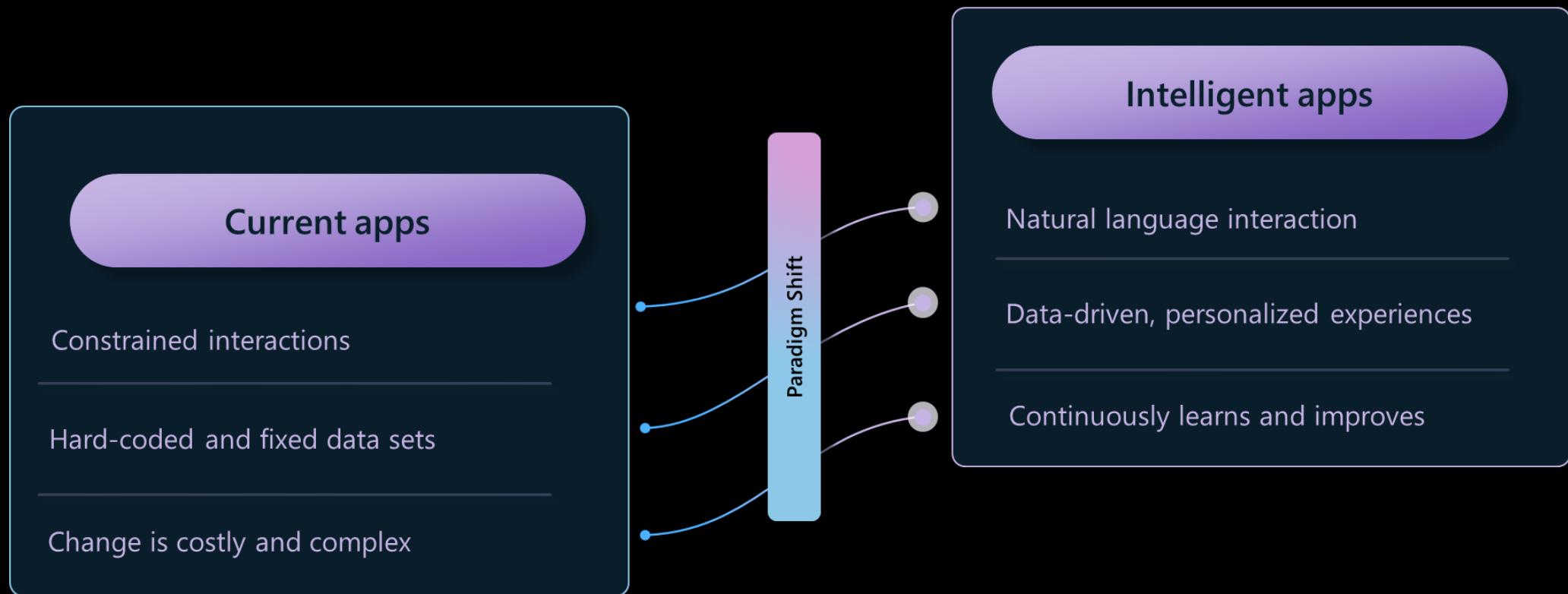
Use AI in existing applications

In the age of AI,
every app should
be intelligent



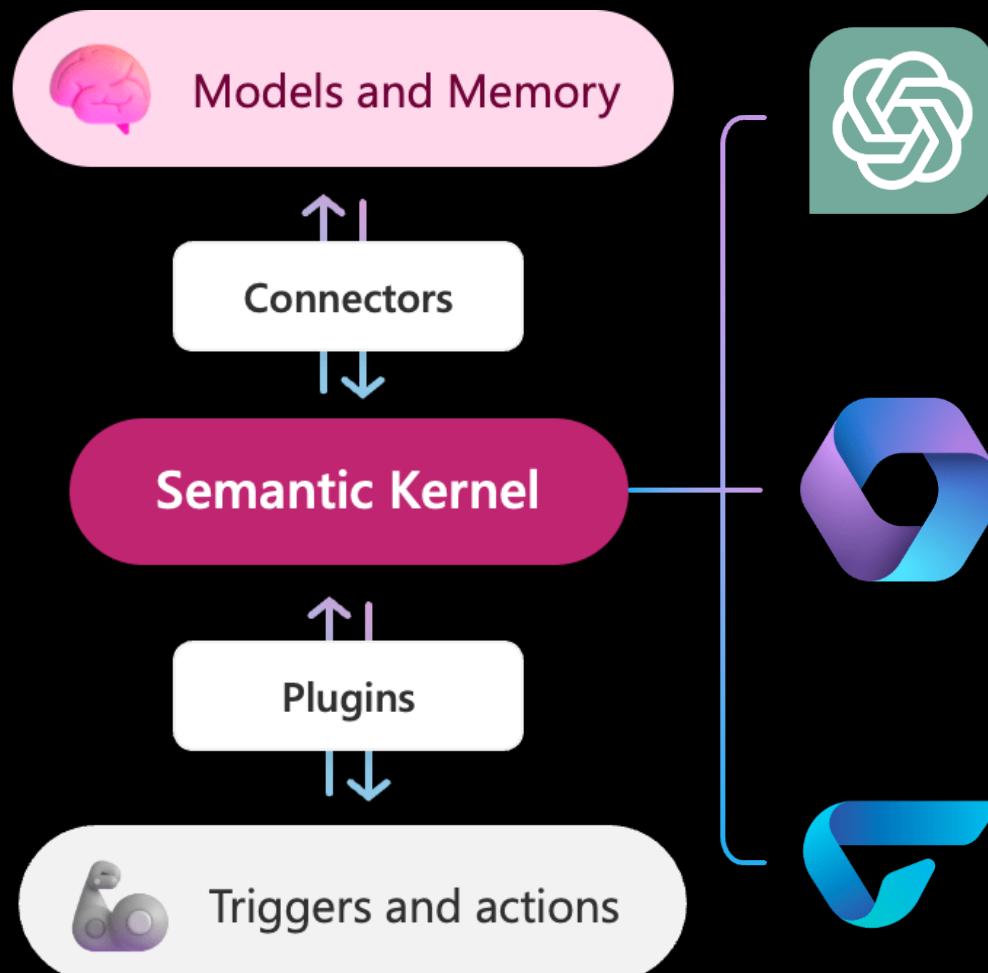


Use AI in existing applications





Semantic Kernel



Semantic Kernel
Provides the bridge
between enterprise
developer and Ai
researchers



Welcome to GloboTicket

Time to start your onboarding



RENÉ VAN OSNABRUGGE
ORGANIZER



GitOps Commands

- **/help** - Alex will give a tip or hint. This can be a blog, a link or just hints
- **/expert-tip** - A detailed Wiki page with step-by-step instructions will be presented to finish the challenge
- **/verify** - The acceptance criteria will be checked. Jordan Sparks will show what has been checked, what is OK and what is not OK. Verify will also add the score to the scoreboard
- **/fix** - A PR with code and instructions will be presented for the merge
- **/finish** - Close the issue, finish the challenge and go back to the intranet



Sign In and Join Your Team

A screenshot of a web page titled "Welcome to the Global DevOps Experience". The page features the event's logo at the top center. Below the logo, the text "Welcome to the Global DevOps Experience" is displayed in large, bold, white font. A paragraph of smaller text explains the next steps: "Time to get started with the event! You will need to log in using your GitHub account which you used to register yourself. When logged in, you can select a team to join." At the bottom of the page is a blue button with the text "Join the event!" in white.

Sign-in with
Your **GitHub**
Handle and
Pick your team

<https://www.globaldevopsx.com/event-day>