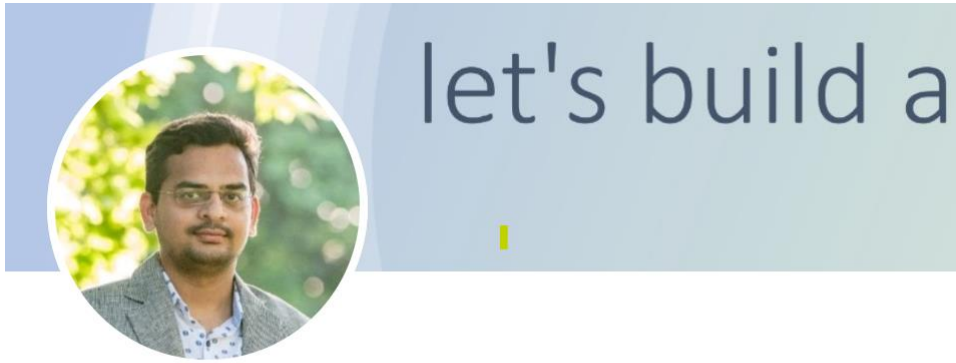
Insight

Platform Engineering using GitHub

Introductions

Stephen Tulp
APAC Technology Strategist - Cloud Infrastructure and Security at Insight

Greater Perth Area · **Contact info**



Ram Mara (He/Him) · 1st
Microsoft Azure Professional

Greater Perth Area · **Contact info**

Insight

Agenda


Insight
01
Part One
Platform Engineering


Insight
02
Part Two
GitHub


Insight
03
Part Three
Landing Zone Vending Machine


Insight
04
Part Four
Demo

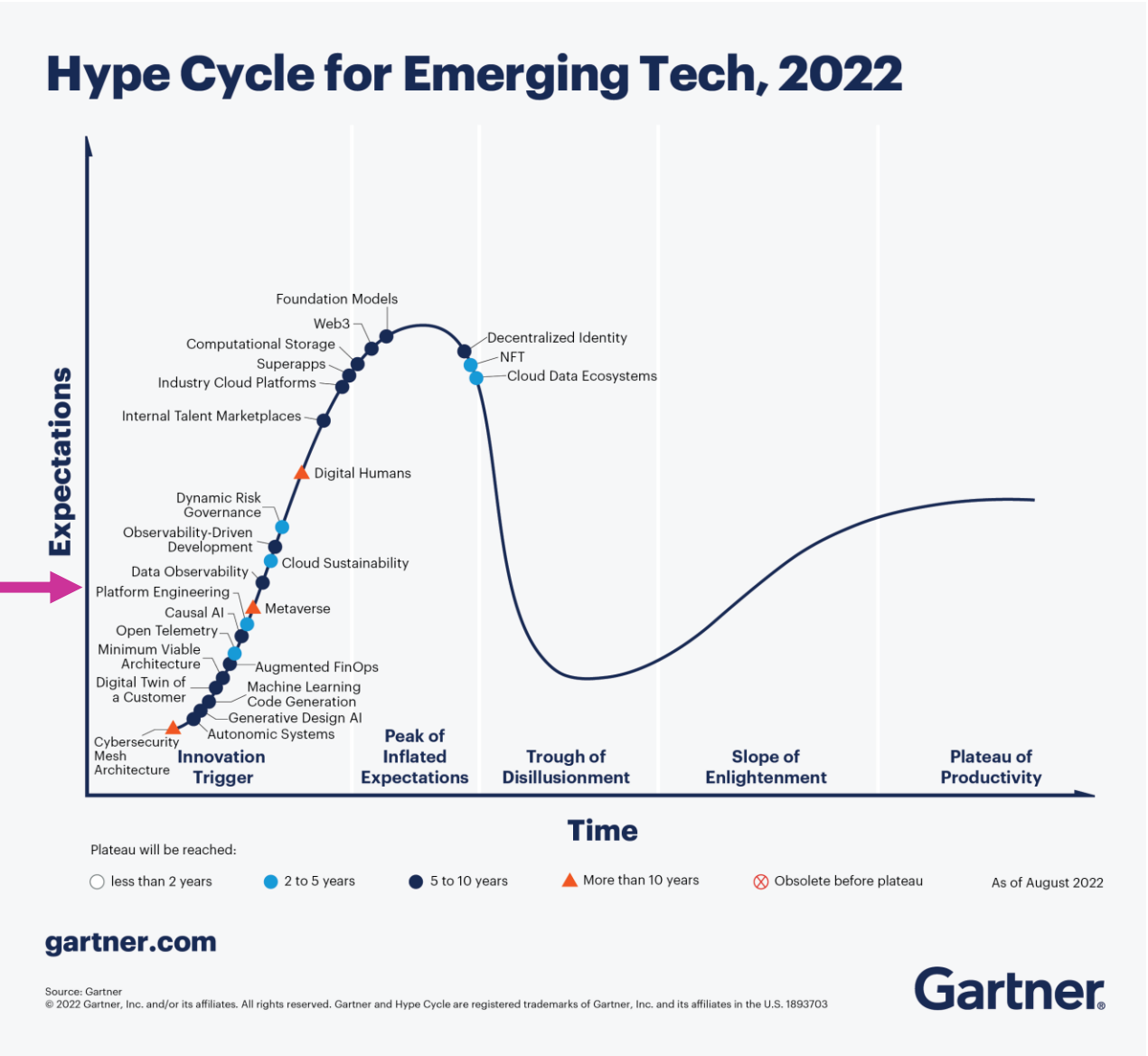
Insight

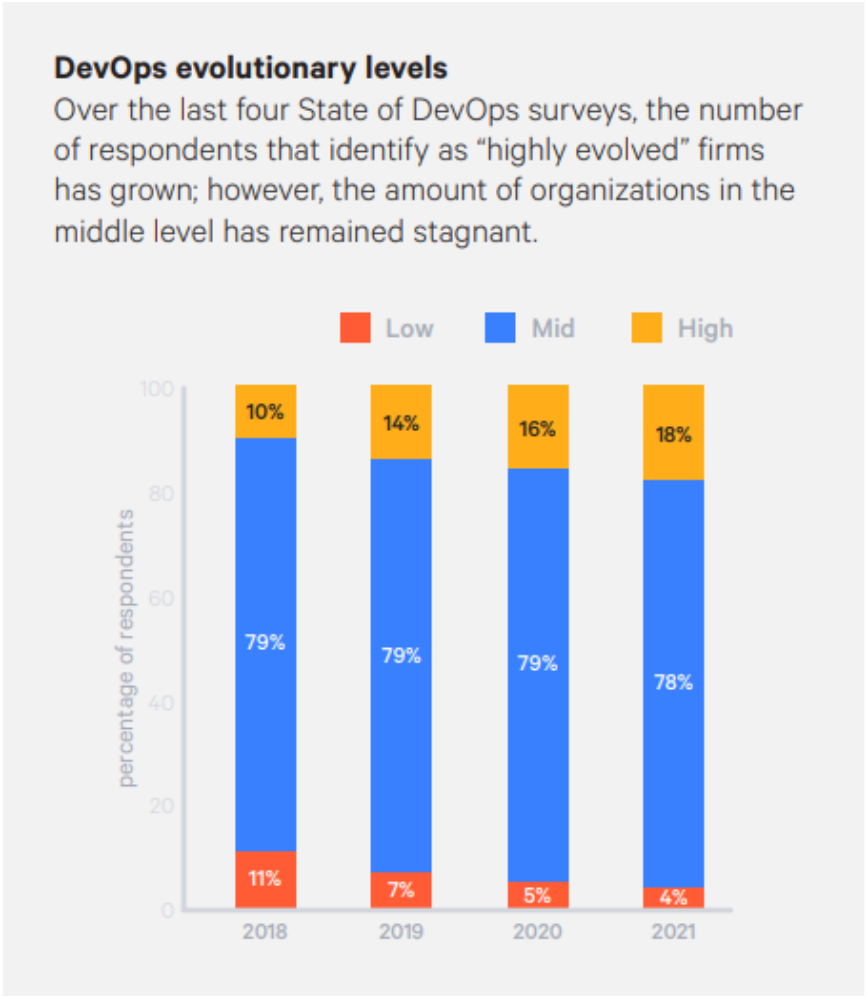
Part One

Platform Engineering


THEY'RE ALREADY SEVEN IN THERE!
YOU SHALL NOT PASS !!!
DevS
Infra

Why are we talking about Platform Engineering?





Full LifeCycle Dev:

Business
FrontEnd
BackEnd
Infrastructure
Monitoring
QA
Security
Data
Support
...

Cognitive load!!!

Insight

What is Platform Engineering? (Chat GTP Version)

“Platform engineering is an emerging discipline focused on enhancing developer productivity by reducing the complexity and uncertainty of modern software delivery. It addresses some of the biggest challenges of doing DevOps at scale, including aligning development practices with business priorities and reducing the burden of managing a complicated web of tools and infrastructure across the application lifecycle”.

What is Platform Engineering? (My Version)

“Platform engineering enables self-service and promotes golden paths with recommended tools & security best practices built-in, thus helping reduce cognitive load for developers in the cloud native era”.

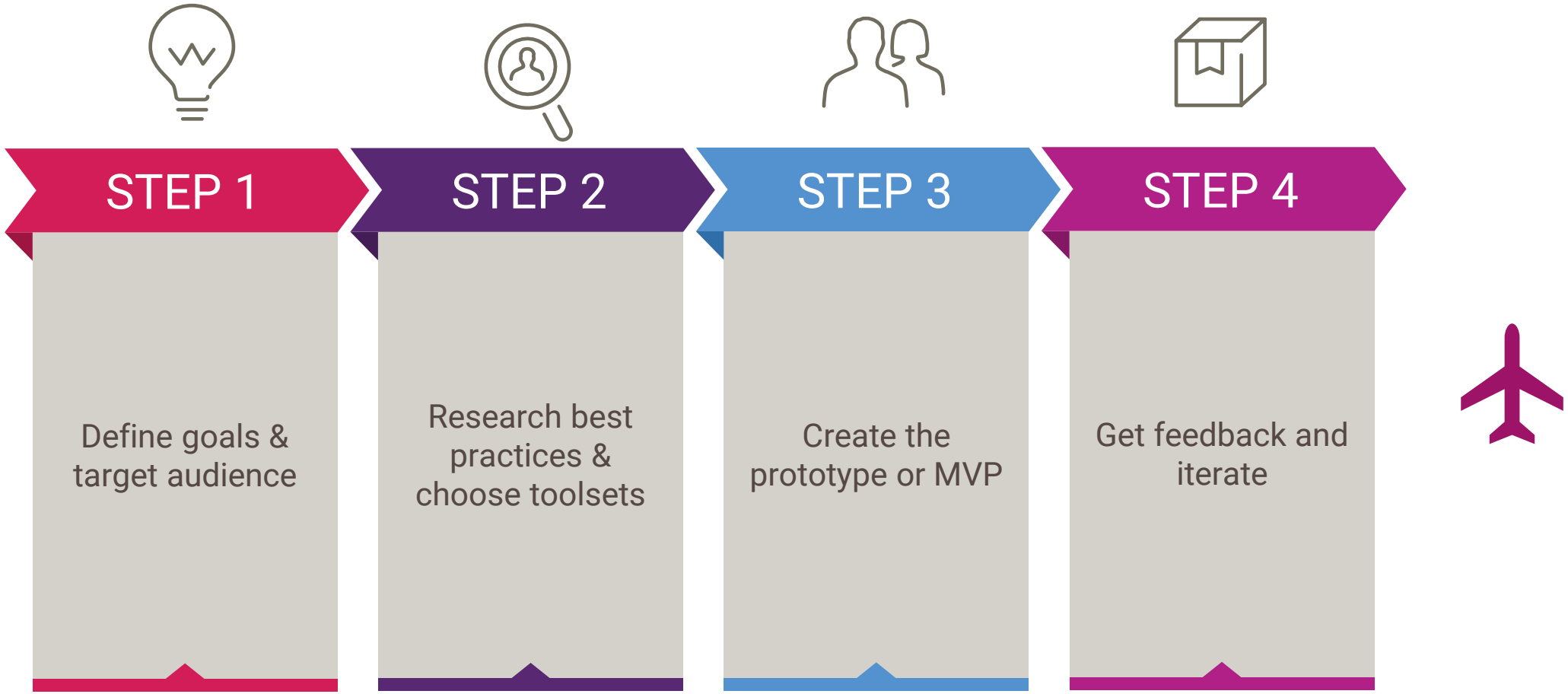
“Platform engineering teams can be viewed as internally focused DevSecOps teams with very specialized skills: infrastructure knowledge, automation knowledge, security knowledge and the ability to write code.”

Juan Orlandini, Insight Distinguished Engineer

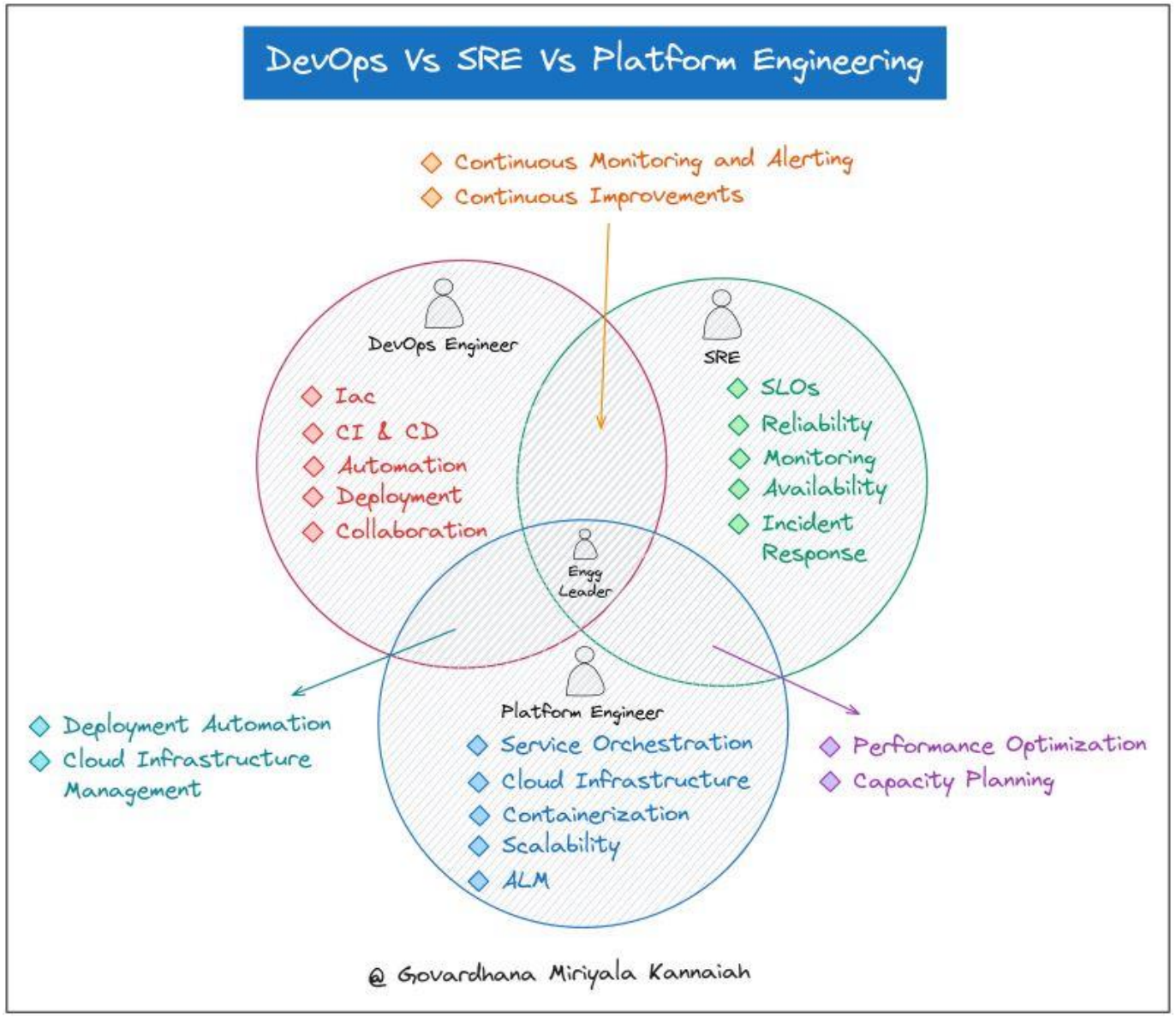
What are benefits of Platform Engineering?

- Developers can focus exclusively on developing and don’t need to be involved in provisioning the platform infrastructure, thus reducing the cognitive load of developers/consultants.
- Address security, compliance and monitoring up front.
- Provide golden paths, an opinionated set of processes, tools and technologies that provide standardization and reusability and minimise anti-patterns.

Approach towards Platform Engineering?



DevOps vs. SRE vs Platform Engineering



- **DevOps Engineers** drive efficient and reliable software delivery by bridging the gap between development and operations teams
- **Site Reliability Engineers** concentrate on optimizing the reliability, performance, and efficiency of software systems
- **Platform Engineers** design, build and maintain the infra and tools for supporting software development, deployment, and operations

Insight

Lessons Learnt from the Journey so Far…..

“Platform engineering is what happens when DevOps engineers start to talk to each other”

Insight

““The biggest mistake would be for a platform engineer to build something and say to developers ‘*look at what I built you*.’ Instead, platform engineering teams should engage with DevOps, FinOps and business leaders — and decide which systems and platforms should become common territory. Then they can create a first version, test, verify, correct or add functionality, and repeat. It’s a never-ending process. That’s what makes it a product. Not a project.”

Insight

02

Part Two

GitHub

GitHub

- Acquired by Microsoft in June 2018
- 100 million developers use GitHub across the globe
- End of 2022, 413 million open-source contributions on the platform.
- 500 languages are used across the platform to build software.

InnerSourcing

The InnerSource working model applies the open-source product delivery model to private, internal teams within an organisation.

- Like open-source, InnerSource is a decentralized product delivery model based on open and transparent collaboration.
- Peer production network of self-organizing communities of interest and practice.
- Community-driven enterprise aimed at converting users into contributors.

InnerSourcing also depends on lightweight documentation in order to market to consumers and encourage community contributions.

How Does Insight use GitHub and Innersourcing

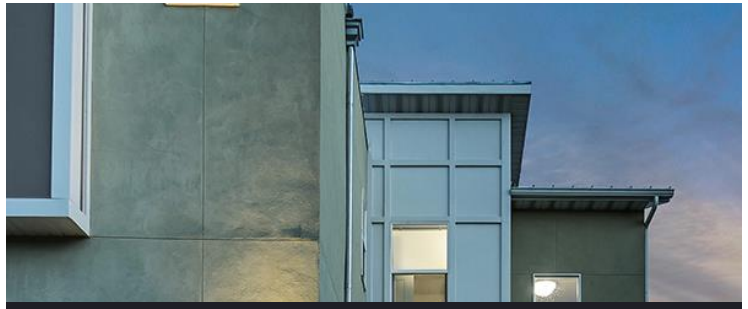
- Repos for all our key platform solutions (*Landing Zones, Integration and Data Platforms*)
- Actions for CI/CD (Build, Validation, Testing and Deployment)
- Issues for *Bugs*, *Feature Requests* and *General Feedback*.
- Project Boards to track *Issues and PRs*
- Wiki for engineering standards, guidelines and examples.
- README and Contribution guides to get started and help contribute back into the solution.

Insight

Part Three

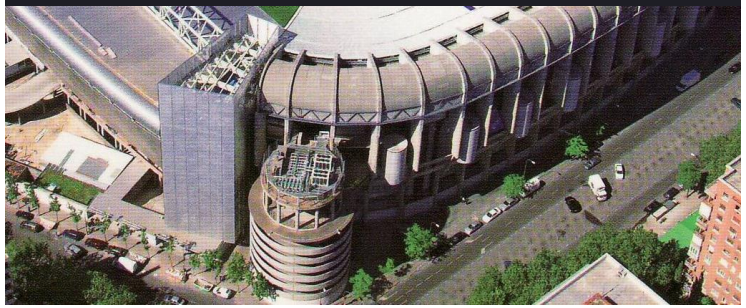
Landing Zone Vending Machine

What are you building?

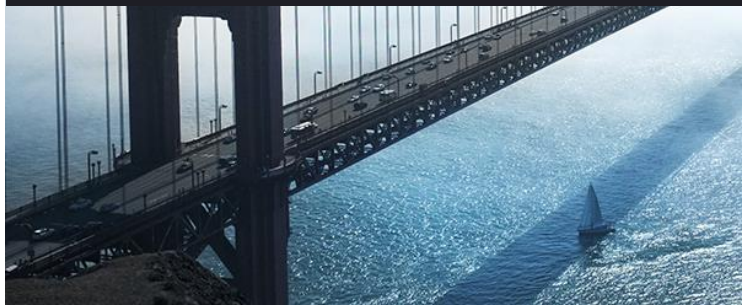


House

Stadium

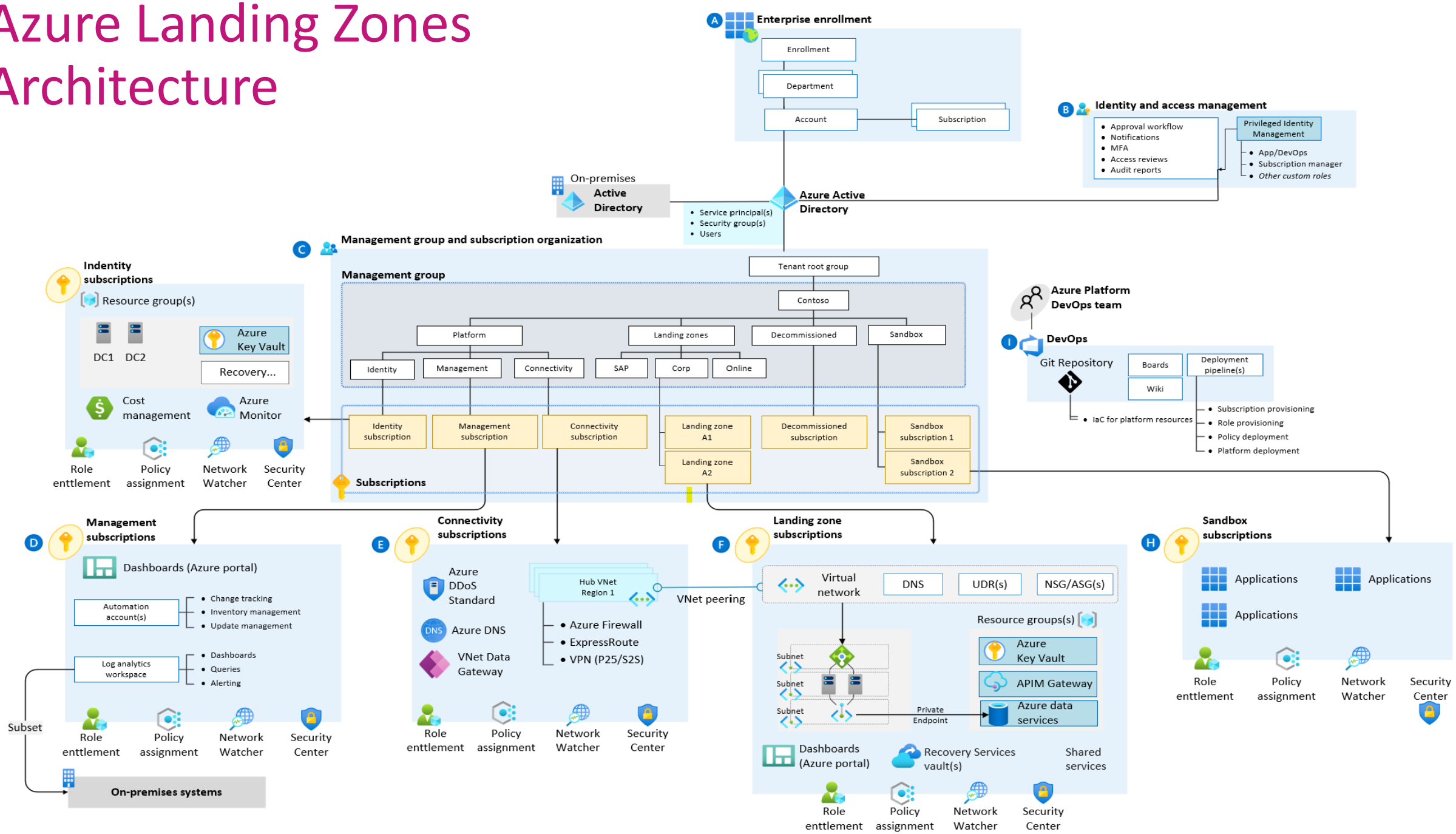


Bridge

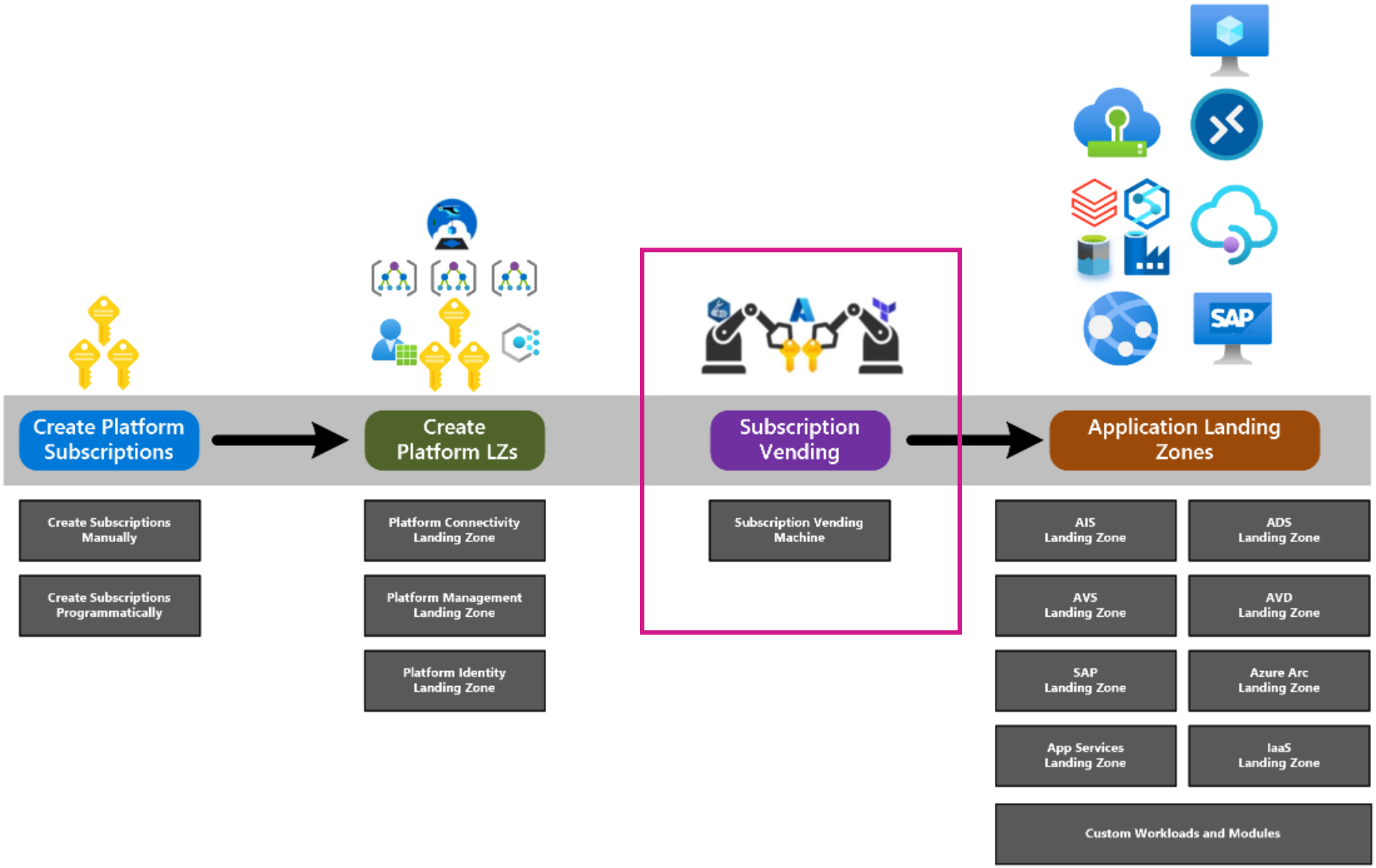


Insight

Azure Landing Zones Architecture

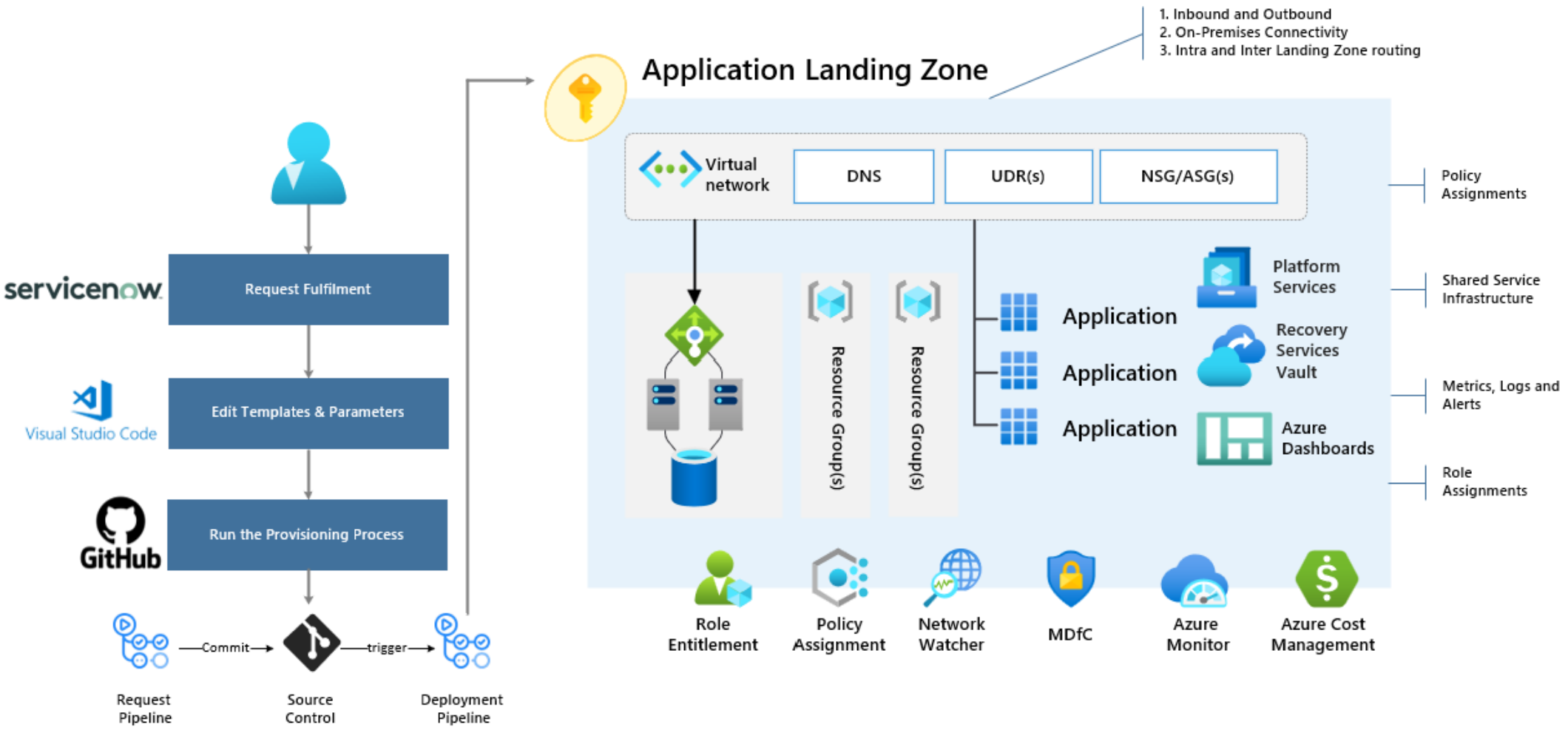


Azure Landing Zones Journey

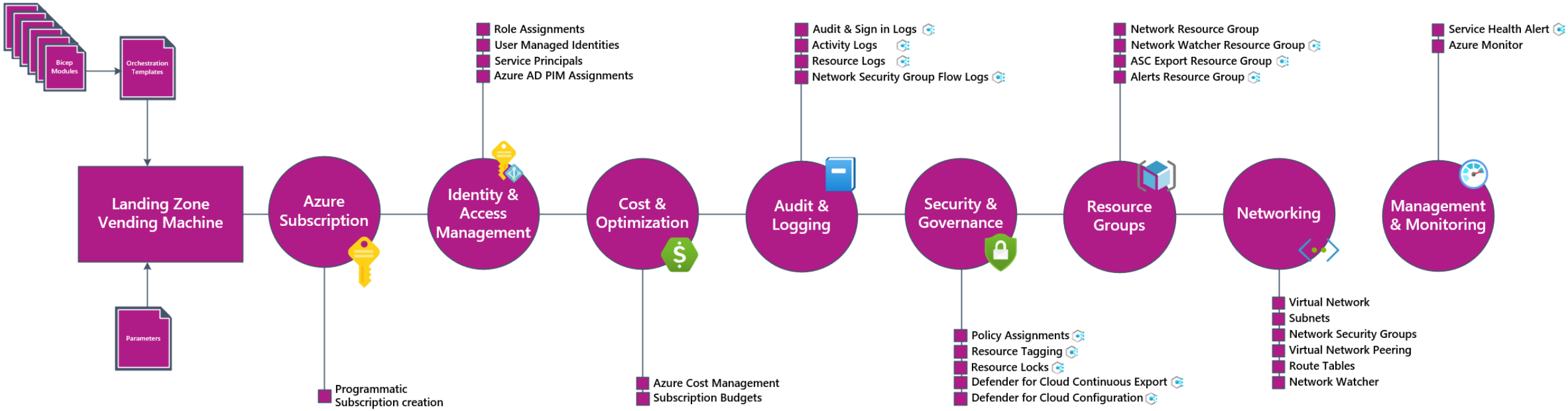


Insight

Subscription Vending Provisioning Process



Insight


Bicep Modules
Orchestration Templates
Parameters
Landing Zone Vending Machine
Azure Subscription
Programmatic Subscription creation
Identity & Access Management
Role Assignments
User Managed Identities
Service Principals
Azure AD PIM Assignments
Cost & Optimization
Azure Cost Management
Subscription Budgets
Audit & Logging
Audit & Sign in Logs
Activity Logs
Resource Logs
Network Security Group Flow Logs
Security & Governance
Policy Assignments
Resource Tagging
Resource Locks
Defender for Cloud Continuous Export
Defender for Cloud Configuration
Resource Groups
Network Resource Group
Network Watcher Resource Group
ASC Export Resource Group
Alerts Resource Group
Networking
Virtual Network
Subnets
Network Security Groups
Virtual Network Peering
Route Tables
Network Watcher
Management & Monitoring
Service Health Alert
Azure Monitor

Insight

Insight

Part Four

Demo