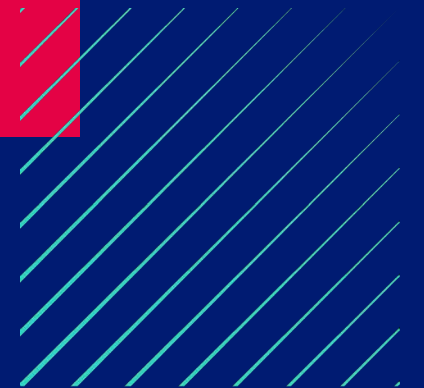


**SOFTRAMS**



**0 to 60 on Day 1 :  
The Promise Of Software Factories**



# About



- CTO, [Softrams](#)
- Founder & CEO, [Teaching For Good](#)
- Building software and teams for 24 years
- Telecom, Supply Chain, EdTech, Healthcare and Civic Services
- LinkedIn: [/mkmurali](#)
- Twitter : [@mkmurali](#)



Avery



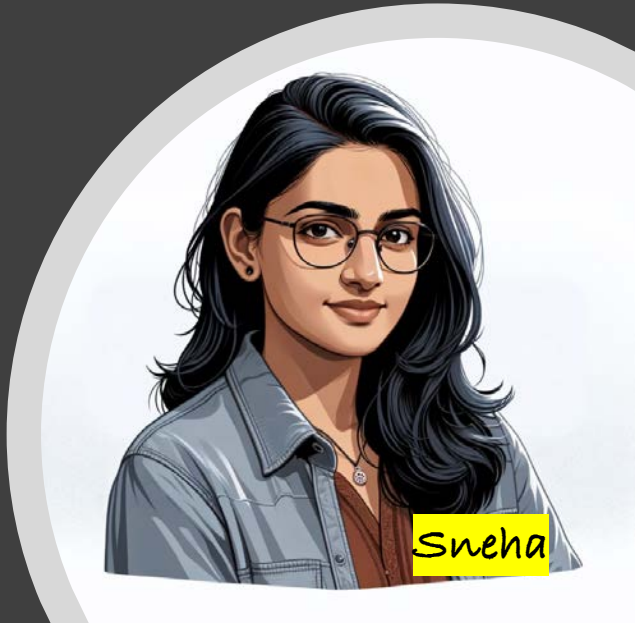
Layla



Lucy



Emma



Sneha

# Personas





# New Idea

- Emma, the Product Manager
- Emma got a great new idea for a new service

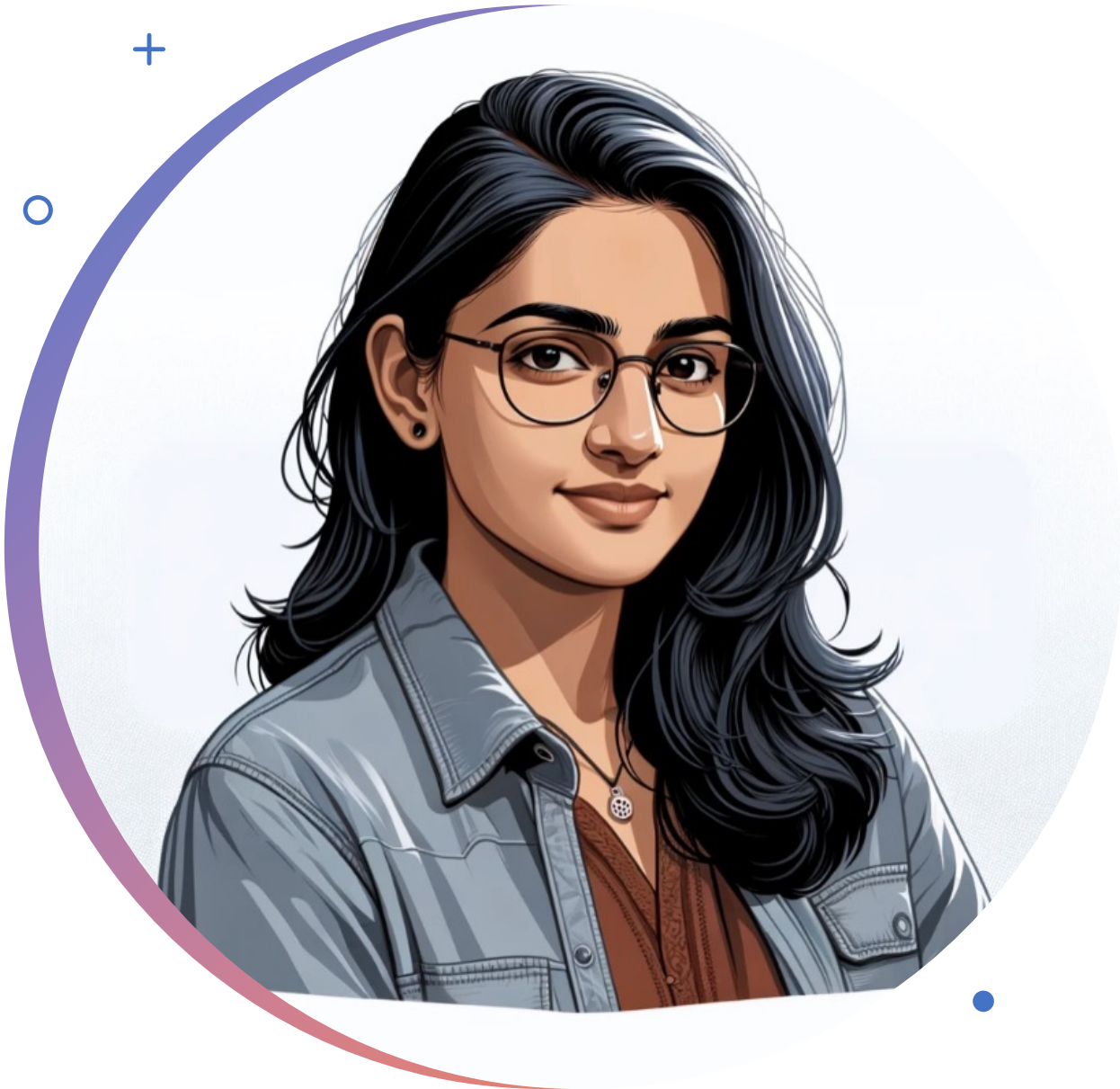




What does it take to go from **Idea**  
to first **demo**?



What does it take to go from **Idea**  
to first **launch**?



# Experiments

- Sneha, the Full Stack Developer
- Want to experiment building a custom ChatGPT to enhance the product



What does it take to go from **Idea**  
to first **demo**?





# Onboarding

- Avery, the new Engineer on team
- Want to quickly learn about the ecosystem of libraries, services and components



What does it take **Avery** to  
contribute her **first PR**?



Our Journey



**Murali Mallina** 10:44 AM

May 8th, 2018 ▾

Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I request a new project, specify a name, it should create a repo (copy of that API service boiler plate), create dev branch, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say [softrams.io](https://softrams.io)) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I request a new project, specify a name, it should create a repo (copy of that API service boiler plate), create dev branch, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.





Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, specify a name, it should create a repo (copy of that API service boiler plate), create dev branch, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should create a repo (copy of that API service boiler plate), create dev branch, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service boiler plate), create dev branch, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), create dev branch, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, setup all hooks needed for the repo for CI/CD. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.





Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. Builds the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. **Builds** the sample repo, tests and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. **Builds** the sample repo, **tests** and deploy the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. **Builds** the sample repo, **tests** and **deploy** the sample service container. Configure a cname for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.



Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. **Builds** the sample repo, **tests** and **deploy** the sample service container. Configure a **cname** for the sample service. (say softrams.io) Final output will be the URL where we can access the service healthcheck.





Can we put this on our calendar to do an end-to-end demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. **Builds** the sample repo, **tests** and **deploy** the sample service container. Configure a **cname** for the sample service. (say [softrams.io](https://softrams.io)) Final output will be the **URL** where we can access the service **healthcheck**.



Can we put this on our calendar to do an **end-to-end** demo. I will give you the API service sample repo that we can use for testing. If I **request a new project**, **specify a name**, it should **create a repo** (copy of that API service **boiler plate**), **create dev branch**, **setup all hooks needed for the repo for CI/CD**. **Builds** the sample repo, **tests** and **deploy** the sample service container. Configure a **cname** for the sample service. (say softrams.io) Final output will be the **URL** where we can access the service **healthcheck**.

# Capabilities



- Security
- FinOps
- Data

Governance

- Services
- Libraries
- Templates

Knowledge



Delivery

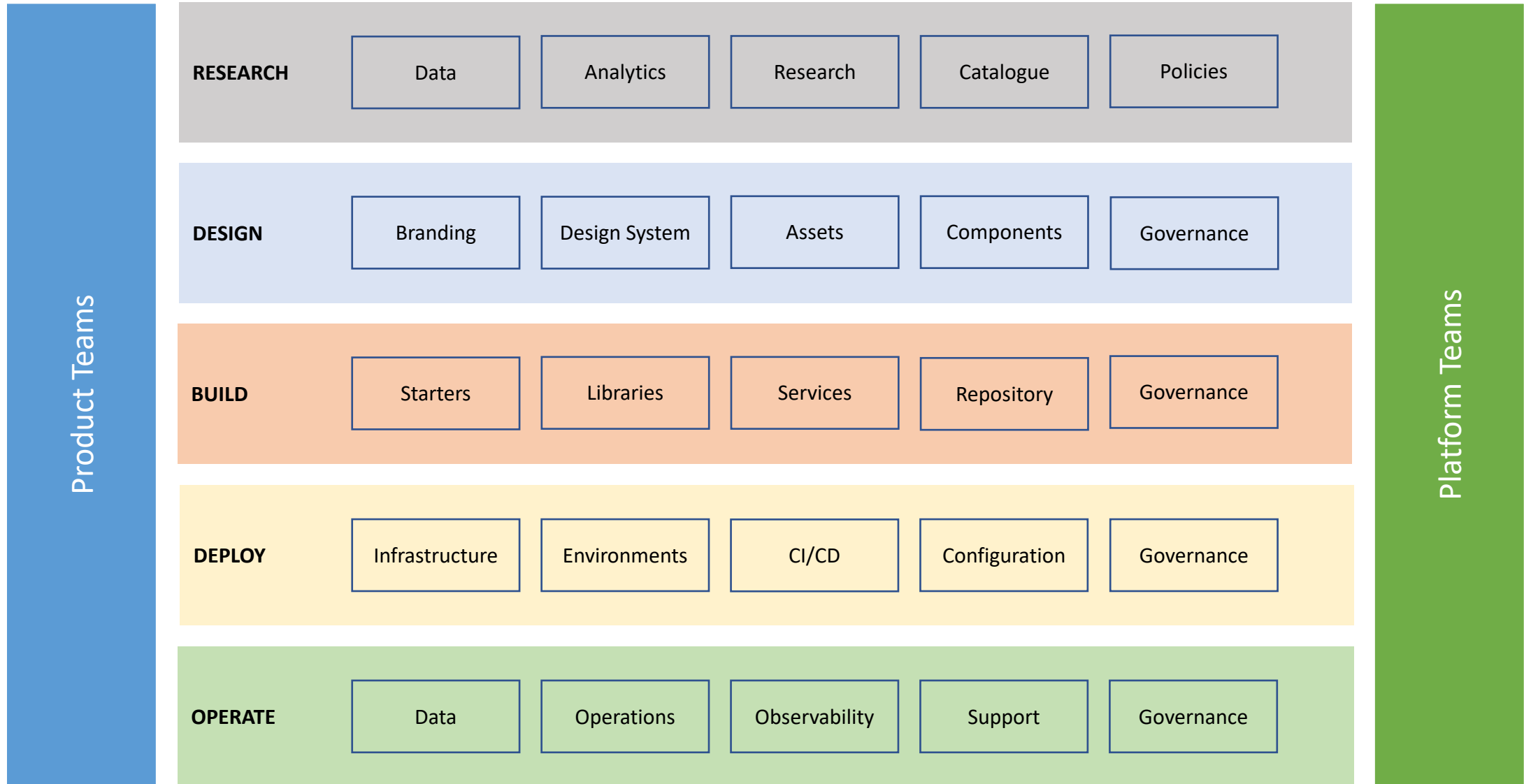
- CI/CD
- Configuration
- Observability

Infrastructure

- Repositories
- Infrastructure
- Environments



# Capabilities For The Full Life Cycle



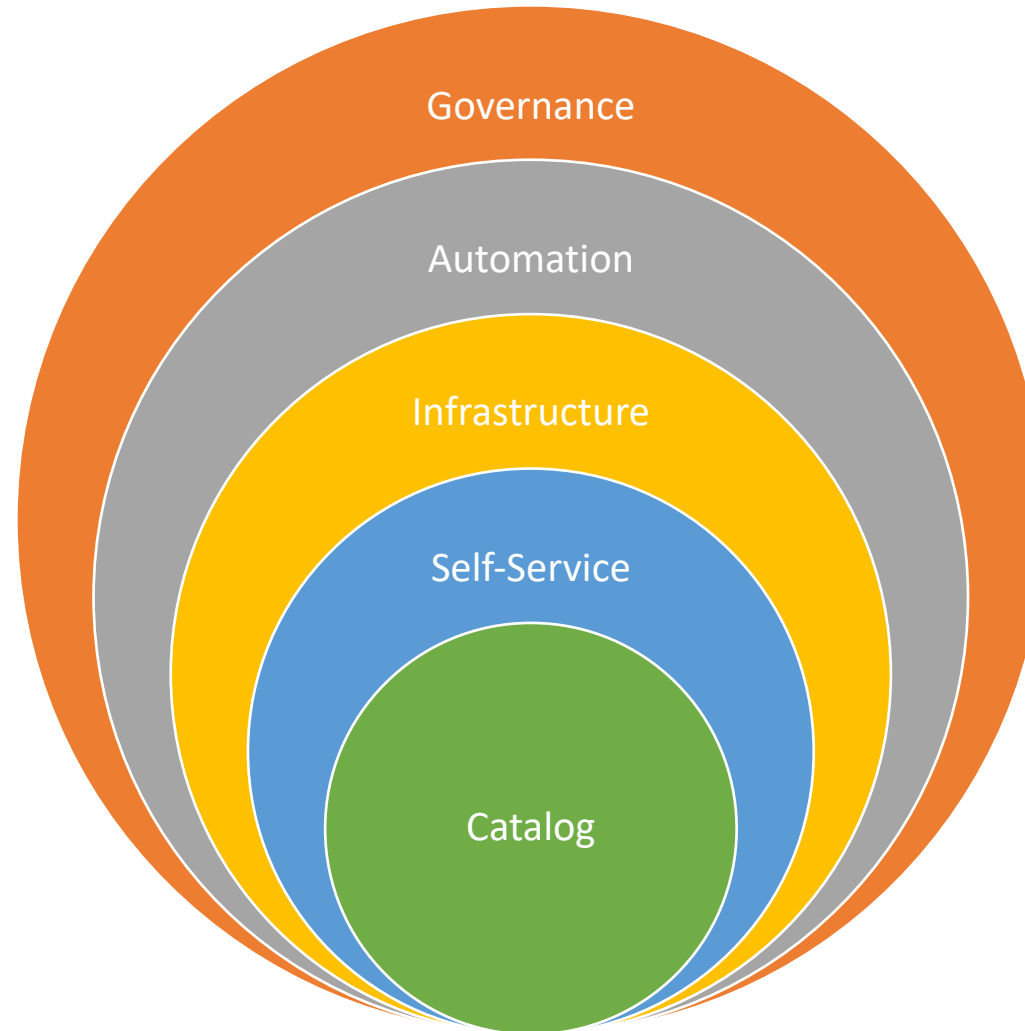


“In software engineering and enterprise software architecture, a software factory is a software product line that configures extensive tools, processes, and content using a template based on a schema to **automate** the development and maintenance of variants of an archetypical product by adapting, assembling, and configuring framework-based components”

- Wikipedia ([https://en.wikipedia.org/wiki/Software\\_factory](https://en.wikipedia.org/wiki/Software_factory))



- Software Factory*
- *Platform Engineering*
  - *Internal Developer Portal*

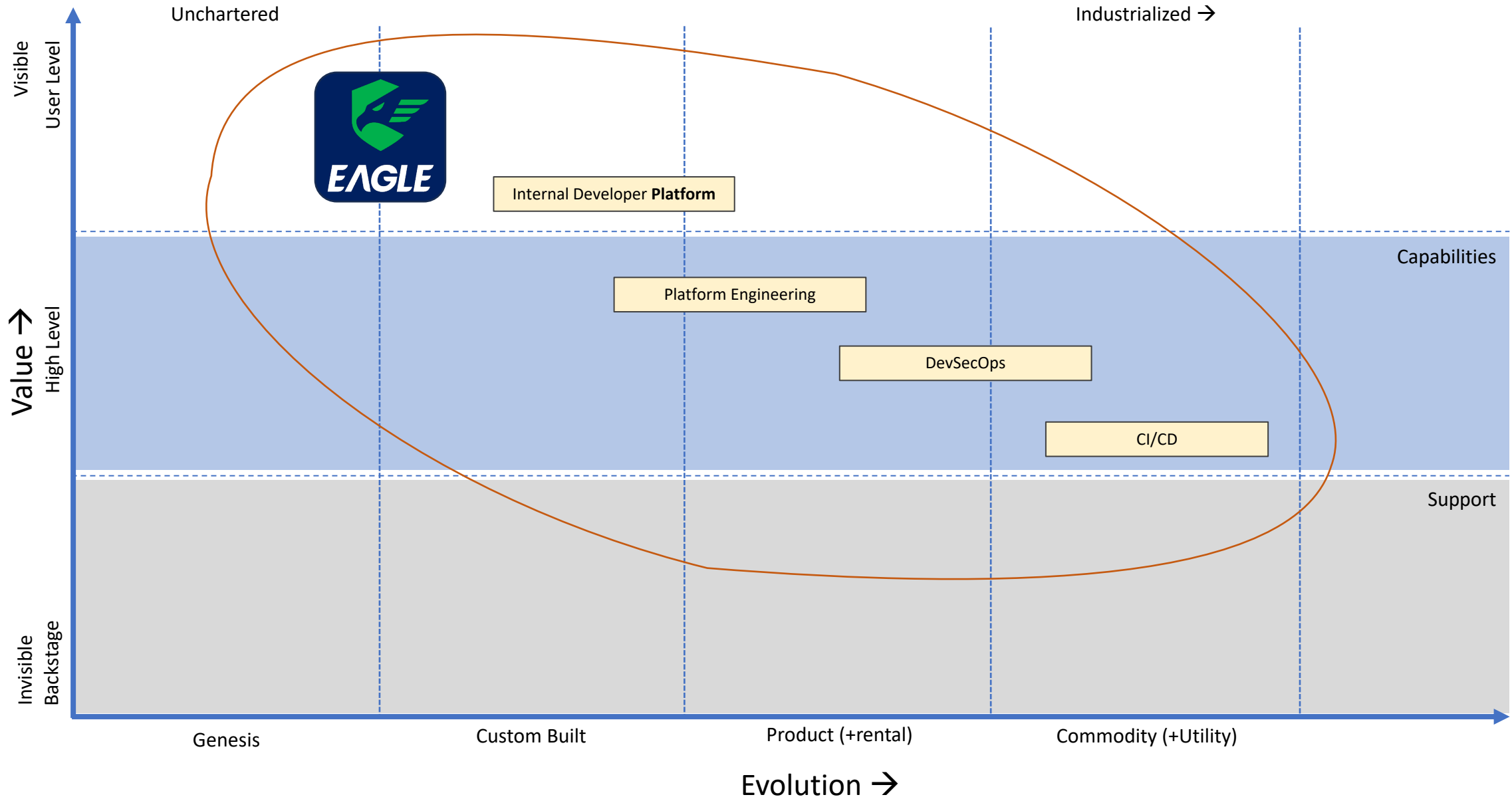


- Evolution*
- *Developer Experience*
  - *Platform Maturity*

# Practices - Wardley Map

○ Solution/Component

▭ Value/Need



# Building A Software Factory

# Low Code Application Platforms



Figure 1: Magic Quadrant for Enterprise Low-Code Application Platforms



Source: Gartner (August 2022)

*Scripts, Tools & Systems*

- *Design Systems*
- *Generators (Angular Schematics, hygen, projen, amplify, cdk)*



AWS Service Catalog

CLOUD FOUNDRY

OPENSIFT

HEROKU

Qovery

OpsLevel

humanitec

Coherence

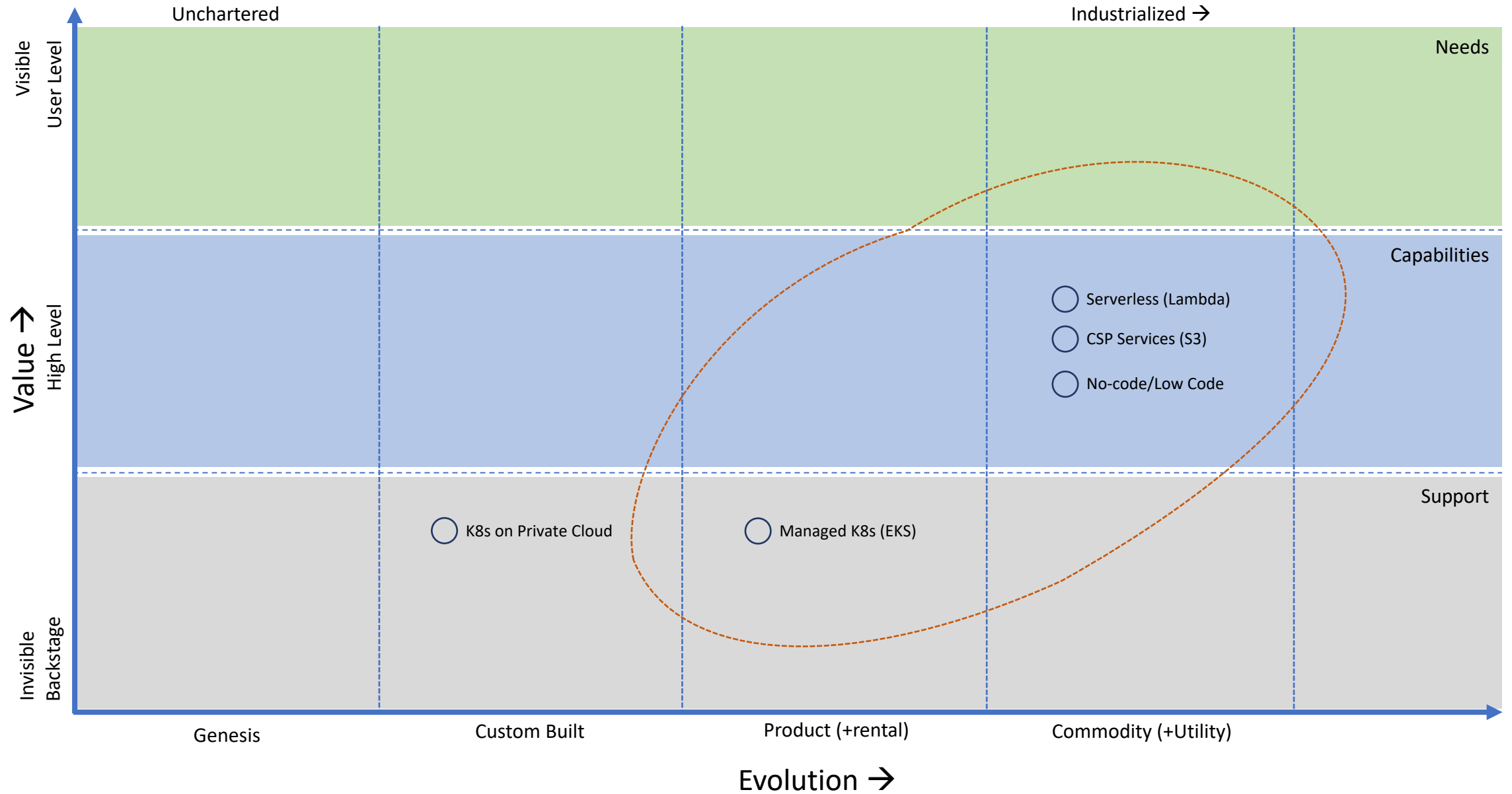


EAGLE

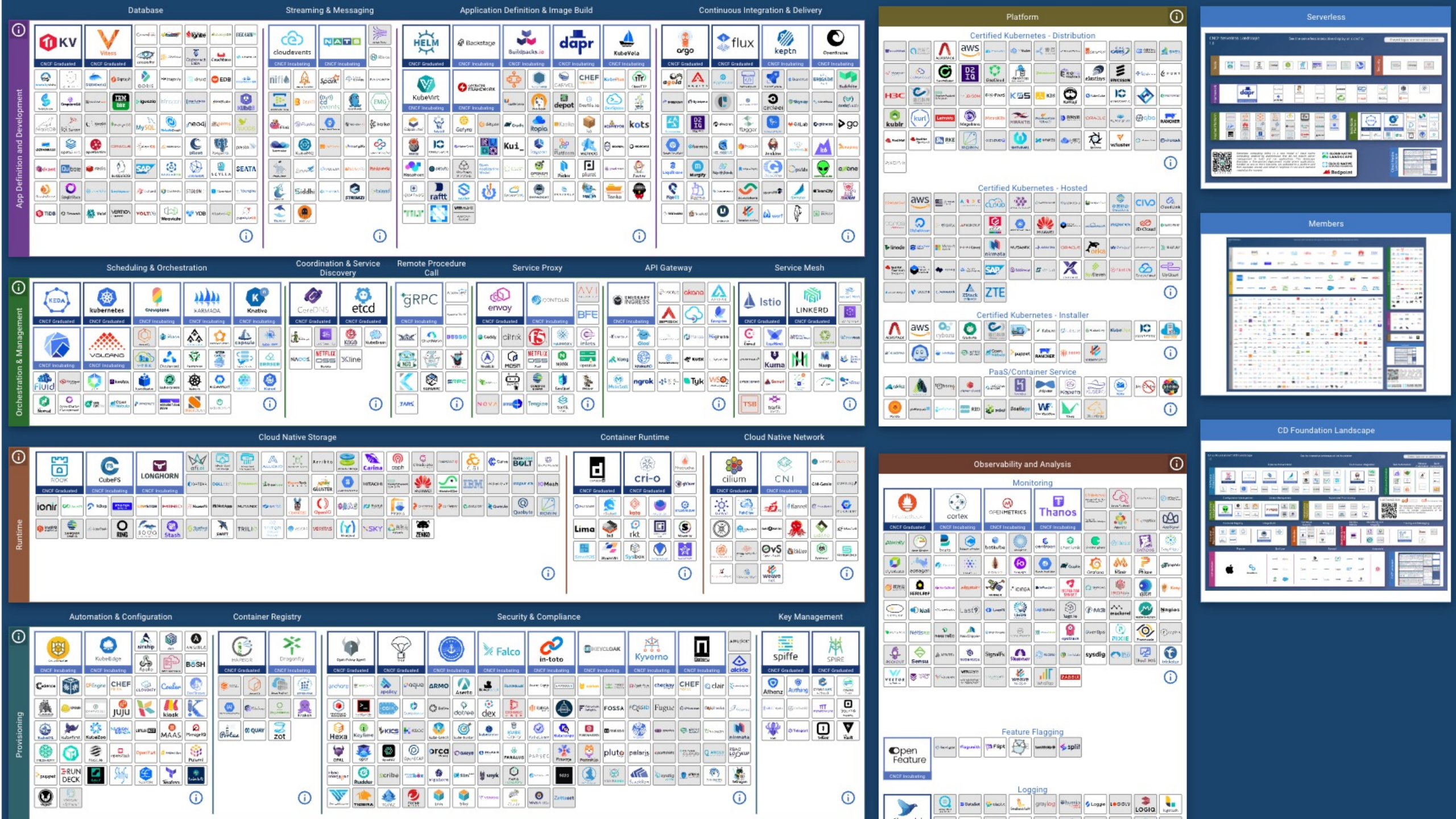
# Workloads - Wardley Map

○ Solution/Component

Value/Need







Database

Streaming & Messaging

Application Definition & Image Build

Continuous Integration & Delivery

Platform

Serverless

App Definition and Development

Scheduling & Orchestration

Coordination & Service Discovery

Remote Procedure Call

Service Proxy

API Gateway

Service Mesh

Orchestration & Management

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Members

CD Foundation Landscape

Runtime

Cloud Native Storage

Container Runtime

Cloud Native Network

Observability and Analysis

Provisioning

Automation & Configuration

Container Registry

Security & Compliance

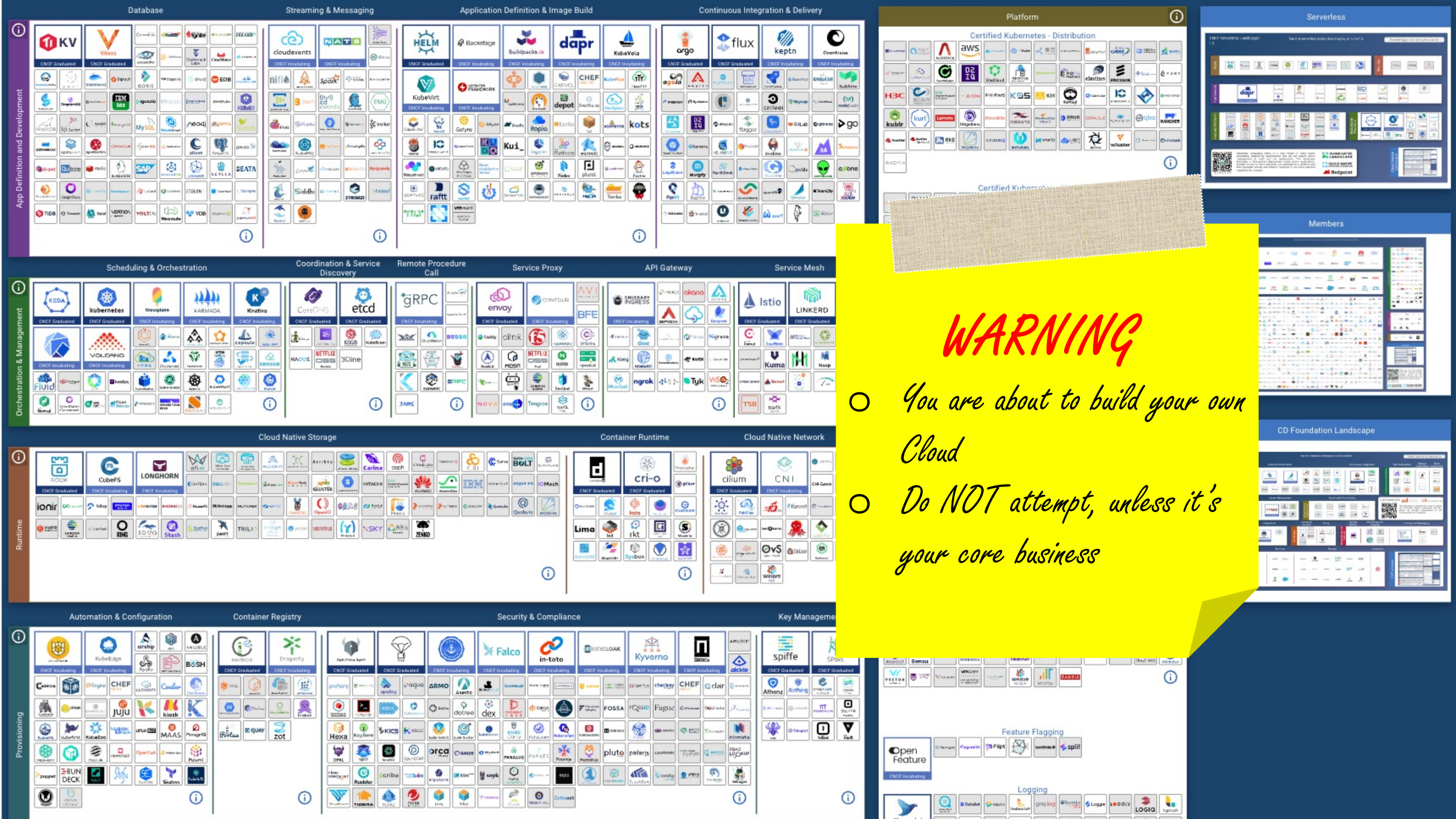
Key Management

Monitoring

Feature Flagging

Logging





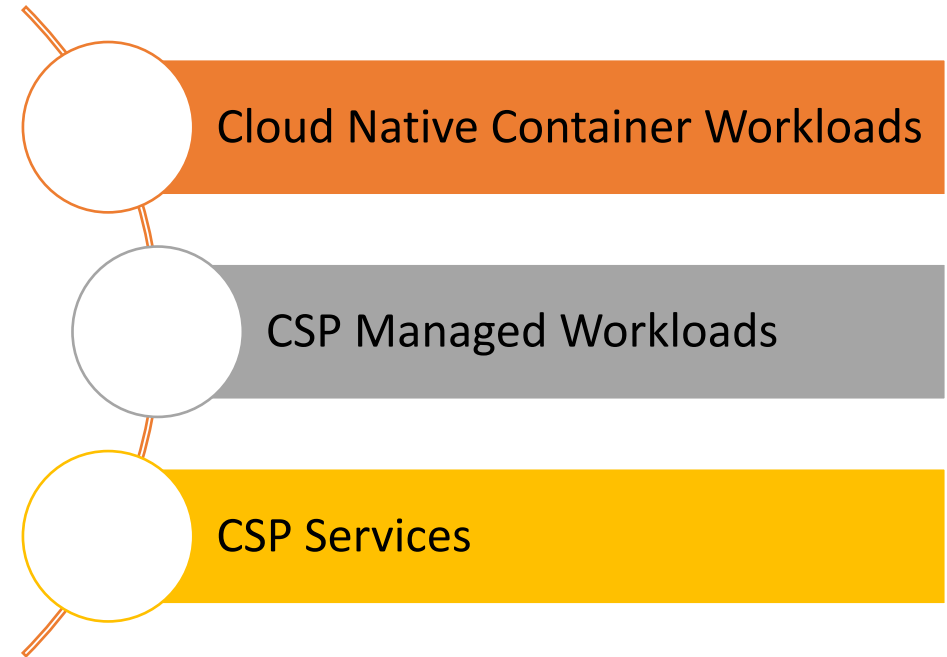
# WARNING

- You are about to build your own Cloud
- Do NOT attempt, unless it's your core business

# Unified Workflow



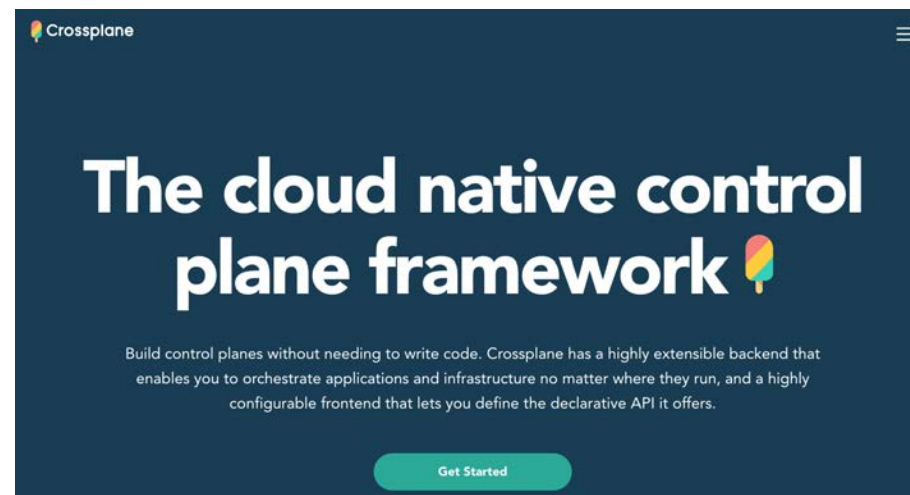
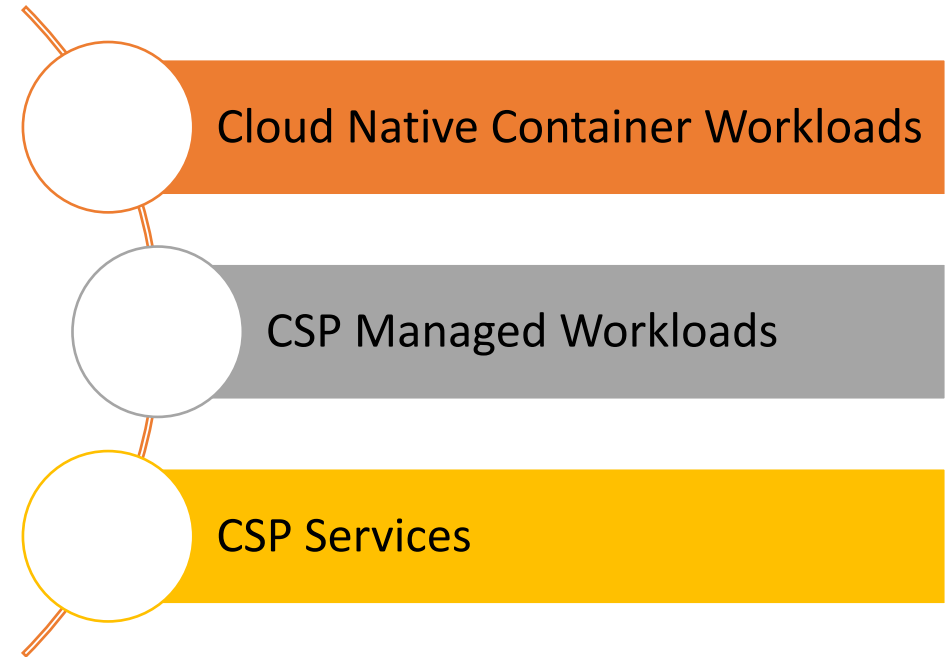
- Cloud agnostic container-based workloads
  - Cloud agnostic, based on Kubernetes
- CSP managed workloads
  - Serverless (AWS Lambda, Batch, Aurora ...)
- CSP managed services
  - S3, SQS, EventBridge, Kinesis ...
  - AI/ML Services



# Unified Workflow



- Cloud agnostic container-based workloads
  - Cloud agnostic, based on Kubernetes
- CSP managed workloads
  - Serverless (AWS Lambda, Batch, Aurora ...)
- CSP managed services
  - S3, SQS, EventBridge, Kinesis ...
  - AI/ML Services



# What Works



- Do **NOT** build your own Cloud
- **Control** and **craft** are “*exponentially expensive*”
  - Focus on “*value*”
- Start with the most important, “*golden path*”
  - Create a Journey/Value Stream map
  - Document the journey
  - Automate “most expensive” steps
  - Operate, monitor and learn
  - Optimize
  - Rinse. Repeat
- Leverage public cloud
  - Serverless
  - Fully managed services



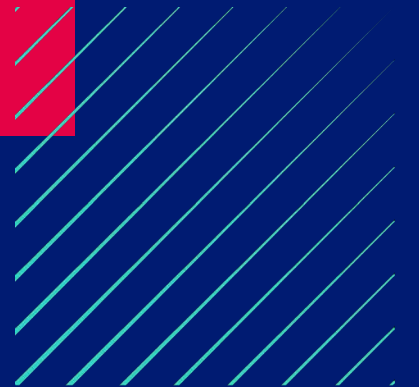
## *Research & Planning Tools*

- *Journey Map*
- *Value Stream Mapping*
- *Service Blueprint*



*"You never change things by fighting the existing reality. To change something, **build a new model** that makes the existing model obsolete."*

R. Buckminster Fuller



**SOFTRAMS**

**Thank You!**

<https://bit.ly/conf42-murali>

