

# Evolute Migrate

## AI-based Containerization

---

Kristopher Francisco



krfrancisco@evolute.io



408.314 3523



2445 Augustine Drive  
Santa Clara | CA | 95054  
United States



# Agenda



Overview



Evolute Migrate

- BCD Framework
- Demo



Scalability Patterns



# Overview

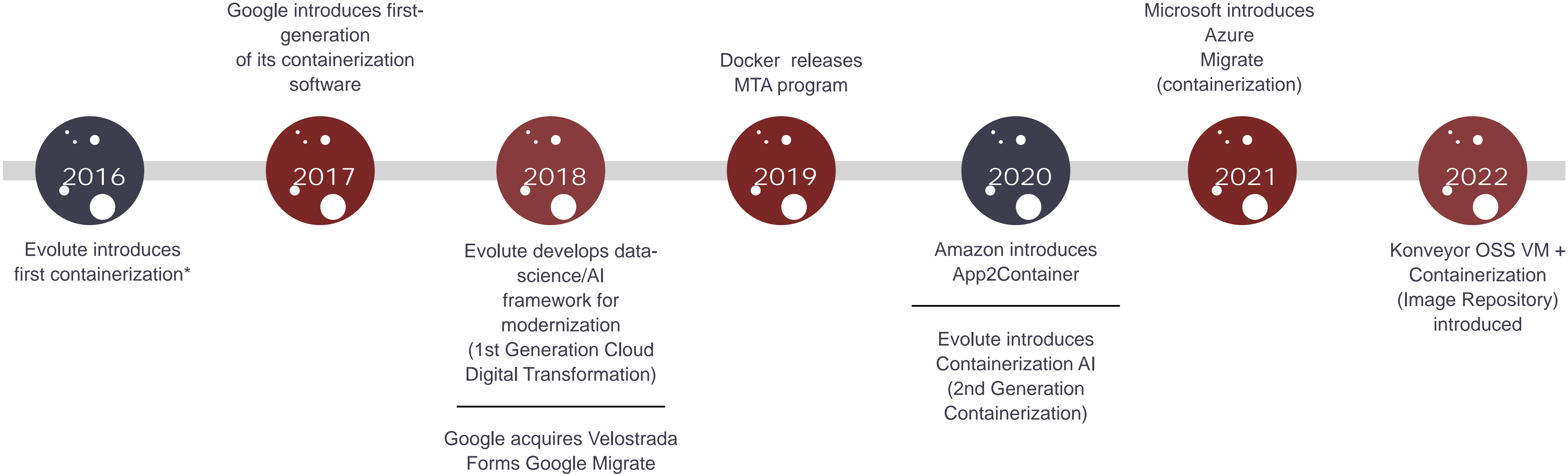


History



# Timeline

## First and Most Comprehensive



\* USPTO 62797200





# Evolute Migrate

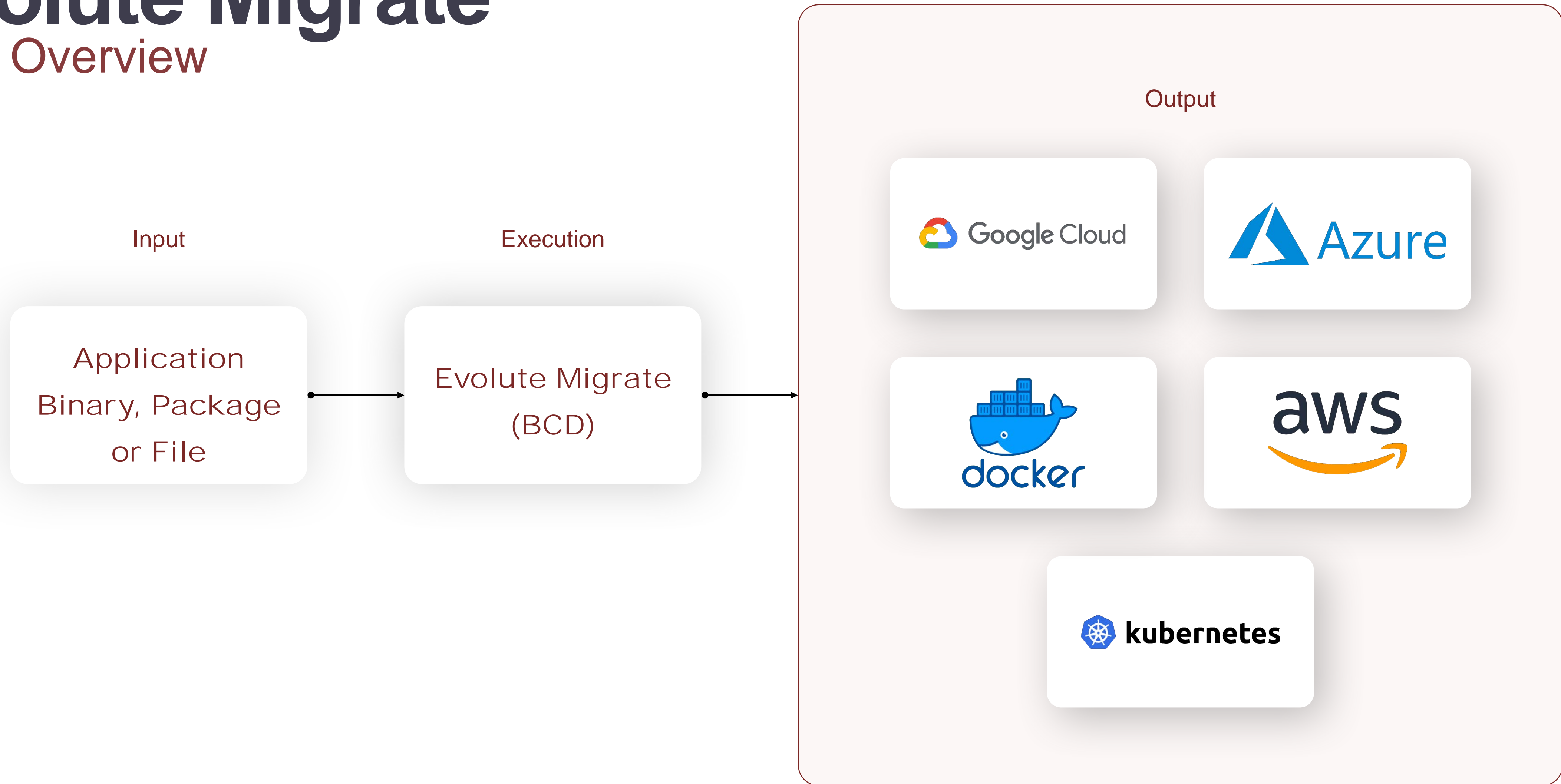


## BCD Overview



# Evolute Migrate

## BCD Overview





# Demo

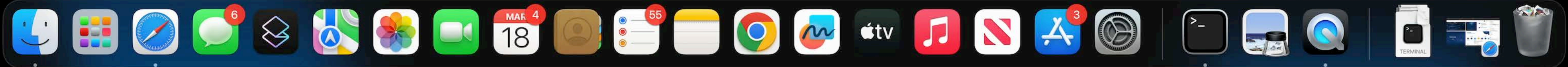


kriss — root@evo-mgr-0009:~ — ssh evo-mgr-0009 -l ec2-user — 92x53

```
(p36) [root@evo-mgr-0009 ~]#
```

kriss — root@evo-gapps-0008:~ — ssh -l ec2-user guest-apps0008 — 107x53

```
[root@evo-gapps-0008 ~]#
```





# Evolute Migrate



BCD Deep Dive



# AI Framework

## BCD

### Classification

Leverages K-Means Clustering via Unsupervised ML

### Containerization

- Leverages Decision Trees via Supervised ML
- ~3100 Base Paths (OS, Classification Input, Output Artifact, Destination Platform)

### Runtime

- Run In Any Target Docker/K8S Platform
- Start Services
- Separate Components



# Evolute Migrate Model

## BCD - Binary Configuration Data



### CLASSIFICATION

Binary, Package or File

Example binary - mysqld

Example package -  
ftp-0.17-67.el7.x86\_64.rpm

Example binaries(nginx, mssql,  
customer vendor software)

PROFILE



### CONFIGURATION

Binary Configuration Data Mapping

Software Language Evaluation

Library and  
Dependencies Mapping

OS Source and Destination  
Mapping

Software Static and Dynamic  
Variable Analysis

Storage and Data Transformation

TRANSFORM



### CONTAINERIZE

Properly Instantiate

Separate Services

Start Order

Proper Clustering

Scalability

RUN



# Evolute Migrate



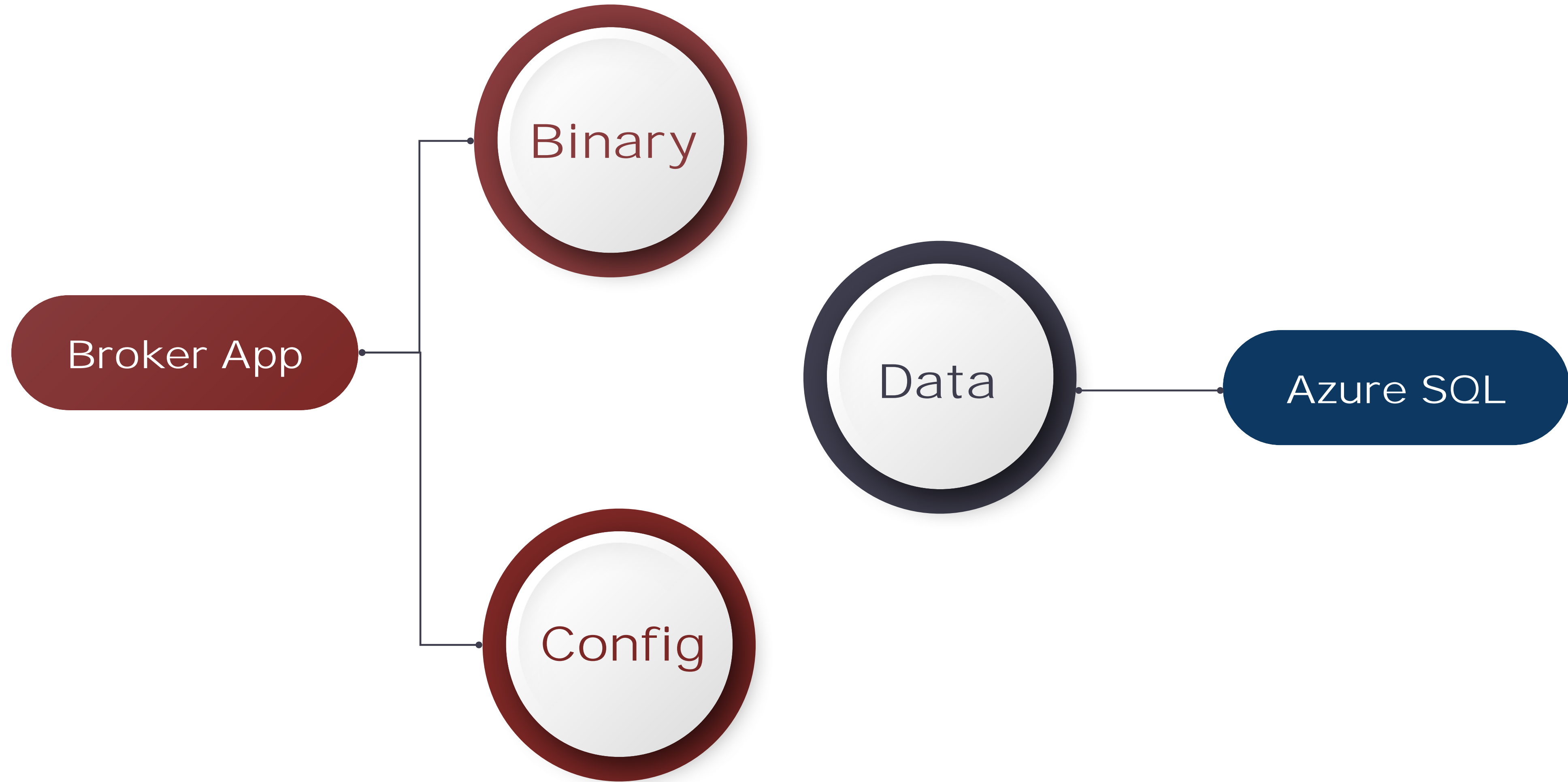
Example App

In-House Developed Software



# Evolute Migrate

Example Software



Note: Not all components are depicted (e.g. client side software, serverless API, et al)



# Evolute Migrate

## Software Manifest

### Broker App

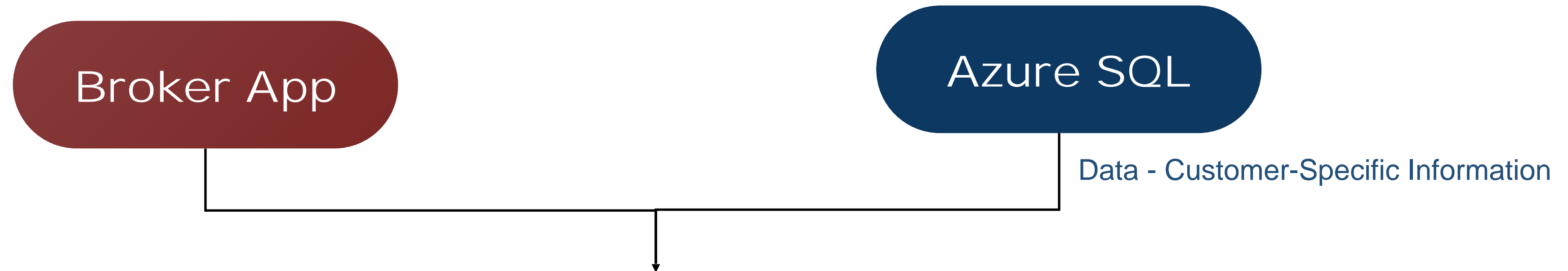
- Binary - Software Runtime
- Configuration - Unique Variables Make Instance Unique

```
1. FROM mcr.microsoft.com/windows/servercore:ltsc2022
2. SHELL ["cmd", "/c"]
3.
4.
5. # Runtime dependencies
6. ADD ["https://download.microsoft.com/download/9/3/F/93FCF1E7-E6A4-478B-96E7-D4B285925B00/vc_redist.x64.exe" , "C:\\\\msvcrt-2015-x64.exe"]
7. RUN C:\msvcrt-2015-x64.exe /quiet /install /wait && del C:\msvcrt-2015-x64.exe || exit 0
8. ADD ["https://download.microsoft.com/download/9/3/F/93FCF1E7-E6A4-478B-96E7-D4B285925B00/vc_redist.x86.exe" , "C:\\\\msvcrt-2015-x86.exe"]
9. RUN C:\msvcrt-2015-x86.exe /quiet /install /wait && del C:\msvcrt-2015-x86.exe || exit 0
10.
11. # Application: Broker Services
12. RUN md "C:\\Program Files (x86)\\Custom Company" || exit 0
13. COPY ["C:\\Program Files (x86)\\Custom Company\\Broker Services", "C:\\Program Files (x86)\\Custom Company\\Broker Services"]
14. RUN sc create "BrokerService" binpath="\"C:\\Program Files (x86)\\Custom Company\\Broker Services\\Custom Company.OutlookIntegration.BrokerService.exe\" start=auto DisplayName="Broker Broker Service"
15.
16. # Setup/Create MSMQ Queues
17. SHELL ["powershell", "-command"]
18. RUN Set-Item Env:PSModulePath -Value 'C:\Windows\system32\WindowsPowerShell\v1.0\Modules;C:\Program Files\WindowsPowerShell\Modules;C:\Users\ContainerAdministrator\Documents\WindowsPowerShell\Modules'
19.
20. RUN Install-WindowsFeature MSMQ-Services
21. RUN New-MsmqQueue -Name "broker" -Label "broker" -QueueType Private -Transactional | Set-MsmqQueueAcl -UserName "Everyone" -Allow FullControl
22. RUN New-MsmqQueue -Name "booking" -Label "booking" -QueueType Private -Transactional | Set-MsmqQueueAcl -UserName "Everyone" -Allow FullControl
23. RUN New-MsmqQueue -Name "messageprocessor" -Label "messageprocessor" -QueueType Private -Transactional | Set-MsmqQueueAcl -UserName "Everyone" -Allow FullControl
24.
25.
26. #Create Services
27. SHELL ["cmd", "/c"]
28. RUN sc create "DataPullService" binpath="\"C:\\Program Files (x86)\\Custom Company\\Broker Services\\Custom Company.OutlookIntegration.DataPullService.exe\" start=auto DisplayName="Broker Data Pull Service"
29. RUN sc create "MessageProcessorService" binpath="\"C:\\Program Files (x86)\\Custom Company\\Broker Services\\Custom Company.OutlookIntegration.MessageProcessorService.exe\" start=auto DisplayName="Broker Message Processor Service"
30. RUN sc create "DataWriterService" binpath="\"C:\\Program Files (x86)\\Custom Company\\Broker Services\\Custom Company.OutlookIntegration.DataWriterService.exe\" start=auto DisplayName="Broker Data Writer Service"
```



# Evolute Migrate

## App Config (Example)



Config

```
<connectionStrings>  
<add name="BrokerOUContainer" connectionString="metadata=res://*/Entities.Custom CompanyOutlookIntegrationModel.csdl|res://*/Entities.Custom CompanyOutlookIntegrationModel.ssd|res://*/Entities.Custom CompanyOutlookIntegrationModel.msl;provider=System.Data.SqlClient;provider connection string="data source=tcp:companyBrokerbroker-dbsrv-poc.database.windows.net,1433;initial catalog=3041_BrokerOutloInteg;User id=Brokeruser;password=V2jLqM9kPb3xRtG7c;multipleactiveresultsets=True;App=EntityFramework"; providerName="System.Data.EntityClient"/>  
</connectionStrings>
```



# Architecture + Scalability

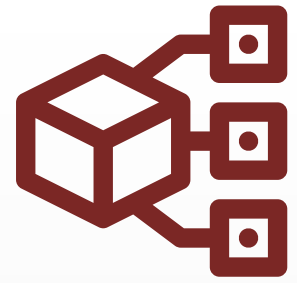


## Scalability Patterns



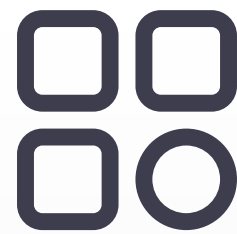
# Scalability Patterns

## Deterministic Architectures



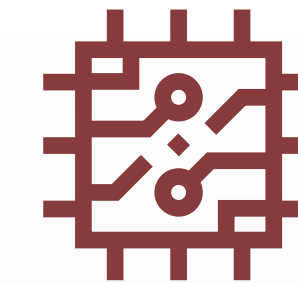
By Application  
Function

ex: Shopping Cart  
function on e-  
commerce app



By Resource  
Demand

ex: component requiring  
more hardware/network  
resources



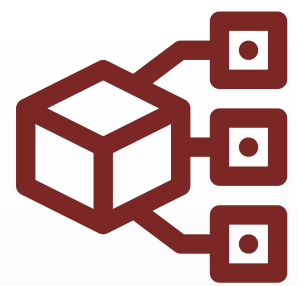
By Component

ex: nginx package  
installed on OS



# Scalability Patterns

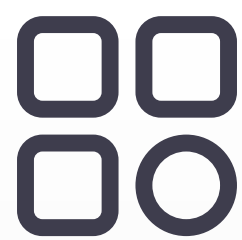
## Deterministic Architectures



By Application  
Function

Scalability Pattern:  $m \times n$

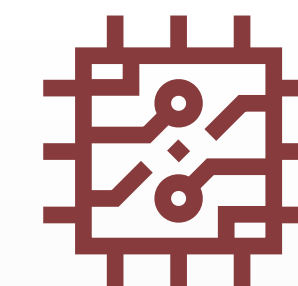
Implementation: e.g.,  
**Deployments + Services**



By Resource  
Demand

Scalability Pattern:  $n+1$

Implementation: e.g. **Pod  
Autoscaler**



By Component

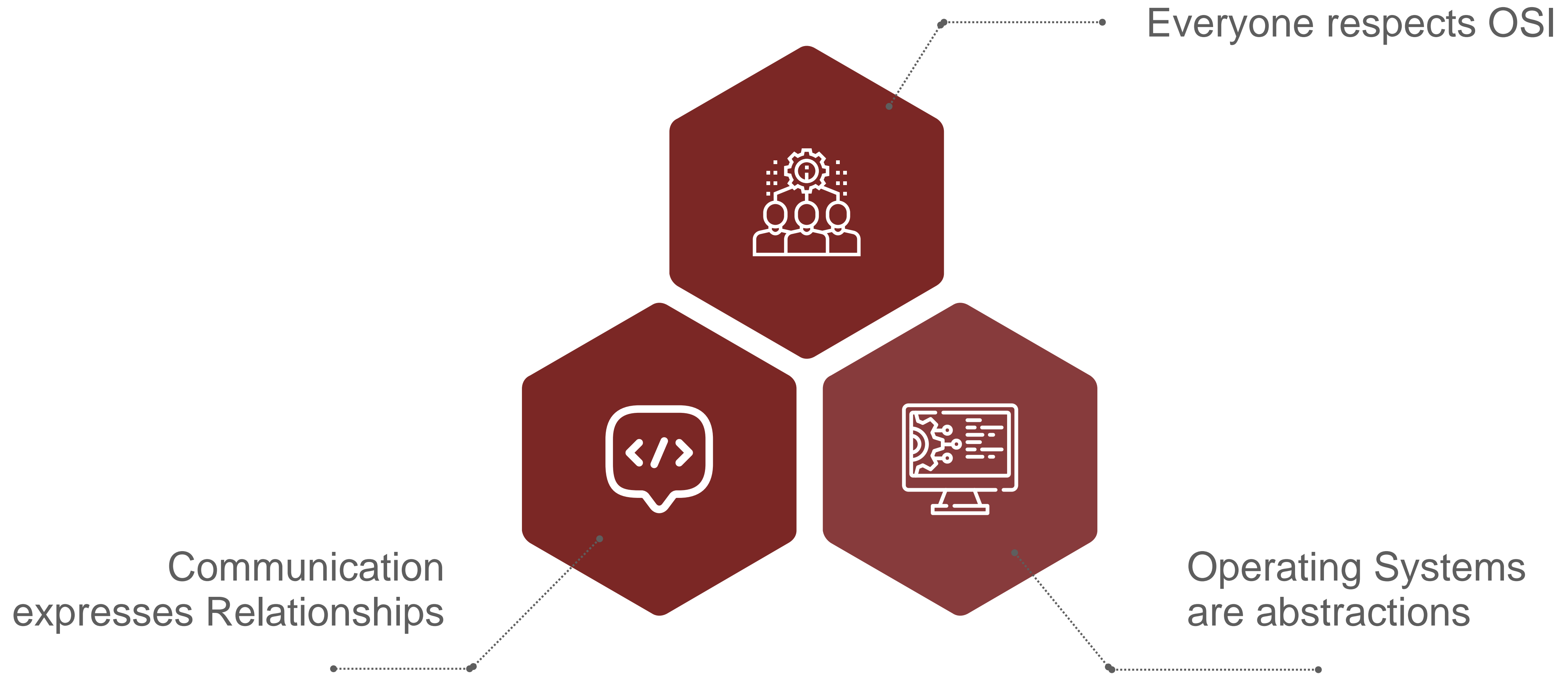
Scalability Pattern:  $n+1$

Implementation: e.g.,  
**Deployment Count**



# Scalability Patterns

## A Few Computational Truths

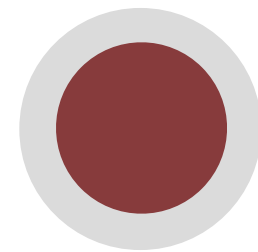
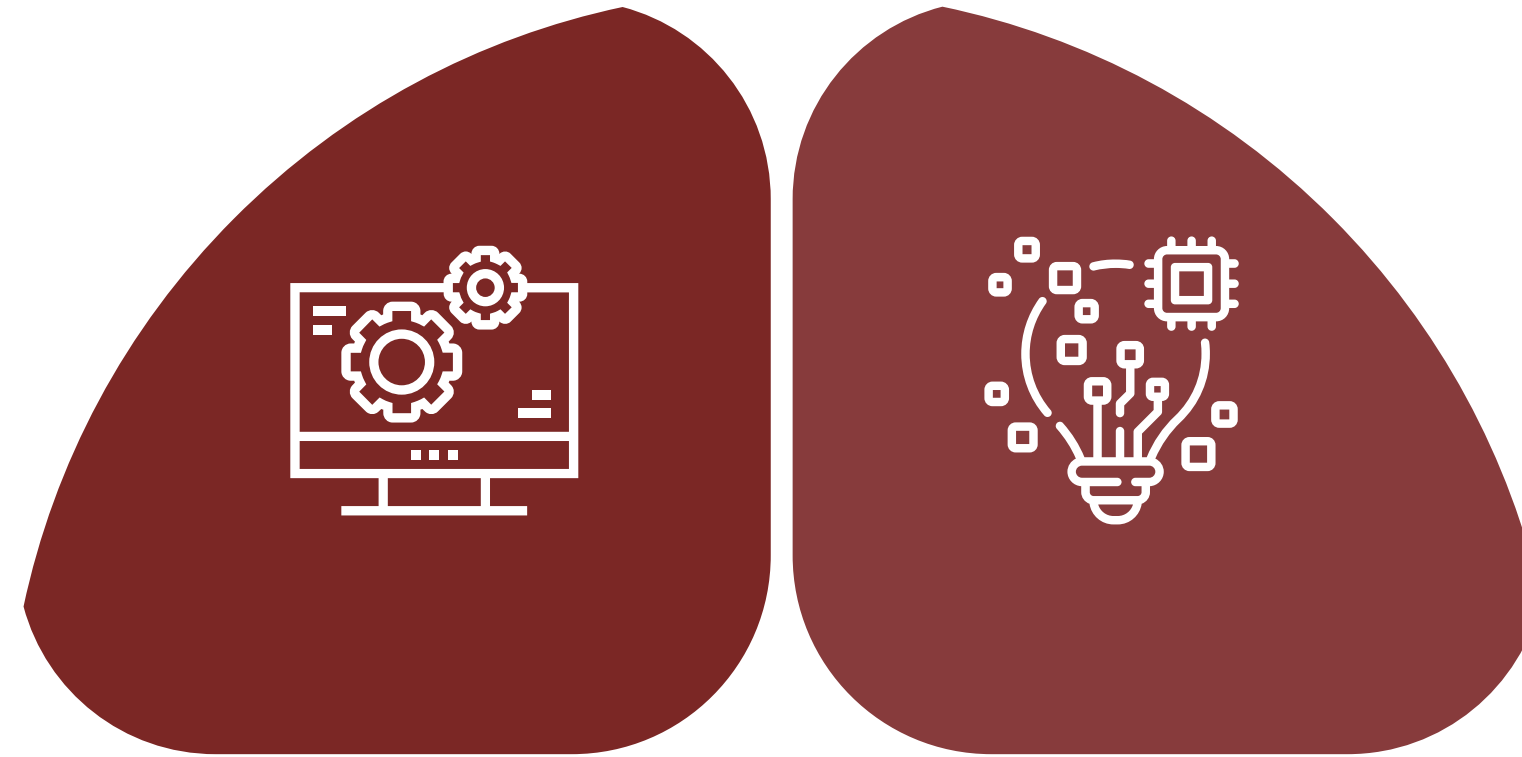
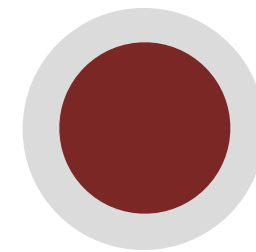




# Scalability Patterns

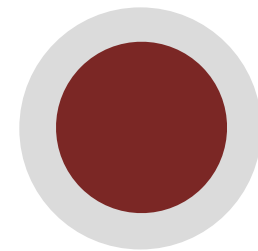
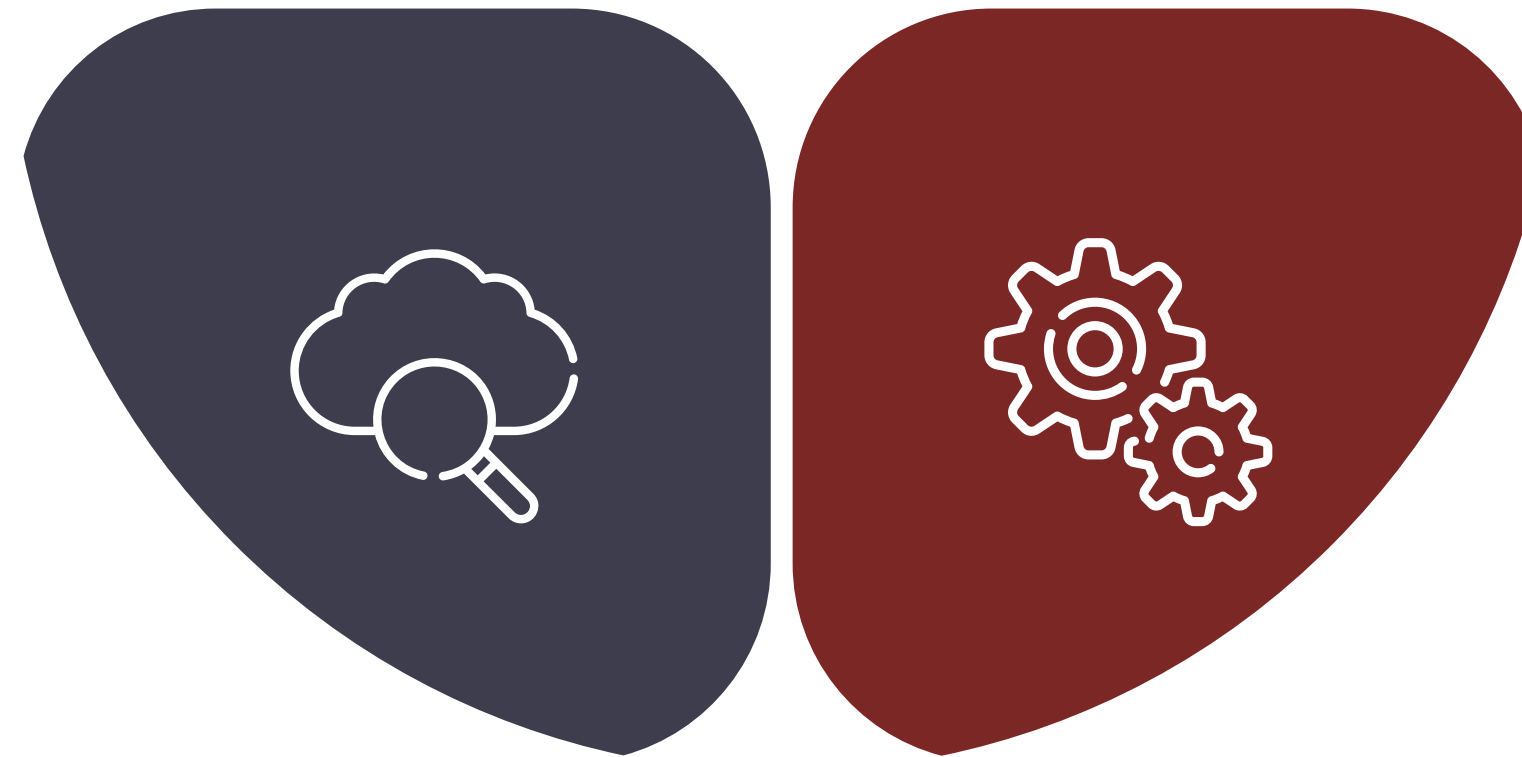
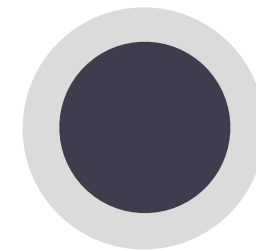
## Use Cases

Edge Computing  
MEC  
Distributed Workloads



Enterprise Digital  
Transformation  
Software Modernization  
OS Modernization

Hybrid/Multi Cloud  
Public Cloud (AWS, Azure, GCP)  
Private Cloud



Function Driven  
Environments  
Serverless/Lambda  
No Code/Low Code



# Thank You

---

Kristopher Francisco



krfrancisco@evolute.io



+1 650.382.4058



2445 Augustine Drive  
Santa Clara | CA | 95054  
United States

CONTACT US TO  
LEARN MORE!