Considerations & pitfalls when migrating from on-premise to public cloud

SACON NY | February 28, 2018
WHO AM I?

Rosemary Wang
@joatmon08
Platform & Infra @Thoughtworks
WHAT’S “SOFTWARE ARCHITECTURE”?

Software = Applications

Architecture = Technical
WHAT’S “SOFTWARE ARCHITECTURE”?

**Software** = Applications
+ Domains + Infrastructure

**Architecture** = Technical
+ Data + Security + System + Operations
OUR CLIENTS MOVE TO PUBLIC CLOUD BECAUSE...

“We want to change quickly.”

“We want to elastically scale.”

“We want to utilize cloud/SaaS offerings.”
OTHER MIGRATION PATTERNS

On-Premise to Public Cloud

Hybrid Cloud

Traffic Shifting
Pick the First Candidate.
YOUR FIRST CANDIDATE...

**Important but not Critical**

- Greenfield vs. Brownfield
- Lift & Shift vs. Cloud-Native
- Client-Facing vs. Internal
- Monolith vs. Microservice
Important but not Critical

prove VALUE

Greenfield vs. Brownfield

Lift & Shift vs. Cloud-Native

Client-Facing vs. Internal

Monolith vs. Microservice
FOR DATA, CONSIDER...

regulatory limitations

data transport

dataset size

cost

performance

backups
Remember the Domain.
REMIND YOURSELF...

*business domain*

*user flow*

*secret sauce*

*not the*

*technical solution*
A MISCONCEPTION

Public Cloud Migration is **not** Digital Transformation.
Use Build Pipelines.
“..is an automated manifestation of your process for getting software from version control into the hands of your users.”

Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation
manifest = technical architecture
"we are using containers"

control = auditing architecture
"no direct access"
When I need to deploy anything, then I should do it through a pipeline.
PIPERINES EVOLVE OVER TIME

Template Generators ➔ Shared Libraries ➔ Pipeline-as-Code ➔ Pipeline-as-Service
Apply Patterns to Infrastructure.
THE ANTI-PATTERN

On-Premise Architecture = Public Cloud Architecture
SELF-SERVICE

Infrastructure Services → Platform Services → Self-Service
<table>
<thead>
<tr>
<th>unit test</th>
<th>deploy</th>
<th>integration test</th>
<th>smoke test</th>
</tr>
</thead>
<tbody>
<tr>
<td>inspec exec -t docker://</td>
<td>docker push</td>
<td>docker compose</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="InSpec Logo" /></td>
<td><img src="image" alt="Docker Logo" /></td>
<td><img src="image" alt="Docker Compose Logo" /></td>
<td></td>
</tr>
<tr>
<td>terraform plan</td>
<td>terraform apply</td>
<td>inspec exec</td>
<td>“I should be able to deploy a ___ on my platform.”</td>
</tr>
</tbody>
</table>
Standardization over Customization.
REPLACE HOMEGROWN SOLUTIONS WITH INDUSTRY TOOLS

Homegrown Image Builder

→

Industry Tool

CHEF

HashiCorp Packer
Security First.
WE RECOGNIZE THAT...

On-Premise Security ≠ Public Cloud Security
The Anti-Pattern

We think this policy or configuration works...

(we’ll figure it out later!)

* add to the list of tech debt
THE ANTI-PATTERN

We think this policy/configuration works... (we'll figure it out later!)

* add to the list of tech debt

Misconfigured Amazon Web Services **bucket** exposes 12000 social ...
SC Magazine - Feb 5, 2018
Another misconfigured Amazon Web Services (AWS) **S3** cloud storage **bucket** has been left insecure this time exposing the sensitive data of 12,000 social...
Researchers noted that while no financial information was **compromised**, the leak damages the brands credibility and more importantly exposes...
EVERYTHING SECURITY-RELATED...

Identity Provider

Secrets Management

Artifact Management

should be

Self-Service

Pipelines

Cloud-Friendly
Reduce the Risk.
Is your system **performant**?

Is your system **resilient**?

Where are your **bottlenecks**?

Which **parts** need to be highly available?
TAKE SOME MEASUREMENTS

client experience (business metrics)  mean time to recovery (devops metrics)
Create a Cloud Culture.
product teams learn operations & observability
infrastructure & security teams learn development & testing
EVERYONE (PRODUCT, INFRA, & SECURITY)...

uses similar **developer workspaces**.

learns **containers**.

learns **pipelines**.

**breaks habit** of direct access.
Pick the First Candidate.
Remember the Domain.
Use Build Pipelines.
Apply Patterns to Infrastructure.
Standardization over Customization.
Security First.
Reduce the Risk.
Create a Cloud Culture.
A comprehensive look at the path to cloud migrations

Despite roadblocks in the process, cloud migrations can bring significant rewards. To successfully move to the cloud, organizations need to craft a migration strategy, assess risks and plan for surprises.

Cloud Design Patterns

Depicts common problems in designing cloud-hosted applications and design patterns that offer guidance.

Gartner Identifies Five Ways to Migrate Applications to the Cloud

Migrating your Existing Applications to the AWS Cloud

A Phase-driven Approach to Cloud Migration
maps can be useful

journeys will always differ

move only what needs