Developing a Successful Cloud Strategy

Hemant Kishan
VP of Client Engagement and COO

Tanya Atanasova
Cloud Solutions Engineer

3/23/2017
Agenda

• Intros and objectives

• The Why

• The How
  • Cloud Journey (Business Case, Strategy and Implementation)

• Demo

• Q&A
Objective

• The purpose of this workshop is to explore some of the critical factors to consider in framing a successful Cloud Strategy
  • More on the “How” not the “Why”

• Context: RightBrain Networks is an Ann Arbor-based Cloud Consultancy that is an Advanced Partner to AWS and a Silver Partner to Azure

• The strategy is based on real-world examples and client situations based on our experience in delivering cloud services
  • “YMMV”
Moving to the Cloud -- The Why
Moving to the Cloud -- The Why

• Cloud economics
• Speed and agility
• Security and risk management
• Impact to IT budget headline numbers
  • Total spend
  • Mix of spend
• Cloud Transformation -- leverage related initiatives into a broader Program
  • DevOps
  • Application modernization
  • Innovation
  • Big Data and other initiatives ....
Moving to the Cloud -- The How
4-Step Phased Cloud Journey

1. PoC and overall Cloud Business Case

2. Gaining experience through initial Cloud Deployments

3. Comprehensive Strategy with known end-state
   a. Hybrid vs. “All In”

4. Implementing a Cloud Transformation Program
1. Proof-of-Concept and Business Case
Assessing Cloud Features through a PoC

• Key objectives:
  • Validate networking and security
  • Stand up services (compute, storage, network)
  • Assess learning curve

• Critical Success Factors:
  • PoC success metrics (speed, cost, etc)
  • Select a relatively simple workload for the PoC (eg., simple app, test environment, etc.)
PoC Examples

• Technology evaluation
  • Product assessment

• Solution prototyping

• Test environment

• Simple application deployment
PoC Example
Business Case

• Current TCO of applications and workloads

• Projecting future state Cloud cost footprint

• Operational efficiency

• Ability to serve the business
2. Gaining Experience with the Cloud
2. Initial Cloud Deployments

• Assumes business case is promising and PoC was successful

• Example of initial production deployments:
  • Backup
  • DR site
  • Public website
  • Dev/Test environments
  • New Initiatives
    • Mobile apps
    • Big Data and BI

• Startups are “all in” from inception
3. Comprehensive Strategy
3. Comprehensive Strategy

• Large enterprises can have thousands of applications
  • That differ in:
    • RoI (TCO before and after)
    • Tech stack
    • ALM stage
• Migration Roadmap
  • What are we trying to optimize?
    • Cost, speed, risk, etc.
  • How are we selecting the migration approach?
• End-state
  • “All In” or Hybrid
  • Weight of the legacy footprint

Key Questions -- avoid “boiling the ocean”:
• Where do you start?
• What makes the most business sense?
• What migration approaches fit which applications?
Migration Approaches

• Lift & Shift -- replace like for like
  • Least effort but with minimal ability to take advantage of Cloud features
  • Can be done in least amount of time

• Cloud Optimize -- One or both of the following:
  • DevOps: construct deployment pipeline
  • Architecture enhancement: use of ASG, AZ’s or other Cloud features

• Cloud Native -- refactor/rewrite to leverage cloud native services (lambda, etc)
  • Most investment but also maximum benefits
APM and ALM

• Assessing application characteristics through Application Portfolio Management and Application Lifecycle Management is a key part of building the roadmap
  • Intersection with EA
  • Understanding whether an application is strategic or approaching EOL
  • Factoring in business criticality of an application
• For simplicity, we will bucket applications into two buckets:
  • Keep: Strategic application with anticipated continued investment
  • Deprecate: Application that is approaching EOL and/or known be replaced or retired
Summary of Factors to consider in building the Roadmap

• Inputs from ALM
  • Classified as:
    • **Deprecate**: Targeted for retirement, or replacement
    • **Keep**: Business standard and/or strategic

• Inputs from APM
  • TCO and expected Cloud savings

• Cloud Migration Approaches
  • Lift & Shift -- replace like for like
  • Cloud Optimize -- One or both of the following:
    • DevOps: construct deployment pipeline
    • Architecture enhancement: use of ASG, AZ’s or other AWS features
  • Cloud Native -- refactor/rewrite to leverage cloud native services (lambda, etc)
APM/ALM: Key Information for each app

- Application technology
- COTS or custom -- candidate for SaaS?
- Business capability mapping
- Capacity requirement: compute, storage, network
- TCO
- Number of instances
- Utilization
- EOL status

- Identified successor app
- Business strategic
- Business criticality
- BCP requirement
- Number of users
- Stateless behavior
- Usage pattern (steady, predictable peaks, unpredictable)
- Releases/year
- Deployment pipeline
- Future-state:
  - Target architecture
  - Next state TCO, Optimized state TCO

Use ALM insights to broadly classify apps as:

- Keep -- planned to be part of future app portfolio
- Deprecate -- planned to be retired or replaced
Migration Priority: Which applications do we migrate first?

- **Expected Cloud Savings**
  - **High**
  - **Low**

- **Deprecate**
- **Keep**

### Why bother?
- Identify number of applications in each quadrant
- Calculate total savings per quadrant

### 1. Expected High Savings
- These are "keeper" applications that will deliver the highest savings

### 2. Expected Low Savings
- Keep

### 3. Expected High Savings
- Deprecate

### 4. Expected Low Savings
- Why bother?

**Output:**
- Prioritized set of apps by BU and expected savings
Migration Approach: Which approach makes the most sense?

<table>
<thead>
<tr>
<th>New Apps</th>
<th>Existing Apps</th>
<th>Deprecate</th>
</tr>
</thead>
<tbody>
<tr>
<td>COTS</td>
<td>CO</td>
<td>L&amp;S</td>
</tr>
<tr>
<td>Custom</td>
<td></td>
<td>“Appliance”</td>
</tr>
</tbody>
</table>

Notes:
- Appliance mode: zero to minimal “care and feeding” -- for those apps that were in Category 3. Some exceptions can be CO.
- Most Category 1 and 2 apps will be CO; a small subset of custom apps may be candidates for CN (e.g., Stateful to Stateless)
- For new custom apps: In-going assumption is for CN (a small subset could be CO)

Legend:
- CO: Cloud Optimize
- CN: Cloud Native
- L&S: Lift and shift

Legend:
- CO: Cloud Optimize
- CN: Cloud Native
- L&S: Lift and shift
Summary

• Rigorous application portfolio rationalization will allow us to separate the “wheat from the chaff”

• Based on a nuanced understanding of the application’s strategic value to the business and calculated cost savings, we can construct a roadmap for each Category of app (category 1, 2, 3 and 4)

• Based on application category, a default approach can be established for existing as well as new apps
4. Implementing a Cloud Transformation Program
Building Blocks of the Model: CTMO, CCoE, and CIF

1. Cloud Transformation Management Office
   • The “control center” that drives the program

2. Cloud Center of Excellence
   • An Enterprise capability of enablers

3. Cloud Implementation Factory
   • The execution arm
Overall Implementation Model

Govern

CTMO

Enable

CCoE

- “Force Multiplier”
- Drive efficiency and consistency

Execute

CIF

- Plan, measure
- Drive collaboration and learnings

- BU teams
- Tactical execution
CTMO (cloud transformation management office): The “Control Center”

• The CTMO plans, measures, governs and executes the Transformation Program:
  • Governance
    • Decision body, communication and escalation
  • Application Portfolio Assessment
  • Program Management
    • Plans, workstreams and resources
  • Business case and financial plan
    • Current state, next state, optimized state
CTMO: Governance

• Recommend:
  • Model: Federal/State
    • Federal: Focus on Common standards, shared services, standardized tooling. Optimize for CommerceHub
    • State: Focus on BU-specific needs and differentiated features
    • 2-Tier: Advisory and Management-level

• Governance output:
  • Mandated: Decisions, Prescriptive guidance
    • Applies to all stakeholders
  • Recommended: Guidelines, design patterns
    • Best practice
  • Exception process:
    • Allowance for unique circumstances
Cloud CoE: Common Set of Enablers

• Establish a CCoE to address the following:
  
  • Cloud Readiness
    • Readiness tollgate process
    • Migration Toolkit
  
  • Cloud Engineering
    • Standards and Toolbox
    • Sandbox environment
    • Cloud SME’s
  
  • Engagement Model
    • CCoE and CIF alignment and collaboration
Cloud CoE: Main Components

Migration Enablers
- Cloud Readiness Process
- Cloud Migration Toolkit

Future State Enablers
- Reference Architectures
- Common Standards, Services and Toolbox

Execution Enablers
- Cloud Sandbox
- Cloud SME's

CCoE and CIF Engagement Model
CCoE Details

• Acts as a force multiplier:
  • Provide expertise through SME’s
    • Translate good intentions into tangible value
    • Provide resources to meet needs of BU Teams
      • Standards & Toolbox
      • Cloud migration toolkit
  • Establish Sandbox environment for trying out ideas and building tools and services
  • Define a tollgate-based process for assessing cloud readiness
    • “Open book” format – known expectations and enabling resources
  • Underpin CoE model with robust collaboration
    • Defined engagement model
Cloud Implementation Factory: Execution Arm

• The CIF is a virtual team that implements the migration by leveraging the CCoE enablers and executing a “play” from a pre-defined Playbook:

  • Playbook
    • Lift and Shift
    • Cloud Optimize
    • Cloud Native
Execution @ the BU Team Level

Key Artifacts
- Current State architecture
- Future State architecture
- Cost Model (before/after)
- Readiness assessment plan
- Cloud migration plan

BU Team Resources
- Arch
- Engr Mgr
- Prod Mgr
- DBA
- DevOps
- Devs

Aligned Resources
- Ops
- CTMO
- CCoE
Execution Team

• Detail:
  • CTMO rep assigned to align with overall program plan
  • Cloud CoE provides shared resources (SME’s)

• Eg. of SME role: DevOps Engineer Role
  • Own CI and CD process and associated artifacts
  • “Infect the Host” model for disseminating knowledge re. core tools (eg., Chef, Bamboo, CloudFormation, Packer, etc.)
  • Responsible for coordination with Cloud CoE with regard to standards and best practices
  • Provide automated tools/scripts to developers for facilitating deployments as part of development lifecycle
Agenda

• Big Data example
• Best Practice discussion
Best Practice Examples

Typical patterns:

• IAC (or SDI)
• Scale out
• Scale up
  • Predictable scaling -- seasonal or scheduled
• On-demand environments
  • CloudFormation and automation
• Engineering deployment pipelines
• NCA (native cloud application) development
Questions, Comments

Thank You!