

Importance of the Well-Architected Framework

# **Migration Scenario: Migrating Web Applications to the AWS Cloud**

# 

**Use case**

Figure 1: CompanyACloud.com Architecture (Before and After Migration)

CompanyACloud.com is a customer-facing web application of company A, which serves as a marketing portal and a customer management system. Customers, partners and employees use the web application to collaborate with each other using a rich web interface that can be viewed in a standard internet browser. CompanyACloud.com lists a complete catalog of products and their details. As new product announcements are made, marketing campaigns generate substantial amounts of traffic to the site resulting in periodic spikes. Outside of the timeframes caused by these spikes, CompanyACloud.com experiences a fairly steady and predictable traffic load, which is characteristically high on weekdays and low on weekends. The website is currently hosted on dedicated infrastructure at the company’s headquarters.

## 



Well-Architected Framework and migration phases



Assess tools



**Customer cloud readiness**

* Cloud Adoption Readiness Tool (CART)
* Migration Readiness Assessment (MRA)

**Total cost optimization analysis and**

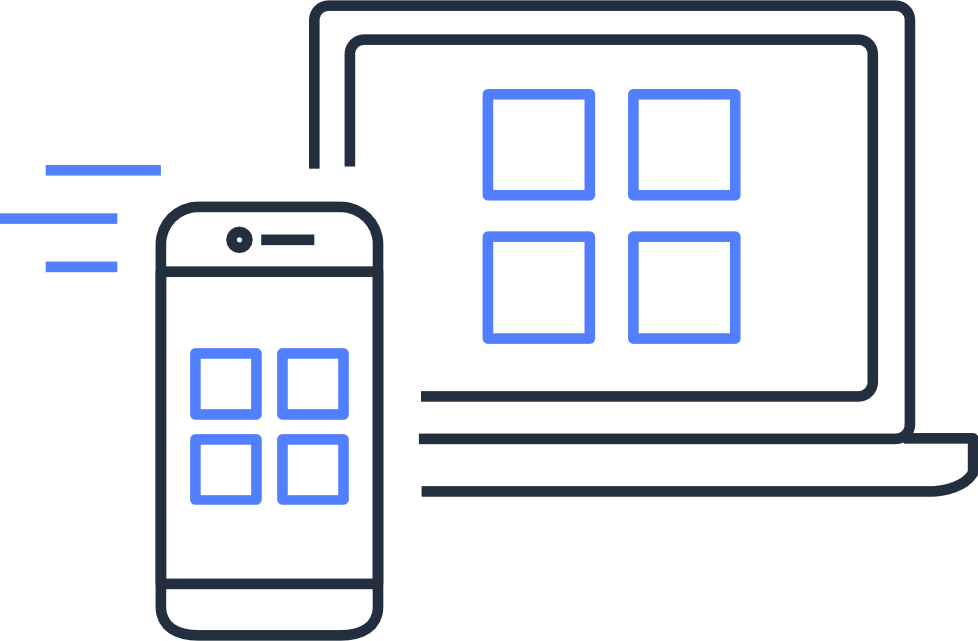
**business case development**

* TSO Logic
* Migration Portfolio Assessment (MPA)

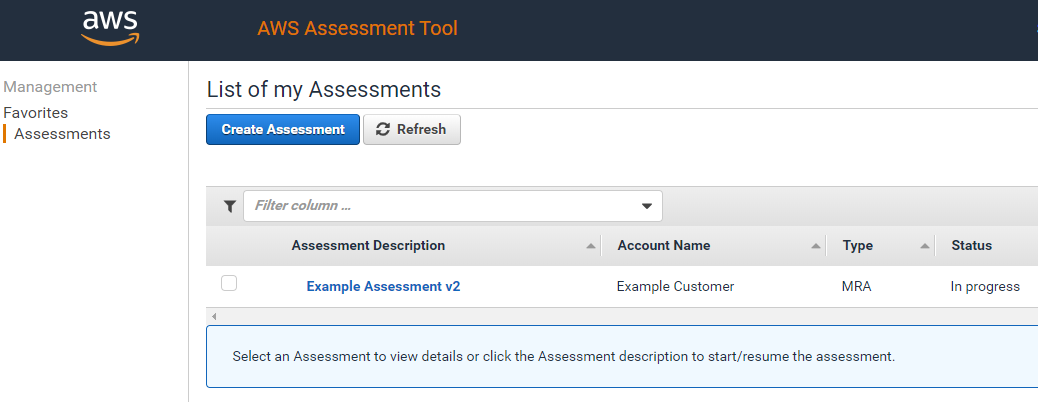


Cloud Adoption Readiness Tool

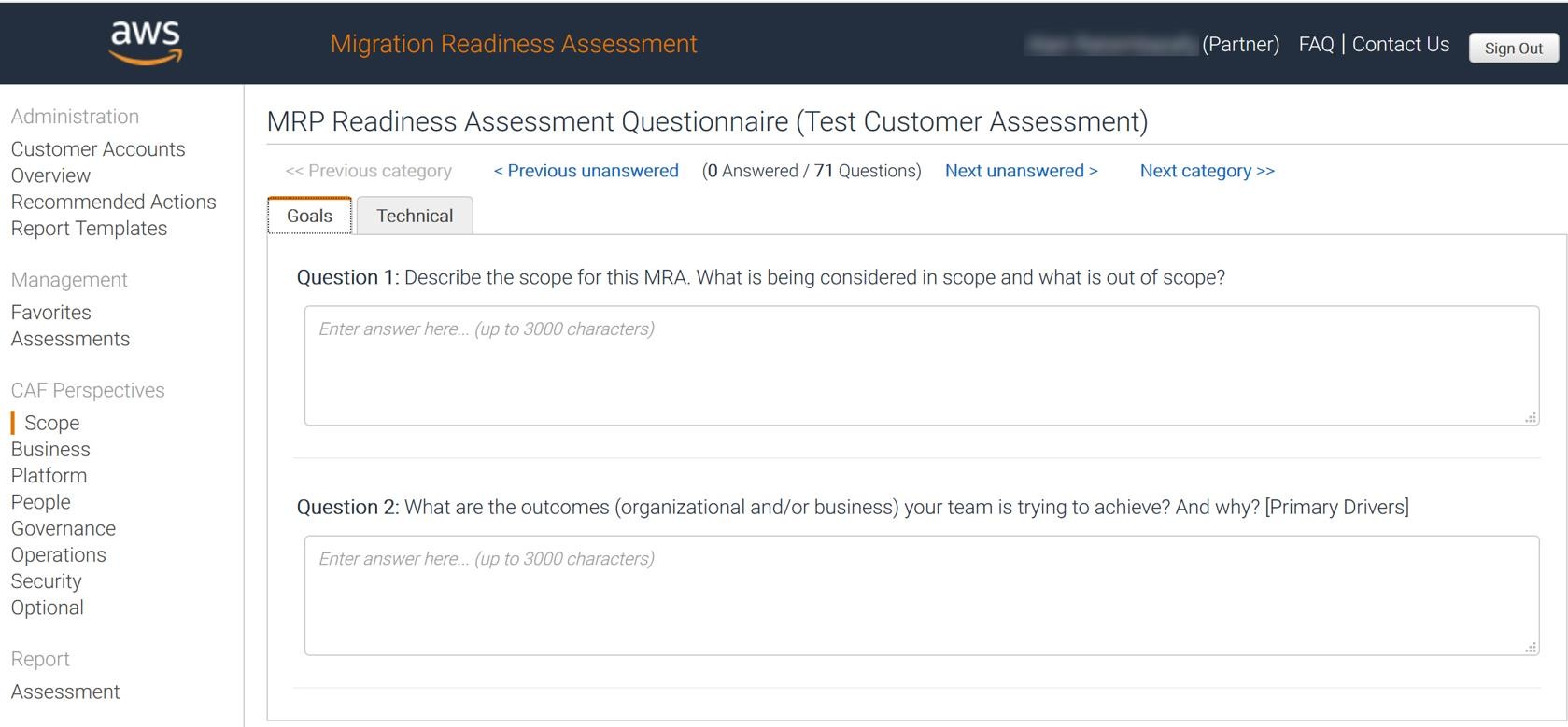
**CART Overview**

* Self-service evaluation
* Plans for cloud adoption
* Surface strengths and weakness

Migration Readiness Assessment

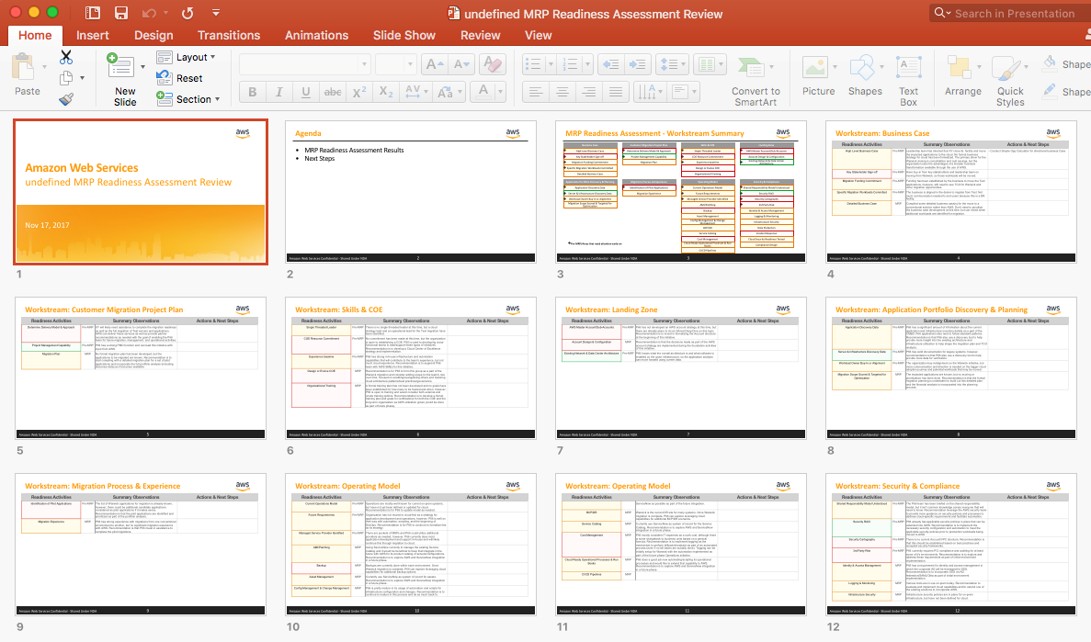
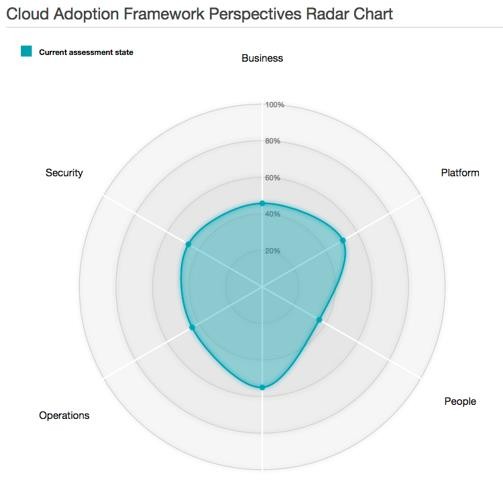
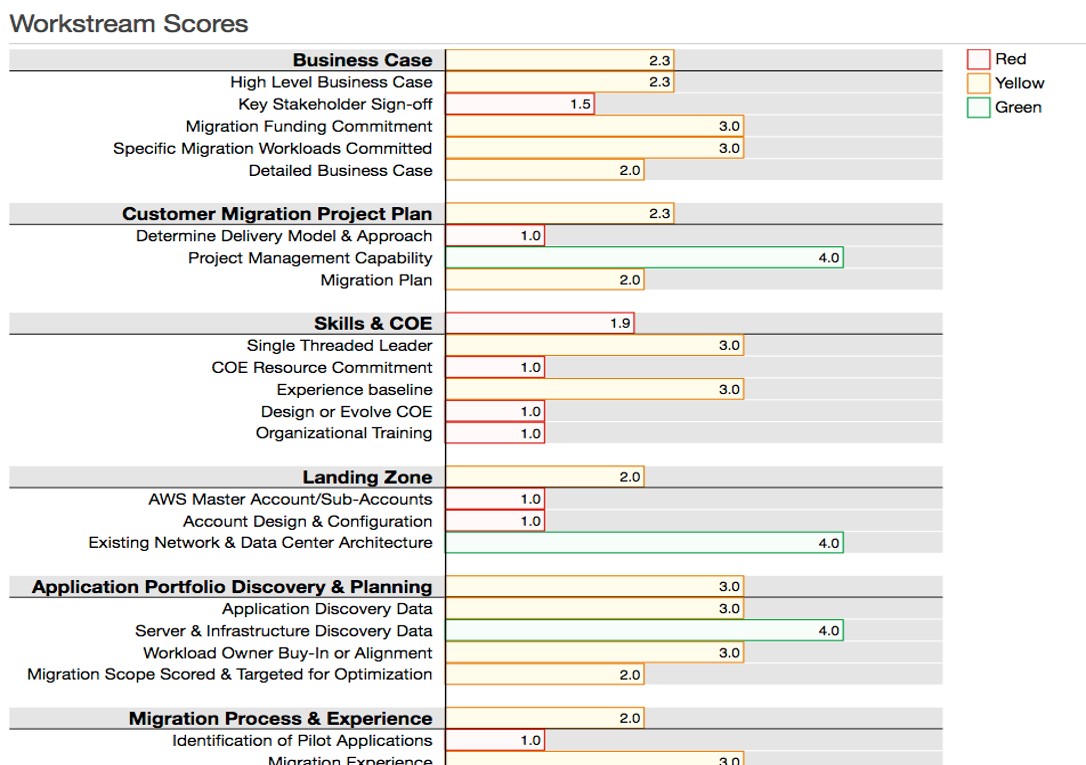
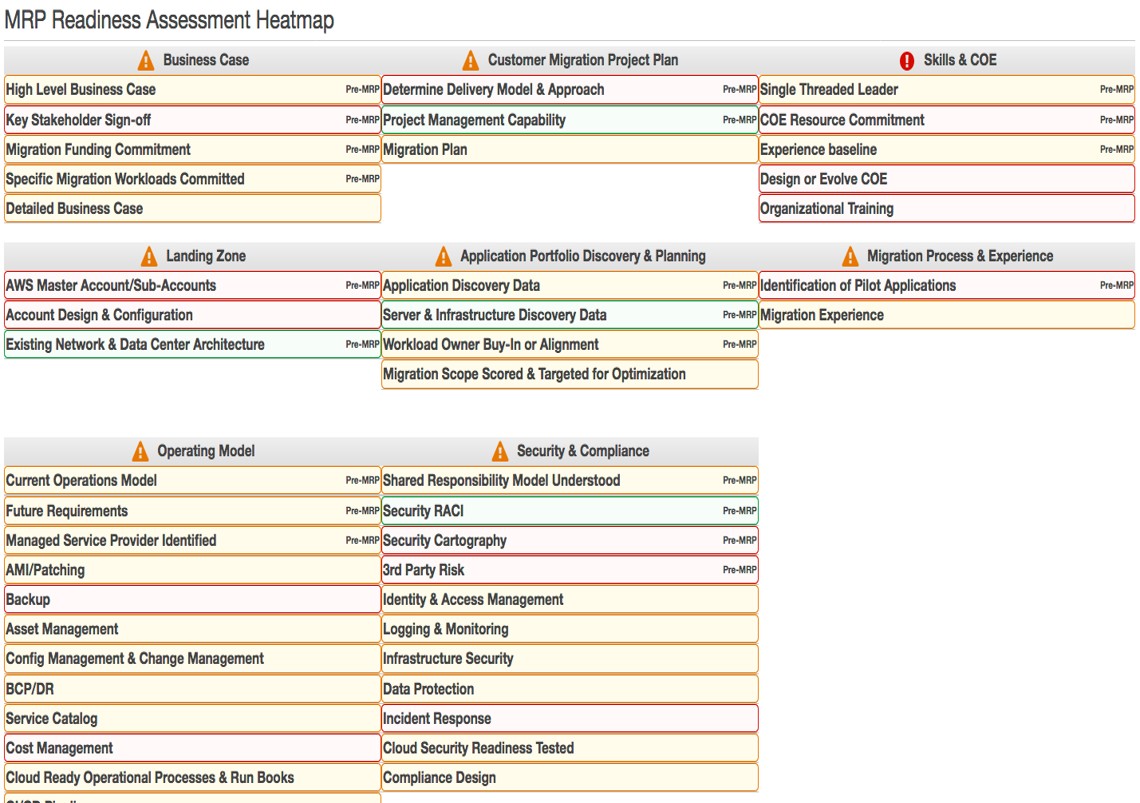
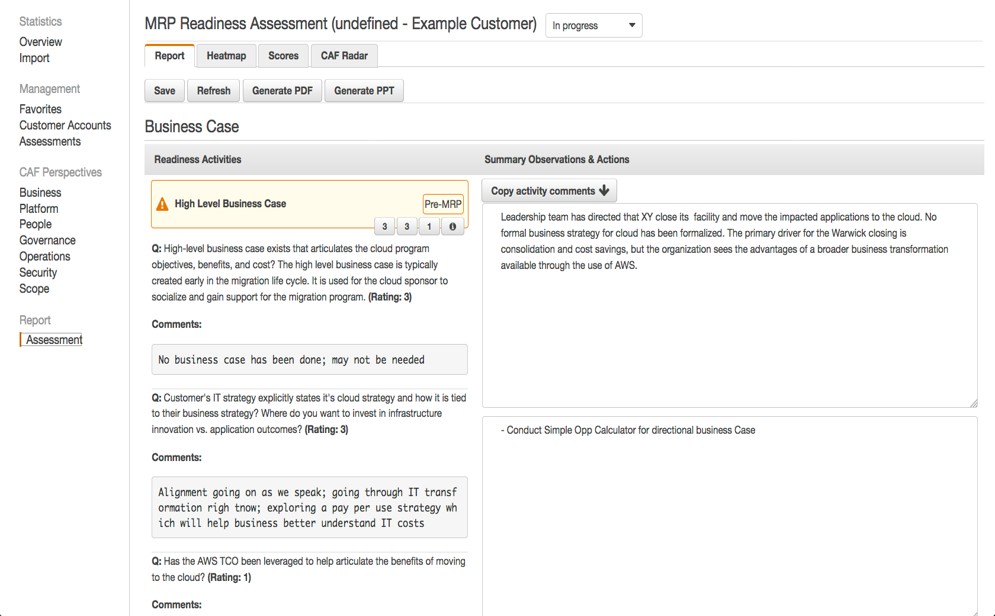


* + Readiness to migrate
  + Conducted by AWS or a partner
  + Identifies priorities



18

1 2 3



4 5

### Provides deep knowledge of customer motivation

* Assists Migration Acceleration Program
* Strengthens customer trust

|  |  |  |
| --- | --- | --- |
|  | **CART** | **MRA** |
| **Access** | * Available for all AWS customers | * Available for AWS Professional Services, AWS solutions architects, and APN Partners. |
| **Engagement** | * Customers can self-assess * Customers control assessment duration | * Workshop style engagement * Typically a 1 day face to face engagement |
| **Content** | * Uses 6 perspectives of the AWS Cloud Adoption Framework * Uses 16 questions | * Uses 6 perspectives of the AWS Cloud Adoption Framework * Uses over 70 questions |
| **Migration Acceleration Program (MAP)** | * Not required for MAP | * Used in the MAP Assess phase |



TSO Logic



Assess phase

On-premises analysis, including TCO

Evidence-based business case

### **Engagement model and period estimate**

### Collector method

|  |  |  |
| --- | --- | --- |
| **Project initiation** |  | **Deliverables** |
| Scope confirmation VM provisioning | Collection period request for business data | Viewer access summary report |
| **1 Week** | **2 Weeks** | **1 Week** |

Using existing data

|  |  |
| --- | --- |
| **Project initiation** | **Deliverables** |
| Receipt and review of dataset | Viewer access summary report |
| **2 Days** | **1 Week** |



Migration Portfolio Assessment

A web application to simplify the portfolio assessment process

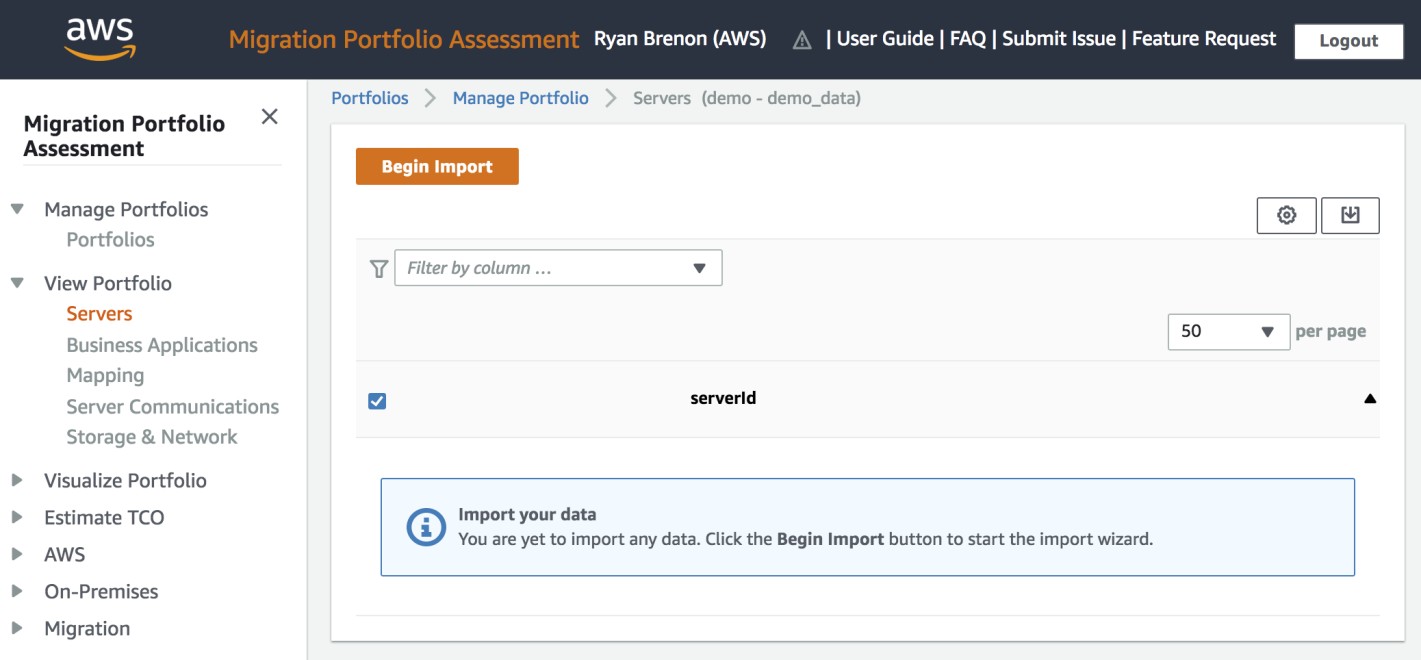
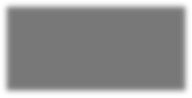
* Shorten the sales and planning cycle, and provide a consistent and scalable solution.

Why

* The customer has discovery output or Configuration Management Database (CMDB) extract, and they want to validate the business case for AWS migration.
* Migration analysis and strategy planning.

Use case

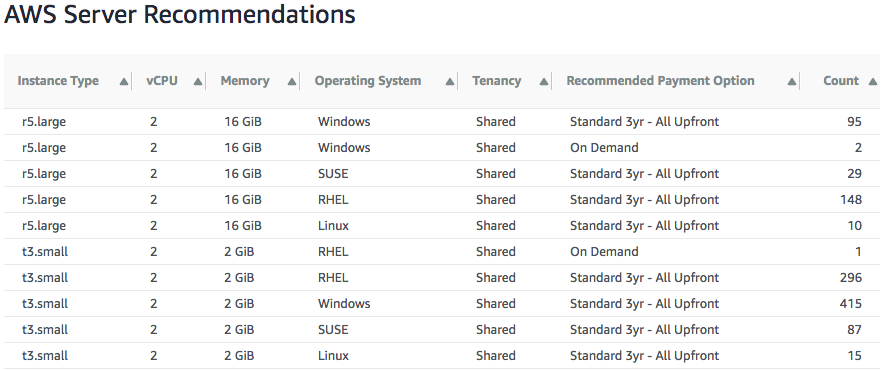
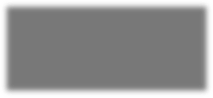
Import CSV or Excel files that contain:



* Server
* Application
* Database
* Server communication
* Server-to-application mapping
* Database-to-application mapping
* Application-to-application dependency
* Shared storage and bandwidth data

Optimize infrastructure spend:

* Provision



* Turn off equipment
* Get the lowest cost

MPA target

AWS Server Cost

Type

D es cr iption

Cost

EC 2

10000 instance[s) - Per k Utilizat.ion

543 58B,108

Reserved Instance Volum e Discount -$4133,432

Residual value Residual value of Reserved InEtance(s) $0

Total Cost

Manual Adjustment Adjusted Total

$39,454,67G

$O

$39,454,67G

Details

Cost Per Instance Typ e

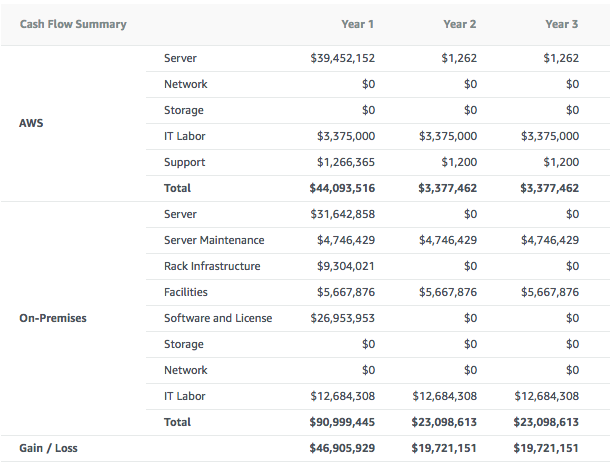
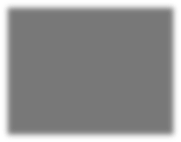
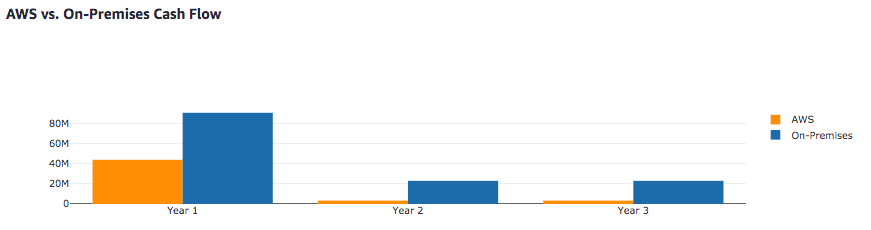
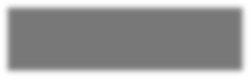
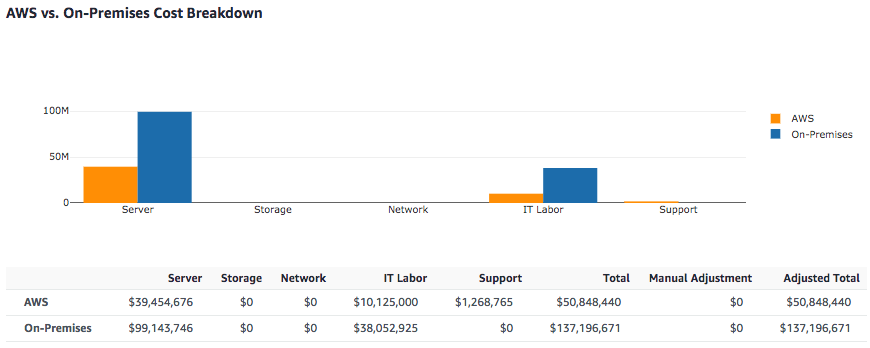
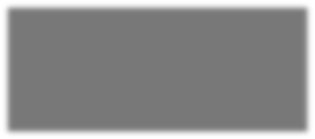
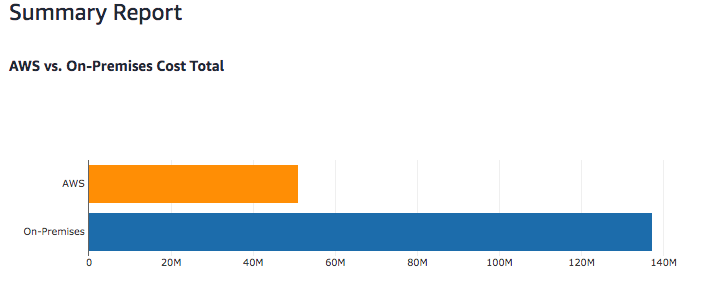
Cash FI ow Per Server

Payment Options Per Server

Reserved Instance Volume Discount

reserved.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pcymc n t Type SM mm ary | Payment Type DeMiI | | | | |
| P aym ent Type | Total U Chant Co st a | Total k1o nth ly Co st a | RI Tie r Disc ount a | Total Cost y |  |
| Standard 3yr - All Upfront | 543,594.139 | $0 | -$4.134, 414 | 539,459.725 |  |
| Standard 3yr - Parti al Upfront | S15,580.018 | S29,643,161 | - $4.27 7, 518 | S4O,745.861 |  |
| Standard 3yr - N o Upfront | $0 | 547,03 8,957 | -$4,478894 | 542,560, 044 |  |
|  |  |  |  |  | 33 |



### Right-sizing

* On-premises and AWS comparisons
* Total cost of ownership
* Migration project cost estimates

35

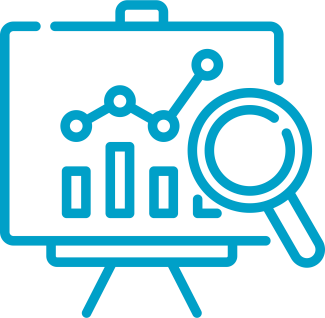
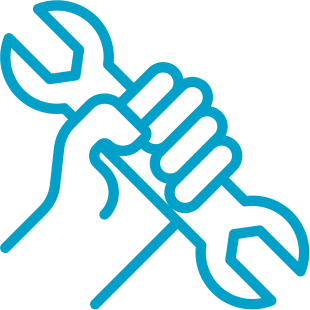
|  |  |  |
| --- | --- | --- |
|  | TSO Logic | MPA |
| Engagement model | * Offered as a service by TSO Logic project   managers and data analysts.   * TSO Logic is available through APN Partner Central via [How to Engage](https://w.amazon.com/bin/view/TSO_Logic/HowToEngage/); or contact PDM for external partners | * No engagement is requiredSelf-service thru AWS accelerate - * <https://accelerate.amazonaws.com/> |
| Data collection | * Offline manual data transformation and upload   by TSO Logic data analyst   * Agent-less method | * Guided self-service process to import discovery result, Configuration Management Data Base (CMDB) or manually gathered data |
| Right sizing with Licensing analysis | * Right size includes Operating System (OS) licensing analysis support * CPU and memory utilization considering age of processor | * Right size recommendations do not include OS licensing analysis support * CPU and memory utilization considering age of processor |
| Network and labor cost analysis | * Not supported | * On-premise and AWS estimate analysis for shared storage, network, labor costs, and support plan |

36

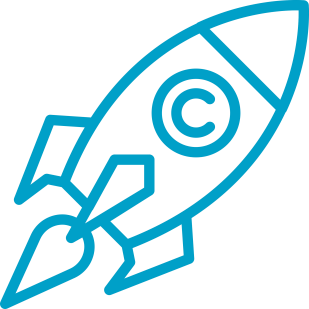
****

Mobilize Phase

* Describe how to gather information about a customer’s application portfolio data
* Discuss how migration strategies affect architectural decisions
* Determine the right migration strategy for a customer’s scenario
* Identify how to set up an AWS multi-account baseline using best practices

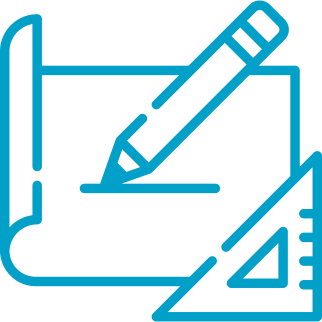
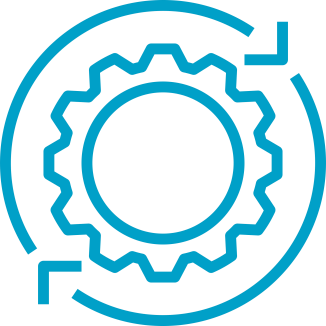
Skills/



Discovery and planning AWS landing zone

center of excellence

Migration experience

Business case Migration plan Operating Model Security and compliance



AWS Cloud



AWS Application Discorvice

Amazon Athena, third-party visualization tools



Connector and agents



Application Discovery Service dependency

|  |  |  |
| --- | --- | --- |
| **Supported Server Types** | Discovery Connector | Discovery Agent |
| * VMware virtual machine | Yes | Yes |
| * Physical server | No | Yes |
| **Deployment** |  |  |
| * Per server | No | Yes |
| * Per vCenter | Yes | No |
| **Collected data** |  |  |
| * Static configuration data | Yes | Yes |
| * VM usage metrics | Yes | No |
| * Time series performance information * Network inbound/outbound   connections   * Running processes | No | Yes – export only |



Application Discovery Service benefits

**Application Discovery Service**

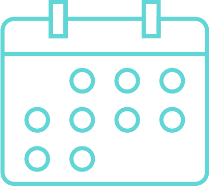
 

AWS Application Discovery Service

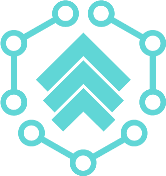
AWS Migration Hub

**Application Discovery Service**

**Discovery flexibility – agent or agentless**

**Migration planning**

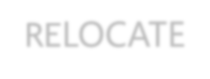
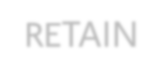
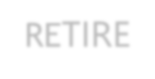
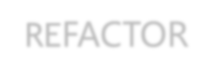
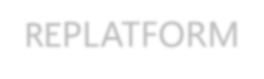
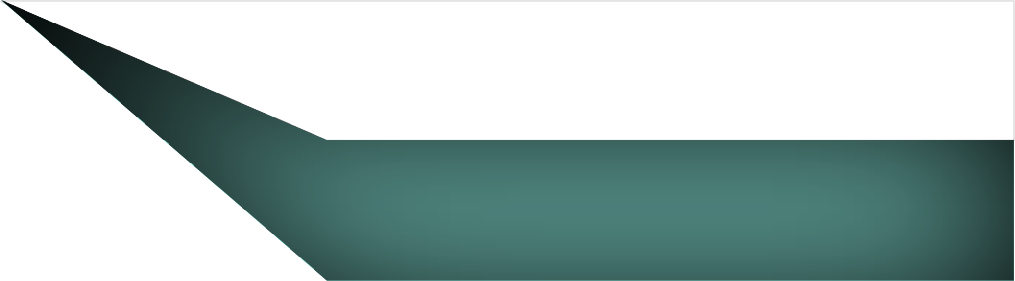
**Securely capture discovery data**

**Runs on AWS Migration Hub**

Migration strategies



Application migration strategies



REHOST REPLATFORM REPURCHASE REFACTOR

RETIRE RETAIN RELOCATE

# AWS Control Tower

###### AWS Control Tower Automate setup



of landing

zones using

blueprints

Apply guardrails for ongoing governance

Automate account provisioning workflows

Get visibility into metrics via a dashboard

* + Landing zone
  + Account factory
  + Guardrails
  + AWS Organizations

AWS Management and Governance

**AWS Management and Governance Overview**



Agility

##### Experiment Be productive

Agility and control: Customers want both.

Governance

Enable Provision



Empower a distributed team

Operate



AWS Management and Governance

AWS Management and Governance services

Enable



AWS Control Tower

AWS Well-Architected Tool

AWS Organizations AWS Budgets AWS License Manager

Provision



AWS CloudFormation AWS Service Catalog

AWS

Marketplace

AWS OpsWorks

Operate



Amazon CloudWatch AWS CloudTrail AWS Config AWS Systems Manager AWS Cost & Usage

Report

AWS Cost Explorer

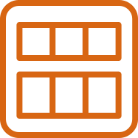
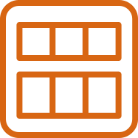
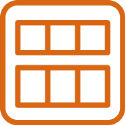
# Migrate servers with AWS

Server Migration Service (AWS SMS)

Automates migration for Hyper-V, VMware, and Azure VMs

77

* Replicates continuously
* Migrates server groups



AWS Cloud

AWS Server Migration Service (AWS SMS)

Amazon Machine Images (AMIs)

VMware, Hyper-V, VMM, or Azure

Virtual

machines (VMs)

Server Migration Connector

VMware, Hyper-V, or Azure



Virtual machine

Schedule

Schedule migration job.

Upload

Take a snapshot.

Export VM to open virtualization format (OVF) template.

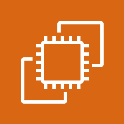
Upload virtual machine disk (VMDK) to an Amazon Simple Storage Service (Amazon S3) bucket.

Convert

Convert VMDK file to an Amazon Elastic Block Store (Amazon EBS) snapshot.

Create

AMI



AWS Cloud

AMI

Create AMI

Amazon EC2

Convert

VMDK

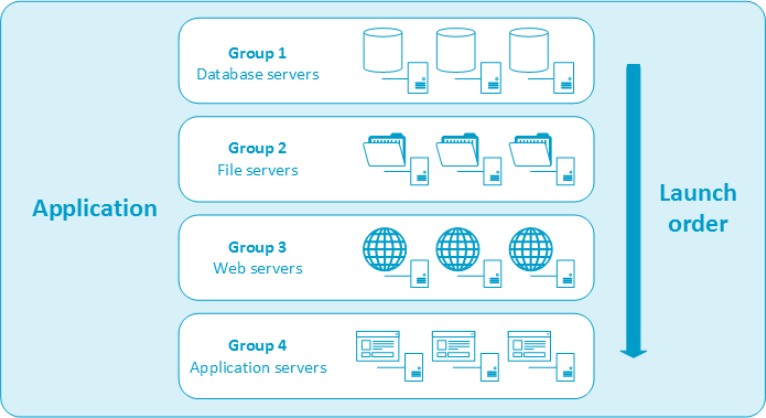
Snapshot

Create an AMI.

* Multi-server application migration
* Coordinated launch

Customer scenario:

* + Tiered application





Migrate servers with CloudEndure (an AWS Company)

82

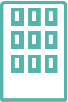
**Cloud Endure Migration and Cloud Endure Disaster Recovery**

#### Automatically migrate and protect applications or databases from any source into AWS.

* Business outcomes:

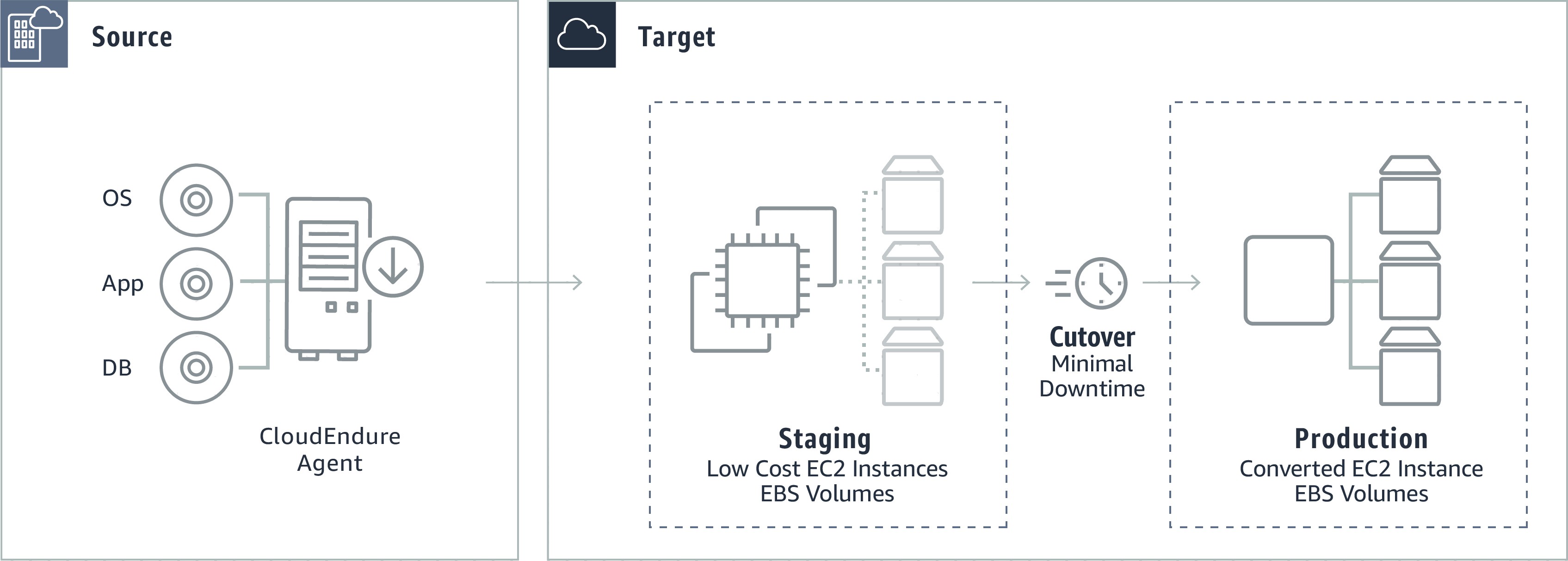
##### CloudEndure Migration – Allows self-service and rapid, reliable migration with

minimal business disruption

* + CloudEndure Disaster Recovery – Meet aggressive recovery objectives while reducing IT resilience and total cost of ownership (TCO)



On-premises to Cloud AWS Regions Cloud to Cloud



Operating system

Application

Database

Flexible

Reliable

Highly automated



Any source migration

Continuous replication

Reduced effort

Wide range of operating system options

Minimal

downtime

Non-disruptive tests

Migrate back

Highly secure

Robust

AWS CloudEndure Wide Platform Support

##### Any Application



Any



Database

x86 Operating



Systems

Source Infrastructure



Physical Data Centers

|  |  |
| --- | --- |
|  | Description |
| Accounts | An organization’s CloudEndure activity |
| Projects | A collection of machines replicating to a specific Target |
| Users | Persons who can administer CloudEndure Projects |
| Licenses | Required for installing Agent and attached to a Project |
| Credentials | Project-specific, AWS access keys that allow control of AWS account |

|  |  |  |
| --- | --- | --- |
|  | Action | Benefit |
| Implementation | 1. Identify the Source machines 2. Group into waves 3. Set cutover date for each wave | Better resource allocation and project continuity |
| Initial Replication | 1. Install Agent in stopped mode 2. Start the replication, one wave at a time | Avoid network overloading and  decrease IT overhead |
| Continuous  Replication | Confirm replication reaches Continuous Data  Protection (CDP) mode | Target machines can only be launched after initial replication is complete |
| Testing | Test target machines 1-2 weeks before planned cutover | Leave time to address any issues that may come up |
| Cutover | Verify that the machines are in CDP mode | Shorten cutover window and minimize cutover downtime |

## **Leveraging the Cloud**

Once the production site was launched, Company A was looking forward to the time when they could use some of the advanced features of AWS. The team automated some processes so that once the server is started it could be easily “attached” to the topology. They **created an Auto Scaling group** of web servers and were able to provision more capacity automatically when specific resources reach a certain threshold (Apache web servers CPU utilization above 80% for 10 min). The team invested some time and resources in **streamlining their development and testing processes** to make it is easy to clone testing environments. They gained lot of experience using AWS resources and also invested time in **leveraging multiple Availability Zones** for even higher availability.

## **Optimization**

During the optimization phase, the development team analyzed their utilization patterns and realized that they could save 30% if **they switched to Reserved Instances4**. They purchased four Reserved Instances (2 for web servers and the other 2 for tomcat servers). They built additional scripts to run their web application in 3 different “modes”: weekend, weekday and promotion mode. These modes defined the minimum number of servers to run. The team also integrated Amazon CloudWatch into their existing dashboards so that they can monitor the system metrics of every instance in their cloud fleet.

# **Conclusion**

The company was able to successfully migrate an existing web application to the AWS cloud. With minimal effort, the team was not only able to free up the physical infrastructure for other projects but also reduce the operating expenditure by 30%. Using the phased-driven approach, the development team was able to resolve all the financial, technical and social-political concerns. Deciding to invest in a proof of concept proved extremely valuable. The resulting architecture was not only elastic and scalable but also flexible and easier to maintain.

5

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc.