

AWS Cost Management Lessons from the Private Sector

J.R. Storment

Chief Customer Officer at **Cloudability**





cloud**ability**

Cloud infrastructure analytics



Infrastructure analytics for scaled web businesses and enterprises.

\$1.5B+ in tracked cloud costs. 18,000+ Users



Why should I care about cost management?



OLD WAY: A few big purchases a year by a few people



NEW WAY: Many little decisions every day by many different people



Why does it need to be cultural?



The cloud is no longer one person's responsibility...



Engineers



Finance



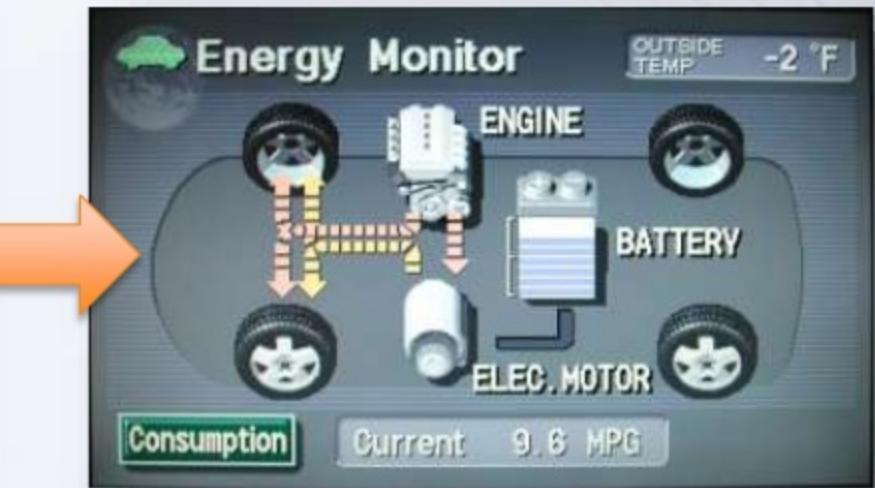
Operations



Capacity



Executive



How do you build the culture?

- Put data in the hands of the people
- Enact policies and evangelize best practices
- Incentivize good behavior

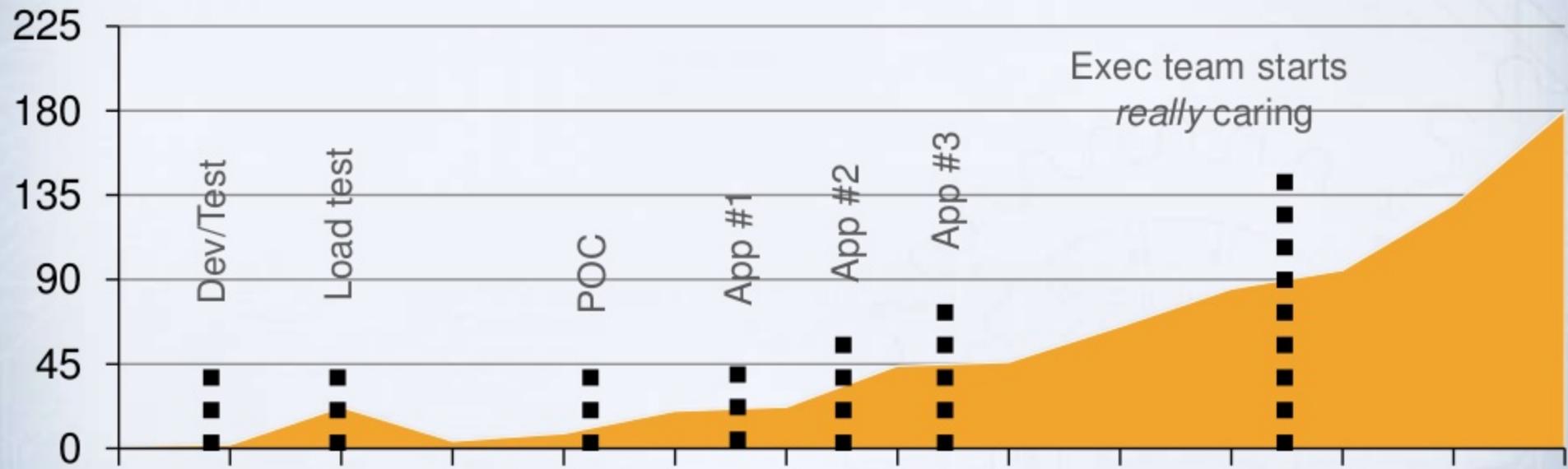


What happens when it works?

- There are no surprises
- Everyone is on the same page
- Make budgeting easier
- Better decisions are made from engineer to CEO
- Save money



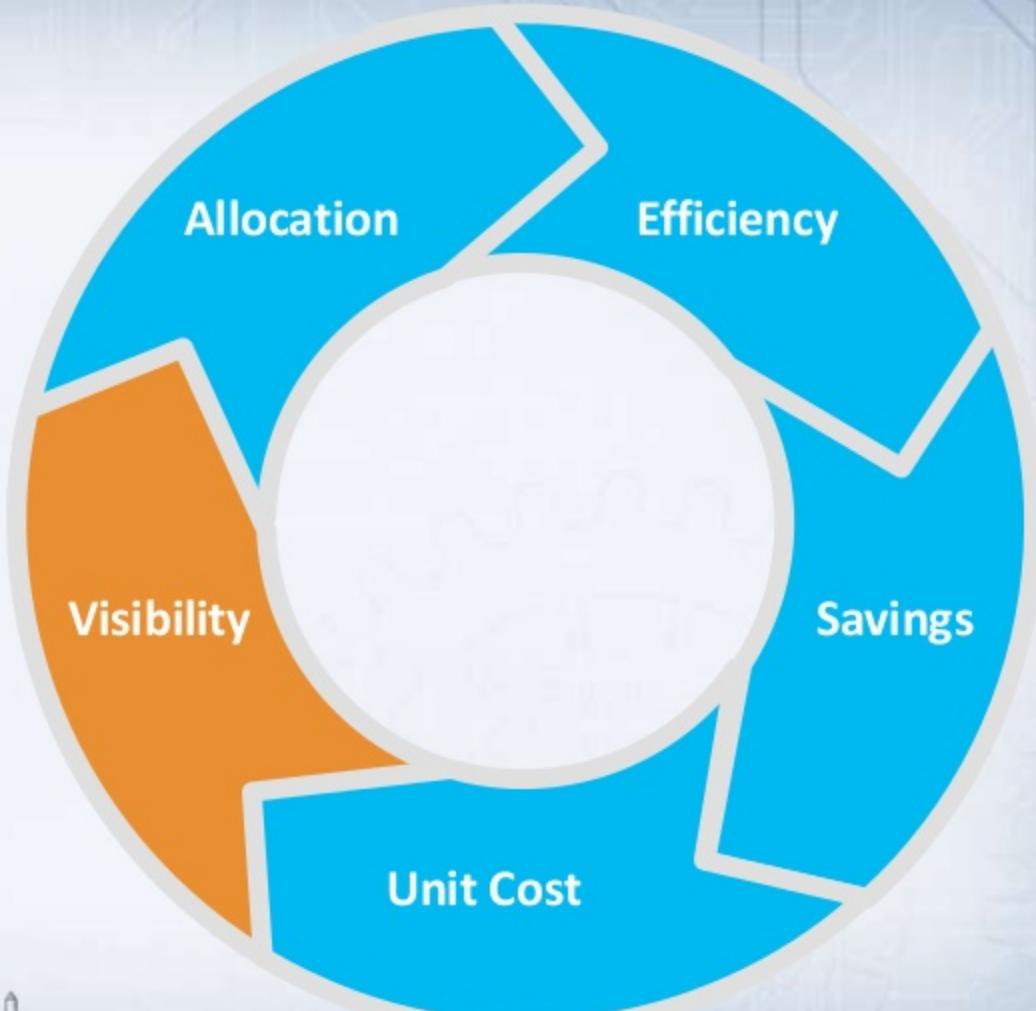
A story of growth







I. Cost visibility



Two types of people



Incur costs



Watch
costs



Two types of people



Incur costs



Watch
costs



Two types of people



Incur costs



Watch
costs



Two types of people



Incur costs



Watch
costs

Two types of people



Manage costs



Watch
costs

Tips for cost visibility

- Get each stakeholder the spending fundamentals daily
- Let each team see other teams' spending habits
- Create broadly available dashboards





cost

usage

manage

Select View

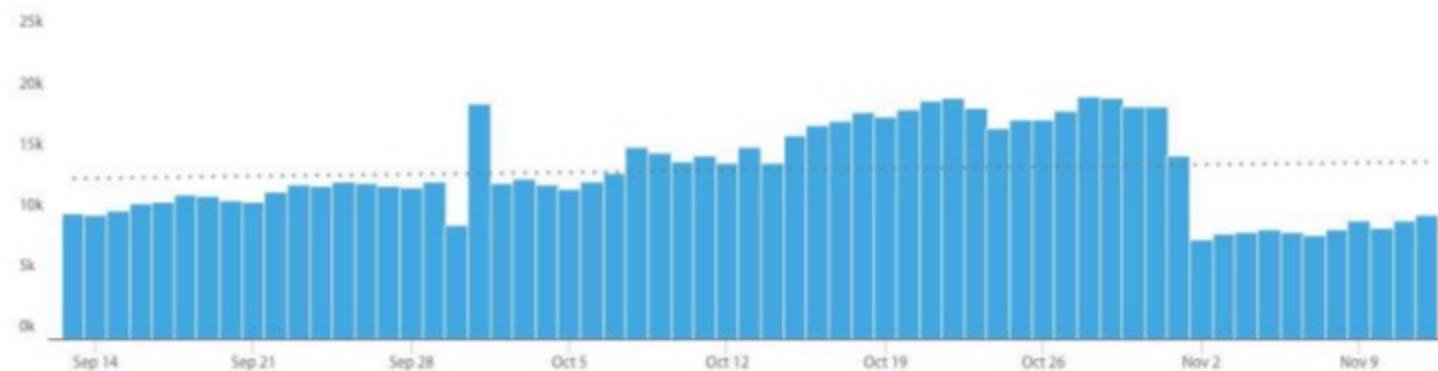
Hi, Amy Smith



New Widget

Annotations

DAILY SPENDING LAST 60 DAYS



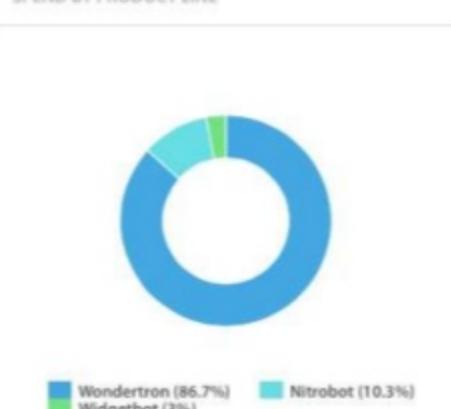
MONTH TO DATE SPEND



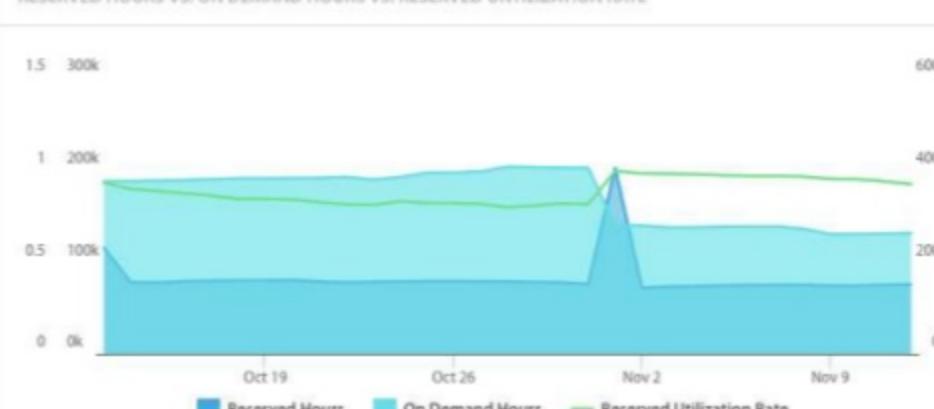
PROJECTED MONTHLY SPEND



SPEND BY PRODUCT LINE



RESERVED HOURS VS. ON DEMAND HOURS VS. RESERVED UTILIZATION RATE



TOP ACCOUNTS

Account Name	Total Invoiced Costs
Globex	\$13,815.42
Roboto	\$9,518.18
Acme	\$2,148.44
Stark Industries	\$1,836.78
Chotchkies	\$22.30
Initech	\$4.01
Transworld	\$1.04
LexCorp	\$0.30

II. Allocation

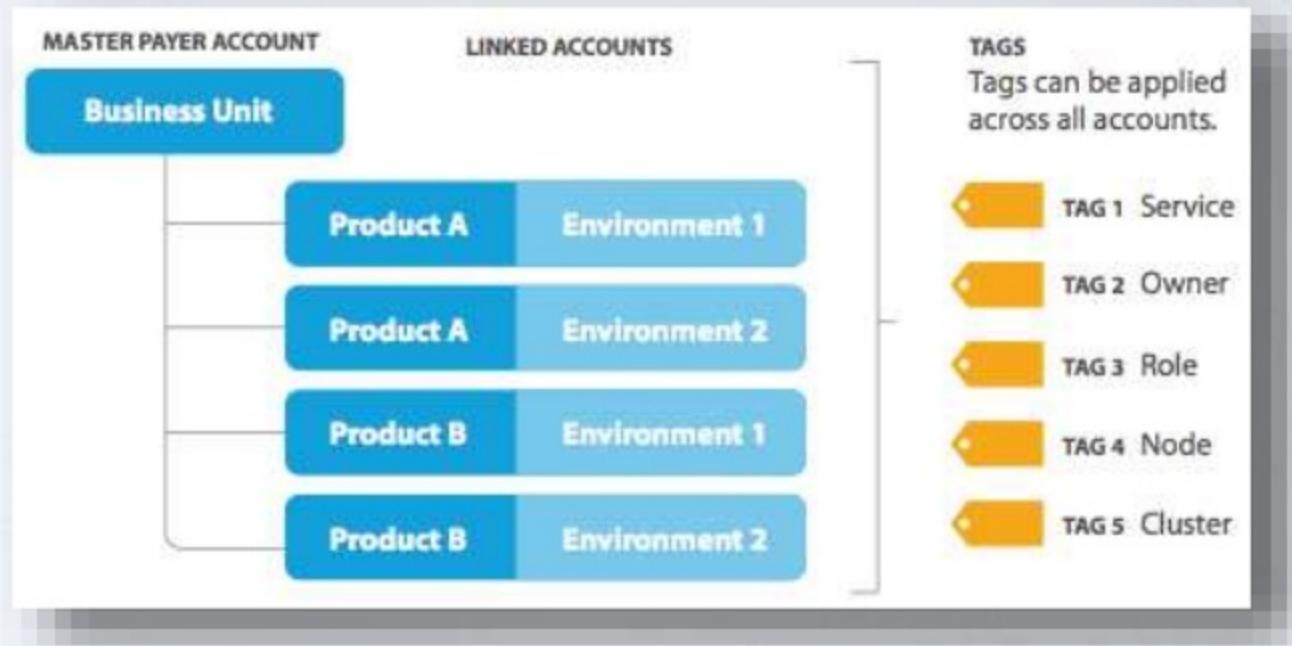


First, get everyone involved

- Multiple major business units
- Lots of products or cost centers
- Thousands of shared AWS resources
- Distributed teams using AWS



Tools for splitting up resources



- Tags are highly flexible, but 100% coverage is difficult due to compliance
- Linked accounts offer clean chargeback but limit reporting options

Add a View

NAME

US Production Spending

VIEW IS PRIVATE?

Division = US ×

Environment (value) contains prod ×

FILTERS

Account ID ≠ 2314-3423-2948-3824 ×

+ Add Filter

Start typing to filter...

Amy Smith

×

Entire Organization

+

Cancel

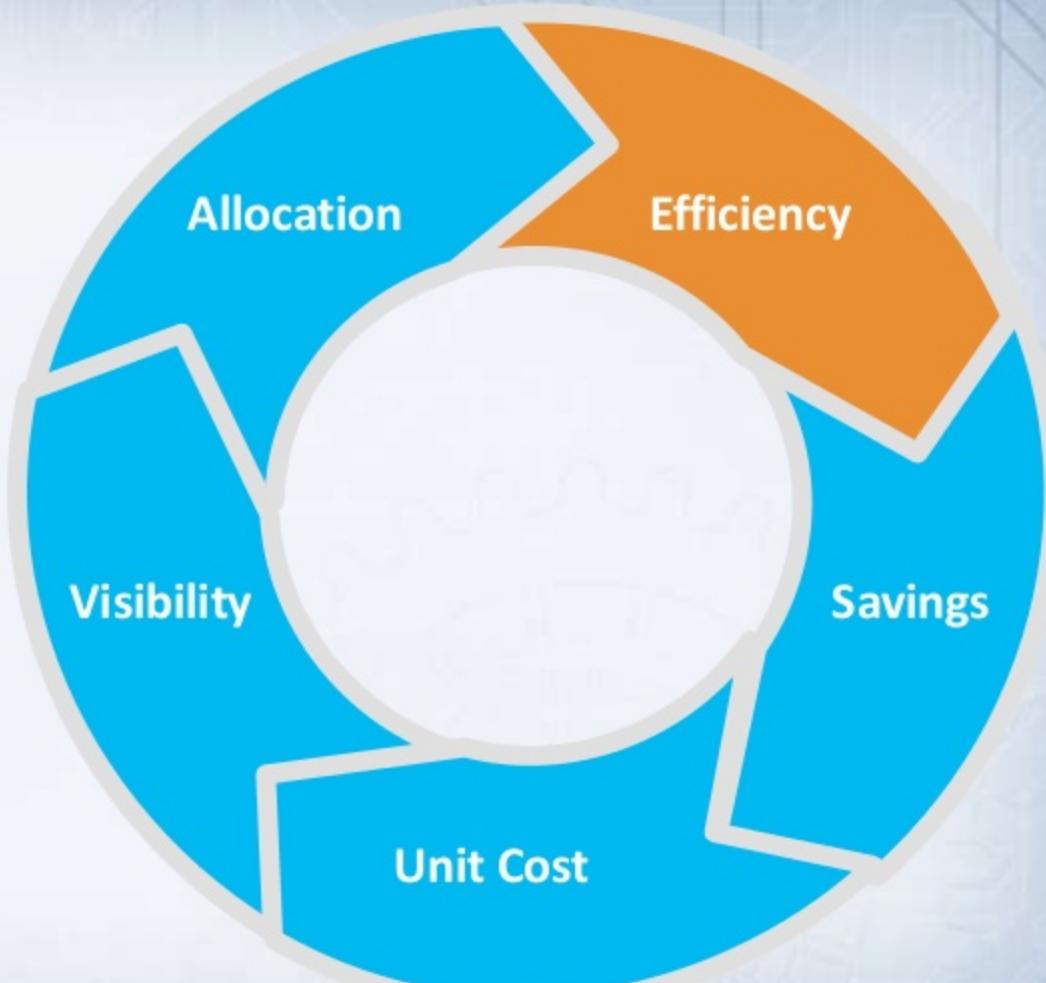
Save

Pro tips: Allocating Costs

- Get consensus on the taxonomy (but let Finance drive)
- Define 2-3 mandatory tags like “project” or “environment”
- Consider a “tag or terminate” rule to enforce compliance



III. Efficiency



168

hours in a
week





Don't run the cloud like a data center:

65% of the hours in a month
are nights and weekends



Three easy wins for Amazon EC2 efficiency...



Turn off underutilized instances

TOTAL BANDWIDTH  **441.06MB**

Avg CPU UTILIZATION  **0.38%**

ESTIMATED TOTAL COST  **\$1,682.62**

Customize this report to add up to 3 dimensions and 3 metrics

[Customize Report](#)

INSTANCE TYPE 	DAYS ALIVE 	INSTANCE NAME 	TOTAL BANDWIDTH 	Avg CPU UTILIZATION 	ESTIMATED TOTAL COST 
<input type="checkbox"/> m2.4xlarge	209	[redacted]	3.19MB	0.00%	\$513.52
<input type="checkbox"/> m2.4xlarge	286	[redacted]	39.81MB	0.00%	\$393.96
<input type="checkbox"/> m3.xlarge	105	[redacted]	8.28MB	0.66%	\$146.72
<input type="checkbox"/> m3.xlarge	104	SOLR1	23.14MB	0.14%	\$146.72
<input type="checkbox"/> m3.xlarge	33	(not set)	35.26MB	0.03%	\$59.92
<input type="checkbox"/> m1.medium	146	amazon-linux-test	8.36MB	0.03%	\$45.59

Low CPU, low bandwidth, low disk I/O, >1 day old | This example: Save \$1,682 per week



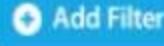
Schedule Dev/Test downtime

HOUR ▲	RUNNING INSTANCES ▼	ESTIMATED TOTAL COST ▼
0 0	578	\$2,724.42
0 1	578	\$2,724.45
0 2	578	\$2,725.67
0 3	581	\$2,729.56
0 4	581	\$2,712.20
0 5	581	\$2,712.20
0 6	581	\$2,733.13
0 7	581	\$2,689.44
0 8	581	\$2,689.27
0 9	578	\$2,688.57



Find legacy instances that could be upgraded

1. Newer instance families are faster/cheaper but have smaller SSDs
2. Look for instances that have attached Amazon Elastic Block Store (Amazon EBS) and low ephemeral disk access

Filters: Instance Type contains m1  Disk Access < 1.00GB  Storage Devices > 1  

3. Move M1 to M3 to save \$0.03-\$0.10

Source: <http://aws.amazon.com/ec2/pricing/>



Get alerts when thresholds are crossed

Report Subscription

REPEAT

ON SU MO TU WE TH FR SA

AT

SEND REPORT EVEN IF THERE IS NO CONTENT

SUMMARY Weekly on Mo at 9:00am

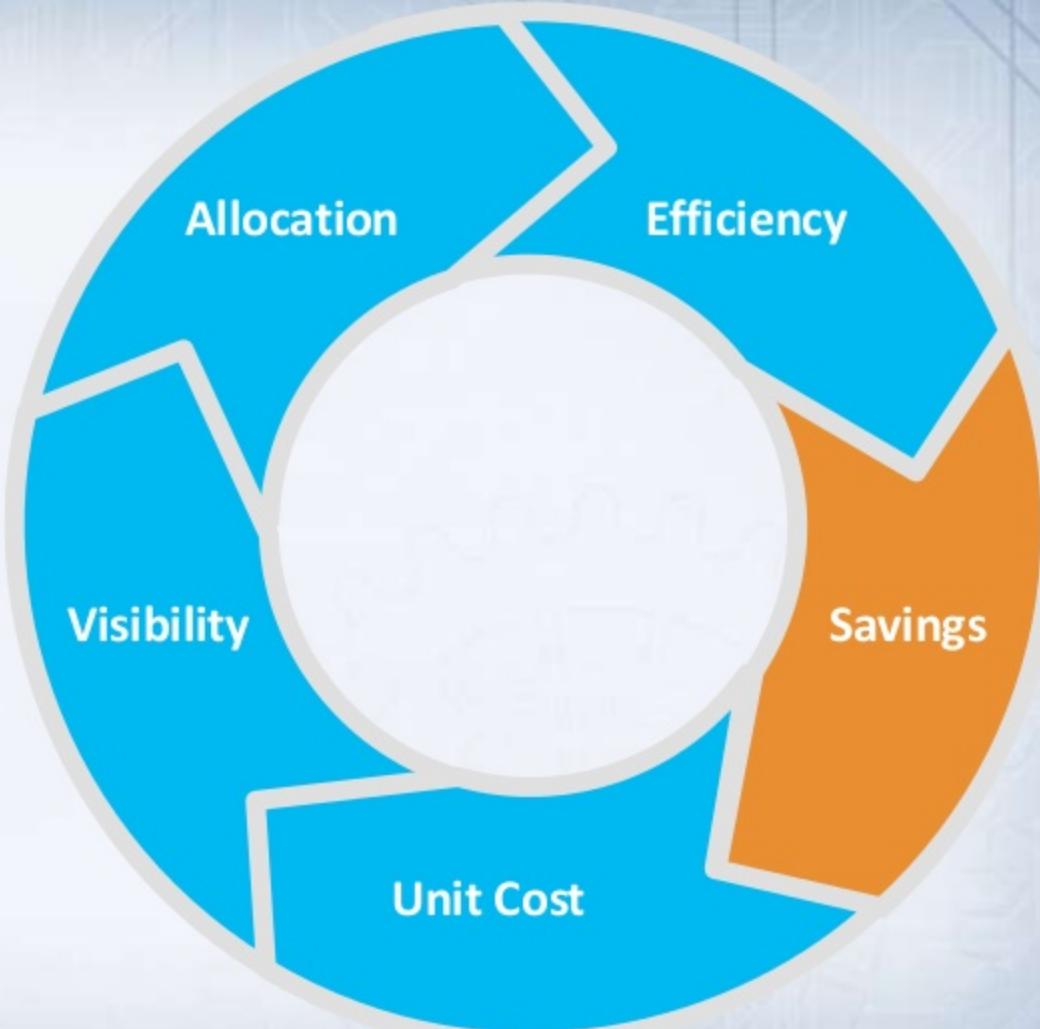


Tips for encouraging efficient behavior

1. Automate weekly waste reporting for each team
2. Gamify cleanup by creating a visible leaderboard
3. Do a monthly, company-wide waste review



IV. Savings



Spot vs Reserved Instances?

Reservations require no engineering,
enabling finance to autonomously reduce costs



Buy Reserved Instances based on current hourly data



Don't wait to buy reservations until you've completed the perfect analysis—you'll waste more money than you save



Reservations

Infrastructure



Aligning Reserved Instances to infrastructure is more important than ever because Reserved Instances are all sunk costs





1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	gather	evaluate	confer	BUY!!		
30						
31						

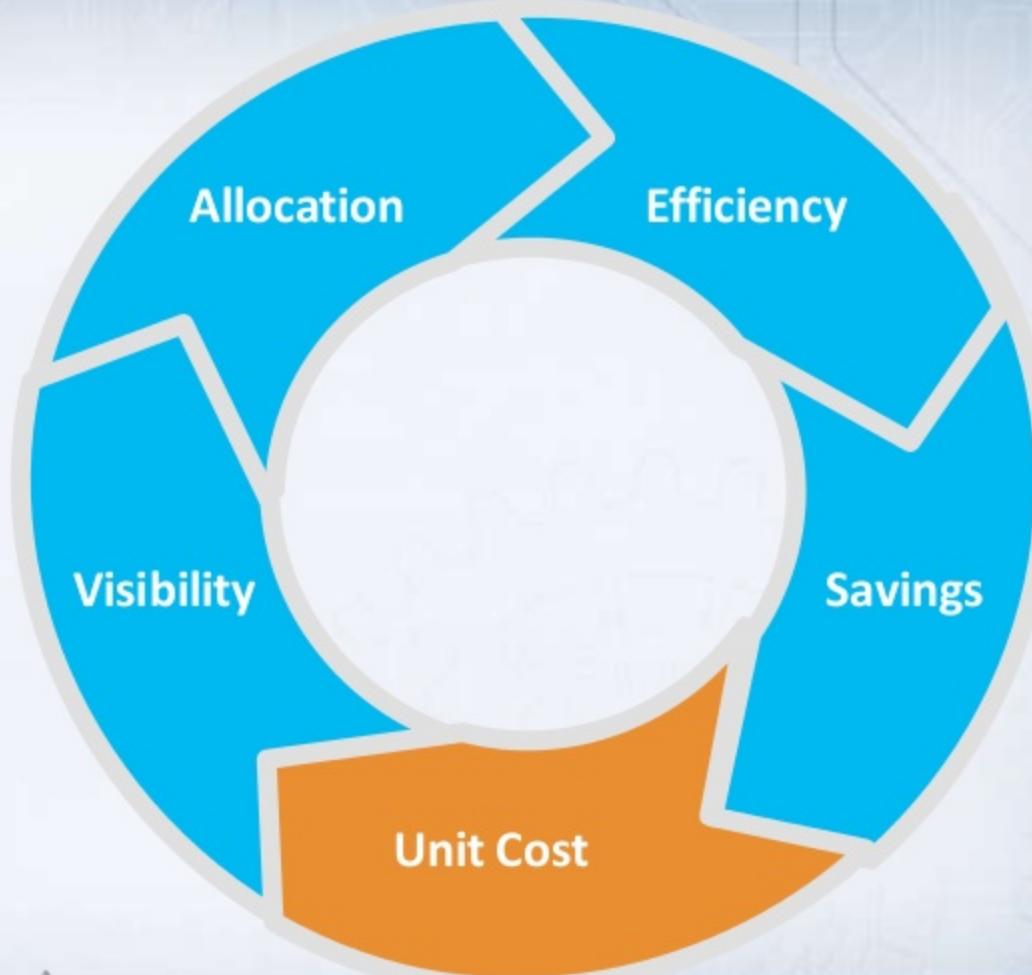


Tips for reservation efficiency

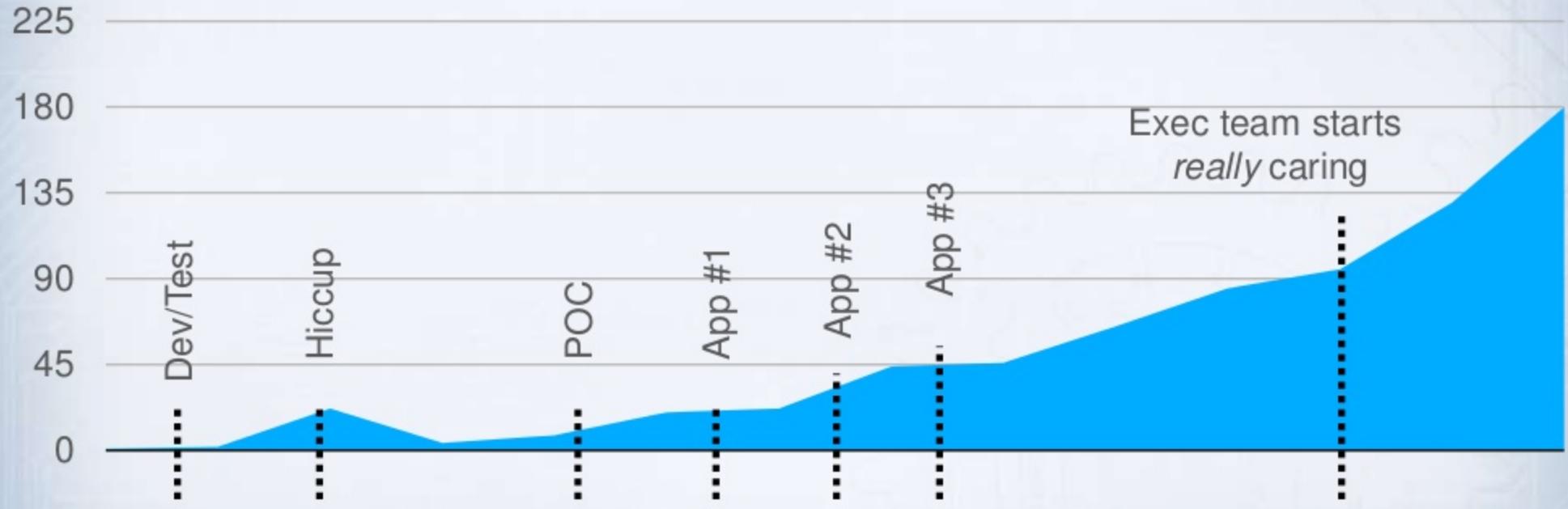
1. Appoint a person to own reservation purchases— often a tech-minded finance person
2. Start with an immediate small and uncontroversial buy
3. Buy iteratively on a schedule that you hold sacred



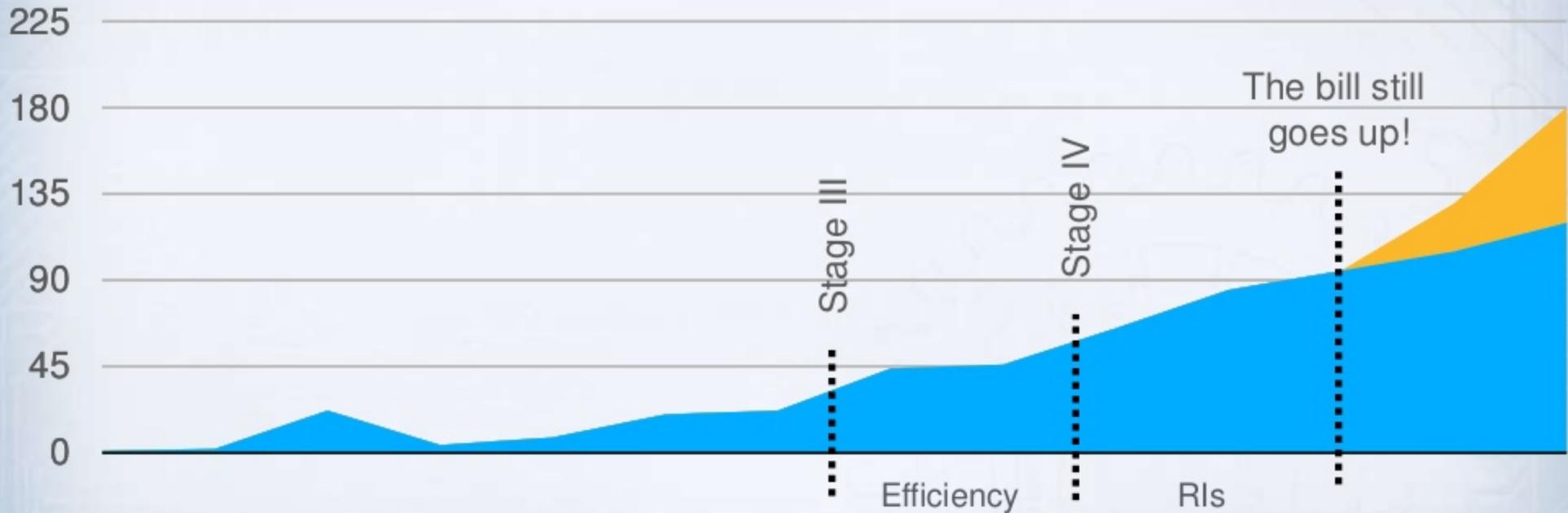
V. Unit Cost



Why does unit cost matter?



Your bill may increase due to growth, not waste

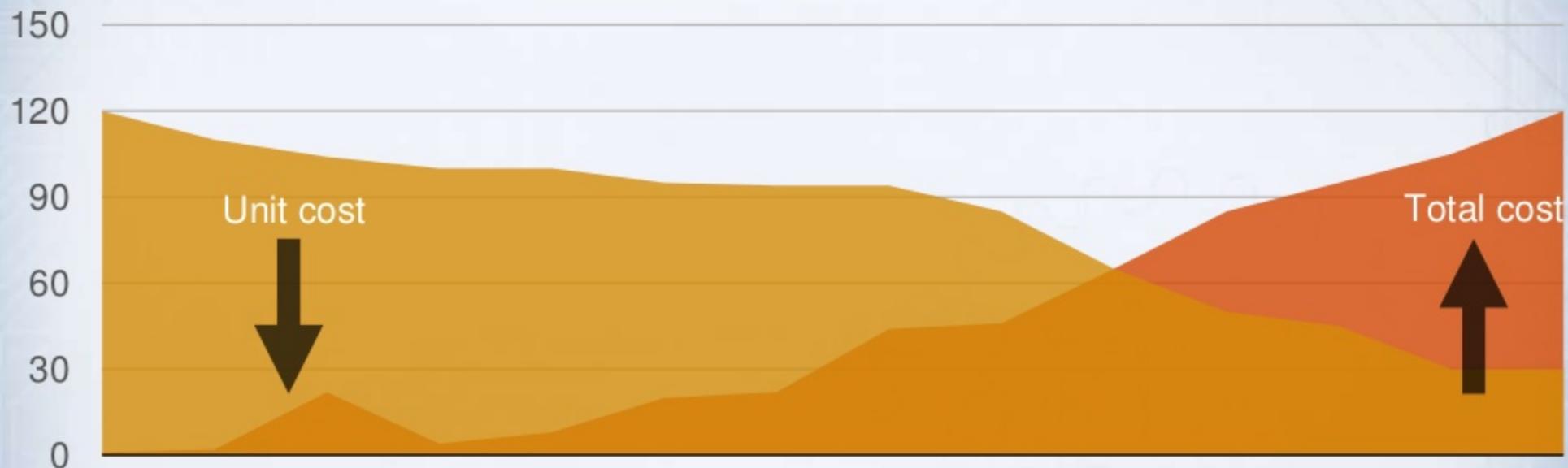


Unit cost is
cost per X

cost per *user*
cost per *subscriber*
cost per *pageview*
cost per *customer*
cost per *API call*



Focus on reducing unit cost, even at total cost grows



Unit cost =
total cost / business metric



$\$1000 / 1000 \text{ customers} =$
\$1 unit cost



What does it look like in the real world?



Case study: Enterprise with 100s of apps

- Split the detailed billing reports into 750 different views broken down by BUs/applications/environments/etc.
- Daily spending updates
800 users get daily emails with just their spending
- Oversight for reporting and control
Execs/finance see the total spending in a simple weekly digest

"Our goal is to put control of AWS resources in each product team's hands. We want to get out of the way but at the same time we still need oversight to stop overspending."



Case study: Scaled web business pushing unit cost

- Product owners track unit cost daily tying spending to revenue
- Operations examines infrastructure changes in terms of unit cost impact
- Finance reports monthly & quarterly on margin impacts based on unit cost
- VPs set goals on unit cost and worry less about setting arbitrary total cost goals

Shortcut: Track your AWS bill as a percentage of revenue



Increasing efficient use among ops/engineers

- Lots of developers using dev/test instances and forgetting to turn them off
- Used Cloudability API data to normalize dev/test environment “health”
- Satisfied management’s need for spending visibility
- Immediately found \$5,000 in savings

Midge's law

$$\text{avg hourly node cost (\$)} \times \text{avg node uptime (\%)} \times \text{sq rt inverse of CPU util (\%)} \times \text{avg node running life (hrs)} \times \text{avg daily unique nodes}^{1.3}$$





Optimizing thousands of reserved instances

- One team purchases Reserved Instances across Adobe using Cloudability
- Implemented consistent monthly schedule for purchases to increase iteration

25th of the month:

Central team drives decisions with minimal input from teams

26th of the month:

Modifications made each month ahead of purchase

28th of the month:

Purchase based on the current month's utilization data

Result: Exceeding 80% RI coverage, saving over 50% on EC2



Getting started on your culture



Things to do right after this summit

- I. Visibility:** Set up a daily spending report for stakeholders
- II. Allocation:** Meet with your finance team to determine the taxonomy for tags & linked accounts
- III. Efficiency:** Look for underutilized instances (low CPU+BW+disk) and M3 candidates
- IV. Savings:** Make an immediate small and uncontroversial Reserved Instance buy
- V. Unit cost:** Determine a top-line business metric, divide last month's costs by it



Contact me

- Booth 23
- @cloudability
- jr@couldability.com





cloudability

Coming to DC and NYC AWS Summits this Summer

START A FREE 14-DAY TRIAL AT CLOUDABILITY.COM

Thank You.

This presentation will be loaded to SlideShare the week following the Symposium.

<http://www.slideshare.net/AmazonWebServices>

