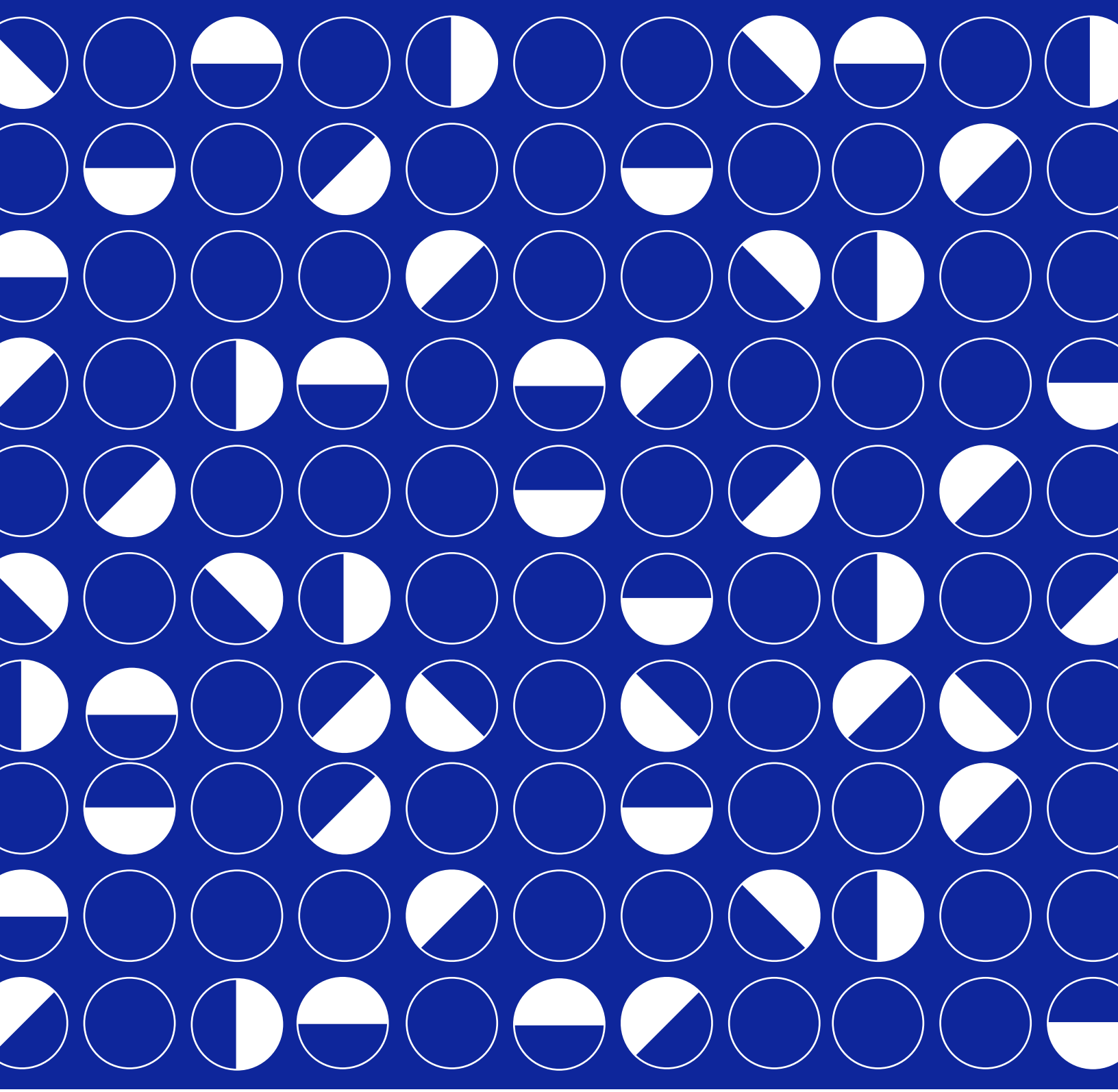


# AWS Tagging Strategy Best Practices

Using Tags & Consolidated Billing to Lower Your AWS Spend

# Introduction

A clear understanding of your cloud usage and costs is critical to getting the most out of your AWS cloud. To achieve that goal, every team in your company — from Finance and Operations to Engineering and Upper Management — needs accurate data to answer the cloud questions most relevant to them. Finance might want data that allows them to control costs and apply chargebacks to the correct business unit, while Operations would need breakdowns of cloud spending by project compared to their budgets. Engineering could want to make sure they're using cloud resources efficiently, and upper management probably wants to understand the costs and savings, not to mention how to ascertain proof of ROI.

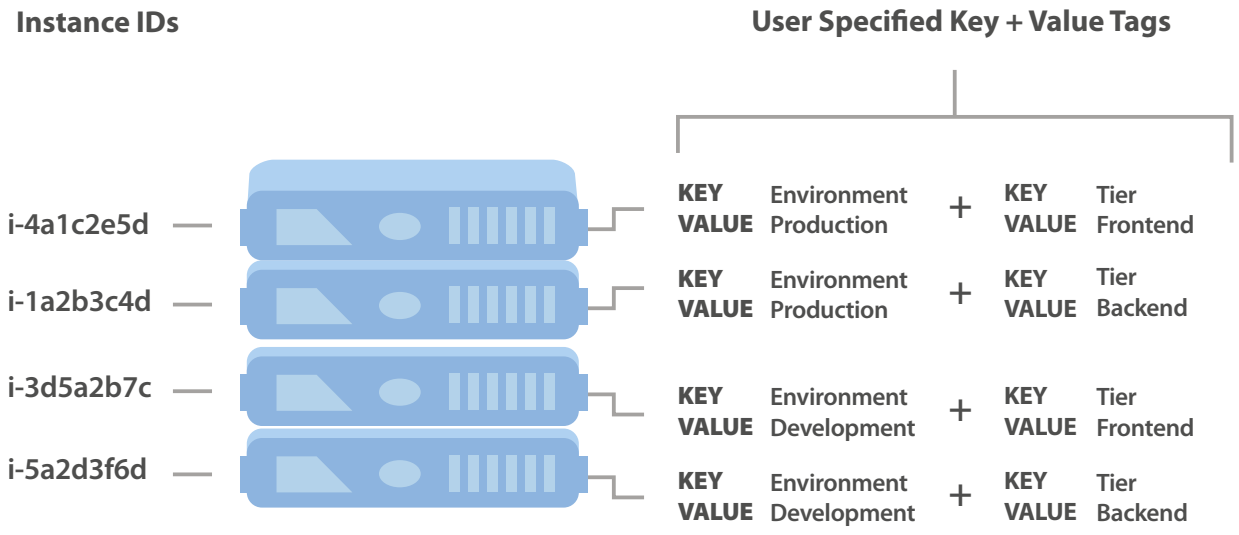
How can you get this kind of cross-team visibility? It comes down to combining the right tracking methods with the right tools. Two of the most important tracking methods are built into AWS: tags and consolidated billing. When combined with the right tools (especially in the form of a cloud cost management platform), you can gain the insights you need from the data.

In this e-book, we'll go over creating a strategy to use these building blocks so all your stakeholders can work together to optimize how your company uses the cloud. Using the Cloudability platform, we'll also show you how tags and consolidated billing can be used with a cloud cost management platform to give you the visibility you need.

# What Are Tags?

Tags are metadata labels (each with a customer-defined key and a value) that you assign to your resources. The key is like the column heading on a spreadsheet, and a value is tied to that key. Think of it like describing a bunch of shirts. Each shirt might have a key for **color**, then a value of **red**, **blue** or **green**. Using this system, you can sort your AWS resources.

Truth be told, it doesn't matter what key and values you use. To AWS, tags are simply strings of characters with no semantic meaning. Because tags are only meaningful to you, they can be whatever you want them to be. This is why it's so important to craft your tag strategy around your business systems.



This illustration shows four AWS instances, each with two tags: an **Environment** tag and a **Tier** tag. **Environment** and **Tier** are the keys, each with its own set of values. The **Environment** key can have values of either **Production** or **Development**, while the **Tier Key** can have values of either **Frontend** or **Backend**.

If you break down spending by **Environment**, the first two instances would be grouped together as **Production** and the second two would be grouped together as **Development**. If you want to look at spending by Tier, the first and the third instances would be grouped together as **Frontend**, and the second and the fourth instances would be grouped together as **Backend**.

## Tagging Tips

Here are a few things to remember about tags.

**Tagging isn't retroactive**
Tags begin to organize and track data as soon as they are applied, but they aren't applied retroactively. This means you must plan your tagging strategies ahead of time for comprehensive reporting. For example, creating an EC2 instance on January 1st without assigning a tag to it until February 1st leaves all of the instance's January data untagged and untracked.

**Automate as much as possible**
Tags need to match exactly. In the eyes of AWS, a tag named **Enterprise** is different from a tag named **enterprise**. Misspellings, variations in case and abbreviations occur all too often when human beings create tags. Instead, try to enforce tagging policy through automation. Infrastructure automation programs such as Chef, Puppet and Ansible can help ensure consistency.

**The Name tag is extremely useful**
The **Name** tag is an underused asset. Use the **Name** tag value across multiple resources to roll up costs into a single line item. Here are some ways to use the **Name** tag:

- Tag S3 buckets with the **Name** tag to track storage and bandwidth.
- Apply the same **Name** tag to EBS volumes that you've applied to its EC2 instance to get total compute cost.
- Tag RDS DB instances, snapshots and replicas.

This example shows how to tag an EBS volumes with the instance ID to get total compute and storage costs for the instance:

```
name=the_id_of_the_instance_the_volume_is_ attached_to
```

AWS has a 50 tag limit per resource.  If you want to use more tags, you can concatenate the **Name** tag with another tag, for example, name=Service-Role-Node.

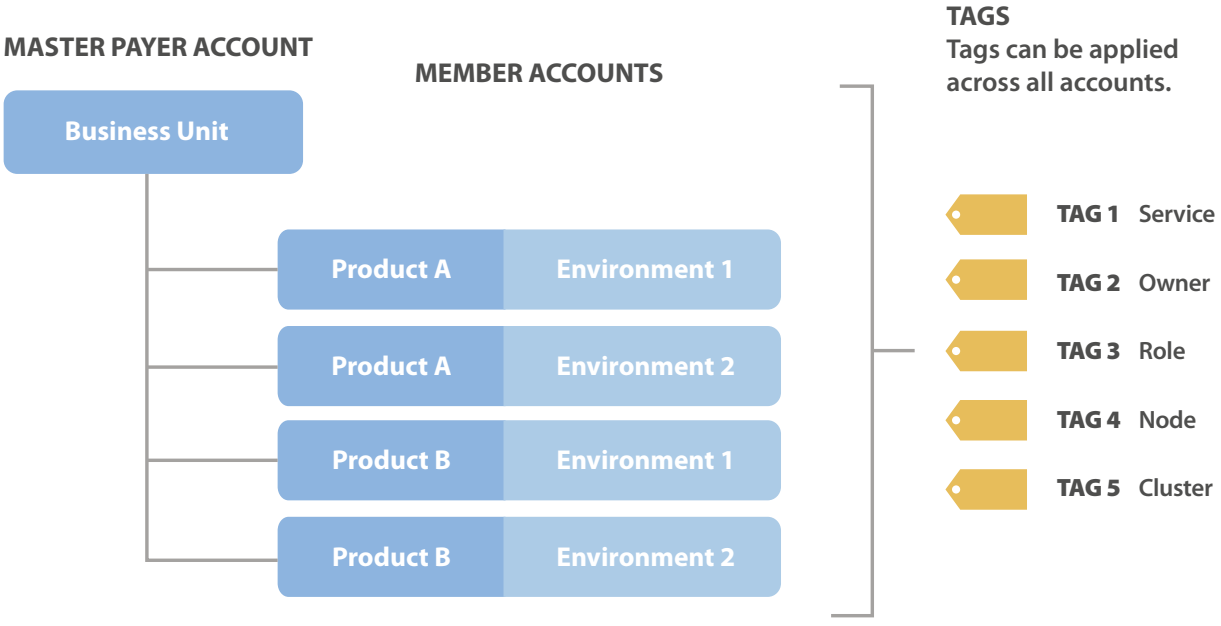
**Tag Everything**

In general, you should tag everything that you can — not just EC2 instances. You want to minimize the amount of spending in untagged resources. After all, if spending is untagged, then it's much harder to track down how it's being used.

# What Is Consolidated Billing?

You can use consolidated billing in AWS Organizations to simplify payments for multiple AWS accounts. Each organization has a master account that pays the charges of all the member accounts.

Consolidated billing is a great way to show a clean division between cost areas. For example, you might want separate accounts for development, staging and production environments for each product that your company offers.



In this example, the master account is the **Business Unit**. Below the Business Unit are member accounts that represent different environments for diffferent products. Consolidated billing gives you discrete divisions without having to spin up several different payer accounts. Tags apply across all accounts, making it easy to allocate and compare costs or usage.

## Pro Tip

Consolidated Billing can also be used to share Reserved Instances (RIs). RIs purchased at the master account level are used as needed by member accounts. RIs purchased at the member level give priority to that account, then are used by other member accounts. See more in our Guide to AWS RIs.

# Using Tags and Consolidated Billing Together

Tags and consolidated billing both have their advantages and disadvantages. It's generally better to use them together rather than to only use one or the other. Tags are highly flexible, but 100% coverage can be difficult. Consolidated billing gives you clean chargebacks, but has limited reporting options. Having more than one or two consolidated billing accounts can be overly complicated and they are difficult to use with VPNs and shared services.

Get the best of both worlds. Use consolidated billing to split out the most important divisions. Use tags to dig down deeper into those accounts for more fine-grained visibility.

# Devising Your Tagging and Consolidated Billing Strategy Together

Devising a strategy for how to tag your resources and group your accounts is at least as much about social interactions as it is about technology. Remember that tags are only meaningful to you and how you organize your cloud resources, so make sure that your tagging strategy is designed to mirror your company structure and reporting system.

## Communication is key

The first step is to get everyone involved. Sit down with all the stakeholders, like Finance, Operations and Engineering, and make sure you understand what everyone needs. If you already have tags and consolidated billing in place, then the first step is to audit what exists. Make sure the structure works for every team.

## Keep it simple

Implementing a reporting strategy can seem overwhelming when you have a complex infrastructure, so keep your initial strategy simple. Start with three to five obvious areas which have costs you want to understand. For example, you might initially focus on the business unit, the product, the owner and the role. Even a few small first steps yield big returns in terms of information.

## Formulate your questions

You're doing all this work because you need answers about how your company is using the cloud. Answers to questions like these:

- What business unit of the organization should this cost be charged to?
- Which cost centers are driving my costs up or down?
- How much does it cost to operate a product that I'm responsible for?
- Are there unused resources in my dev/test environment?

Terms such as "business unit" and "cost centers" should tell you where you want to focus your attention.

## Set up your nomenclature

Once all of the stakeholders have agreed on what needs to be targeted, you can define the specific tags and consolidated billing structure you want to use. Again, where tags are concerned, consistency is key, so consider using automation to apply them.

To get you started, here are some of the most commonly used tags.

| Tag | Explanation | Examples |
|---|---|---|
| **Cost center** | This tag reports on resources used by specific departments or teams within a company. For teams with more nuanced department or team distinctions, you can use more than one tag per instance to identify where the cost and usage data belong.<br>For example, if the goal is to report costs/usage by division and then by department, apply two tags, `division=X` and `department=Y`, to each resource. | `cost center=1006, division=smallbusiness, department=productdevelopment, team=engineeringpodA` |
| **Product or project** | It's critical to monitor and report the costs/usage associated with a specific project or product line. Tagging according to product is handy for tracking costs related to products sold to consumers, while tagging according to project is typically used for tracking R&D efforts. | `product=creditcardprocessing, project=SEO` |

| Tag | Explanation | Examples |
|---|---|---|
| **Application or service** | Service-oriented architectures are quickly becoming the norm for companies with complex applications and infrastructures.<br>The **service** tag is a great way to pinpoint the cost and usage of any resource that is dedicated to providing a particular service. The tag lets you monitor costs per service over time and identify any unexpected changes. | `service=logging, service=authentication, service=notifications` |
| **Employee** | Engineers often create and turn off resources at will for their own development and testing efforts. The assumption is that the usage will be brief rather than long running.<br>The **employee** tag can help monitor this type of usage. An easy approach is to tag each resource according to its owner as it's created.<br>Note: Small teams typically do this manually, while larger teams may write simple wrappers to the cloud vendor's API that applies the tag automatically after any resource is created. | `owner=gob, owner=michael` |
| **Environment** | Monitoring and attributing costs to different computer environments is essential to understanding the total cost of your company's cloud infrastructure.<br>For example, most companies have a production, development, staging and demo environment.<br>Using the environment tag allows you to see the true cost of each specific environment. Accurate tagging helps you avoid the pitfall of artificially inflating production costs by factoring in unintended costs from other environments. | `environment=staging, environment=production, environment=dev` |
| **Role** | Engineers often create and turn off resources at will for their own development and testing efforts. The assumption is that the usage will be brief rather than long running. | `role=webserver, role=customerdatabase` |
| **Software version** | Using the **software version** tag lets you tag resources by specific versions of an app or project. This lets you compare costs between software versions. | `version=august2018release, version=beta` |

# Using a Cloud Management Platform

Once you've tagged your resources and created consolidated billing accounts, you're going to get a lot of useful information. So how can you organize that data into meaningful reports that are easily accessible by your stakeholders? At first, many companies do this with spreadsheets and tools that the cloud vendors provide. But as your infrastructure grows more complex and the amount of data you're processing increases exponentially, these methods become unwieldy.

They become unwieldy because of the insane amount of data that's produced to track AWS cost and usage. Let's look at EC2 as an example. AWS bills are based on usage down to the second, which means 86,400 pieces of data per instance every day. Additionally, there are 115 different EC2 instance types with 10 OS choices and 17 possible region choices, creating 480,000 possible unique prices/SKUs. With modern infrastructures spinning up thousands of instances, the billing data gets very large, very quickly.

When you get right down to it, making the most of your data requires a way to organize it so that you automatically get:

- Full visibility
- Accurate, actionable recommendations
- Easy-to-understand reports and dashboards
- Customizable views to narrow your focus
- A comprehensive view of costs and usage that augments the information you get from tags with other important considerations, such as discounts and credits

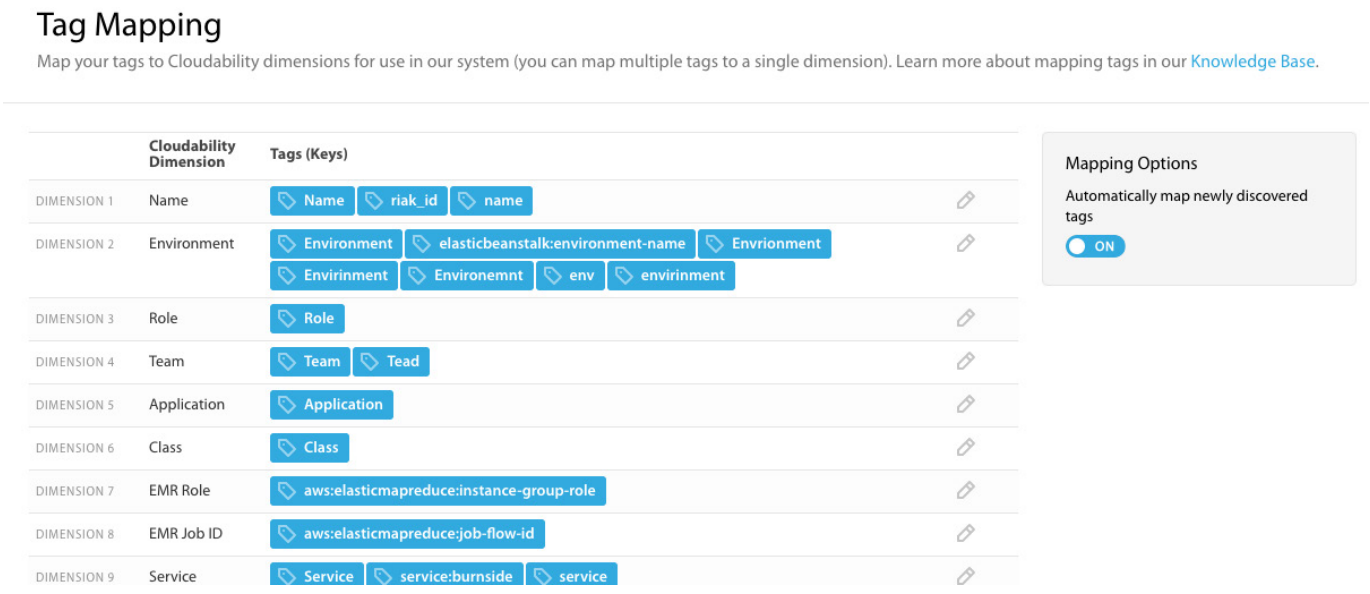
The only way you're going to do that is by applying data science, analyzing the data and presenting it in a way that allows you to make actionable and informed decisions. That's where a cloud management platform like Cloudability comes in. With it, all your raw data is organized and easily accessible. You'll have a single source of truth for how much your cloud infrastructure costs, how it's being used, how you can save more money and how you can increase efficiency.

To help give you an idea of how effective a cloud management platform can be, we're going to walk you through a few of our solutions to getting maximum visibility and cost allocation through tags and consolidated billing.

## Challenge: Getting data reported consistently

### Solution: Tag Mapping

As we mentioned before, tags have to be an exact match to be grouped together, which means even small variations can throw off your reports. To account for this, Cloudability has a feature called Tag Mapping.

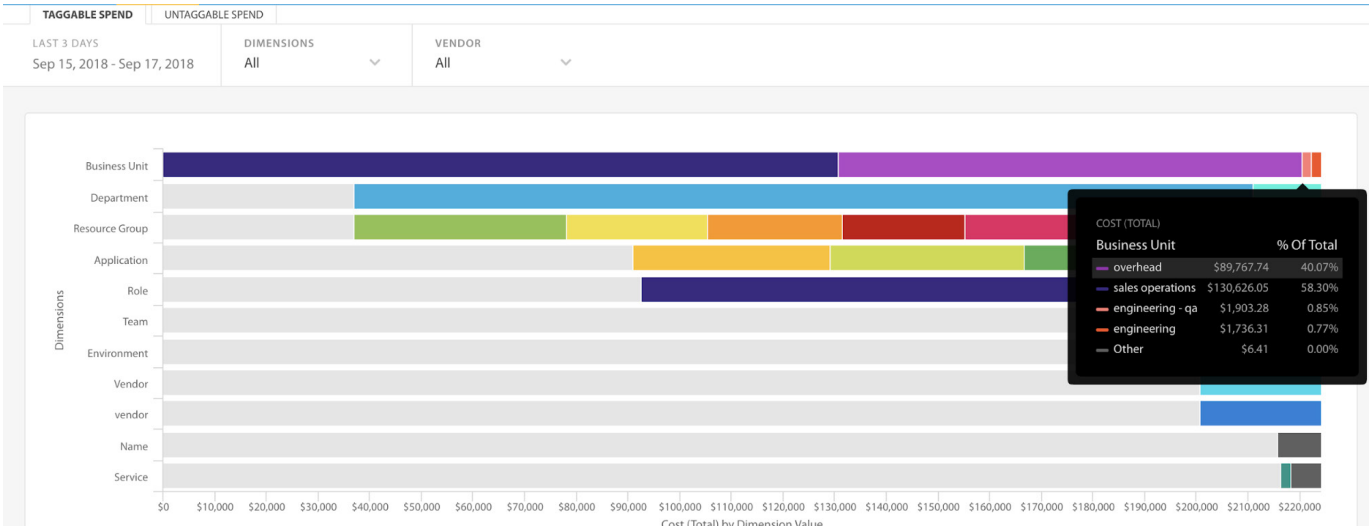



This feature lets you take multiple versions of what should be a single tag and map them to one dimension so you can make sure you're getting all the data that applies to the resource. By using Tag Mapping, small variations like **Environment**, **environment**, **Environement**, or **Environmnet** all get mapped together correctly.

# Challenge: Getting a global view of your tag usage

## Solution: Tag Explorer

A large spreadsheet of tags can make it tricky to get the global view you need for complete visibility. The Tag Explorer feature in Cloudability was designed to give you that global view by sorting all of your resources into tag keys and breaking down those resources by tag values.




The Tag Explorer displays the most widely used tags. The graph makes it easy to quickly see how much each tag is spending. Hovering over a bar displays a breakdown of those costs according to the values associated with the key.

Many customers want to see the most expensive untagged resources. Click on the grey segment of any dimension and you get a detailed report. For example, here's a report of the most expensive untagged resources by department.

Most Expensive Resources where Team = Not Set

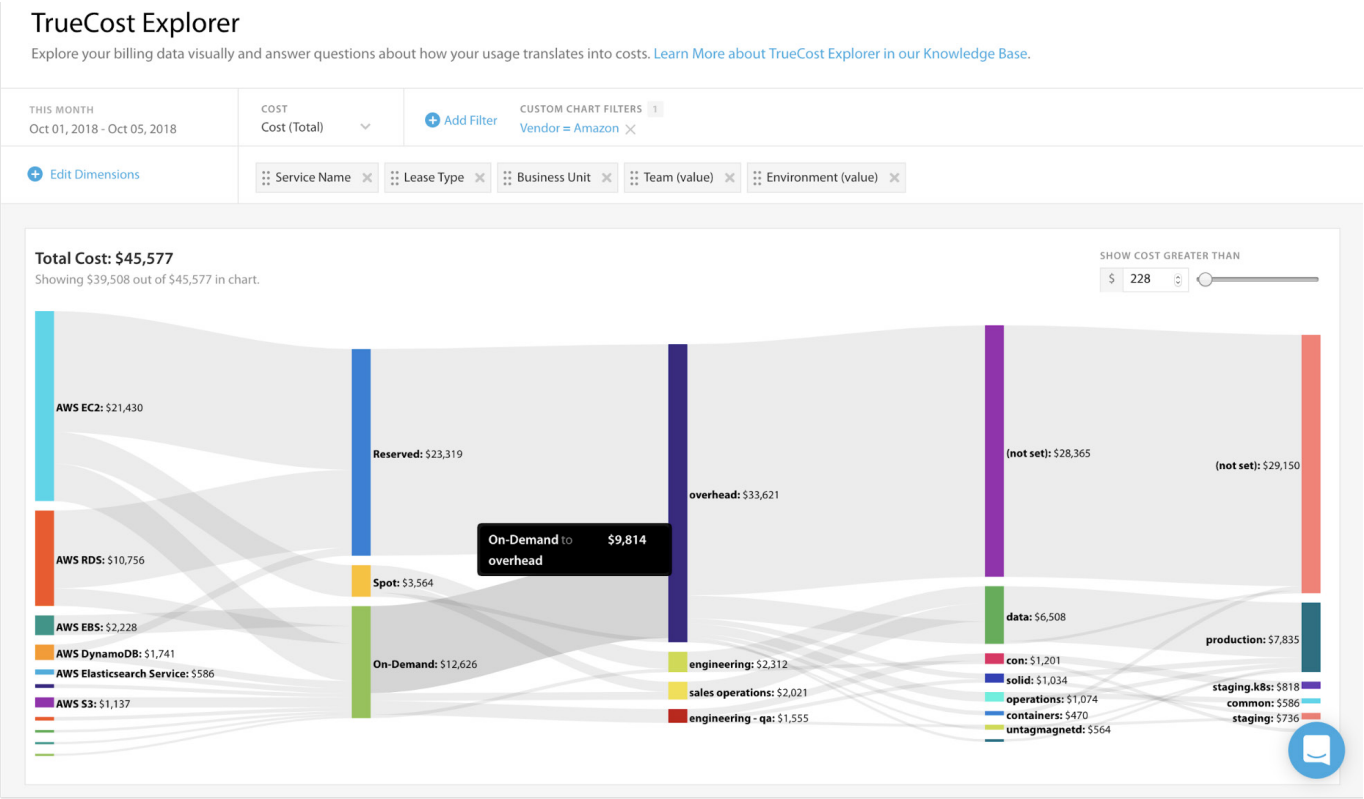
| Dimensions | | | | Metrics |
|---|---|---|---|---|
| Account Name | Region | Service Name | Resource ID | Cost (Total) |
| AWS Production | us-east-1 | AWS RDS | arn:aws:rds:us-east-1:338273444847:cluster:cluster-x75ozgni4u6uyqjkzfer7gh22u | $318.17 |
| AWS Production | us-east-1 | AWS RDS | arn:aws:rds:us-east-1:338273444847:cluster:cluster-6a3sfayl236flzogicnjasrhii | $310.05 |
| AWS Production | us-east-1 | AWS RDS | arn:aws:rds:us-east-1:338273444847:cluster:cluster-tdqn63pqdai5iyi7zfw2ytsijq | $223.50 |
| AWS Production | us-east-1 | AWS VPC | arn:aws:ec2:us-east-1:338273444847:natgateway/nat-0219f3266f2813bc7 | $189.91 |
| AWS Production | us-east-1 | AWS DynamoDB | arn:aws:dynamodb:us-east-1:338273444847:table/pipeline-aws-describe-data-production/backup/01537268407516-9b6e6c25 | $122.58 |
| AWS Production | us-east-1 | AWS S3 | Leg-detailed-line-item-files | $97.78 |
| AWS Production | us-east-1 | AWS S3 | Torpedo-backup | $63.88 |
| AWS Production | us-east-1 | AWS RDS | arn:aws:rds:us-east-1:338273444847:cluster:cluster-puhqkgr4lvpdw53czimlc2gtra | $60.77 |
| AWS Production | us-east-1 | AWS VPC | arn:aws:ec2:us-east-1:338273444847:natgateway/nat-0a03b4a7051208e9b | $59.02 |
| AWS Production | us-east-1 | AWS SQS | arn:aws:sqs:us-east-1:338273444847:production-compute-fetcher-reservations | $51.28 |

Not all AWS resources can be tagged. Click the **Untaggable Spend** tab to see all AWS resources that don't currently support tags and learn what they cost.

# Challenge: Seeing how cost relates to teams and usage

## Solution: True Cost Explorer

It can be incredibly difficult to track how your AWS costs directly relate to your team layout and your cloud usage, let alone to get a visual reference for it. We solved this with our True Cost Explorer feature. The True Cost Explorer works a bit like the pivot tables in Excel or other spreadsheet programs — except it also shows how the compiled data is related. In Cloudability, you can use the True Cost Explorer to summarize all kinds of data and show the relationships between them. That includes tags.




In the screenshot above, you'll see the various AWS resources in play, followed by how they're billed. The last three groupings are for tags based on **Business Unit**, **Team** or **Environment**. Let's follow the path of EC2 instances. If you move from left to right, you'll see that the majority of them are billed as RIs, with some Spot billing for Engineering and Sales Operations. The rest are billed as On-Demand. You can also see that a huge chunk of costs under Overhead aren't tagged to a specific Team or Environment. So in a glance, you can see spots where plugging tagging holes might help to answer questions about lowering overhead costs.

As a note, the True Cost Explorer can also help you drill down to get full visibility on costs down to the Resource ID. Hover over each section, and you can see the amount moving between sections, such as the $9,814 of On-Demand billing that's part of overhead. Click on the bars and you can see options like the top line items, time comparisons or apply filters. You can find out what else the True Cost Explorer can do on our website. It's an extremely powerful tool, and it's an example of the visibility and control you can get when you have a solid tagging strategy.

## Challenge: Comparing cost metrics can be complicated

**Solution: Custom reports using tags**

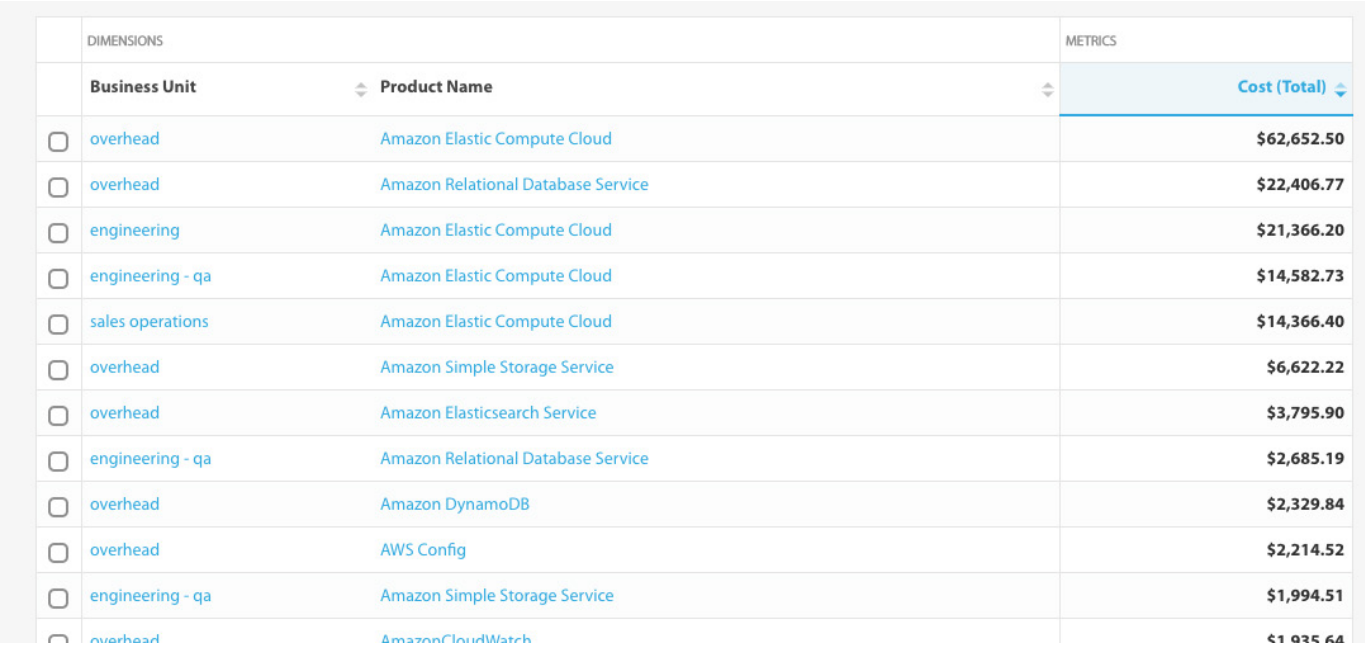
Cloudability ingests all cost and tagging data so you can create detailed cost allocation reports across any of the tagging dimensions. Stakeholders can drill down to the data they need or create simple cost allocation reports that are easy for anyone to understand and use.

The custom report feature in Cloudability is designed for this application. Add the tags you want to track as dimensions in the custom reporting interface, and you can report them with one of Cloudability's cost metrics. Those metrics include:

- Cost (Blended)
- Cost (Total)
- Cost (Adjusted)
- Cost (Amortized)
- Cost (Adjusted Amortized)

(To learn more about how you can use these metrics to learn the true cost of running your cloud, read our blog post, Understanding AWS Cost Metrics: Do You Have a True Cost Metric?)

Once the report is generated, you can view the combined costs of every resource with the selected dimensions. You'll see which cost centers are spending more and by how much.



| | DIMENSIONS | | METRICS |
|---|---|---|---|
| | Business Unit | Product Name | Cost (Total) |
| ☐ | overhead | Amazon Elastic Compute Cloud | $62,652.50 |
| ☐ | overhead | Amazon Relational Database Service | $22,406.77 |
| ☐ | engineering | Amazon Elastic Compute Cloud | $21,366.20 |
| ☐ | engineering - qa | Amazon Elastic Compute Cloud | $14,582.73 |
| ☐ | sales operations | Amazon Elastic Compute Cloud | $14,366.40 |
| ☐ | overhead | Amazon Simple Storage Service | $6,622.22 |
| ☐ | overhead | Amazon Elasticsearch Service | $3,795.90 |
| ☐ | engineering - qa | Amazon Relational Database Service | $2,685.19 |
| ☐ | overhead | Amazon DynamoDB | $2,329.84 |
| ☐ | overhead | AWS Config | $2,214.52 |
| ☐ | engineering - qa | Amazon Simple Storage Service | $1,994.51 |
| ☐ | overhead | AmazonCloudWatch | $1,935.64 |

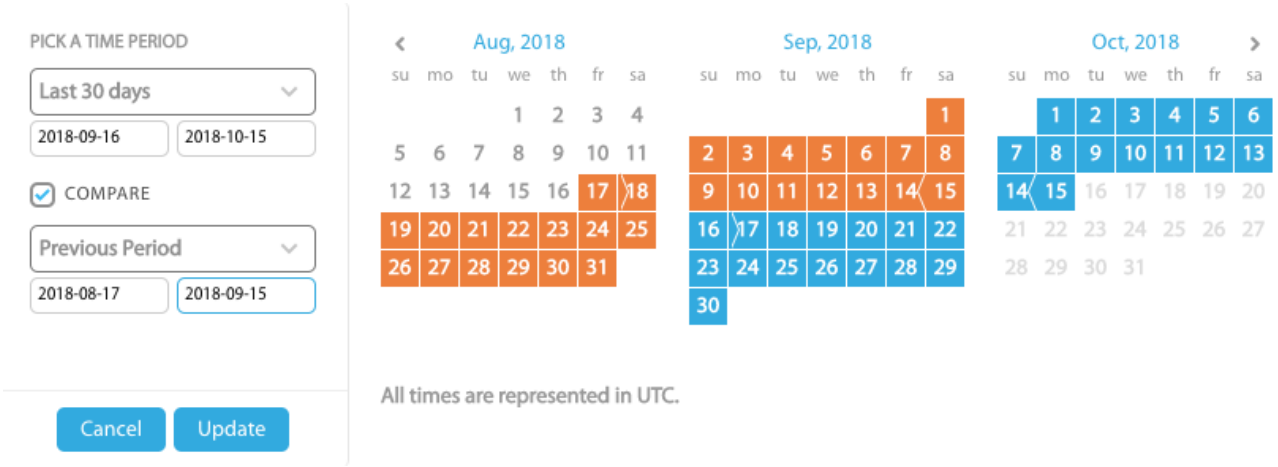
This report shows the Cost (Total) for the previous 30 days, broken down by which AWS product it's applied to. You can see what each account is spending and what to charge back.

You can use Cloudability to get a more complete understanding of costs by adding additional information such as blended versus unblended rates. This is important if you're using reserved instances across multiple consolidated billing accounts. You can sometimes buy reservations in one consolidated billing account but, if it's not being used for a particular hour, it can be applied automatically by AWS to other consolidated accounts under the same master payer structure. You'll be able to unravel where you've paid for the reserved instances, see where you're getting secondary benefits, and know that you're allocating that cost back to the right businesses.

## Challenge: Tracking use over time in a flexible infrastructure

### Solution: Using tags to create trends over time

If you want to know which cost centers are driving your costs up or down, then you need to be able to compare two similar time periods. In Cloudability, we built time period comparison functionality into our dashboard widgets and our reports. You select a range of days you want to evaluate, then select the Compare option. Going forward, the chart will compare the current 30 days to the last 30 days.




*Selecting 30-day comparison periods.*

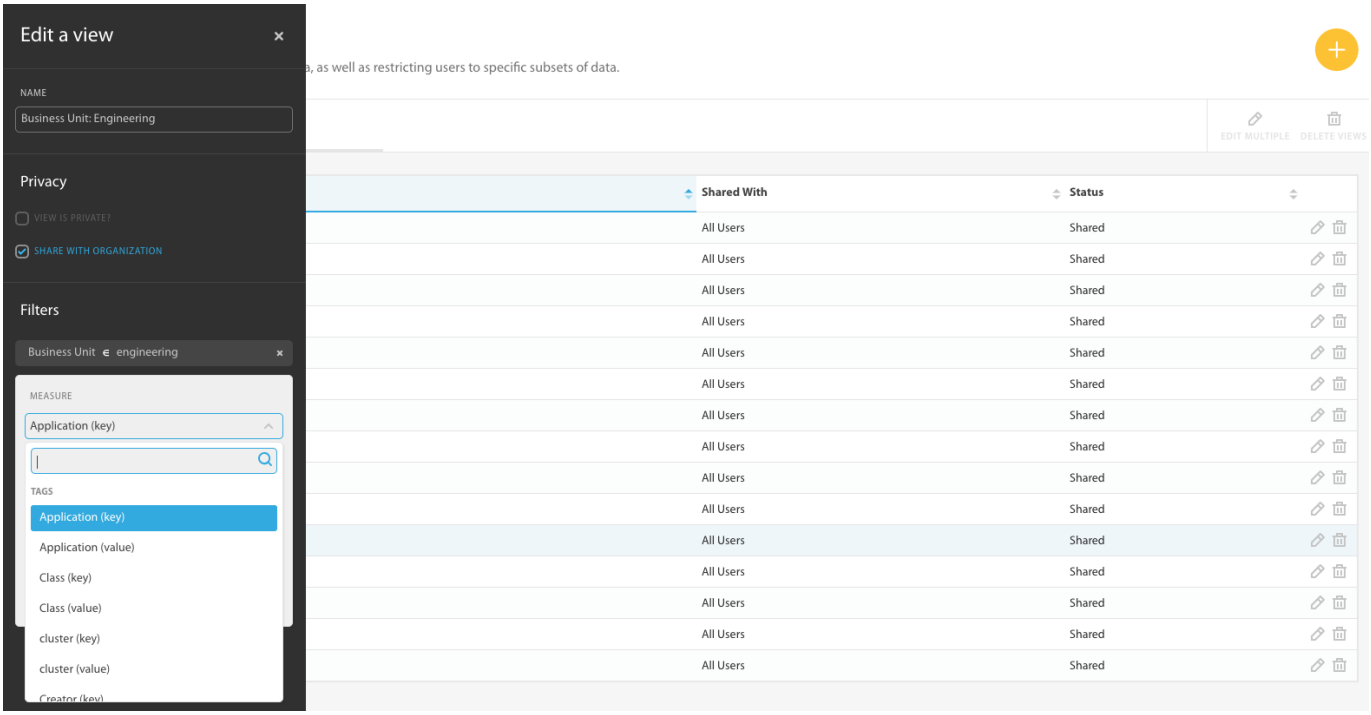
| DIMENSIONS | METRICS | | | | |
|---|---|---|---|---|---|
| **Environment (value)** | **Usage Hours** | **Cost (Total)** | **Cost (Amortized)** | **Cost Adjustment** | **Cost (Adjusted)** |
| production | 182,190.450707 | $18,504.67 | $18,504.67 | $379.80 | $18,884.47 |
| Sep 01, 2018-Sep 30, 2018 | 5,184,099.441695 | $216,510.14 | $216,510.14 | $-12,027.86 | $204,482.28 |
| Aug 02, 2018-Aug 31, 2018 | 5,001,908.990987 | $198,005.46 | $198,005.46 | $-12,407.65 | $185,597.81 |
| staging.k8s | 29.575833 | $-100.33 | $-100.33 | $64.32 | $-36.01 |
| Sep 01, 2018-Sep 30, 2018 | 6,993.320833 | $3,836.84 | $3,836.84 | $-1,503.73 | $2,333.11 |
| Aug 02, 2018-Aug 31, 2018 | 6,963.745 | $3,937.17 | $3,937.17 | $-1,568.06 | $2,369.11 |
| production.k8s | 0.264723 | $-274.47 | $-274.47 | $286.04 | $11.58 |
| Sep 01, 2018-Sep 30, 2018 | 6,479.00 | $3,563.83 | $3,563.83 | $-1,241.10 | $2,322.74 |
| Aug 02, 2018-Aug 31, 2018 | 6,478.735277 | $3,838.30 | $3,838.30 | $-1,527.14 | $2,311.16 |
| standby | 0.00 | $0.00 | $0.00 | $0.00 | $0.00 |
| Sep 01, 2018-Sep 30, 2018 | 4,320.00 | $171.65 | $171.65 | $-60.82 | $110.83 |
| Aug 02, 2018-Aug 31, 2018 | 4,320.00 | $171.65 | $171.65 | $-60.82 | $110.83 |
| common | -7.033889 | $23.61 | $23.61 | $-16.50 | $7.11 |
| Sep 01, 2018-Sep 30, 2018 | 20,133.00 | $3,930.40 | $3,930.40 | $-790.75 | $3,139.65 |
| Aug 02, 2018-Aug 31, 2018 | 20,140.033889 | $3,906.79 | $3,906.79 | $-774.25 | $3,132.54 |
| operations | -976.308334 | $-143.68 | $-143.68 | $105.88 | $-37.80 |
| Sep 01, 2018-Sep 30, 2018 | 7,328.124442 | $1,498.31 | $1,498.31 | $-531.00 | $967.31 |
| Aug 02, 2018-Aug 31, 2018 | 8,304.432776 | $1,641.99 | $1,641.99 | $-636.88 | $1,005.11 |
| development | -2,054.593846 | $-742.60 | $-742.60 | $118.82 | $-623.79 |
| Sep 01, 2018-Sep 30, 2018 | 1,462.86589 | $16.35 | $16.35 | $-2.62 | $13.74 |
| Aug 02, 2018-Aug 31, 2018 | 3,517.459736 | $758.96 | $758.96 | $-121.43 | $637.52 |

In the example above, the report shows costs broken down by the **Environment** tag. The **production**, **staging.k8s**, **production.k8s** and other environments are all shown. You see not only what each environment costs, but what the change in spending is between time periods. If you look at a dashboard and realize that spending for a particular business unit is starting to go up, you can pull up a report very quickly and see there's a large increase in the TEST environment. You don't know at this point if it's a good increase or a bad increase, but you can at least identify the change so you know where to look.

# Challenge: Seeing only the cost & usage data you need to see

## Solution: Customized Views

Product managers often want to know how much it costs to operate the products that they manage — and only their products. By using the tags and consolidated billing combined with the Cloudability Views feature, you can create a customized View with specific tags and member accounts.



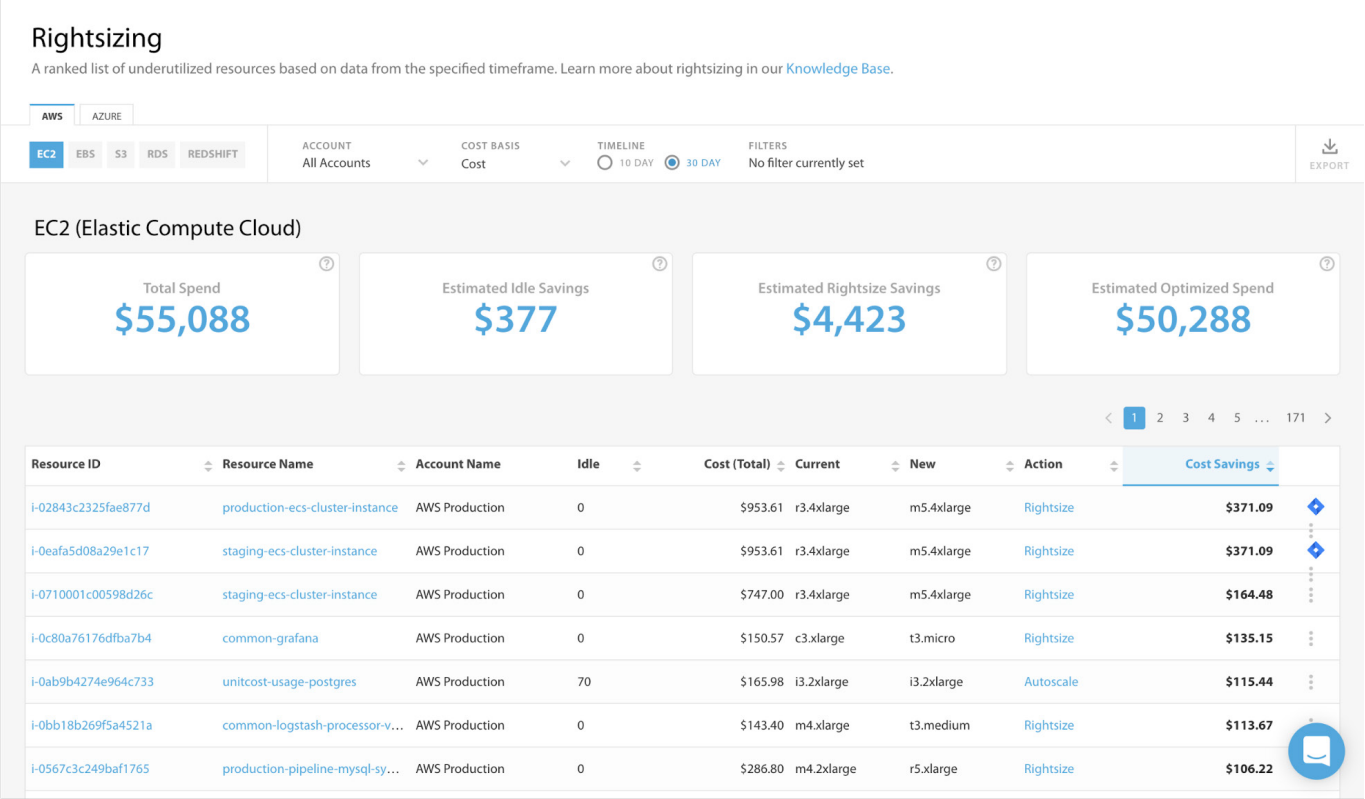

A manager can create a View where people in the Business Unit of **Engineering** could see their costs — and only their costs.

The product manager can limit the view to just the applicable data by selecting a drop-down menu that allows them to select the view that interests them.

# Challenge: Identifying unused and over-provisioned resources

## Solution: Rightsizing by account

Two of the most common reasons for wasted cloud spend are unused resources and over-provisioned resources. The Cloudability Rightsizing feature was developed to find those areas of waste and make recommendations that can help you save money without any drop in performance. Using a combination of data science and machine learning, the recommendations are based on your past use cross-referenced with years of billing data to compute the possible risk for each rightsizing action.




A well-planned Consolidated Billing setup integrates extremely well with Rightsizing. The third column (Account Name) is tied to the member account responsible for that instance. Rightsizing recommendations can be filtered to specific accounts to help focus your optimization efforts.

## Key takeaways

Get all the stakeholders involved when you're devising your tagging and consolidated billing strategy.

Drive your strategy by formulating the specific questions you want answered.

Use automation to enforce your tagging policy.

Tag every cloud resource that you can, and use consolidated billing to cover the gaps that tagging can't.

A cloud management platform is one of your best options for analyzing your data and making the most out of it.

## Getting More from Your AWS Cloud

In the end, your ability to effectively manage your cloud costs will directly impact how much value you get for your cloud use. Over the years, we've found that the savings our customers get are often turned around and invested back into new projects. In essence, they've managed to increase their cloud budgets without increasing their costs.

Think of it this way: the combination of a solid tagging strategy, a deliberate consolidated billing structure, and Cloudability can give you the insights needed to lower your cloud costs by 30% without losing any performance. If your cloud spend is $5 million every year, then you'll get the same resources for only $3.5 million. That's an additional $1.5 million that you can now use to add those competitive features that will make all the difference.

And it starts with tagging and consolidated billing. Together, they're one of your most valuable cloud cost management tools. Use them well.

# What's Next

## About Cloudability

Cloudability helps IT, Finance and Business teams manage the variable spend model of cloud with a FinOps platform that uses data science, machine learning and automation. With over $9 billion in cloud spend under management, we enable customers to create financial accountability and lower the unit economics of cloud.

Get the resources you need at cloudability.com/resources

## About FinOps

FinOps is a combination of best practices, culture and systems that enable distributed IT, Finance and Business teams to tune cloud deployments for speed, cost or quality. The FinOps journey consists of three iterative phases — Inform, Optimize, Operate.

Learn about FinOps by reading FinOps: A New Approach to Cloud Financial Management.

### Get Your Cloud Under Control

Whether you're a cloud-native company moving quickly or an enterprise looking to migrate to the cloud, there's a complex journey ahead. Get the resources to learn more about building and managing a cost-efficient cloud.

cloudability.com/resources