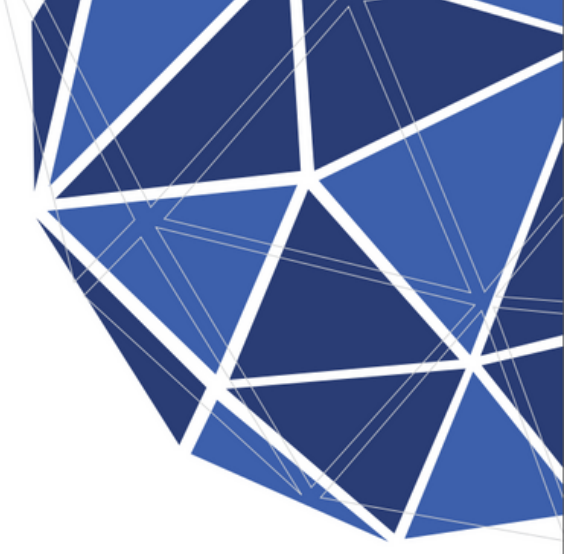
conveyor
by dataminded

A **4 simple step guide** to reduce your cloud costs by

More than 50%

www.dataminded.com

Introduction



Building data platforms in the cloud is changing. Gone are the days when you would manually set up a few EC2 instances and run some modest data processing on them. Solutions in this space have moved up the value chain, going from IaaS to PaaS to SaaS. This made it a lot easier for data teams to get started with data. The so-called “Modern Data Stack” is defined in terms of this movement.

> The most important difference between a modern data stack and a legacy data stack is that the modern data stack is hosted in the cloud and requires little technical configuration aby the user. These characteristics promote end-user accessibility as well as scalability to quickly meet your growing data needs without the costly, lengthy downtime associated with scaling local server instances.— Fivetran

The modern data stack is hosted in the cloud and requires little technical configuration, making it easier for data teams to get started with data. Of course, this also came with price increases, by handing out more of the management of your data platform you are paying more to compute your data pipelines. Often, your license cost will be 2x - 4x or even more the price of the pure VM costs.

In this whitepaper, we have tried to summarize the present scenarios of the data cloud technologies and compare different pricing options. This whitepaper also focuses on our unique 4 step process of saving your cloud costs.

Let’s dive in!

Table of Content

What is the present scenario?

Big vendors often claim they have outstanding performance. But in most real-world examples there are only a few heavy-lifting data jobs and hundreds of smaller, almost trivial, jobs. Snowflake and Databricks are often seen as the biggest cost drivers in data platforms, but they are not alone. Azure Synapse and Google Bigquery also come with significant costs.

The one thing they all have in common is the pay-as-you-go pricing model. This is good since you do not have to pay for what you do not use. However, it disincentivizes planning and removes cost predictability.

The lack of predictability can result in bill shock as your cloud costs might suddenly double from one month to the next. It would not be the first time that a one-off report results in 2 years worth of data needing to be reprocessed, inflating your bill accordingly. Events like these starkly contrast with the expectation that cloud costs will only go up when more and more data pipelines are added to your platform.

Another problem that occurs is that, because of the rising costs, it takes longer to have some return on investment. And at some point, your company needs to see this to continue investments. When it is realised too late, further improvements to your data platform can be blocked.

4 steps that will help you out

Moving back to on-premise and ditching the cloud can be a way forward to make your IT costs more predictable. But, for most companies, this is not realistic. It takes a big upfront investment and a long-term commitment to go back on-premise. There is a better solution that is proven to help you out on this topic. We recommend using our unique 4-step process to reduce your cloud costs for your data workloads.

Le's deep dive into the four-step process on the next pages.

1. Right-size your resources

This sounds like a very obvious tip, but you would be surprised how easy it is to find wrongly configured jobs. There might be several reasons:

- An engineer copy pasted the config from another job and simply forgot to run a smaller cluster for a simple job. This results in small jobs, requesting a large amount of resources.
- A job was going out of memory because of an issue in the code. Since the business needed the new data today, an engineer increased the memory configured for the job. Later the bug was fixed but the engineer forgot to decrease the memory.
- There is no easy way to monitor resource consumption, and thus developers who are always under pressure to deliver new features do not configure resources until the bill becomes too big.

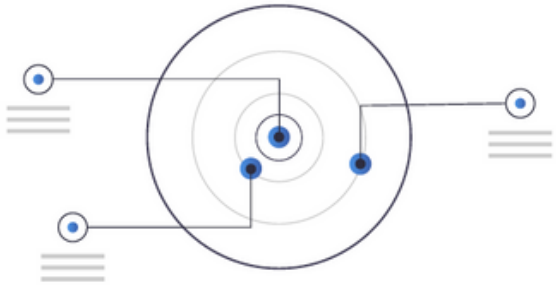
To combat this you should have a solution in place to monitor the resource consumption of your jobs. Jobs that don't use the resources provided should have their configuration changed to use fewer resources.

Another way to avoid wasting resources is to properly configure autoscaling for long-running clusters. Autoscaling will automatically add more resources to a cluster when needed, but it will also remove those resources when they are no longer needed. Spending some time configuring autoscaling can often result in significant cost savings.

Another common mistake is using the wrong tools for the job. Not every service is suited for every job. In the past, we have seen multi-node clusters being used to schedule a simple shell script because it was the standard way of doing things. This might be fine if you have only one such job, and all your other jobs use a cluster. But if you have multiple jobs following this pattern, you should select a more appropriate (and cheaper!) method to run your small jobs.

For example, if all your jobs are Spark jobs being run on an EMR cluster, it might be easiest to run your single-node python job on an EMR cluster too. However, at some point, it is more cost-efficient to run this on AWS Batch.

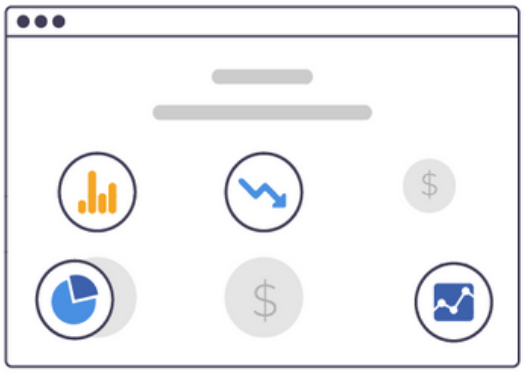
2. Go for SPOT instances



Spot instances are a way to save money on cloud costs, as cloud vendors offer heavy discounts on their spare capacity. You can get a discount of 60-80% compared to on-demand instances. However, there is a risk of losing your machine to a spot interrupt, which happens when the cloud vendor needs it for something else. To avoid this, you can use specific strategies to choose the correct spot instances. These strategies can help minimize spot interrupts.

Still, it's important to plan for redundancy and resiliency in your data platform when using spot instances, since spot interrupts may still occur. Also when not running spot instances this is important, as other issues can arise. One way to achieve this is by running your data platform on Kubernetes clusters. Keep in mind that there is an overhead to running a Kubernetes cluster, but it can result in a significant reduction in instance costs.

3. Reduce the license costs



These days, license costs can make up more than 50% of your cloud bill. Examples of data services that can become very expensive are Databricks, Snowflake, AWS EMR (Serverless), AWS Glue, AWS Redshift, Azure Data Factory, etc. There are more cost-effective services available.

If you have already started right-sizing your resources, your license costs will probably have been reduced quite a bit already. To reduce them even further there is the option to build it yourself. Don't do it to save $500 on your monthly bill. But if your yearly spend on service is high enough, it can warrant the investment to build it yourself. This option cuts out the license cost completely, but it is replaced by an upfront investment to build an alternative and do not forget about the maintenance cost.

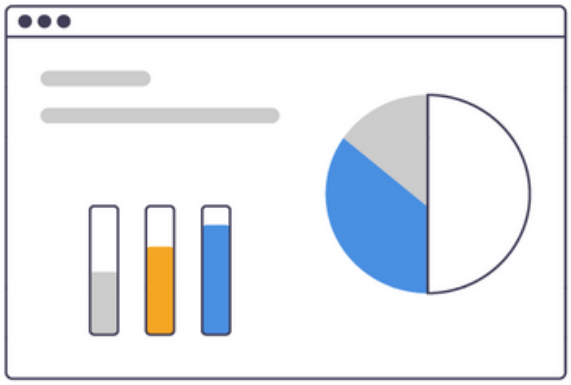
Most of the time you can also reduce your license cost by using services in a smarter way. For example, turning off unused compute. A notebook can be shut down when developers are not working and there are many other development resources that can be paused during the night. Setting up autoscaling correctly for these services will also reduce your license cost. Some of these tactics are handled in step 1, right-size your resources.

Another option is to use low-cost (managed) alternatives on the market. They don't work for every use case, but in many cases it is an option. When you are serious about reducing your cloud bill, you should put them on the table and find out which one can work for you. You can think of alternatives like Aiven Kafka, Airbyte, DuckDB, Ahana Cloud and Conveyor. You should evaluate how many features you use and need from your current service and if you can live without them.

Check out two of our blogs where we give some tips about this topic.:

- Customizing SageMaker Notebook Instances
- Cloud story: Spin down unneeded infrastructure quickly with Terraform to save OPEX

4. Monitor, analyse, iterate



Reducing costs is a continuous process, you need to have certain capabilities in place to easily do this. The most important is the ability to attribute cloud costs to individual projects. Attributing to individual jobs and pipelines would be even better. You can have a much more focused discussion on where you're spending most, where to act, and whether those actions actually have an impact.

You can also have a discussion about the value vs cost of certain projects, which might result in projects with negative ROI being canceled. Often we see certain pipelines being supported forever at companies when the value that was delivered is already long gone!

You can achieve this on AWS for example by correctly tagging all resources with the project name and department. After that in AWS cost explorer, you can easily get a view of how much each project is costing. Some resources might be shared (like an s3 bucket or EKS cluster), for these resources you might just split the cost evenly. Or start looking at more specialised tools like kubecost (https://www.kubecost.com/) to properly divide these costs.

Conclusion

Cloud computing costs can be a major expense for businesses. There are many strategies that can help to reduce the amount of money spent on cloud services.

First, it's important to right-size your application, and use autoscaling. This will reduce your cloud costs significantly when not already done. Secondly SPOT instances are an effective way to save money, using Kubernetes can help you use SPOT instances effectively. Thirdly look at cost-effective alternatives to the most expensive data solution such as Databricks and Snowflake. Finally, monitoring and analysing usage are key to finding the most cost-effective solution. By implementing these measures, businesses can save more than 50% of their total cloud bill.

Conveyor enables you to take all these 4 steps easily.

Conveyor enables you to focus on what really matters and iterate quickly. It is a managed compute and workflow platform that helps you develop, manage and monitor your data. With just a few taps you can build, deploy and scale your data projects right from the ideation to the production stage. Using Conveyor enables you to make faster releases of your use cases to production. All you have to do is write the data pipelines in the language of your choice and deploy your projects within seconds.

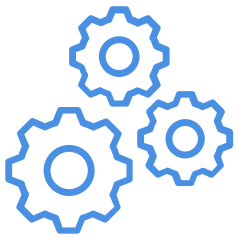
To know more about how Conveyor works check our website;

www.dataminded.com/conveyor >

Conveyor helps in 3 ways

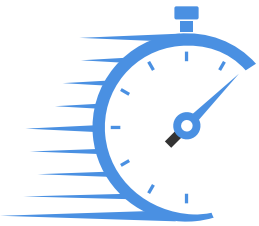
By simplifying data engineering

- You can get started with a single command. Also, scaffold your batch or stream processing project from a template.
- You can choose your compute size and deploy in seconds.
- You can extend your use cases with your own tools and libraries. Containerization is at the core.



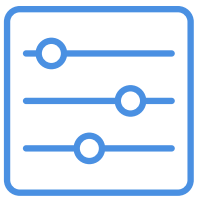
By providing velocity

- It's an easy journey from notebooks to production.
- You can experiment and test in isolated environments. You can also spin up new personal environments in seconds with a dedicated workflow manager.
- Metrics and logs are out of the box. You can track performance and errors with live access to metrics and logs.



By enabling you to control your costs

- You can analyse cost per project with cost monitoring & optimize where it best suits you.
- Enables by default to spot. Using spot results in typical savings of 70- 90% over on-demand pricing.
- It's fully managed. Services and infrastructure are always up-to-date. No patching, no management, and no worries!



Success story

Client in the R&D space

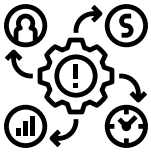
Situation

This organisation was building out its data platform and was already delivering its first use cases to businesses. Their Databricks costs reached over EUR 10K per month and grew rapidly, and this created pressure on the data organisation to reduce their monthly spending.

Solution

We worked with the client to identify their most expensive jobs and moved those to a containerized solution, running on spot instances. We measured cost savings made, and this encouraged us to move more workloads away from Databricks notebooks until all jobs were migrated.

Impact



We managed to bring their computing cost from **EUR 10K to EUR 3K**, which removed the pressure from the data team. This enabled them to build several more use cases instead of focusing on managing notebooks or worrying about increasing cloud costs.

conveyor
by dataminded

Our team offers the capability, technology, and way of working for organizations to collect high-quality data and find actionable insights. We assist you in becoming more data-minded and self-reliant to evolve from traditional reports to machine learning and intelligent applications. By building robust and scalable data solutions, Data Minded empowers teams to make more impactful decisions and achieve results beyond imagination.

Reach out to us.

Talk with the experts: Click here!

Our website: Visit now.

Contact Info

dataminded

info@dataminded.be

Vismarkt 17

3000 Leuven, Belgium