**11**

**Cost Considerations**

One of the primary goals of any business is to increase profitability while serving its customers. Cost is the key parameter of discussion when a project is initiated. Having an application upgrade and adding new product features heavily depends upon the amount of funding available. The product's cost is everyone's responsibility and needs to be considered in every phase of the product life cycle (from planning to post-production). This chapter will help you understand the best practices for optimizing costs for your IT solutions and operations.

**Cost optimization** is a continuous process and needs to be managed carefully without sacrificing customer experience. Cost optimization *doesn't mean* cost reduction but reduces the business risk by maximizing **return on investment** (**ROI**). You will need to understand your customer needs before planning any cost optimization and act accordingly. Often, if customers are looking for quality, they are ready to pay a higher price.

In this chapter, you will learn various design principles for the cost optimization of your solution. The cost aspect needs to be considered for every phase and component of the architecture. You will get an understanding of the right selection of technology to ensure cost optimization at every layer. You will learn the following best practices of cost optimization in this chapter:

* Design principles for cost optimization
* Techniques for cost optimization
* Cost optimization in the public cloud

By the end of the chapter, you will have learned about various techniques to optimize cost without risking business agility and outcome. You will have learned different methods to monitor the cost and apply governance for cost control.

**Design principles for cost optimization**

Cost optimization includes increasing business value and minimizing risk while reducing the cost of running a business. You need to plan by estimating the budget and forecasting expenditure. To realize cost gain, you need to implement a cost-saving plan and closely monitor your expenditure.

You can follow several principles that help you achieve cost optimization. The following sections talk about the common design principles that allow you to optimize cost. You will find that all cost optimization design principles are closely related and complement each other.

**Calculating the total cost of ownership**

Often, organizations tend to overlook the **total cost of ownership** (**TCO**) and decide based on the upfront cost to acquire software and services, known as **capital expenditure** (**CapEx**). While the upfront cost determination is essential, in the long run, the TCO matters the most. The TCO includes both CapEx and **operational expenditure** (**OpEx**), covering all the dimensions of the application life cycle. The CapEx cost consists of the price organizations pay upfront to acquire services and software, while OpEx includes the cost of operation, maintenance, training, and retirement of software applications. It would be best to consider all associated costs to help make more strategic decisions while calculating your ROI in the long run.

For example, when you buy a refrigerator, which will run 24/7, you look for an energy-saving rating to keep your electricity bill low. You are ready to pay a higher price up front as you know the total cost over time will be lower due to the saving in the energy bill.

Now let's take an example of a data center. There is an upfront hardware acquisition cost involved, the CapEx. However, the data center setup requires additional ongoing costs, the OpEx, which includes heating, cooling, rack maintenance, infrastructure administration, security, and so on.

For a typical use case, when you are purchasing and implementing software, consider the following costs to calculate the TCO:

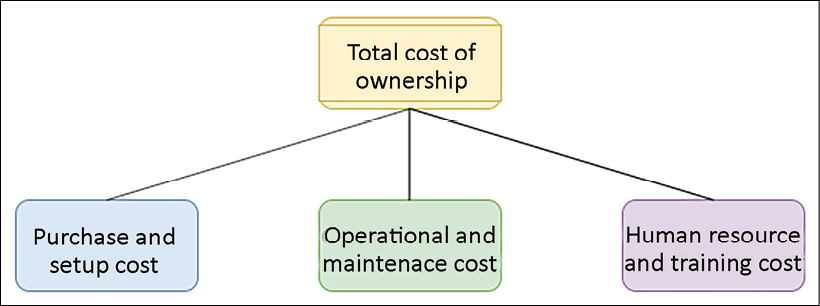


Figure 11.1: TCO for software

Let's look at this at a more granular level. Each TCO component has the following common costs involved for *off-the-shelf* software such as Oracle or an MS SQL database:

* **Purchase and setup costs**:These are the upfront cost to acquire the software and the services to deploy software. This includes the following:
  + Software price includes software with user licenses
  + Hardware cost includes purchasing a server and storage to deploy software
  + Implementation cost consists of the time and effort to get it ready for production
  + Migration cost includes moving data to the new system
* **Operational and maintenance costs**: This continues the cost of service to keep the software running for the business use case. This cost includes the following:
  + Software maintenance and support
  + Patching and updates
  + Enhancement
  + Data center cost to maintain hardware server
  + Security
* **Human resources and training costs**: This is the overhead cost to train staff to use the software to address business activities. This cost includes the following:
  + Application admin staff
  + IT support staff
  + Functional and technical consultant
  + Training cost and training tools

When looking for a solution, you will have multiple choices (such as taking out a subscription for a **Software as a Service** (**SaaS**) product such as **Salesforce CRM**). The SaaS model is mostly subscription based, so you need to determine whether you are getting the desired ROI for a more significant number of uses. You can take a hybrid approach and use the cloud to handle your hardware by choosing the **Infrastructure as a Service** (**IaaS**) option and installing off-the-shelf software. Overall, if the available software doesn't meet your requirements, you can choose to build it yourself. In any scenario, you should calculate the TCO to decide where you can make the maximum ROI. Let's look at budget and forecast planning, which can help to control TCO and achieve ROI.

**Planning the budget and forecast**

Every business needs to plan its expenditure and calculate ROI. Planning the budget gives guidance to organizations and teams on cost control. Organizations plan a long-term budget for 1-5 years, which helps them to run the business based on the funding required. These budgets then come down to the individual project and application levels. During solution design and development, the team needs to consider the available budget and plan accordingly. The budget helps to quantify what the business wants to achieve. The forecast provides an estimate of what the company is making.

You can consider budget as important strategic planning in the long run, and the forecast provides an estimate at a more tactical level to decide the business direction. In application development and operation, you can quickly lose track of the budget and overrun expected costs in the absence of a budget and a forecast. These two terms may be confusing, so let's understand the clear difference between a budget and a forecast:

|  |  |
| --- | --- |
| Budget | Forecast |
| Represents future results and cash flow for business objectives that you want to achieve | Represents revenue and the current situation of the business |
| Plans for the long term, for example, 1-5 years | Plans month to month or quarterly |
| Is adjusted infrequently, maybe once a year, based on business drivers | Is updated regularly based on actual business progress |
| Helps to decide business directions such as organization restructuring based on actual cost versus budgeted cost | Helps to adjust short-term operational costs such as additional staffing |
| Helps to determine performance by comparing planned cost versus actual cost | Isn't used for performance variation but for streamlining progress |

Forecast information helps you take immediate action, while the budget may become unachievable due to changes in the market. As shown in the following diagram, while you are working on day-to-day solutions, developments based on historic expenditure forecasts can prompt you to adjust the next month's cost:

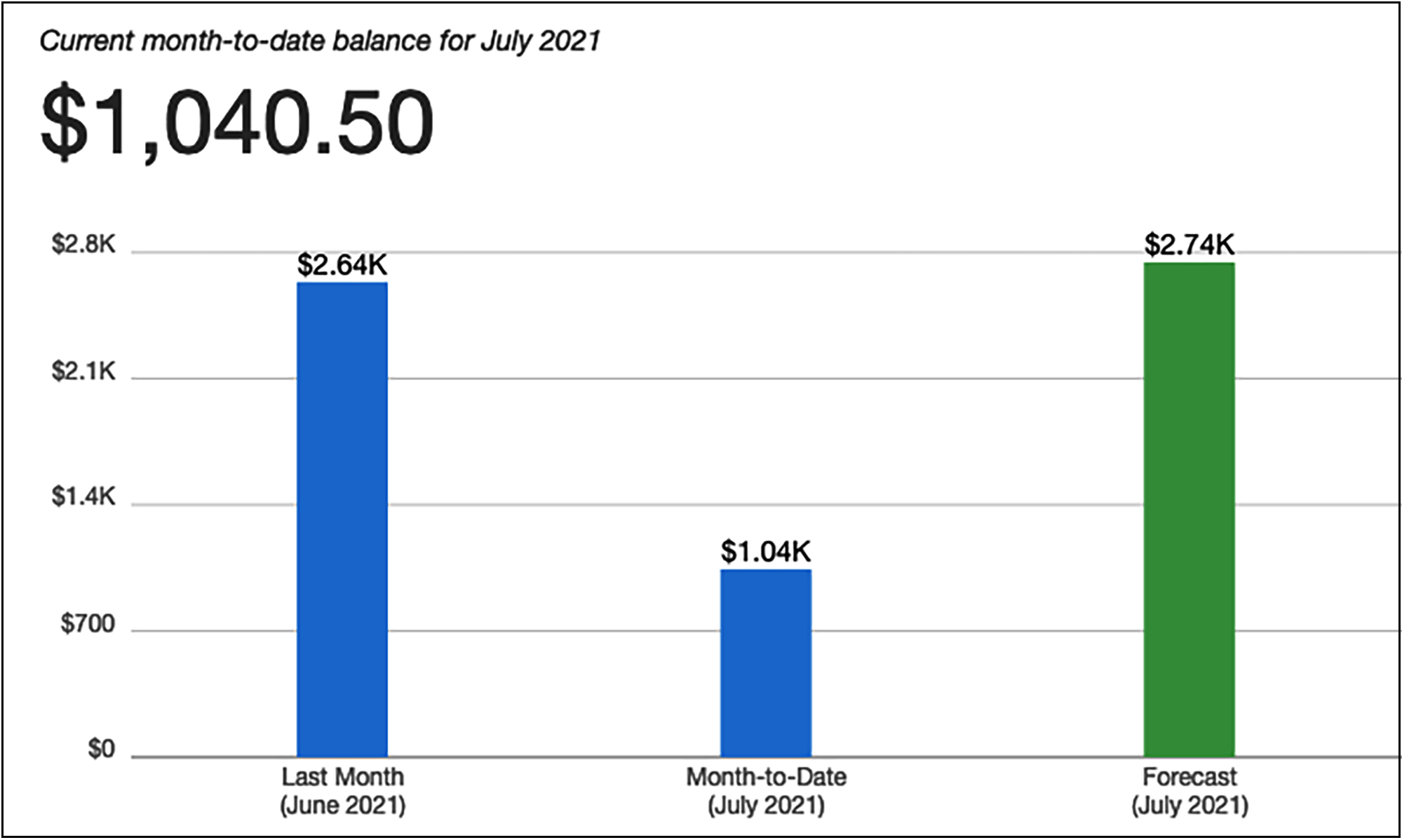


Figure 11.2: Billing and forecast report

In the preceding billing and forecast report, your monthly budget is $2,000, and the forecast shows that you will overrun your budget by the end of the month. Here, the forecast helps you act and control costs to stay within budget. Let's look at the mechanism to improve cost efficiency by managing demand and services in the next section.

**Managing demand and service catalogs**

Almost every organization has a centralized IT team, which works with internal business partners such as the application development team and the support teams of various business units. The IT team manages the demand for IT infrastructure, including the cost of all software and hardware, as well as support to manage application hosting. Often, business partners lack understanding of the cost drivers for the IT services that they use. For example, application development teams tend to overprovision their development or test environment, causing an additional cost.

Other factors that get the right sizing and demand forecasting from various organization units can help to match supply and demand. By consolidating all requirements in one place, an organization can benefit from economies of scale. You may achieve a lower variable cost because a large contract can achieve higher economies of scale. The right demand from all organization units is aggregated, which translates into lower prices.

Organizations can take one of the following two approaches to manage demand and services:

* **Demand management**: To save costs in your existing IT environments (where you may observe overspending may be prevalent), you can take the *demand-led approach*. It helps to improve cost efficiency in the short term, as you are not introducing many new services. You can analyze historical data to understand factors that are driving demand and capture cases of overprovisioning. You should establish a process between the IT team and business partners to streamline operational costs for IT services.
* **Service catalog management**: If there is a demand for new services and you don't have much historical data, you can take the *service-led approach*. In this approach, you need to understand the demand for the most frequently used services and create a catalog. For example, suppose the development team asks for a Linux server with a MySQL database to create a dev environment. In that case, the IT team can create a service catalog that helps the dev team acquire a small Linux and a database server. Similarly, the IT team can identify the most common set of services and attach a granular cost.

Each approach can have a significant cost saving in the short and long term. However, these transformations present substantial challenges as you need to change the project planning and approval process. The business and finance team need to align and understand the clear relationship between business growth and increased IT capacity. The cost model needs to be built around the most efficient approach by combining offerings from the cloud, on-premises, and off-the-shelf.

**Keeping track of expenditure**

You can find individual system costs by tracking expenditure and linking it to the system or business owner. Transparent expenditure data helps identify ROI and reward owners, optimizing resources and reducing cost. It can help you to determine what it costs every month for a department or project.

Saving cost is a shared responsibility, and you need to have a mechanism to hold everyone accountable for cost-saving. Often, organizations introduce a **show-back** or **charge-back** mechanism to share cost responsibility between organizational units.

The centralized billing account informs each organization unit regarding their expenditure in the show-back approach but doesn't charge the actual amount. Each business unit within an organization manages its budget under a master payee account in the charge-back mechanism. The master account charges back the amount to the business units as per their IT resource consumption every month.

When starting cost control for your organization, it is better to start with show-back as a stepping stone and move to charge-back as the organizational model matures. For each business unit, you should create expenditure awareness by configuring notifications so that teams get an alert as they approach the forecasted or budgeted amount of consumption. You should create a mechanism to monitor and control your cost by appropriately assigning them to the right business initiative. Provide visibility to create accountability for cost expenditures for each team. Cost tracking will help you to understand team operations.

Each workload is different; you should use the pricing model that suits your workload to minimize cost. Establish mechanisms that ensure business objectives are achieved by applying cost optimization best practices. You can avoid overspending by defining a tagging strategy to link business units with specific expenditure and using the **check-and-balance**approach.

**Continuous cost optimization**

If you follow cost optimization best practices, you should have a good cost comparison with existing activity. It's always possible to reduce the cost of your applications that are migrated and matured over time. Cost optimizations should never end until the cost of identifying money-saving opportunities is more than the amount of money you will save. Until that point is reached, you should continually monitor your expenditure and look for new ways to save on cost. You should keep finding an area to save costs by removing idle resources.

For an architecture that is balanced in terms of its cost and performance, ensure that the cost paid for resources is well utilized and avoids any significantly underutilized IT resources such as server instances.

A biased utilization metric showing exceptionally high or low cost will harm your organization's business.

Application-level metrics for cost optimization need to be considered carefully. For example, introduce archival policies to control data storage capacity. To optimize the database, you should check for appropriate database deployment needs, such as if multi-location deployments for the database are essential or whether provisioned **Input/Output Operations Per Second** (**IOPS**) are applicable as per your database utilization needs. To reduce your administrative and operational overhead, you can use the SaaS model to help your employees focus on applications and business activities.

To identify a gap and apply necessary changes for cost-saving, you should implement resource management and change control processes during the project life cycle. Your goal is to help your organization design the architecture as optimally and cost-effectively as possible. Keep looking for new services and features that might directly reduce your costs.

Let's learn some techniques that can help you to optimize the cost and increase ROI.

**Techniques for cost optimization**

To gain a competitive edge and keep up with rapid growth, enterprises are investing more in technology. With economic instability, cost optimization becomes an essential but challenging task. These companies spend a lot of time and research on reducing costs in the procurement process, operation, and vendors. Many companies even share data centers, call centers, and workspaces as a cost-saving method. Sometimes organizations delay upgrades to avoid buying new expensive hardware.

The organization can save more if it takes a broader look into the overall information technology architecture across its business units. Improving existing architecture can open doors to bring more opportunities and business to the company, even if it requires a bit more adjustment in the budget. Let's identify the focus area where companies can save money and gain more revenue with techniques such as moving to the cloud, simplified architecture, virtualization, and shared resources.

**Reducing architectural complexity**

Organizations often lack a centralized IT architecture, resulting in each business unit trying to build its own set of tools. Lack of overall control causes a lot of duplicate systems and data inconsistency. IT initiatives in individual business units are driven by a short-term goal.

In such cases, business units are not well aligned with long-term organizational vision, such as the digital transformation of the entire organization. Further, it adds complexity to maintain and upgrade those systems. Taking a simple step to define set standards and avoid duplication can help to save costs.

In the following diagram, you can see a complex architecture on the left-hand side, where business units are working in their own application without any standardization, causing duplicate applications with a lot of dependencies. This kind of architecture results in high costs and risks. Any new experiment takes a long time to market, which results in losing the competitive edge. A standard process can provide a holistic view and high flexibility to create an agile environment by applying automation, which helps to reduce the overall cost and results in a more significant ROI:

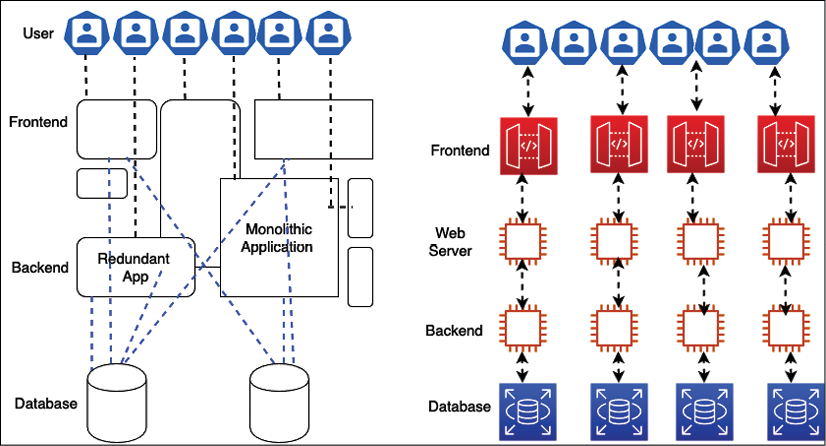


Figure 11.3: Architectural standardization

The first thing is to eliminate duplication and identify the function reuse across the business unit to reduce the architectural complexity. During *gap analysis* of existing architecture, you will find there is so much code, so many existing components, and a project that can be reused across the organization to support your business requirement. To reduce the complexity of IT architecture, think of an out-of-the-box solution that fits your business needs and provides an ROI. Customization should be your last approach if no other option is available.

Any new application needs to have an accessible integration mechanism to interact with the existing system using **service-oriented architecture** (**SOA**). Harmonizing the user interface design across the application provides a set of standard UI packages, which can be reused for any new application.

Similarly, other modules can be reutilized with service-oriented design. You learned about SOA patterns in *Chapter 6*, *Solution Architecture Design Patterns*, which help you keep all the different pieces of software working separately and still communicating with each other to build an entire system.

In the modular approach, each team is responsible for developing a service, which every team across the organization can use to avoid duplication. As an architect, you should help the team create a service-oriented design, where each team handles individual architecture components as a service that can be developed independently. With the help of microservices architecture, you can deploy an entire application in a modular way. If one component is not working, you can rework it without impacting the whole application. For example, a payment service developed to collect payment from a customer visiting an e-commerce website can also be utilized to make payments to vendors in the vendor management system.

Once you set up a centralized IT architecture, taking a modular approach helps you keep the cost down. Empowering your IT architecture team can help to align organizational units with the company's vision and support other parallel projects to follow the overall strategy. It also helps to provide consistency in other critical services that are often overlooked, such as legal, accounting, and human resources.

With the help of the IT architecture team, you can get excellent feedback and make sure that projects are aligned with business needs and requirements. By overseeing the overall architecture across teams, an architect can advise whether there is any duplicate effort, project, process, or system that is not aligned with the business need. The centralized architecture will reduce complexity and tech debt, bring more stability, and increase quality. The overall idea of centralized architecture is to increase IT efficiency, so let's learn more about that.

**Increasing IT efficiency**

Nowadays, every company uses and consumes IT resources. A lot of funding is consumed by too many servers, laptops, storage capacity, and software licenses. The license is one of the resources that is sometimes underused, undiscovered, idle, or installed incorrectly and consumes a lot of funding. A centralized IT team can lead the effort for license optimization by keeping track of used software licenses and retiring additional licenses. They can save costs by negotiating a bulk discount with the vendor.

To increase IT efficiency, you may want to cancel non-compliant projects that take additional funding and resources. Also, you should help teams to revisit the strategy to support or terminate any unused and non-aligned projects continuously. The following methods can be considered for cost optimization:

* Re-evaluate projects with high costs, as they may not be well aligned with the business vision. Reshape projects that have high value but no direct impact on the IT strategy.
* De-prioritize projects that have little to no business value even though they are aligned with the IT strategy.
* Cancel non-compliant projects with low business value.
* Decommission or retire unused applications.
* Replace old legacy systems by modernizing them to reduce maintenance costs.
* Avoid duplicate projects by reutilizing existing applications.
* Wherever possible, consolidate data and develop an integrated data model. You will learn about maintaining a centralized data lake in *Chapter 13*, *Data Engineering for Solution Architecture*.
* Consolidate vendor procurement across the organization to save costs on IT support and maintenance expenditure.
* Consolidate any system that does the same thing as payment and access management.
* Eliminate costly, wasteful, overprovisioned projects and expenditure.

Moving to the cloud can be an excellent consideration to increase IT resources efficiently and reduce costs. The public cloud providers, such as **Amazon Web Services** (**AWS**), offer a *pay-as-you-go* model that means you only pay for what you are using. For example, the developer desktop can shut down during non-working hours and weekends, reducing workspace costs by up to *70%*. The batch processing system needs to be brought up only to process jobs and can be shut down immediately afterward. It works just like any electrical appliance that you switch off when not required to save electricity costs.

Applying automation is a great mechanism to increase overall IT efficiency. Automation helps eliminate costly human labor and reduces the time spent performing daily routine jobs without error. Automate things wherever possible to provision servers, run monitoring jobs, and process data.

Make sure to do the right trade-off to improve results while deciding to optimize costs. Let's take an example. If you are going to a theme park where you want to go on lots of good rides, you are willing to pay a higher price to see the value of the money you are spending. To attract more customers, if the vendor decided to reduce the price and make adjustments by lowering the number of enjoyable rides, there is a chance that you will go to another theme park as you are looking for a good time. Here, competitors will gain an edge and attract existing customers, while the current vendor will lose business. In this case, cost reduction is adding business risk, which is not the right cost-optimization approach.

Your goal should be measurable, and these measures should focus on both business output and the cost of your system. The quantitative measure helps you understand the impact of increasing output and reducing the cost. The organizational and team-level goals must align with the end users of the application. At the organizational level, the goals will be across organizational business units. At the team level, they will be more aligned with individual systems. You can set up a goal at the business unit level, for example, to reduce the cost per transaction by 10% every quarter or 15% every six months. Defining the goal ensures that systems improve over their lifetime.

**Applying standardization and governance**

Organizations need a strategy to analyze misalignment and overconsumption, reduce complexity, and define guidelines to use appropriate and efficient systems and implement a process wherever it is required. Creating and implementing these guidelines will help companies develop a standard infrastructure and reduce duplicate projects and complexity.

To implement governance, you need to set up resource limits across the organization. Putting the service catalog in place with **Infrastructure as Code** (**IaC**) helps ensure that teams are not overprovisioned with resources beyond their allocated capacity. You should have a mechanism to understand and take action on business requirements quickly. Take both resource creation and decommissioning into account when applying resource limits and defining the process to change them.

Businesses operate multiple applications by various teams. These teams can belong to different business units within their revenue stream. Determining resource costs to the application and business unit or team drives efficient usage behavior and helps reduce cost. You can define resource capacity based on cost attribution and the group's, organization unit's, or department's requirements. To organize the cost structure, you can use resource tagging along with account structuring.

As shown in the following screenshot, you can organize your accounts in different organization units such as HR and Finance, and each department under the organization unit can have its own accounts. For example, here, **HR** has separate accounts for **Payroll** and **Marketing**, while **Finance** has individual accounts for **Sales** and **Marketing**:

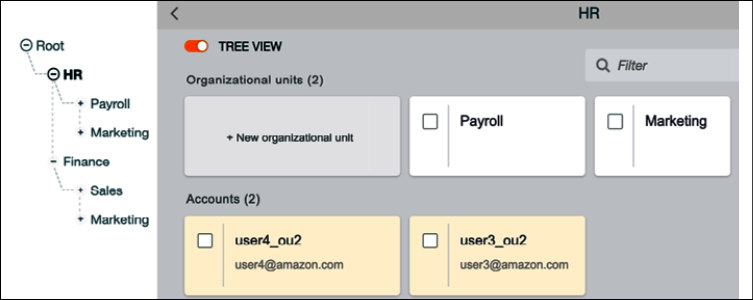


Figure 11.4: Enterprise account structure for organization units (OUs)

In the preceding account structuring strategy, you can control costs at each business unit and department level. Adopting a charge-back mechanism for each department increases accountability for cost at a more granular level, which helps to optimize cost. Account structuring helps you to apply high security and compliance standards across the organization. As each account is linked to a parent account, you can significantly deal with the mass utilization of resources from vendors by consolidating expenditure across the organization.

**Resource cost tagging**

As shown in the following screenshot, to get complete cost visibility and consolidation across resources, you can tag each resource provisioned at the team level, which provides more granular control:

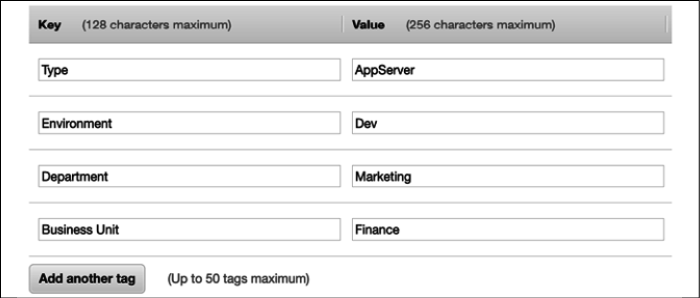


Figure 11.5: Resource tagging for cost visibility

In the preceding diagram, you can see the tagging strategy, which indicates that the given server is for application deployment and is utilized by the development team. This server is owned by the marketing department of the Finance business unit. In this way, the organization can get a granular level of cost expenditure visibility, and the team will be more frugal in their spending. However, you may want to adopt the show-back mechanism at the team level and the charge-back mechanism at the department and business unit level.

You can define your mechanism for tagging, where you can attach a name and value, such as resource name and owner name, to any resource. Almost every public cloud provider provides tagging capabilities out of the box. For on-premises, you can embed server metadata such as DNS name or hostname. Tagging not only helps you to organize costs but also to define a capacity limit, security, and compliance. It can be an excellent tool for inventory management and keeping an eye on the growing need for resources at every level of the organization. The following screenshot shows the cost sorted by the **aws:createdBy** tag, which helps to determine the cost of each resource that is auto-created by AWS:

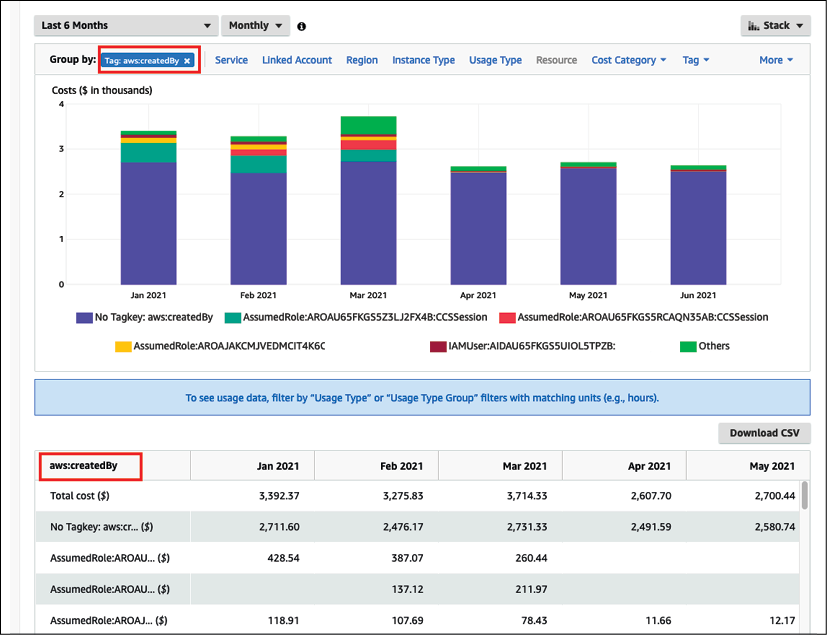


Figure 11.6: Resources amount expenditures dashboard for a cost tag

Business leaders should evaluate the overall requirement to create efficient IT architectures. Collaboration is required to develop a robust IT architecture and define governance across functional teams to set up accountability. Also, set up a standard to review the architecture, create the baseline for any new project initiative, and explain the process that will ensure that the system complies with the correct architecture and identify the route to improvement.

Engage all impacted stakeholders within your business in usage and cost discussions. The CFO and application owners must understand resource consumption and purchasing options. Department owners must understand the overall business model and the monthly billing process. This will help to set the direction for the business units and the whole company.

Make sure third-party vendors are aligned to your financial goals and can adjust their engagement models. Vendors should provide a cost analysis of any application they own and develop. Each team within the organization should be able to translate business, cost, or usage factors from management into system adjustments, which help the application implement and achieve the company's desired goals.

**Monitoring cost usage and report**

Accurate cost factors help you to determine the profitability of business units and products. Cost tracking helps you to allocate resources at the right place to increase ROI. Understanding cost drivers helps you to control your business expenditure.

To optimize costs, you must know your expenditure pattern across the organization. You need to have visibility of IT expenditure over a period of time to determine cost-saving opportunities. You can take the required steps for cost optimization and understand the impact by creating a visualization of cost trends, which shows historical costs and forecasts by resources and departments across the organization. Your team needs to gather data by logging all data points, analyzing them with monitoring, and creating a visualization report.

To identify cost-saving opportunities, you need a detailed insight into your workload resource utilization. Cost optimization depends on your ability to forecast future spending and put methods in place to align cost and usage as per your forecast. The following are the primary areas where you should have data visualizations for cost-saving:

* Determine the most significant investments in resources
* Analyze and understand your expenditure and usage data
* Budget and forecast
* Receive an alert when you exceed your budgeted or forecasted thresholds

The following report is showing resource expenditure over six months in AWS. Looking at the visualization, you can see that the graph database service **Amazon Neptune**, represented by the second bar in each month, is consuming the highest costs with an increase in the trend starting in March 2021. As the business unit can visualize the high cost in May and June, it prompts the system admin to take an in-depth look at cost optimizations and find overprovisioned resources. The admin performs cleanup by stopping additional servers, which will bring down the cost:



Figure 11.7: Resource cost and usage report

The preceding report helps business owners to understand cost patterns and take a reactive approach to cost control. The reactive approach caused hidden costs, which went undetected for a specified period. With the proactive approach, the forecast can help business owners to make a decision ahead of time.

The following report shows daily cost expenditure in the filled bars and forecast spending in the empty bars with estimated ranges. Looking at the report, you can see that it is likely that cost may increase in the next couple of months, and you can take action to understand cost attributes and control costs:



Figure 11.8: Cost trend and cost forecast report

Monitoring your costs against the budget can give you another proactive measure to control costs. Setting up an alert when expenditure reaches a certain proportion of the budget (for example, 50% or 80%) helps you review and adjust your ongoing costs.

In the following report, you can visually determine the current cost against the budgeted cost, which was very high a year ago with an actual cost of $7,397 compared to the budgeted $2,500. Based on the following report, IT admins were able to take action to optimize the cost and bring it down to within the monthly budget.

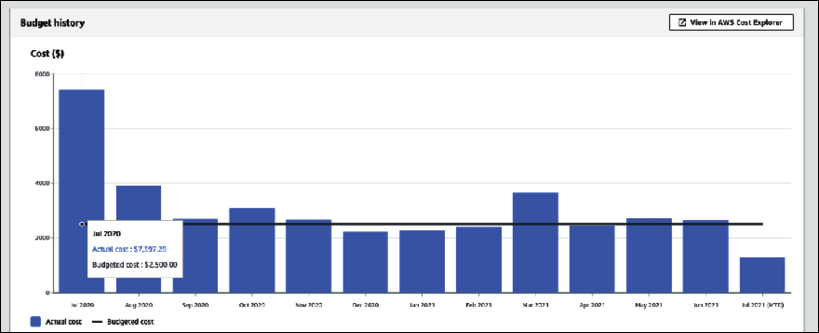


Figure 11.9: Cost and budget report

Cost and budget reports help you to control costs by taking proactive action. Combining your actual running costs with budgets and forecasts provides a great deal of cost control daily.

You can also set up an alert when your actual cost reaches a certain threshold in the budget or forecast. It alerts you proactively via email or mobile message and tells you to take proactive action to control costs.

In the following screenshot, you can see an alert has been set up for when the actual cost reaches 80% of the budget. You can set up multiple alerts to get information when the cost reaches, for example, 50% and 80% of the budget or forecast:

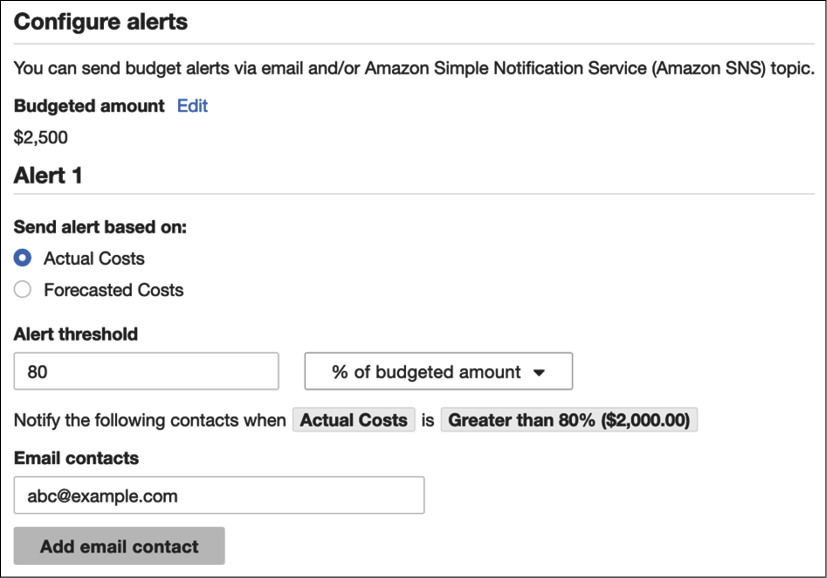


Figure 11.10: Alert based on actual cost

One way to do cost control is by right-sizing your environment with resource monitoring and trigger alarms for over or underutilization. Analysis of resources can be performed using monitoring tools such as **Splunk** or **CloudWatch** and custom logs, where customized metrics such as application memory utilization of your system can be monitored to perform right-sizing. Low utilization of a resource could be a criterion for identifying opportunities for cost optimization. For example, CPU utilization, RAM utilization, network bandwidth, and the number of connections to the application can be analyzed and monitored.

You need to be careful when resizing your environment to ensure you are not impacting the customer experience. The following are best practices to apply when you perform right-sizing:

* Make sure monitoring reflects the end user experience. Select the correct period. For example, performance metrics should cover 99% of the user's request-response time rather than taking an average response time.
* Select the correct monitoring cycle, such as every hour, day, or week. For example, if you are conducting daily analyses, you might miss a weekly or monthly cycle of high utilization and under provision your system.
* Assess the cost of change against the cost-saving. For example, you may have to perform additional testing or engage resources to perform resizing. This cost-benefit analysis will help you to assign resources.

Identify application utilization against your business requirement, for example look at how many user requests are expected to come by the end of the month or during peak season. Identifying and optimizing the utilization gap allows you to save costs. To do this, use the right tool that covers all dimensions from cost-saving to system utilization and the impact on customer experience due to changes, and then utilize reports to understand the business ROI impact due to cost changes. The public cloud follows a different cost model, and it's often an on-demand pay-as-you-go cost structure.

You have to be very diligent when using cloud resources as every second counts toward your cost, and it can be costly if you overlook cost optimization and monitoring. Let's learn more details about cost optimization in the public cloud.

**Cost optimization in the public cloud**

Public clouds, such as **AWS**, **Microsoft Azure**, and **Google Cloud Platform** (**GCP**), provide a great deal of cost optimization with the *pay-as-you-go* model. The public cloud cost model allows customers to trade CapEx for variable expenses, paying for IT resources as they consume them. OpEx is usually lower due to economies of scale. It could be cost-effective to be in the cloud and benefit from continued price reductions that occur over time. The other advantage is that you get additional tools and functionality out of the box with a cloud provider such as AWS, which helps you achieve better agility.

You need a different mindset when defining the cloud cost structure model as it is pretty different from traditional cost models, which most enterprises have been following for decades. You have all the infrastructure available at your fingertips in the cloud, which requires greater control and regulation.

Clouds provide several tools for cost governance and regularization. For example, in AWS, you can set up service limits for each account, so the dev team cannot utilize more than ten servers, and production can have the required number of servers and databases with a buffer.

All resources are associated with accounts in the cloud, so it's easy to keep track of IT resource inventories in a single place and monitor their utilization. In addition to that, you get tools to collect data across various IT resources and provide suggestions. As shown in the following screenshot, **AWS Trusted Advisor** crawls through all resources in the account and offers cost-saving recommendations based on resource utilization:

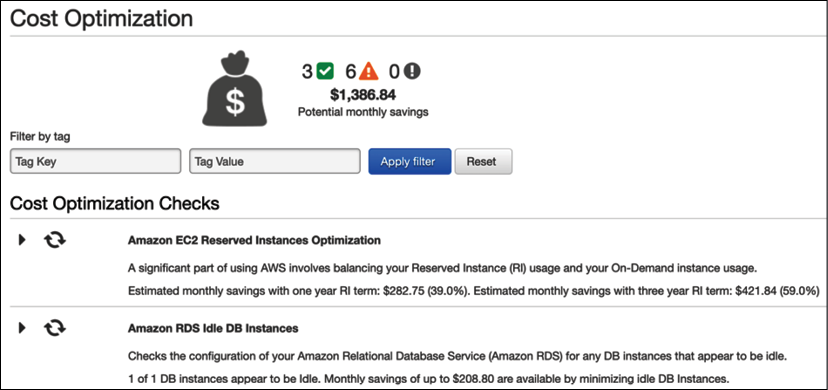


Figure 11.11: Cost-saving suggestions from AWS Trusted Advisor

In the preceding screenshot, AWS Trusted Advisor has detected continuous utilization of the application server (**Elastic Compute Cloud**, **EC2**) and advises buying a reserve instance by paying one year up front with a 40% cost saving. Further checks have identified an underutilized database (Amazon RDS) and suggest shutting it down to make a potential saving.

The cloud can provide an excellent value proposition for cost-saving. To begin with, you can create a hybrid cloud, where you establish connectivity between your on-premises data center and the cloud. You can move development and test servers to the cloud to determine cost structure and potential savings. Once you have set up cost governance in the cloud, move more workload as per the cost-benefit analysis. However, you need to assess your workload and whether it can be moved to the cloud and define a strategy. You learned about cloud migration in *Chapter 5*,*Cloud Migration and Hybrid Cloud Architecture Design*.

Increasingly, public cloud providers are offering managed services, which eliminates any infrastructure maintenance cost and overheads for alert and monitoring configurations. A managed service reduces the total cost of ownership by reducing cost as service adoption increases.

**Summary**

Cost optimization is a continuous effort from application inception (from proof-of-concept to implementation and post-production). You need to review architectures and cost-saving efforts continuously.

In this chapter, you learned about design principles to optimize costs. Before making any purchase decision, you should consider the total cost of ownership for the entire life cycle of software or hardware. Planning a budget and keeping track of forecasts help you to stay on the cost optimization path. Always keep track of your expenditures and look for possible opportunities for cost optimization without affecting user experience or business value.

You learned about the various techniques for cost optimization, which include reducing architectural complexity by simplifying enterprise architecture and setting a standard that everyone can follow. It's recommended to avoid duplication by identifying and consolidating idle and repeated resources to negotiate the bulk purchase cost. Apply standardization across the organization to limit resource provision and develop a standard architecture. Tracking data for your actual costs against budgets and forecasts can help you to take proactive action. You learned about various reports and alerts that can help to control costs. You also learned about cost optimization in the cloud, which can help you further to optimize value.

Automation and agility are major factors that increase resource efficiency, and DevOps can provide a great deal of automation. In the next chapter, you will learn about various DevOps components and DevOps strategies to deploy your workload in the most automated way efficiently.