

# AWS WELL-ARCHITECTED FRAMEWORK: 6 PILLARS AND BEST PRACTICES



In this article, we'll explore the AWS Well-Architected Framework. We'll take a look at the design principles and the six pillars. Then, we'll highlight some things to consider when you're first getting started with the framework.

(This tutorial is part of our [AWS Guide](#). Use the right-hand menu to navigate.)

## What is the AWS Well-Architected Framework?

The AWS Well-Architected Framework is [a set of best practices](#) that guide you in building resilient and agile applications on the cloud. Initially a white paper, the framework soon evolved to a widely accepted cloud deployment methodology thanks to positive reviews and feedback from AWS-trained partners and developers.

At its core, [developers can use](#) the best practices of a Well-Architected framework to:

- Optimize connections and communication between various application components, making the app highly scalable and responsive to customer needs.
- Identify potential technological risks within a cloud ecosystem, helping organizations address faults within an existing framework.

## Well-Architected AWS design principles

These design principles guide where and how to implement the Well-Architected Framework for cloud-based applications:

- Use only as much capacity as your workload requires
- Test workloads and applications at a production-scale before deploying them to production.
- Create an architecture that evolves
- Take advantage of [automation to simplify testing](#)
- Create data-driven architecture
- Schedule live-event simulations to help improve infrastructure

## 6 pillars of well architected framework

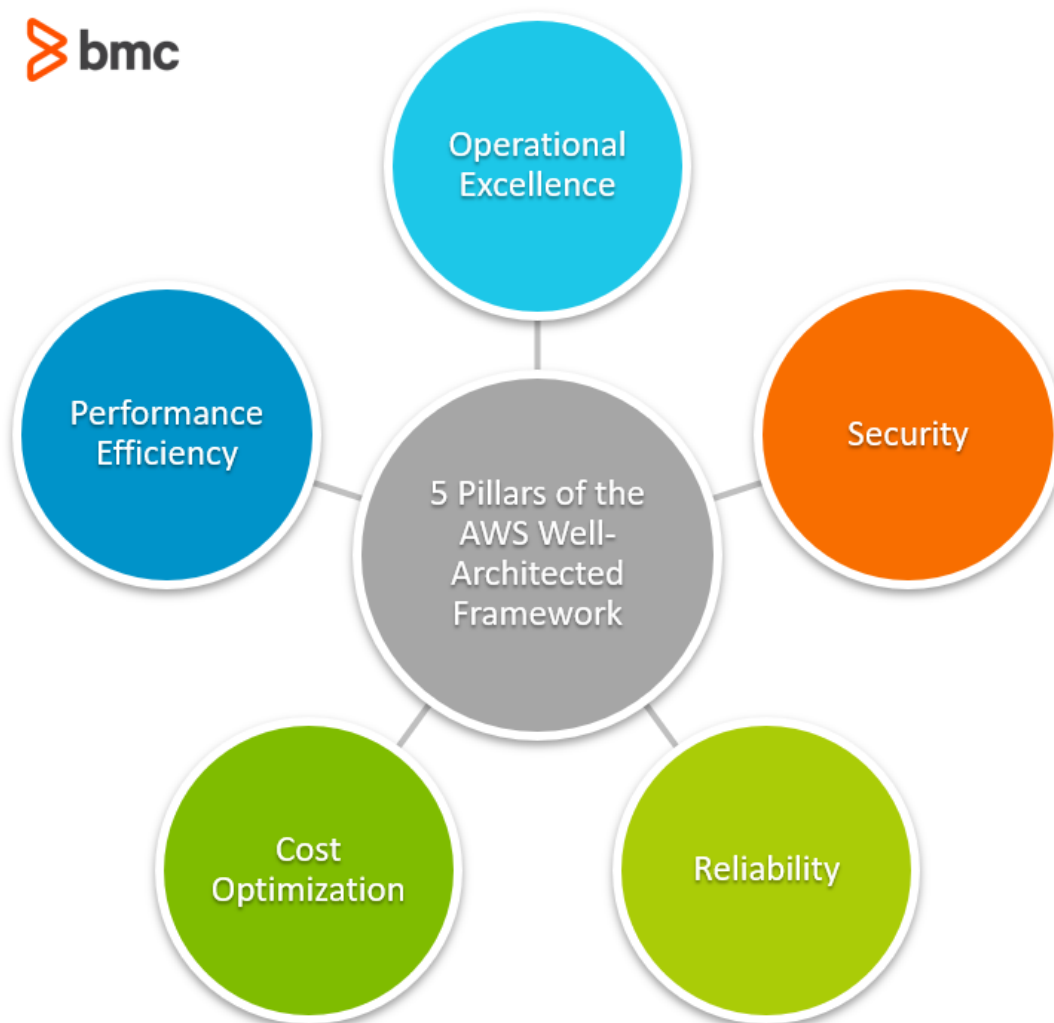
When architecting cloud solutions, consider the six pillars as elements that define the agility and function of applications.

Fundamentally, all Well-Architected best practices are based around these foundational AWS pillars. These guide your application building with the primary goal of creating environments that are:

- Stable
- Efficient
- Highly scalable

These elements also make it possible to test application workloads and code in production-size environments without committing to expensive resources.

Here are the six pillars of an AWS Well-Architected Framework:



# 1. Operational Excellence

This pillar outlines that [developers and operations teams](#) should seek business insights and customer insights to create applications that effectively support production workloads. This means understanding the risks, possible changes, and goals that drive day-to-day business operations.

Each team member should understand the criticality of their role in managing workloads, the entire workload's behavior, and what development decisions will make operations successful.

This AWS pillar also maintains that workloads should be designed to:

- Easily display information on their working statistics
- Enable mechanisms that aid quick feedback, recovery, and [refactoring](#)

# 2. Security

The security pillar emphasizes setting mechanisms that help protect systems, data, and assets. Fundamentally, Well-Architected security revolves around seven design principles:

- Enforce [logging and monitoring](#) for traceability
- Use multiple security controls to ensure security at all application layers
- Automate security best practices
- Protect data both within storage and in-transit
- Adopt an [incident management policy](#)
- Implement a strong foundation of security

Adopting an [Identity and Access Management \(IAM\) policy](#) is critical to ensuring that only the right users can access permitted resources. An IAM policy administers security by implementing an authentication and authorization layer.

Additionally, it is also crucial to implement multi-layered security and control methodologies that enforce infrastructure-wide protection. Enact data protection for application users with a variety of mechanisms such as:

- Versioning
- Logging
- Resilient storage
- Keys
- Regional isolation

Of course, implementing end-to-end security goes much deeper than implementing a user authentication layer. As a result, it is strongly advised to create detection controls and event notifications that help identify and stop security incidents at an early stage.

You can also choose to leverage the benefits of available AWS tools that provide incident response mechanisms, such as:

- AWS Events
- [AWS CloudFormation](#)
- Logging services

### 3. Reliability

A Well-Architected Framework defines mechanisms that help an application handle workloads accurately and consistently. The architecture of the system should be able to prevent and respond to breakdowns automatically.

The design principles for a [reliable system](#) include:

- Automate recovery
- Distribute workloads and requests to eliminate single point of failure
- Use only the capacity you need for your production workload
- Test incidents and recovery procedures
- Manage automation changes

Plan your infrastructure foundations properly in order to handle both current and future workloads. This implies that you should define design and architecture decisions upfront by anticipating workload behavior. While doing so, you'll also want to leverage Service Limits and Resource Quotas to guide the deployment of multi-environment workloads.

Once services are deployed and live, the framework should allow efficient performance and availability monitoring to avoid downtimes or performance failures.

To further ensure your app's reliability, the framework strongly recommends regularly:

- Tracking KPIs
- Backing up data
- Testing your recovery processes

### 4. Cost Optimization

An organization should aim to deliver optimum and resilient business solutions at the least cost to the user. Some design principles that guide the Cost Optimization pillar include:

- Integrate Cloud Financial Management into the organization's overall business objectives
- Pay only for resources your application requires
- Always monitor your application's efficiency (output vs cost)
- Outsource managed services and operating systems to AWS

Cloud Financial Management tools like AWS QuickSight, Cost & Usage Report (CUR), and Cost Explorer offer significant advantages to monitor an organization's cloud costs. These tools also build discipline while creating an organization-wide awareness on cloud expenditure and usage.

A key takeaway of this best practice also advises using only resources that are appropriate for cost-effective workload handling. An organization should consistently evolve its cost optimization policy to take advantage of new services and features within the cloud landscape to reduce costs.

### 5. Performance Efficiency

This pillar guides the effective use of computing resources to meet system and business requirements. Some of the design principles guiding performance efficiency are:

- Delegate complex tasks to third-party vendors who have the required skills to implement

advanced technology easily

- Deploy multi-regional workloads to reduce latency and lower deployment costs
- Test and experiment often to take advantage of virtual resources
- Utilize serverless functions whenever possible

When selecting infrastructure and architecture for your application, take a data-driven approach while exploring multiple options available before settling for the most efficient. This exercise is not a one-time task. Combine multiple approaches and keep reviewing the latest technologies to find an optimum solution.

Lastly, implement policies and procedures to monitor workloads' performance that flags issues before they impact usability.

## 6. Sustainability

This pillar promotes the need to reduce the environmental impact of cloud computing. It begins with an understanding of cloud workload and its impact on the environment. The idea is to optimize resource use in computing, data storage, and networking to avoid waste. It requires designing workloads to minimize carbon footprint. Lastly, it involves being mindful of downstream impacts, like hardware turnover and e-waste. Sustainability is the shared responsibility of data center managers and cloud users.

## Getting started with the AWS Well-Architected Framework

Thinking about implementing the Well-Architected Framework in your [DevOps environment](#)? These are good things to consider.



## The review process

Review your existing architecture consistently to identify bottlenecks and room for improvement. The review process is continuous, just as your organization will always review business and operational goals.

To suit this, the Well-Architected framework aligns seamlessly with built-in AWS system review processes. Besides identifying the most effective infrastructure and resources, frequent reviews will also help identify:

- Team member strengths
- The assignment of responsibilities

As the workload evolves, so should the organization's architecture and technology implementations.

Maintain a consistent review cycle to help your DevOps team achieve deeper insights into the application environment.

## The AWS Well-Architected Tool

The [AWS Well-Architected tool](#) monitors an application's workloads' running state and compares them with the most recent architectural best practices. This tool is based on a Well-Architected Framework that gives access to knowledge in the best way to deploy and maintain applications.

Additional checks on a workload's status to help identify a system's potential risks and laggards is also routinely performed. Best practice guidance is another crucial aspect of the AWS WA tool that comes in handy while an organization looks to build highly efficient, reliable, resilient, and cost-effective applications.

## The AWS Well-Architected SaaS Lens

The AWS WA tool also integrates seamlessly with [AWS SaaS Lens](#), which helps software providers avail hosted and centrally managed applications.

SaaS Lens asks a few questions about your application's intended performance, then helps you create an improvement plan for the application. The questions are, of course, based on the six Amazon pillars of the Well-Architected Framework as well as various suggestions on how to continuously improve the app's performance.

## The AWS Well-Architected Partners Program

[Managed IT service organizations](#) are increasingly adopting the [Well-Architected Partners Program](#). This program enables professionals and organizations to share hands-on experience and in-depth knowledge of building high performing, resilient applications. Both individuals and organizations can [earn AWS certifications](#) to:

- Help customers manage workloads
- Enable application improvements based on best practices of the Well-Architected Framework.

The Partner's Programs also allows one to enlist as a qualified partner who can contribute his experience in developing and deploying highly efficient and optimized cloud projects.

# The future of app development

The Well-Architected framework continues to evolve steadily as a set of best practices to build effective cloud solutions. As one of its core objectives, this framework leverages automation and optimization to help organizations develop highly efficient, reliable, secure, and low-cost applications.

Clients and partners now seek out organizations that fully embrace a Well-Architected framework—the benefits get better over time.

## Related reading

- [BMC Multi-Cloud Blog](#)
- [BMC DevOps Blog](#)
- [Cloud Infrastructure: A Brief Introduction](#)
- [Cloud Monitoring: Choosing the Right Metrics](#)
- [State of the Cloud Today](#)
- [API/Developer Portals: How To Create Great API Portals](#)