**Well Architected Framework**

Organized into 5 pillars

* Security
* Operational Excellence
* Reliability
* Performance Optimization
* Security

**Security**

* Identity foundation
* Security at all layers
* Traceability
* Risk assessment and mitigation strategies

**Operational Excellence**

Addresses the ability to run systems and gain operations to deliver business value.

Includes

* Ability to run and monitor systems
* Continuously monitor supportive process and procedures.
* As the application moves between Deployed->Updated->Operated continuously monitor the system
* Apply engineering process to fix and deploy defects quickly.
* Analyze how application is performing and log application details.

**Reliability**

* Recover quickly from infrastructure or service disruption.
* Dynamically acquire computing resources to meet demand
* Mitigation disruptions such as
  + Misconfigurations
  + Temporary network issues

**Performance**

* Maximizing performance by using computing resources effectively
* Mechanical sympathy is how you can utilize a tool or system with an understanding of how its operates best. e.g. When you understand how a system is designed to be used, you can align with the design to gain optimal [performance](https://wa.aws.amazon.com/wellarchitected/2020-07-02T19-33-23/wat.pillar.performance.en.html).
* Democratize advanced technologies -  Make advanced technology implementation easier for your team by delegating complex tasks to your cloud vendor. Rather than asking your IT team to learn about hosting and running a new technology, consider consuming the technology as a service.

**Cost Optimization**

* Measure efficiency
* Eliminate redundant cost
* Consider using managed service

Best practice for building solution on AWS

Design tradeoffs can increase architecture complexity

**Design tradeoffs**

* Ensure that architecture can handle changes in demand.
* Design the system to scale-in for lowest cost
* Wherever possible automate the provisioning, termination and configuration of resources.
* Treat resources as disposable. Take advantage of dynamically provisioned nature of cloud computing.
* Use loosely couple systems by designing architecture with independent components.
* Design service not servers. Use breadth of AWS services. Do not limit your infrastructure to servers.
* Choose the right database solution based on your requirement. Match the technology to meet your business and not the other way around.
* Avoid single point of failure by assuming everything fails and then design backwards. It can even mean that you need to duplicate everything.
* Optimize for cost – Take advantage of the flexibility of AWS to increase your cost efficiency.
* Use Caching
* Caching minimizes redundant data retrieval operations and improves performance and cost.
* Secure your entire infrastructure
* Build security in every layer of your architecture

AWS Global Infrastructure

* 22 physical regions (geographic locations)
* Each region contains two or more availability zones.
* Each availability zone contains one or more data center.
* You enable and control data replication across regions.
* Regions introduced prior to March 2019 are enabled by default. Asia pacific, Bahrin and Hongkong are disabled by default. You need to enable them prior to using.
* To achieve fault tolerance and stability regions are isolated from each other.
* Resources in one region are not automatically replicated to another region.
* It is your responsibility to replicate data across regions.
* Services are available per region.
* No data centers can be part of two availability zones.
* Replicate across availability zones for resiliency
* Systems can span multiple availability zones.

AWS Local Zone

* Extension of AWS region
* Enable to run latency-sensitive portions of the application (e.g. gaming, machine learning) closer to end users and resources in specific geographies.
* You can use AWS services in geographic proximity to end users.
* Allow placing compute, storage, database and other select services closer to large population where no region exists today.
* Are managed and supported by AWS
* Example Los Angeles AWS Local Zone