Ten design principles for Azure applications

08/30/2018 • 2 minutes to read • Contributors

Follow these design principles to make your application more scalable, resilient, and manageable.

Design for self healing. In a distributed system, failures happen. Design your application to be self healing when failures occur.

Make all things redundant. Build redundancy into your application, to avoid having single points of failure.

Minimize coordination. Minimize coordination between application services to achieve scalability.

Design to scale out. Design your application so that it can scale horizontally, adding or removing new instances as demand requires.

Partition around limits. Use partitioning to work around database, network, and compute limits.

Design for operations. Design your application so that the operations team has the tools they need.

Use managed services. When possible, use platform as a service (PaaS) rather than infrastructure as a service (IaaS).

Use the best data store for the job. Pick the storage technology that is the best fit for your data and how it will be used.

Design for evolution. All successful applications change over time. An evolutionary design is key for continuous innovation.

Build for the needs of business. Every design decision must be justified by a business requirement.