Aspect	Master-Slave Model	Peer-to-Peer (P2P) Model
Architecture & Structure	Single **master** node for write coordination; **slave** nodes handle replication and reads	All nodes are equal peers; any node can handle both reads and writes independently
Data Replication & Consistency	Centralized updates by the master, providing **stronger consistency** but with potential replication lag	Distributed replication; usually **eventual consistency** via quorum protocols
Scalability	Limited by master node's capacity; suitable for **read-heavy** applications	Highly scalable; load is evenly distributed, ideal for **large-scale** deployments
Fault Tolerance & Availability	Dependent on master node; master failure disrupts writes until failover, but reads remain available	High fault tolerance; any node failure is handled by other nodes, **high availability**
Performance	Fast reads via slaves, but **slower writes** due to centralization and replication delays	Both reads and writes are generally **faster** as any node can process them
Use Cases	**Read-intensive** applications, or when strict consistency is needed	**Write-intensive** applications, large distributed systems needing high availability