

Cloud Architectures



Idaho State
University

Computer
Science

Isaac Griffith

CS 3321
Department of Computer Science
Idaho State University

ROAR



Topics Covered

- Cloud Architectures

Architectural decisions for cloud software engineering

Questions to ask when choosing a database organization

- Target customers
 - Do customers require different database schemas and database personalization? Do customers have security concerns about database sharing? If so, use a multi-instance database.
- Transaction requirements
 - Is it critical that your products support ACID transactions where the data is guaranteed to be consistent at all times? If so, use a multi-tenant database or a VM-based multi-instance database.
- Database size and connectivity
 - How large is the typical database used by customers? How many relationships are there between database items? A multi-tenant model is usually best for very large databases as you can focus effort on optimizing performance.
- Database interoperability
 - Will customers wish to transfer information from existing databases? What are the differences in schemas between these and a possible multitenant database?

Scalability and resilience

- The scalability of a system reflects its ability to adapt automatically to changes in the load on that system.
- The resilience of a system reflects its ability to continue to deliver critical services in the event of system failure or malicious system use.
- You achieve scalability in a system by making it possible to add new virtual servers (scaling-out) or increase the power of a system server (scaling-up) in response to increasing load.
 - In cloud-based systems, scaling-out rather than scaling-up is the normal approach used. Your software has to be organized so that individual software components can be replicated and run in parallel.
- To achieve resilience, you need to be able to restart your software quickly after a hardware or software failure.



Using a standby system to provide resilience



Resilience

- Resilience relies on redundancy:
 - Replicas of the software and data are maintained in different locations.
 - Database updates are mirrored so that the standby database is a working copy of the operational database.
 - A system monitor continually checks the system status. It can switch to the standby system automatically if the operational system fails.
- You should use redundant virtual servers that are not hosted on the same physical computer and locate servers in different locations.
 - Ideally, these servers should be located in different data centers.
 - If a physical server fails or if there is a wider data center failure, then operation can be switched automatically to the software copies elsewhere.



System structure

- An object-oriented approach to software engineering has been that been extensively used for the development of client-server systems built around a shared database.
- The system itself is, logically, a monolithic system with distribution across multiple servers running large software components. The traditional multi-tier client server architecture is based on this distributed system model.
- The alternative to a monolithic approach to software organization is a service-oriented approach where the system is decomposed into fine-grain, stateless services.
 - Because it is stateless, each service is independent and can be replicated, distributed and migrated from one server to another.
 - The service-oriented approach is particularly suitable for cloud-based software, with services deployed in containers.



Cloud platform

- Cloud platforms include general-purpose clouds such as Amazon Web Services or lesser known platforms oriented around a specific application, such as the SAP Cloud Platform. There are also smaller national providers that provide more limited services but who may be more willing to adapt their services to the needs of different customers.
- There is no 'best' platform and you should choose a cloud provider based on your background and experience, the type of product that you are developing and the expectations of your customers.
- You need to consider both technical issues and business issues when choosing a cloud platform for your product.



Utah State
University
Computer
Science

Technical issues in cloud platform choice

ROAR



Indiana State
University
Computer
Science

Business issues in cloud platform choice

ROAR



Key points

- The key architectural issues for cloud-based software are the cloud platform to be used, whether to use a multitenant or multi-instance database, the scalability and resilience requirements, and whether to use objects or services as the basic components in the system.



Are there any questions?