EBU7501: Cloud Computing
Week 2, Day 5: Enterprise Design Patterns and Monitoring Systems

Dr. Gokop Goteng

Queen Mary
University of London

Lecture Aim and Outcome

- Aim
  - The aim of this lecture is introduce students to enterprise design patterns and distributed monitoring systems
- Outcome
  - At the end of this lecture students should be able to:
    - Know the different enterprise design patterns
    - Identify which design pattern to use based on different scenarios
    - Know the different monitoring systems available

Enterprise Design

- Enterprise design is the process of aligning the information technology (IT) requirements of an organisation with its business and strategic needs.
- The IT requirements usually aim at satisfying the following
  - Scalability
  - Performance
  - Modularity
  - Flexibility
  - Maintainability
  - Portability
  - Security

Enterprise Design

- The business needs usually consist of the following
  - Cost savings
  - Profitability
  - Staff performance
  - Competitive advantage over competitors
  - Better service delivery
  - Innovation and creativity
  - New products
- These are usually called DESIGN GOALS or PRINCIPLES

Enterprise Design Patterns

- Enterprise design patterns are the different methods used to design an enterprise application
- These include
  - Model View Controller (MVC)
  - Business delegate
  - Service locator
  - Transfer object
  - Intercepting filter
  - Front controller

MVC - Model View Controller

- There is a clear separation and modularity in the codes that implement the model, controller and view
  - loosely coupled
- Components can be reused
- The views can change independently from the controller and model
- It improves maintainability as changes on any of the components do not require changes in the entire application
- There is some level of increased complexity in the implementation of MVC

Business Delegate

- This system provides proxy services by implementing remote interfaces
- Initiation of communication with remote services
- It handles communication details and errors/exceptions
- Receives requests from controller components and translates/forwards the requests to the business service
- Translate the response and sends it back to the controller
- It hides complexity of communications
- Minimise the work of the web tier
- Adds a layer to the application which may increase complexity

Front Controller

- This provides a centralised enterprise system which handles all tasks from a single point
- When there is problem, it will affect the entire system
- It is a good method for secure systems such as ATMs, Credit/Debit cards
- It can be implemented with other patterns

Hybrid Design Pattern

- When you combine two or more of the enterprise design patterns, you are implementing a hybrid enterprise design pattern
- Most real world applications are hybrid in nature

Monitoring Systems

- In distributed systems, the hardware, software, data, databases, networks and users need to be monitored so that failures and security issues can be solved within reasonable time
- Cloud systems need monitoring system to help track faults and use log information for security
- It helps in determining when to scale up a cloud system
- The information obtained is used for decision making
- It helps to troubleshoot and administer systems remotely in cloud systems

Examples of Monitoring Systems

- Ganglia
  - It is used in monitoring high performance computing (HPC) systems and cloud systems
  - It provides information about the usage (memory, CPU, storage), systems failures, users
  - This is an open source system
- Nagios
  - It is also an open source system that is used for monitoring HPC and cloud systems
- Rackspace Cloud Monitoring
  - Used for monitoring globally distributed cloud infrastructure
- Amazon CloudWatch
  - This is a commercial monitoring system for Amazon AWS global infrastructures

Amazon AWS CloudWatch and CloudTrail

- AWS CloudWatch
  - Amazon CloudWatch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers, and IT managers.
  - CloudWatch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health.
- AWS CloudTrail
  - AWS CloudTrail is a service that enables governance, compliance, operational auditing, and risk auditing of your AWS account.
  - It enables you to monitor the API calls made by uses e.g deletion of buckets in S3.

Class Task

- What design pattern allows you to make changes to different applications for different departments gradually without affecting the entire application or business?
- Which design pattern is most secure and why?
- Why do you need monitoring system