Object Oriented Analysis and Design with Java

UE19CS353

Prof. Sudeepa Roy Dey and Prof. Vinay Joshi
Department of Computer Science and Engineering

Acknowledgements: Significant portions of the information in the slide sets presented through the course in the class, are extracted from the prescribed text books, information from the Internet. Since these are only intended for presentation for teaching within PESU, there was no explicit permission solicited. We would like to sincerely thank and acknowledge that the credit/rights remain with the original authors/creators only

UE19CS353: Object Oriented Analysis and Design with Java

OO Development process
System Design and Frameworks

OO Development process

What is OO development?

- The essence of OO development is the identification and organization of application concepts rather than their final representation in programming language.
- It encourages software developers to work and think in terms of application throughout the software life cycle.
- It is a conceptual process independent of the programming language until final stages.
- It is a way of thinking and not a programming technique.
- Its greatest benefit comes from helping developers, customers express abstract concepts clearly and communicate them to each other.
- The OO concepts are used to express a design and also provide documentation.

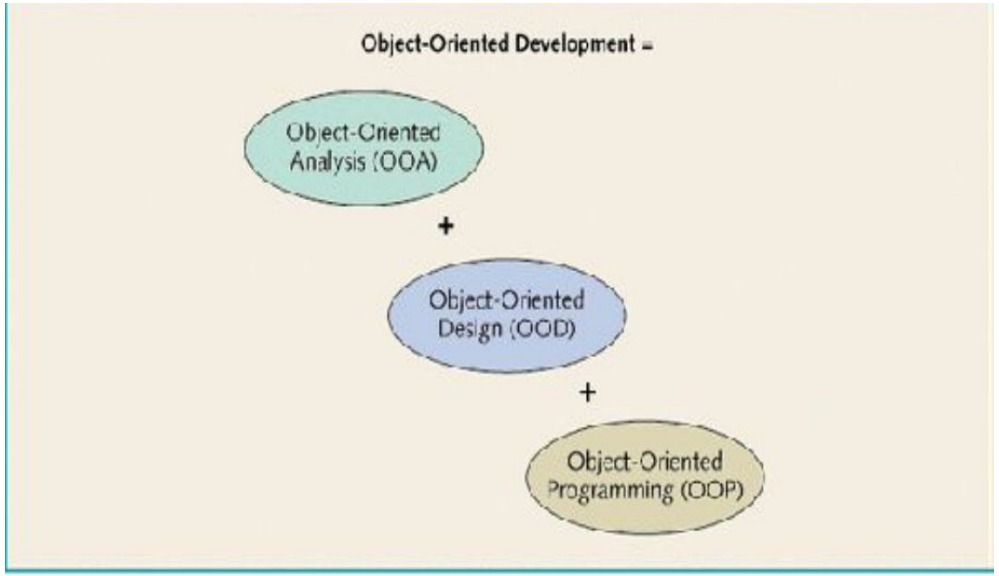
Object Oriented Analysis and Design with Java

OO Methodology

The process for OO development and graphical notation for representing OO concepts consists of building a model of an application and then adding details to it during design.

The methodology has the following stages:

- System conception : Software development begins with business analysis or users conceiving an application and formulating tentative Requirements
- Analysis : The analyst must work with the requestor to understand the problem, because problem statements are rarely complete or correct. The analysis model is a precise abstraction of what the desired system must do, not how it will be done. It should not contain implementation decisions.

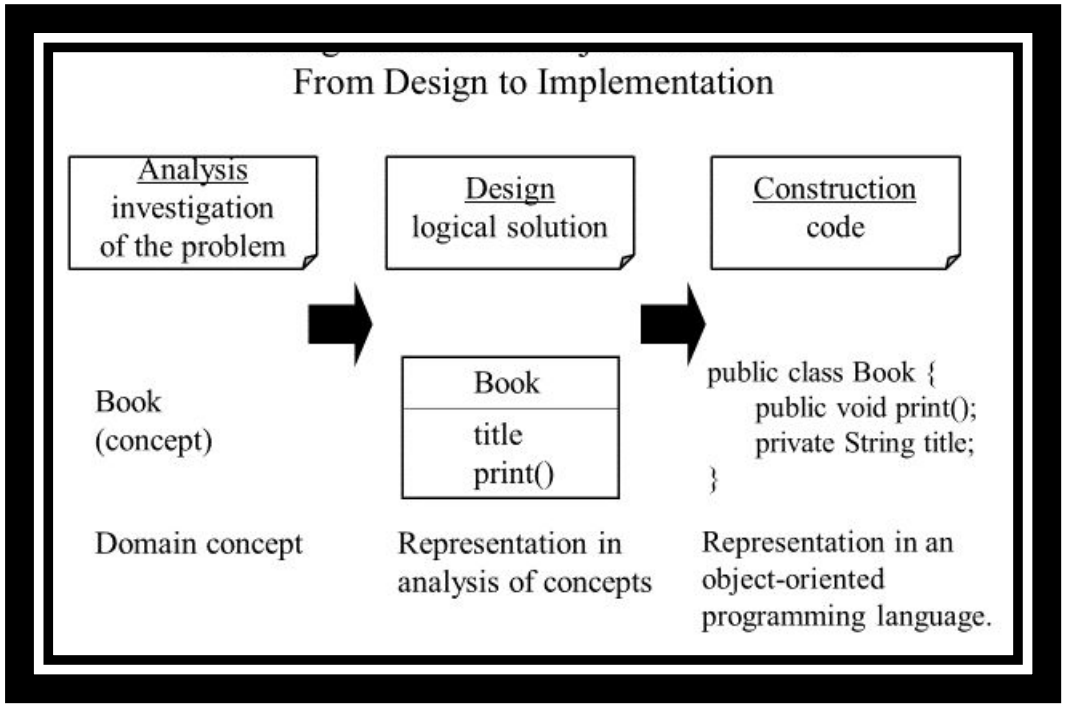


OO Methodology

The System analysis model has 2 parts:

Domain model - a description of the real-world objects reflected within the system Eg: Domain objects for a stockbroker

Application model - a description of the parts of the application system itself that are visible to the user. Eg:- Application might include stock, bond, trade and commission. Application objects might control the execution of trades and present the results.



Object Oriented Analysis and Design with Java

OO Methodology

- **System design:** The development teams devise a high – level strategy called the system architecture for solving the application problem.

- **Class design :** The class designer adds details to the analysis model in accordance with the system design strategy. The focus of class design is the data structures and algorithms needed to implement each class.

- **Implementation :** Implementers translate the classes and relationships developed during class design into particular programming language, database or hardware.

During implementation, it is important to follow good software engineering practice so that traceability to the design is apparent and so that the system remains flexible and extensible

In the next slide we focus on system design in details

Object Oriented Analysis and Design with Java

System Design

- During System design, decide how the problem will be solved.
- Need to apply high level strategy – *System Architecture* – for solving the problem and building a solutions.
- During System design, developer decide the overall structure and style.
- System architecture provides the organization of the system into subsystem.

To construct system architecture, following decision must be made: we take first two

1. Estimating performance
2. Making a Reuse plan
3. Breaking system into sub-system
4. Identifying Concurrency
5. Allocation of Sub Systems
6. Management of Data Storage
7. Handling Global Resources
8. Choosing a Software Control strategy
9. Handling Boundary Conditions
10. Common Architectural Style

i)Estimating Performance

- You should prepare rough performance estimate.
- Purpose of this task to determine if the system is feasible.
- You will have to make simplifying assumptions.(i.e. assume factors)
- Don’t worry about detail – just approximate, estimate and guess

  - For ATM ex Consider following case
    - Bank has 40 branches and no. of terminals.
    - On a busy day half the terminal are busy at once.
    - Suppose each customer takes one min to perform a session and most transaction involve a single transaction. i.e. 40 transaction a minute.
    - You can perform similar estimate for data storage.
    - Count the no. of customer, estimate amount of data for each one and multiply. i.e. 40 *20 =800.

System Design-Reuse Plan

- Two different aspect of reuse are possible:-
  - Using existing thing
  - Creating reusable new things.

- Much easier to reuse existing things than to design new things.
- Most developer reuse existing thing and only a small fraction of developers create new things.
- Creating reusable new things is not easy task. It require good experience.

Reusable thing includes mainly:-

1. Libraries
2. Frameworks
3. Patterns

System Design-Reuse Plan

Libraries: it is a collection of classes that are useful in many contexts. Classes must be organized, so users can find them. Classes must have accurate and thorough description to help users determine their relevance.

Several qualities of good class libraries:

- Coherence: organized well focus
- Completeness: complete behavior
- Consistency: consistent names and signature.
- Efficiency: provide alternative to implement
- Extensibility: able to define subclasses
- Genericity: should be parameterized class where appropriate.

Libraries performs a set of specific and well-defined operations. Examples : Network protocols, compression, image manipulation, string utilities, regular expression evaluation, math etc

Object Oriented Analysis and Design with Java

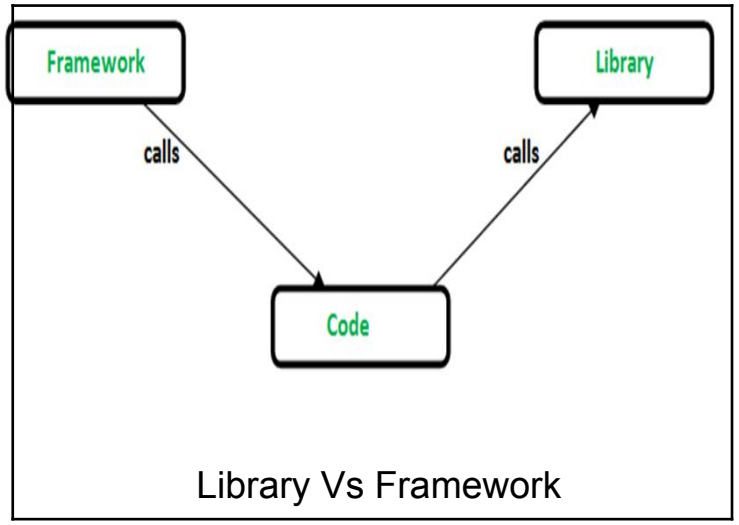
System Design-Reuse Plan

Framework: it is skeletal structure of a program that must be elaborated to build complete application.

- Framework consists of more than just the classes. It is more specific to individual application.
- Framework class libraries are typically application specific and not suitable for general use.

Patterns: A pattern is proven solution to a general problem.

- There are patterns for analysis, architecture, design and implementation.
- A pattern comes with guideline on when to use it, as well as exchange on its use.
- Software Architecture Patterns :Layered Pattern, Client-Server Pattern
- Design pattern: Gangs of Four(GOF) The GoF Design Patterns are broken into three categories: Creational Patterns for the creation of objects; Structural Patterns to provide relationship between objects; and finally, Behavioral Patterns to help define how objects interact.



Library Vs Framework

Frameworks: Introduction

- Definition: “A Framework is a set of cooperating classes that makes up a reusable design for a specific class of software.”
- A framework provides architectural guidance by partitioning the design into abstract classes and defining their responsibilities and collaborations.
- A developer will normally customize a framework to a specific application by “subclassing” and composing instances of framework classes.
- Frameworks are larger architectural elements than design patterns.
- A typical framework contains several design patterns, but the reverse is never true
- Framework are the bodies that contains the pre-written codes (classes and functions) in which we can add our code to overcome the problem. We can also say that frameworks use programmer's code because the framework is in control of the programmer. We can use the framework by calling its methods, inheritance, and supplying "callbacks", listeners, or other implementations of the Observer pattern.

Frameworks in Java

- Java Framework is the body or platform of pre-written codes used by Java developers to develop Java applications or web applications. In other words, Java Framework is a collection of predefined classes and functions that is used to process input, manage hardware devices interacts with system software. It acts like a skeleton that helps the developer to develop an application by writing their own code.

Some of the most popular Java frameworks are:

- Spring
- Hibernate
- Grails
- Play
- JavaServer Faces (JSF)
- Google Web Toolkit (GWT)
- Quarkus

Frameworks in Java

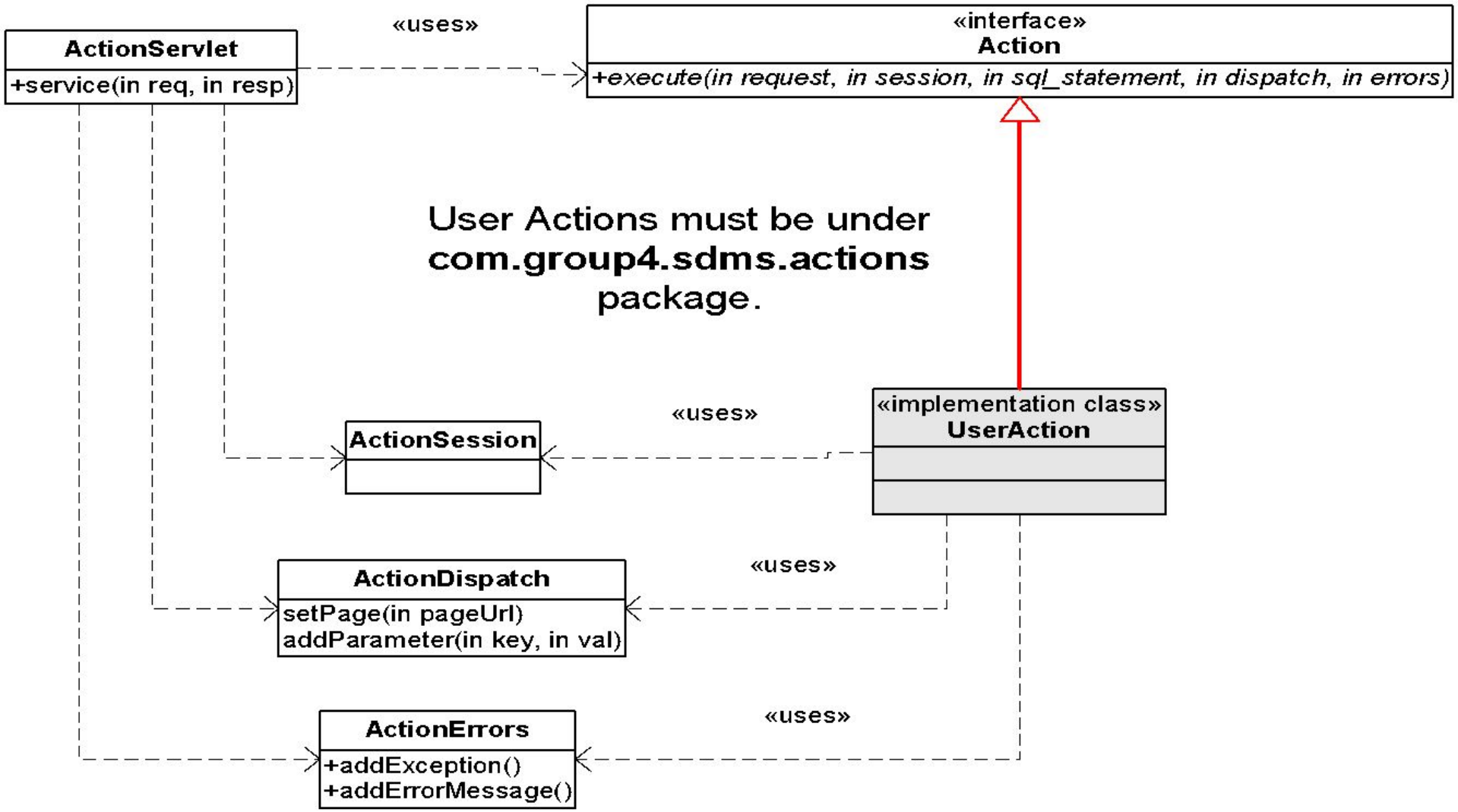
Some more Examples of Frameworks in Java

- In Java, Collection is an example of the framework. It reduces the programming efforts because it provides useful data structure and algorithms. It is referred to as library that do not provides inversion of control.
- Java Collection means a single unit of objects. Java Collection framework provides many interfaces (Set, List, Queue, Deque) and classes (ArrayList, Vector, LinkedList, PriorityQueue, HashSet, LinkedHashSet, TreeSet).

- Another example of GUI based framework is, Swing and AWT classes. Awt is an abstract window toolkit that provides various component classes like Label, Button, TextField, etc., to show window components on the screen. All these classes are part of the Java.awt package.
- On the other hand, Swing is the part of JFC (Java Foundation Classes) built on the top of AWT and written entirely in Java. The javax.swing API provides all the component classes like JButton, JTextField, JCheckbox, JMenu, etc.

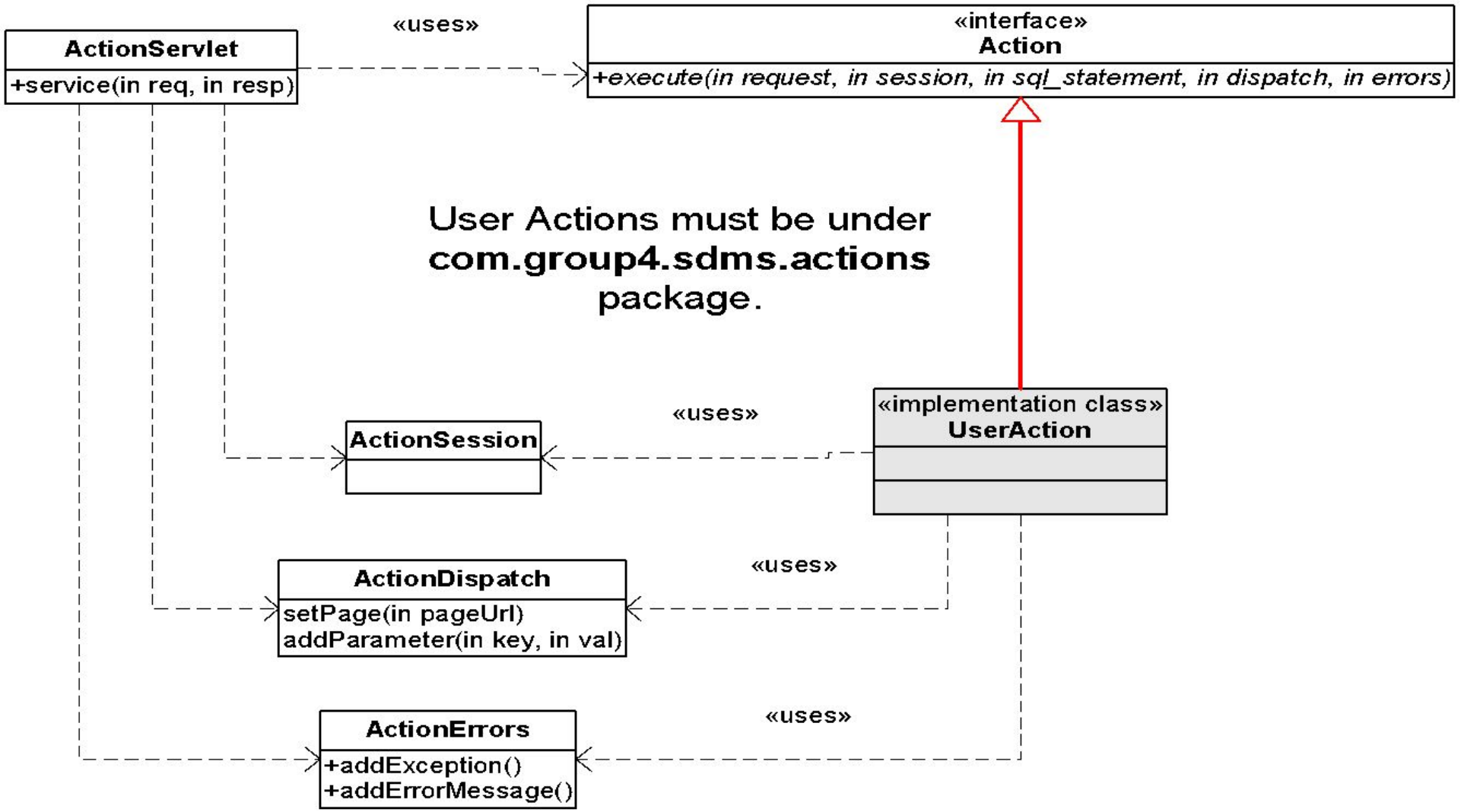
Frameworks in Java: an example

A framework relies on the 'Hollywood Principle' – 'don't call us, we'll call you."
This means that the user-defined classes (for example new subclasses) will receive messages from the predefined framework classes.
These are usually handled by implementing super class abstract methods. (like via generalization)



Frameworks in Java: an example

The framework provides all the classes you see except the <<implementation>> class that fills the 'User Action" role.
This is the whole idea about Frameworks.
The framework "calls" the UserAction code, which is something that application developers provide.



References

1.Object - Oriented Modeling and Design With UML by RUMBAUGH and BLAHA ,Chapter 1 and 14
2.Applying UML and patterns by Craig Larman,chapter-34
3.https://www.javatpoint.com/what-is-framework-in-java

THANK YOU

Prof. Sudeepa and Prof Vinay Joshi

Department of Computer Science and Engineering