5 Software Reuse -Anuj joshi

- Systems are designed by composing existing components.
- To achieve better s/w, more quickly at lower cost

Reuse- based Software Engineering

1. Application system reuse.
    - The whole of an application system may be reused either by incorporating it without change into other system(COTS reuse) or by developing application families.
2. Component reuse
    - Components of an application from sub- systems to single objects may be reused.
3. Object and function reuse
    - Software Components that implement a single well-defined object or function may be reused.

Advantages

- Increased reliability- components exercised in working system
- Reduced process risk- Reuse components instead of people.
- Standard compliance- Embed standards in reusable components.
- Accelerated development- Embed standards in reusable components.
- Effective use of specialists- Reuse components instead of people

5.1 Reuse landscape

- Although reuse is often simply thought of as the reuse of the reuse of system components, there are many different approaches to reuse that may be used.
- Reuse is possible at a range of levels from simple functions to complete application systems.
- Covers the range of possible reuse technologies

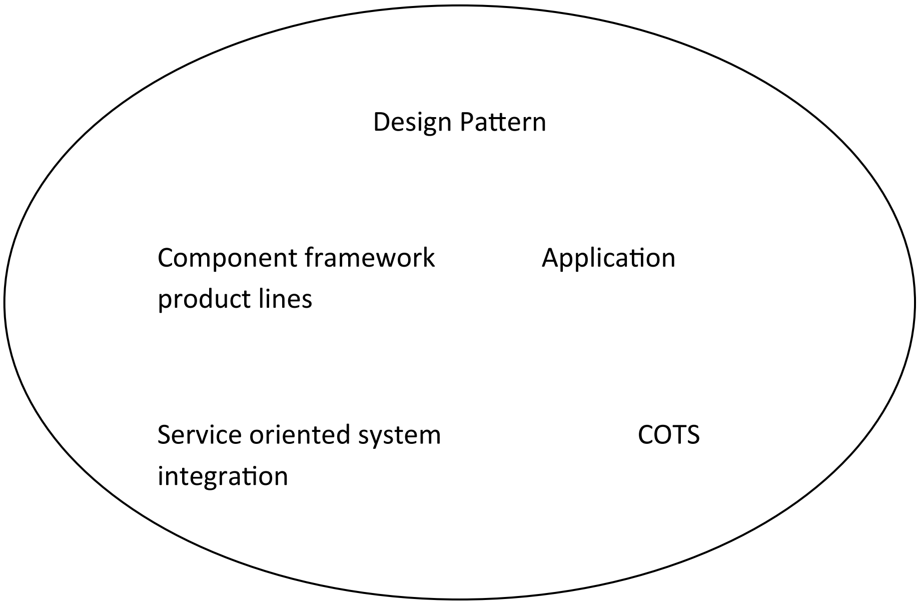

Design Pattern
Component framework product lines
Application
Service oriented system integration
COTS

Fig: **Reuse Language**

Concept Reuse

- When we reuse program, we have to follow the design decision made by the original developer of the program
- This may limit the opportunities for reuse
- However, more abstract form of reuse is concept reuse when a particular approach is described in an implementation independent way and an implementation is then developed.
- Approach to concept reuse is
    1. Design Patterns
    2. Generative Programming

5.2 Design Pattern

- Way of reusing abstract knowledge about a problem and its solution.
- A pattern is a description of the problem and the essence of its solution.
- It should be sufficiently abstract to be reused in different settings.
- Pattern often rely on object characteristics such as inheritance and polymorphism.

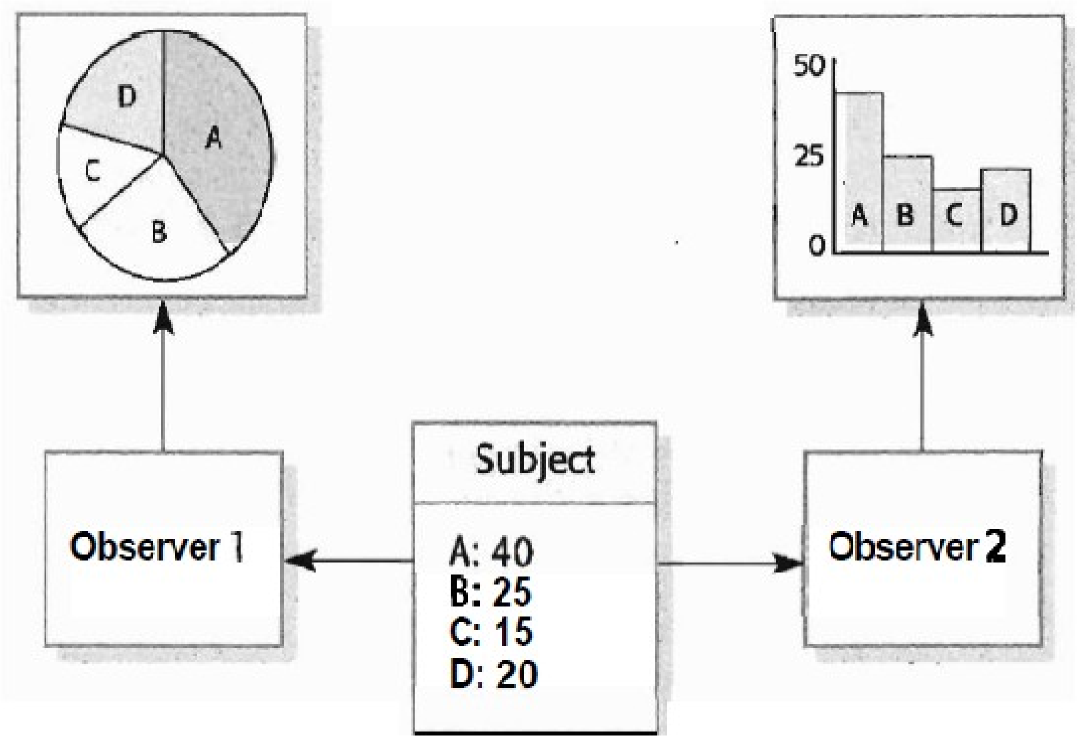
Pattern element

- ➢ Name
    - A meaningful pattern identifier
- ➢ Problem Description
- ➢ Solution

- Not a concrete design but a template for a design solution that can be installed in different ways.

- Consequences

- The results and trade-offs of applying the pattern.



5.3 Generator- based reuse

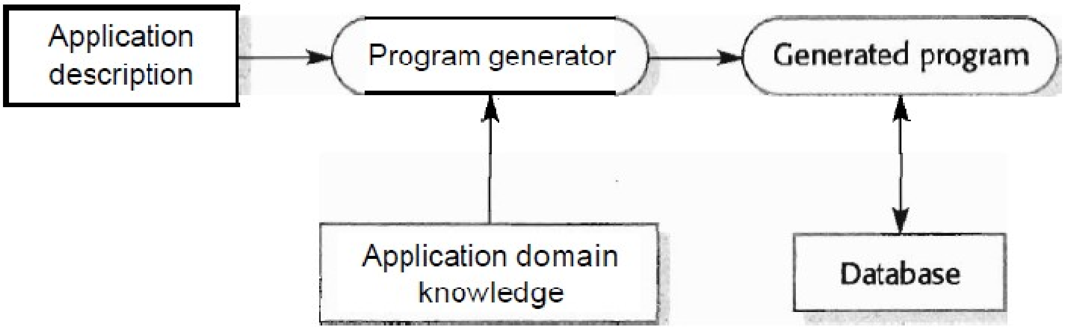
- Program generators involve the reuse of standard program and algorithms.
- These are embedded in the generator and parameterised by user command a program is then automatically generated.
- Generator-based reuse is possible when domain abstractions and their mapping to executable code can be identified.
- A domain specific language is used to compose and control these abstractions.

Types of program generator

- Types of program generator
  1. Application generator for business data processing.
  2. Parser and lexical analyser generator for language processing.
  3. Code generators in CASE tools.
  - Generator- based reuse is very cost-effective but its applicability is limited to a Relatively small number of application domains.

- It is easier for end-users to develop programs using generators compared to other
Component- based approaches to reuse.

Reuse through program generation



5.4 Application Framework

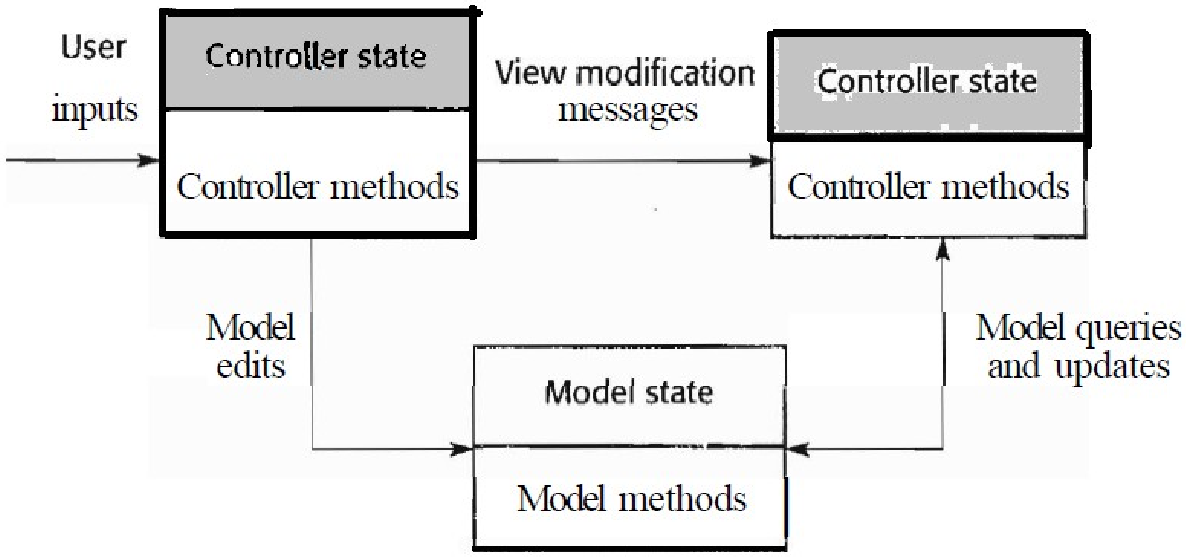
- Generic structure that can be extended to create a more specific sub- system or application
- Frameworks are sub-system design made up of collection of abstract and concrete classes and the interfaces between them.
- The sub-system is implemented by adding components to fill in parts of the design and by instantiating the abstract classes in the framework.
- Frameworks are moderately large entities than can be reuse.

Framework classes

1. System infrastructure framework
    - Support the development of system infrastructures such as communications, user interface and compilers.
2. Middleware integration framework
3. Enterprise application
    - Standards and classes that support component communications and information exchange.

MVC (Model- View Condition)

- One of the best known used frameworks for GUI design.
- MVC framework involves the instantiation of a number of patterns.
- Allows for multiple presentations of an object and separate interactions with these presentations.



5.5 Application system reuse

- Involves the reuse of entire application system either by configuring a system for an environment or by integrating two or more system to create a new application.
- **Two approaches**
  i. COTS product integration
     - COTS(Commercial on the self system) are usually complete application systems that often is an API (Application Program Interface)
     - Benefits in faster application development lower cost.

  ii. Product line development

     e.g.: E- Procurement system

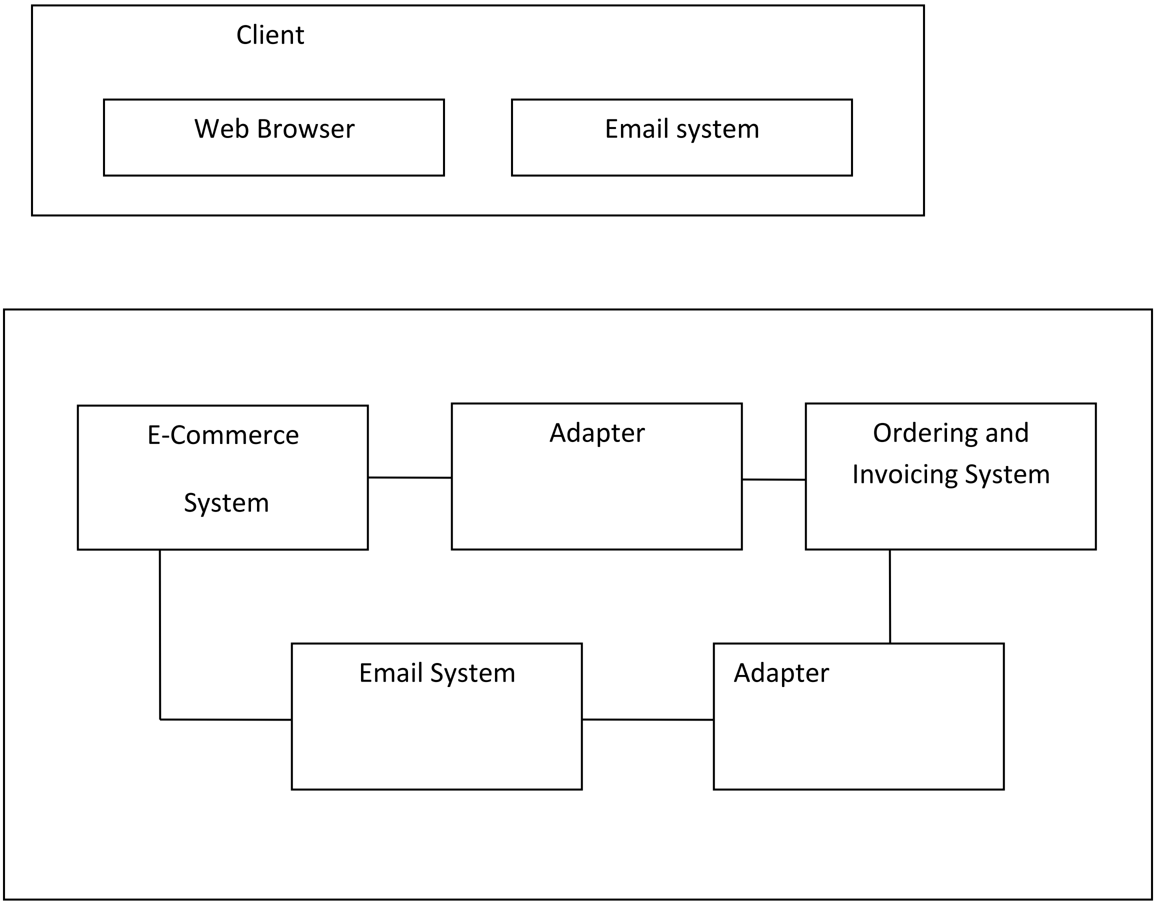


Fig: **E-Procurement System**

COTS product reused

- On client, standard e-mail and web browsing program are used.
- On server, an e-commerce plat form has to be integration with existing ordering system.
    - This involves writing an adaptor so that they can exchange data.

Software Product lines (Application Families)

- Application with generic functionality that can be adapted and configured for use in specific content.
- Adaptive involves
    1. Component and system configuration
    2. Adding new components to the system.
    3. Selecting from a library of existing components
    4. Modifying components to meet new requirements

E.g.; ERP system (Enterprise Resource Planning)

ERP system

- An Enterprise Resource Planning (ERP) system is a generic system that supports common business processes such as ordering and invoicing, manufacturing, etc.
- These are widely used in large companies- they represent probably the most common form of software reuse.
- The generic core is adapted by including modules and by incorporating knowledge of business process and rules.

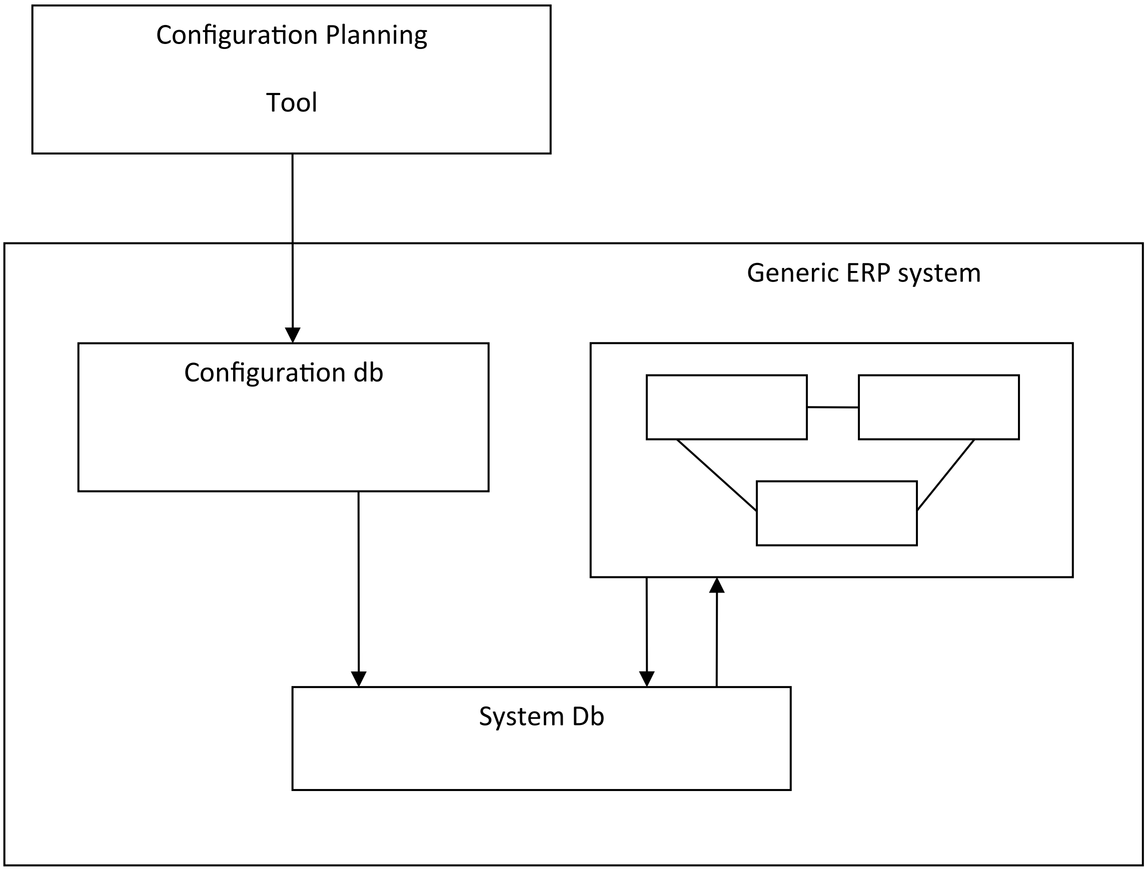


Fig: **ERP System Organisation**

Key points

- Advantages of reuse are lower costs, faster software development and lower risks.
- Design patterns are high-level abstractions that document successful design solutions.
- Program generators are also concerned with software reuse- the reusable concepts are embedded in a generator system.
- Application frameworks are collections of concrete and abstract objects that are designed for reuse through specialisation.
- COTS product reuse include lack of control over functionality, performance and evolution and problems with inter- operation.
- ERP systems are created by configuring a generic system with information about a customer's business.
- Software product lines are related applications developed around a common core of shared functionality.