CHAPTER 3:- COMPONENT-BASED
SOFTWARE ENGINEERING
CBSD is a latest technology for the development the complex or large software system with the help of using the COTS software components or reusable components. For huge large and hardly complex application, that time, some components or objects need to be developed separately specifically tailored to the need of the application and some components are selected from the third party repositories. So CBSE is latest technology which is used to improve the reusability functionality to select the optimal software components from components.

Researchers, practitioners and software engineers have been proposing and applying several algorithms for improving software development while focusing on software quality attributes. It is believed that facing some problems with traditional and OO paradigms motivate shift towards CBSE. CBSE emphasizes on building system by reusing high quality configurable software components [Pressman S. R. (2006)]. This reduces its development cost as well as time-to-market and ensures higher reliability, increase productivity, better maintainability; improve reliability and quality by exploiting reusability. This approach, when a software system is going to be developed, the implementation/coding has to be completed from scratch. Object-Oriented Technology (OOT), reusable software components have become an indispensable part of programming language knowledge.

For very large and hardly complex application, some components need to be developed separately specifically tailored to the need of the application and some components are selected from the third party repositories. So CBSE is latest technology which is mainly objective to increase the reusability functionality with the development of CBS from the COTS software components. This chapter presents a new optimal process to select a subset of components for specific application domain or optimal components which fulfill the requirements of client.

3.1 BASIC CONCEPTS

CBD has two mainly concepts. Firstly is the component and second is interface.
3.1.1 Component

A component is something that can be deployed as a black box. It means researcher has not knowledge of implementation. It has an external specification. (2005).

“A component as a nontrivial, nearly independent, and replaceable part of a system that fulfils a clear function in the context of a well-defined architecture”.

Figure 3.1 Component Framework
Figure 3.1 a framework of software component which is identify that how software components to interface with each other. Researcher stated that every component model identifies the outgoing or incoming interfaces.

<table>
<thead>
<tr>
<th>Component Types</th>
<th>Characteristics</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pure computation</td>
<td>Simple input/output</td>
<td>Math functions, filters, Transforms</td>
</tr>
<tr>
<td></td>
<td>Relations, no retained state</td>
<td></td>
</tr>
<tr>
<td>Memory</td>
<td>Shared collection of Persistent</td>
<td>Database, hypertext, file system, Symbol table</td>
</tr>
<tr>
<td></td>
<td>structured data</td>
<td></td>
</tr>
<tr>
<td>Manager</td>
<td>State and closely related Operations</td>
<td>Abstract data type, many Servers</td>
</tr>
<tr>
<td>Controller</td>
<td>Governs time sequences</td>
<td>Scheduler, synchronizer</td>
</tr>
<tr>
<td>Link</td>
<td>Passes information in entities</td>
<td>User interface, Communication link</td>
</tr>
</tbody>
</table>

*Table 3.1 Component Classification*

3.1.2 Interface
Interface has meant to specify the components service and integration of operations. An interface an integration of operations in which specifies their protocols and signatures.

3.2 THE CBSE-PROCESS
CBSE is partially related to Object-Oriented technology. As the following figure, there are parallel occurring two processes [Pressman S. R. (2006)].

1. Domain Engineering
2. Component Based Development

3.2.1 Domain Engineering
Domain engineering is a mechanism used to discover and develop a subset of software modules.
The main aim is to develop a mechanism which helps in identification of software components and to reuse them for CBSD. Domain engineering includes the domain analysis, design and implementation process which helps in identification and selection of specific application domain of component-based software.

3.2.2 **Component-Based Development**

CBSD includes two processes.

- Integration software product from COTS or reusable components.
- Developing software reusable component.

An approach of developing systems as integration of software components may be broadly classified in terms of four activities:

- Component Adaptation
- Component Qualification
- System evolution and maintenance
- Component Assembly
Figure 3.2 CBSE Process