DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIT 3-SERVER SIDE PROGRAMMING

3.1 SERVLET LIFE CYCLE

A servlet life cycle can be defined as the entire process from its creation till the destruction. The following are the paths followed by a servlet.The servlet is initialized by calling the init() method.

- The servlet calls service() method to process a client's request.
- The servlet is terminated by calling the destroy() method.
- Finally, servlet is garbage collected by the garbage collector of the JVM.

3.1.1 The init() Method

- The init method is called only once. It is called only when the servlet is created, and not called for any user requests afterwards. So, it is used for one-time initializations, just as with the init method of applets.
- The servlet is normally created when a user first invokes a URL corresponding to the servlet, but you can also specify that the servlet be loaded when the server is first started.
- When a user invokes a servlet, a single instance of each servlet gets created, with each user request resulting in a new thread that is handed off to doGet or doPost as appropri- ate.
- The init() method simply creates or loads some data that will be used throughout the life of the servlet.
    - The init method

definition looks like this –

```
Public void init() throws
    ServletException {
                    // Initialization code
                    }
```

3.1.2 The service() Method

- The service() method is the main method to perform the actual task.

The servlet container (i.e. web server) calls the service() method to handle requests coming from the client(browsers) and to write the formatted response back to the client.

- Each time the server receives a request for a servlet, the server spawns a new thread and
  calls service.
- The service() method checks the HTTP request type (GET, POST, PUT, DELETE, etc.) and calls doGet, doPost, doPut, doDelete, etc. methods as appropriate.
- Here is the signature of this method –

```
public void service(ServletRequest request,

ServletResponse response) throws ServletException,

                IOException {

                }
```

- The service () method is called by the container and service method invokes doGet, doPost, doPut, doDelete, etc. methods as appropriate. So you have nothing to do with service() method but you override either doGet() or doPost() depending on what type of request you receive from the client.
- The doGet() and doPost() are most frequently used methods with in each service request. Here is the signature of these two methods.

3.1.3 The doGet() Method

- A GET request results from a normal request for a URL or from an HTML form that has no METHOD specified and it should be handled by doGet() method.

```
public void doGet(HttpServletRequest request,

    HttpServletResponse response) throws

    ServletException, IOException {

            // Servlet code

            }
```

3.1.4 The doPost() Method

- A POST request results from an HTML form that specifically lists POST as the METHOD and it should be handled by doPost() method.

```
public void doPost(HttpServletRequest request,
```

HttpServletResponse response) throws ServletException,

IOException {

// Servlet code

}

3.1.5 The destroy() Method

- The destroy() method is called only once at the end of the life cycle of a servlet. This method gives your servlet a chance to close database connections, halt background threads, write cookie lists or hit counts to disk, and perform other such cleanup activities.
- After the destroy() method is called, the servlet object is marked for garbage collection.
  The destroy method

definition looks like this – public

void destroy() {

// Finalization code...

}

3.1.6 Architecture Diagram

- The following figure depicts a typical servlet life-cycle scenario.
- First the HTTP requests coming to the server are delegated to the servlet container.
- The servlet container loads the servlet before invoking the service() method.
- Then the servlet container handles multiple requests by spawning multiple threads, each thread executing the service() method of a single instance of the servlet.

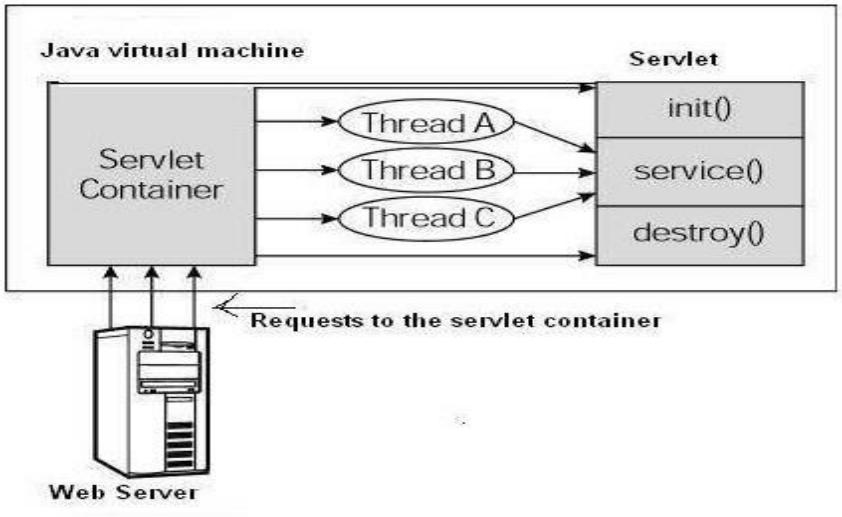

Fig 3.2 Servlet Life Cycle