

Developing Your Own Web Server

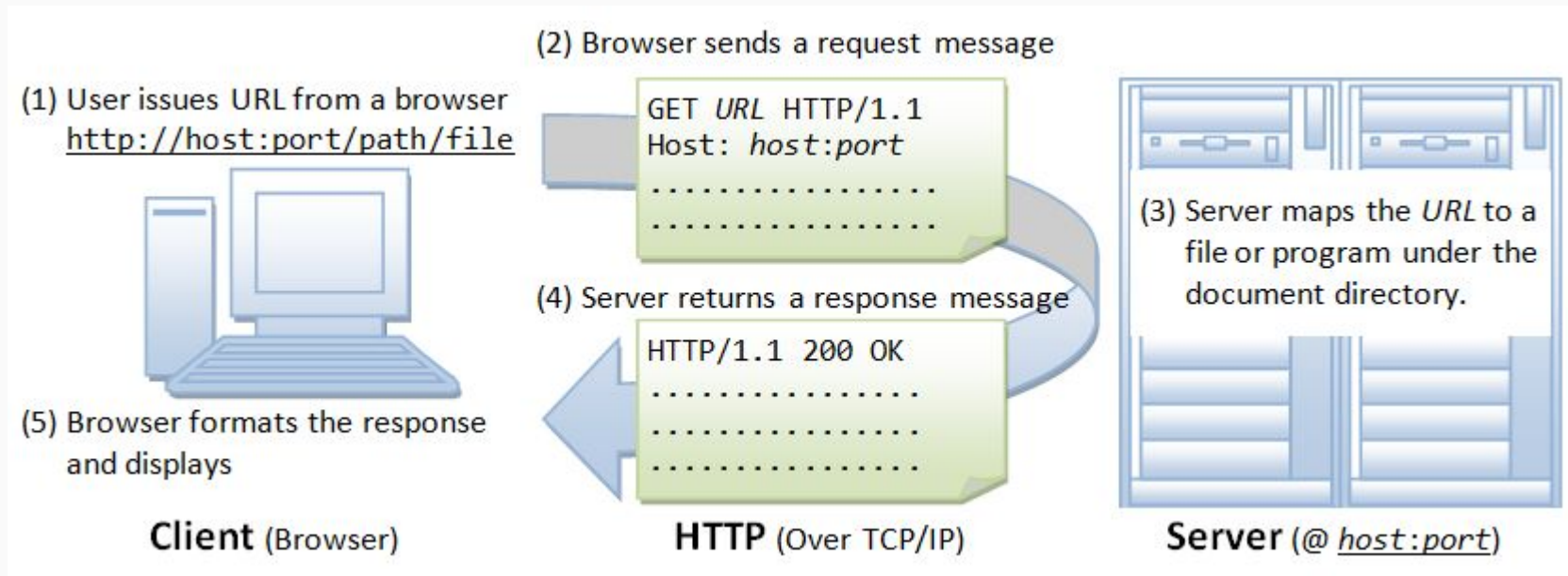
Melnica Web Server Implementation - Batuhan Düzgün - June 2021



Roadmap

- 1- Fundamental topics to develop your basic web server
- 2- General web server architecture
- 3- Tomcat web server architecture
- 4- Melnica web server architecture
- 5- Melnica web server init/start flow diagrams
- 6- Future strategies

Fundamental Topics - HTTP Protocol



Fundamental Topics - HTTP Request

```
GET /doc/test.html HTTP/1.1
```

```
Host: www.test101.com
```

```
Accept: image/gif, image/jpeg, */*
```

```
Accept-Language: en-us
```

```
Accept-Encoding: gzip, deflate
```

```
User-Agent: Mozilla/4.0
```

```
Content-Length: 35
```

```
bookId=12345&author=Tan+Ah+Teck
```

Request Line

Request Headers

Request
Message
Header

A blank line separates header & body

Request Message Body

Fundamental Topics - HTTP Response

```
HTTP/1.1 200 OK
```

```
Date: Sun, 08 Feb xxxx 01:11:12 GMT
```

```
Server: Apache/1.3.29 (Win32)
```

```
Last-Modified: Sat, 07 Feb xxxx
```

```
ETag: "0-23-4024c3a5"
```

```
Accept-Ranges: bytes
```

```
Content-Length: 35
```

```
Connection: close
```

```
Content-Type: text/html
```

```
<h1>My Home page</h1>
```

Status Line

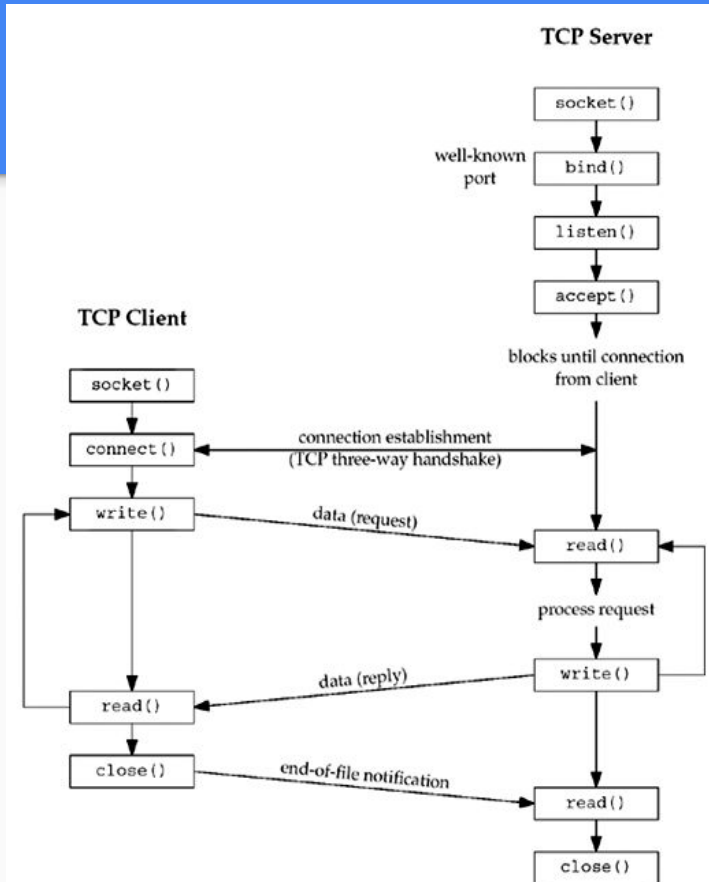
Response Headers

Response
Message
Header

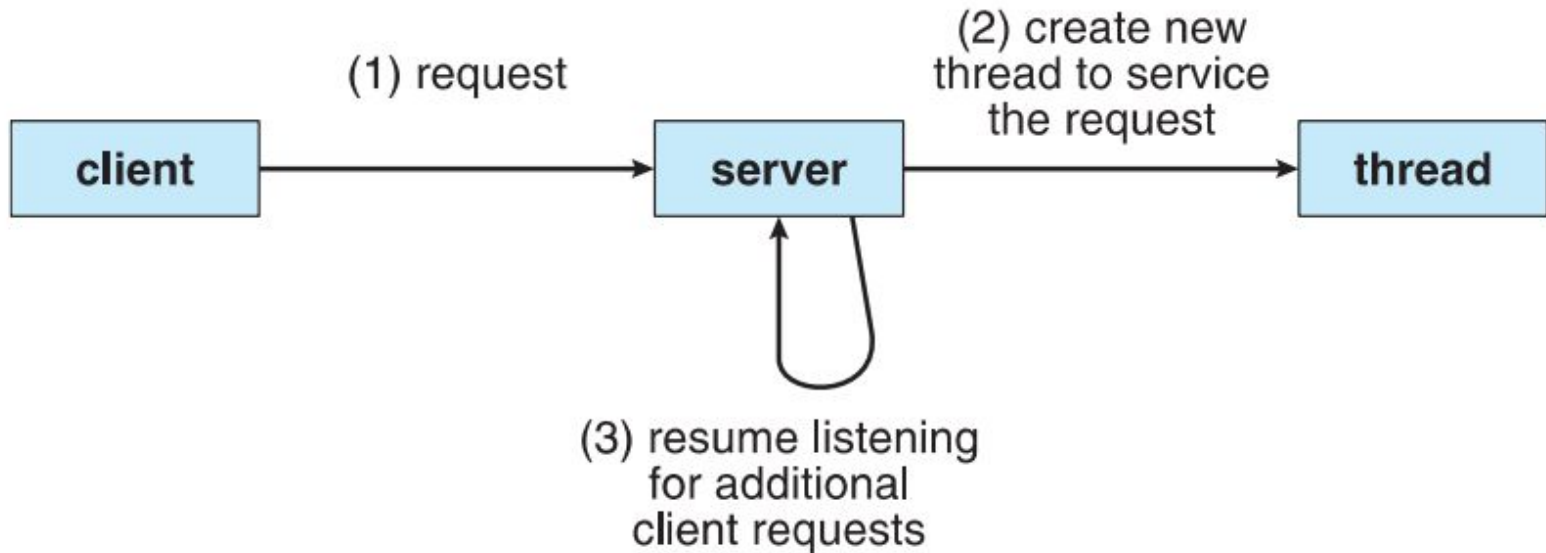
A blank line separates header & body

Response Message Body

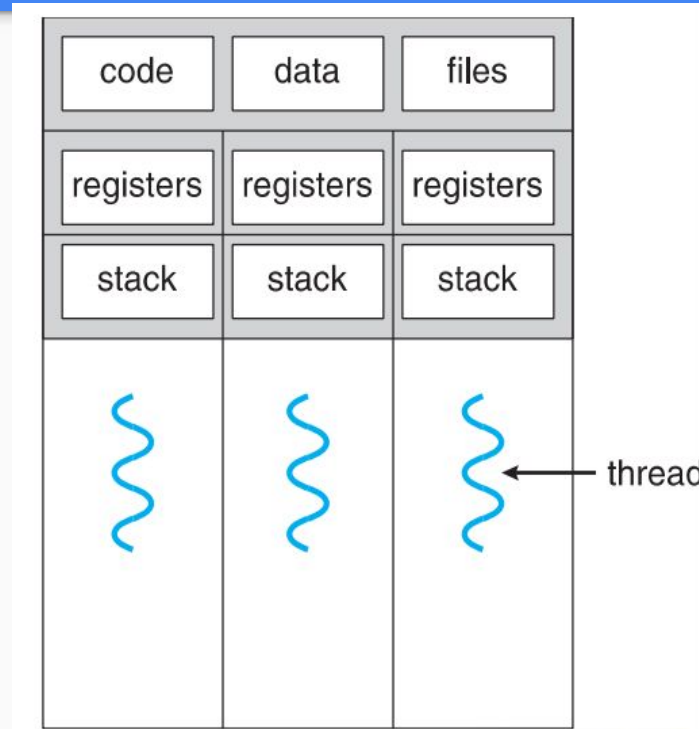
Fundamental Topics - Socket Programming



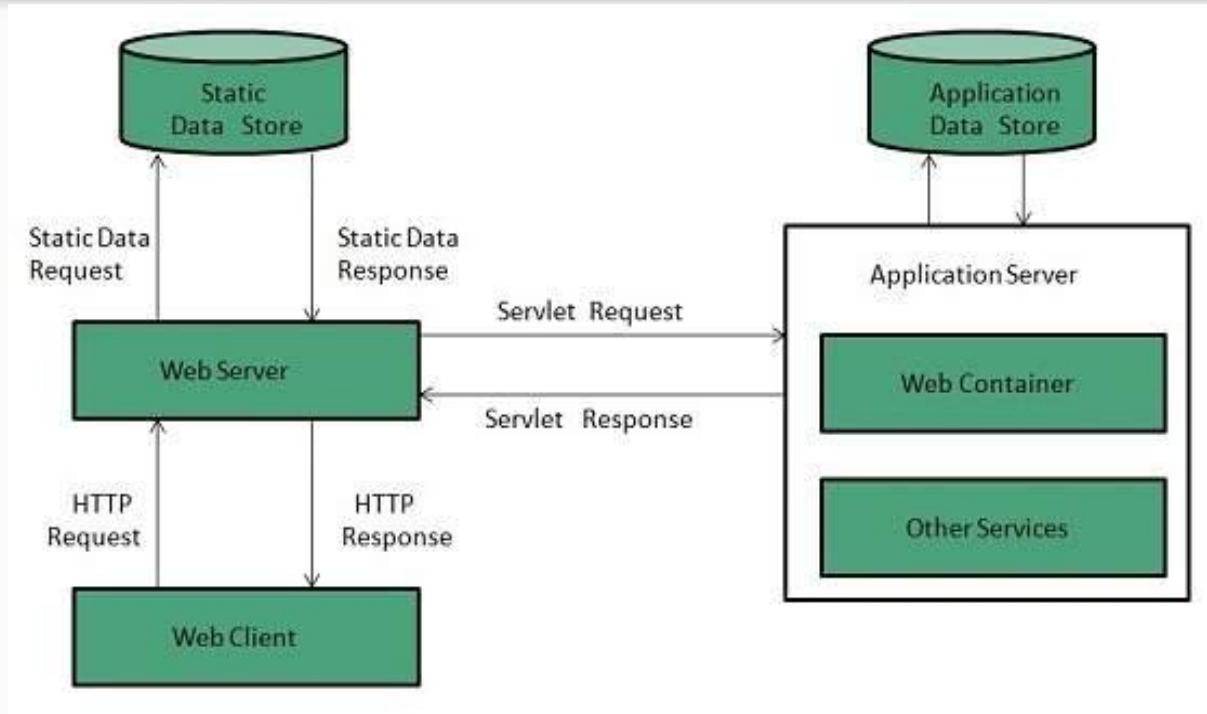
Fundamental Topics - Multithreaded Prog.



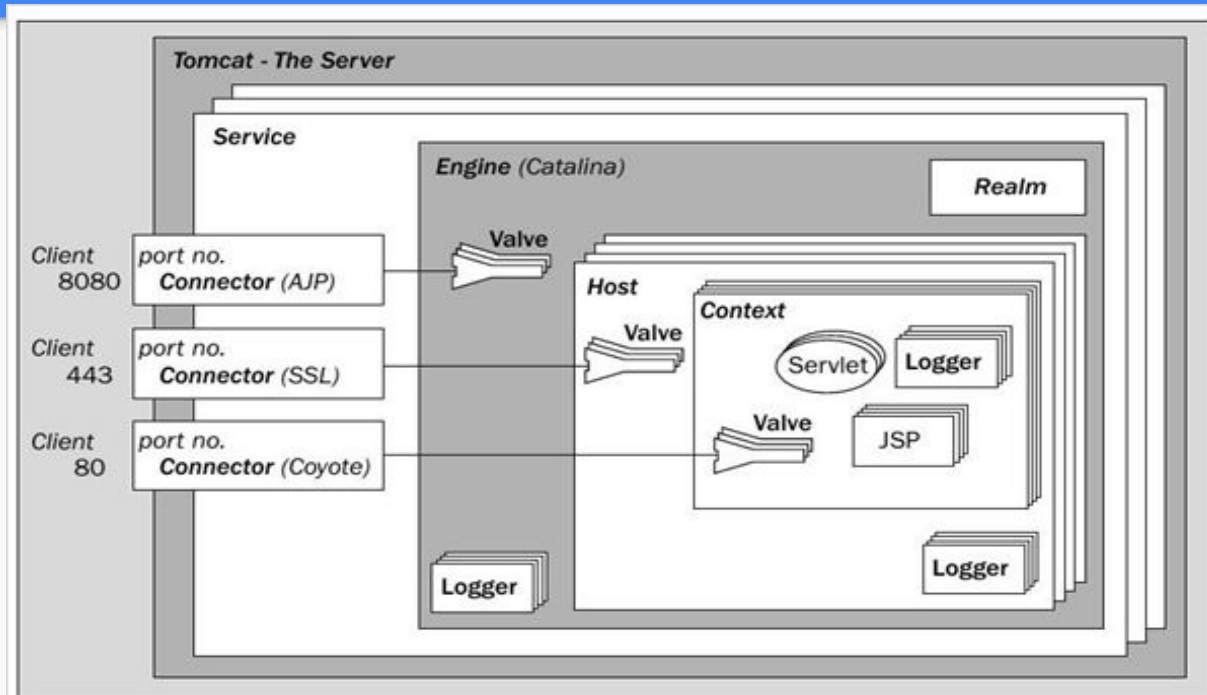
Fundamental Topics - Multithreaded Prog.



General Web Server Architecture

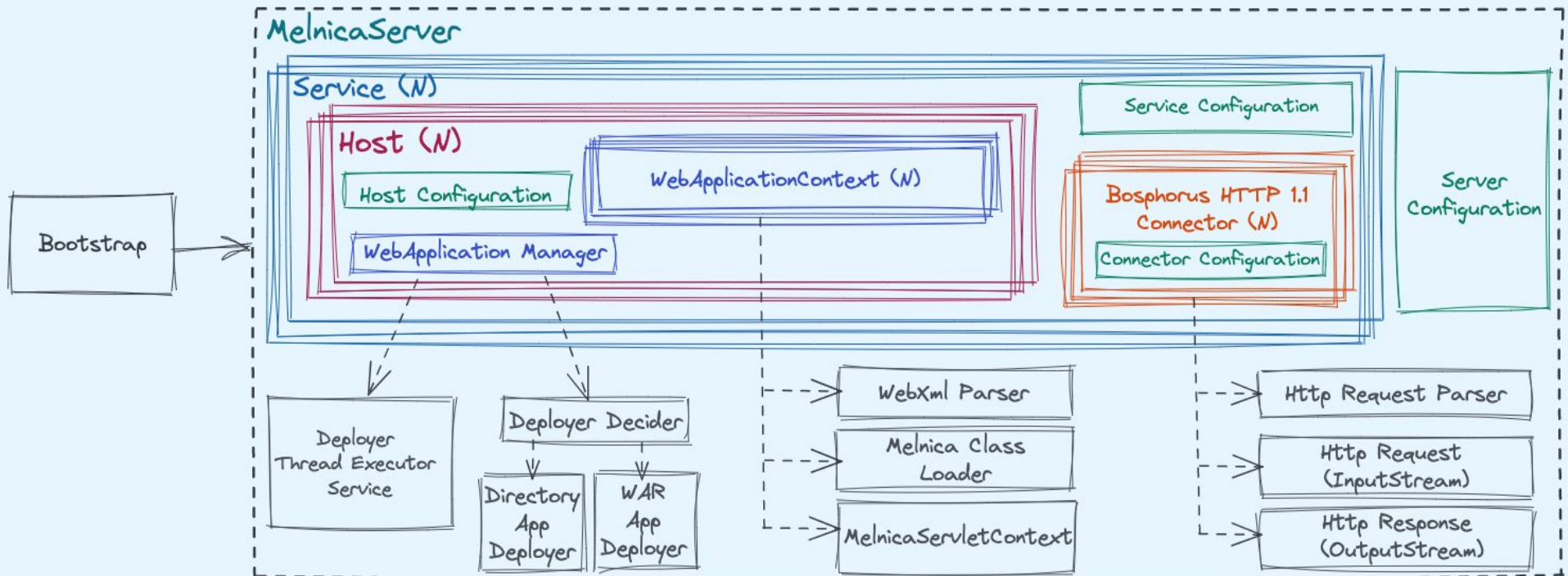


Tomcat Web Server Architecture



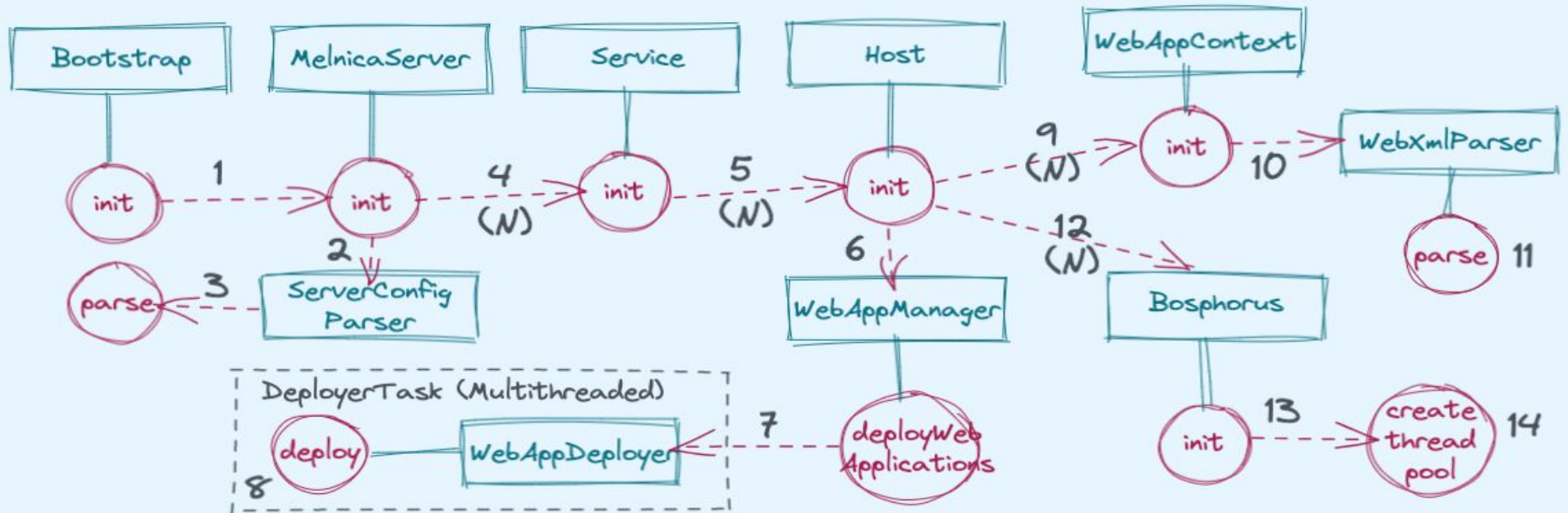
Tomcat's architecture.

Melnica Web Server Architecture



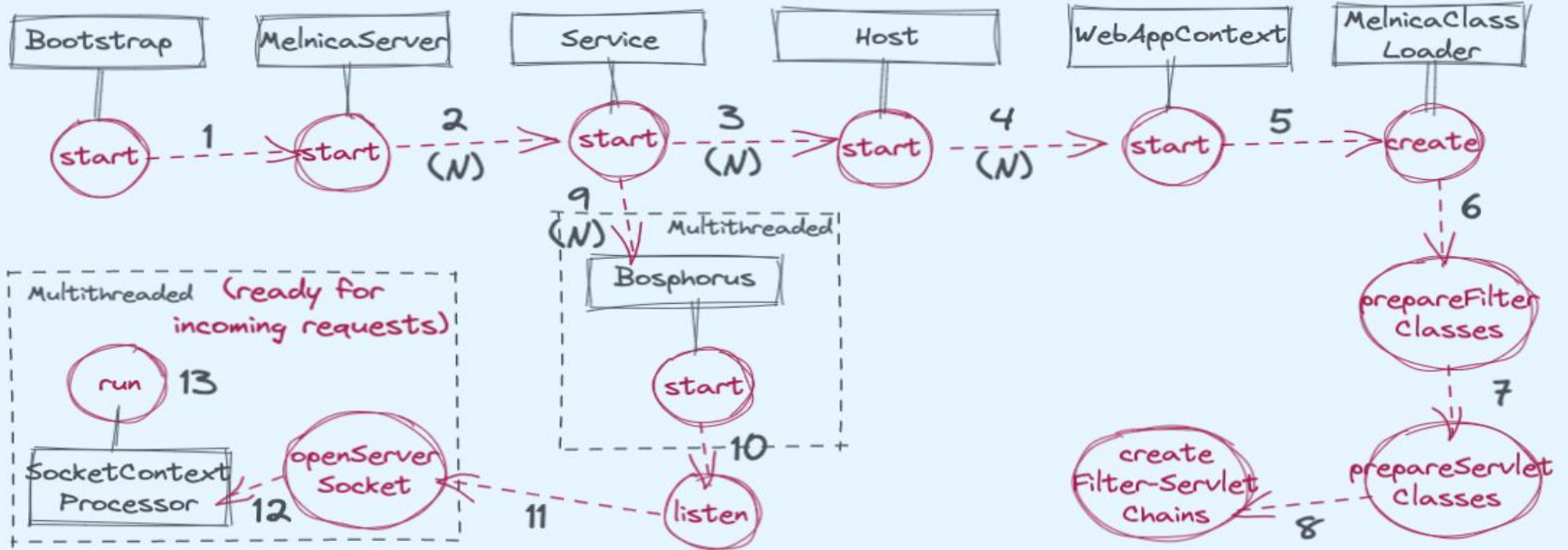
Melnica Web Server Initialize Flow

Melnica Initialize Flow Diagram



Melnica Web Server Start Flow

Melnica Initialize Start Diagram



Future Strategies

- Developing socket context executor which orchestrate all http req/resp in terms of content type
- Refactoring all xml parsers in the project
- Supporting yaml configuration via new yaml parser
- Refactoring the classes which create Http Req/Resp data structures
- Refactoring the HttpRequestParser class