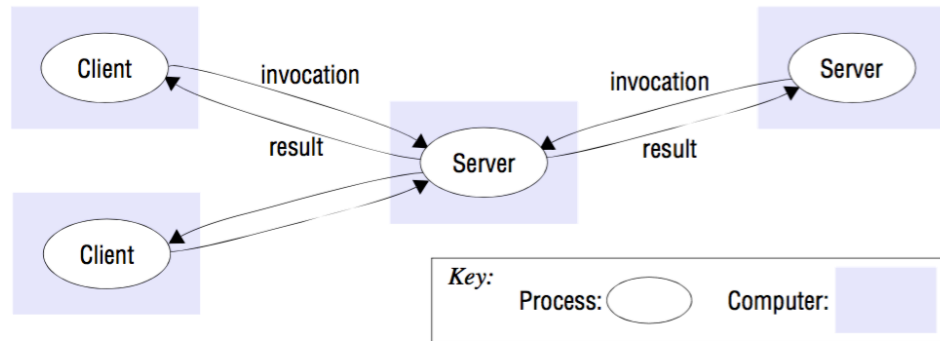


## Chapter 2 System Models

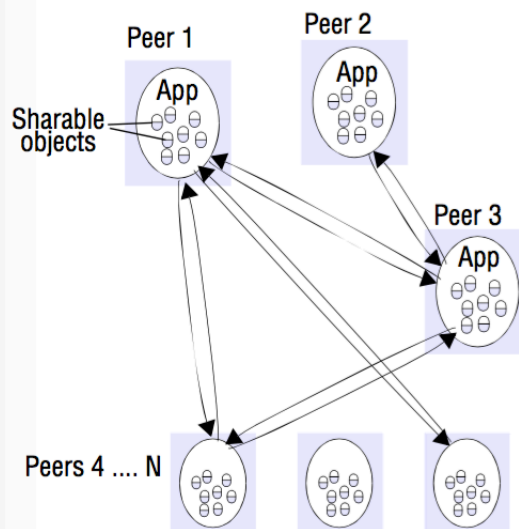
### 1. Architectural Models

#### 1. client-server



- 1.
2. peer to peer

**Figure 2.4a** Peer-to-peer architecture



- 1.
2. interacting cooperatively as *peers* without any distinction between client and server processes or the computers on which they run.

### 2. Fundamentals Models

#### 1. interaction model

1. Computation occurs within processes; the processes interact by passing messages, resulting in communication (information flow) and coordination (synchronization and ordering of activities) between processes. In the analysis and design of distributed systems we are concerned especially with these interactions. The interaction model must

reflect the facts that communication takes place with delays that are often of considerable duration, and that the accuracy with which independent processes can be coordinated is limited by these delays and by the difficulty of maintaining the same notion of time across all the computers in a distributed system.

## 2. failure model

1. The correct operation of a distributed system is threatened whenever a fault occurs in any of the computers on which it runs (including software faults) or in the network that connects them. Our model defines and classifies the faults. This provides a basis for the analysis of their potential effects and for the design of systems that are able to tolerate faults of each type while continuing to run correctly

## 3. security model

1. the modular nature of distributed systems and their openness exposes them to attack by both external and internal agents. Our security model defines and classifies the forms that such attacks may take, providing a basis for the analysis of threats to a system and for the design of systems that are able to resist them

## 3. Agreement in Peperland pg. 65

1. asynchronous Peperland, the messengers are very variable in their speed

## 4. Failure detection in Peperland pg. 69

1. the other division may have been defeated just after it sent the latest messenger.
2. This shows that no protocol that guarantees agreement between the Peperland divisions can exist if messengers can be captured.

## 5. Cloudlet article "Mobile Computing the Next Decade"

## 6. JEE

1. Java Enterprise Edition is the platform for building distributed systems

## 7. Servlets

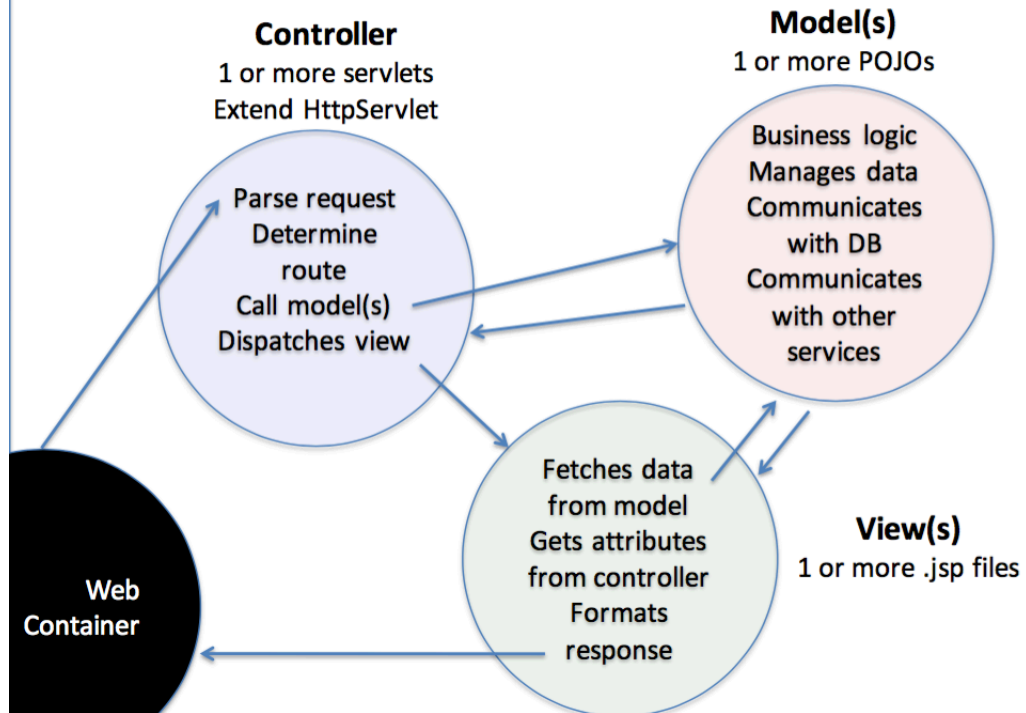
1. the class designed to process HTTP requests
2. life cycle
  1. load class
  2. instantiate servlet
  3. init()
  4. service()
  5. destroy()

## 8. Java Server Pages (JSP)

1. provides a higher-level abstraction to sevlets
2. HTML with java code embedded
3. can run without java

## 9. Model View Controller

# Model – View - Controller



- 1.
10. WWW
  1. tim berners lee
  2. 3 standards
    1. http (hypertext transfer protocol)
    2. **Uniform Resource Locators (URLs)** uri(uniform resource identifiers)
    3. html(hypertext markup language)
    4. W3c IETF
11. HTTP protocol, including request and response header
  1. request and response messages
  2. methods/safety/idempotence
  3. Request Header

General Format	Example
<pre> &lt;method&gt; &lt;resource identifier&gt; &lt;HTTP Version&gt; &lt;crLf&gt; [&lt;Header&gt;: &lt;value&gt;] &lt;crLf&gt; ... [&lt;Header&gt;: &lt;value&gt;] &lt;crLf&gt; a blank line [entity body]           </pre>	<pre> GET /course/95-702/ HTTP/1.1 Host: www.andrew.cmu.edu User-Agent: Joe typing Accept: text/html This line intentionally left blank           </pre>

- 1.
4. Response Header

General Format	Example
<pre> &lt;HTTP Version&gt; &lt;Status&gt; &lt;crLf&gt; [&lt;Header&gt;: &lt;value&gt;] &lt;crLf&gt; ... [&lt;Header&gt;: &lt;value&gt;] &lt;crLf&gt; a blank line [response body] </pre>	<pre> HTTP/1.1 200 OK Date: Mon, 13 Jan 2014 15:43:08 GMT Server: Apache/1.3.39 (Unix) mod_throttle/3.1.2 ... Set-Cookie: webstats-cmu=cmu128.2.87.50.8400; ... Last-Modified: Sun, 12 Jan 2014 21:46:30 GMT Accept-Ranges: bytes Content-Length: 9014 Content-Type: text/html This line intentionally left blank &lt;HTML&gt; &lt;HEAD&gt; &lt;META http-equiv="Content-Type" content="text/html; charset=UTF-8"&gt; ... </pre>

1.

## 12. Safe operations

1. does not change the state of the resources on the server

## 13. Idempotent operations

1. multiple identical requests are the same as a single request
2. safe methods are idempotent

Method	Purpose?	Safe?	Idempotent?
GET	Retrieve a resource	Y	Y
PUT	Insert or replace a resource	N	Y
DELETE	Remove a resource	N	Y
HEAD	Get header information only of a resource	Y	Y
POST	Append to or modify a resource	N	N

14.