

## CS 5523 Lecture 2: Architectural Models

---

- Discussion questions for CDK Chapter 1
- Discussion of Laboratory 1
- What is an architectural model?
- Common architectural models
- A detailed look at web architecture

## Discussion Questions from CDK Chapter 1

---

[CDK 1.9]

Suppose that the operations of the BLOB object are separated into two categories - public operations that are available to all users and protected operations that are available only to certain named users.

- State all of the problems involved in ensuring that only the named users can use a protected operation.
- Supposing that access to a protected operation provides information that should not be revealed to all users, what further problems arise?

## Discussion Questions from CDK Chapter 1

---

[CDK 1.6]

- What is a URL?
- List the three main components of a URL, stating how their boundaries are denoted and illustrating each one from your example.
- To what extent is a URL location independent?

## Discussion Questions from CDK Chapter 1

---

[CDK 1.12]

A server process maintains a shared information object such as the BLOB object of Exercise 1.7.

- Give arguments for and against allowing the client requests to be executed concurrently by the server.
- In the case where they are executed concurrently, give an example of possible 'interference' that can occur between the operations of different clients.
- Suggest how such interference may be prevented.

## Discussion Questions from CDK Chapter 1

---

[CDK 1.7]

A server program written in one language (for example C++) provides the implementation of a BLOB object that is intended to be accessed by clients that may be written in a different language (for example Java). The client and server computers may have different hardware, but all of them are attached to an internet. Describe the problems due to each of the five aspects of heterogeneity that need to be solved to make it possible for a client object to invoke a method on the server object.

## HTTP Primer:

---

- Assumes reliable transport (in practice TCP)
- Request-reply protocol:
  - Client initiates with a request (GET)
  - Server responds with requested document or an error
- Content types identify document types for browser
- HTTP 1.0 - need a separate request for each resource
- HTTP 1.1 - requests pipelined and served by a single connection
- Executables:
  - CGI (Common Gateway Interface) executes on server
  - Java Applets execute on browser

## HTTP Terminology (from W3C RFC 1945):

- *Client* - an application that establishes a connection
- *User agent* - a client that makes requests for services
- *Server* - an application that accepts connections and responds
- *Origin server* - server that has the resource
- *Proxy* - an intermediary between client and server that makes requests on behalf of clients - must parse http
- *Gateway* - a server that acts on behalf of other servers and receives request as though it were the origin server
- *Tunnel* - intermediary that acts as a blind relay
- *Cache* - local store of response messages

## HTTP Server Responses:

- 1xx: informational
- 2xx: action successful
- 3xx: redirection - requires additional action
- 4xx: client error - bad syntax or invalid resource
- 5xx: server error - the server could not fulfil valid request

Example of a server response line

```
HTTP/1.0 404 Not Found
```

## HTTP Transaction:

- *Initial line* (GET, HEAD or POST for client and a status line for server)
- *Zero or more header lines* (giving additional information)
- *A blank line* (contains CRLF)
- *An optional message body* (for the server response it will be the item requested and could be binary)

*Initial and header lines are tokenized ASCII separated by LWS (linear white space)*

## Demo of TCPClient and TCPServer

```
vip.cs.utsa.edu/classes/cs5523s2002/lectures/src/TCPClient.java
```

- Compile with

```
javac TCPClient.java
```
  - Run with

```
java TCPClient "Help me Help me" pandora.cs.utsa.edu
```
- ```
vip.cs.utsa.edu/classes/cs5523s2002/lectures/src/TCPServer.java
```
- Compile with

```
javac TCPServer.java
```
  - Run (on pandora) with

```
java TCPServer
```

## Request Line Syntax:

```
Method SP Request-URI SP HTTP-Version CRLF
```

Examples:

```
GET /classes/cs5523s2001/home.html HTTP/1.0
GET http://www.cs.utsa.edu/index.html HTTP/1.0
```

Full Header Example:

```
GET /classes/cs5523s2001/home.html HTTP/1.0
User: krobbins@cs.utsa.edu
CRLF
```

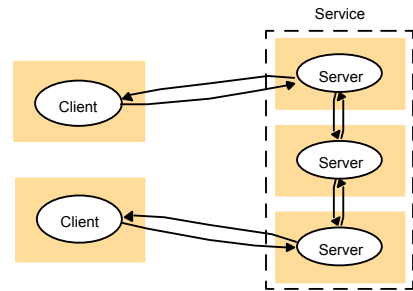
## What is an architectural model?

- *Structure in terms of separately specified components.*
- *Placement of components across a network of computers*
- *Interrelationships between components*
- *Division of responsibilities among components*

## Terminology

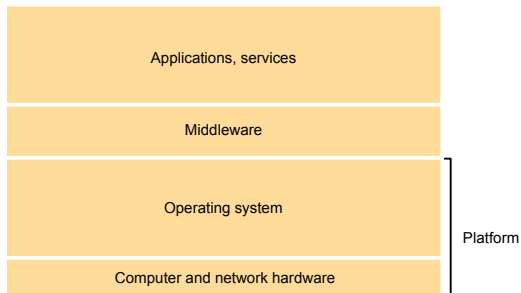
- server - process that accepts requests from other process.
- service - provided by one or more servers
- platform - hardware and underlying operating system
- middleware - software layer that masks heterogeneity and provides a programming model

Figure 2.3 (CDK)  
A service provided by multiple servers



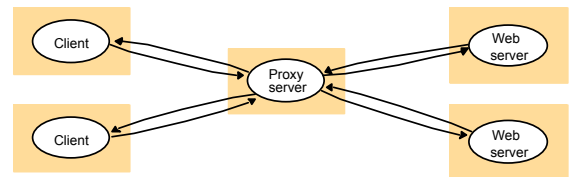
Instructor's Guide for Condonet, Delfmann and Kindberg Distributed Systems: Concepts and Design Edn. 3  
© Addison-Wesley Publishers 2000

Figure 2.1 (CDK)  
Software and hardware service layers in distributed systems



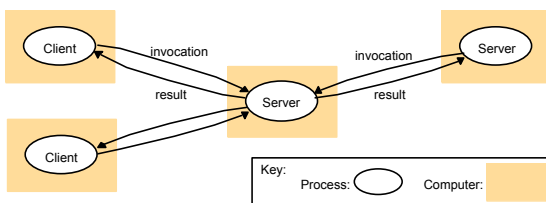
Instructor's Guide for Condonet, Delfmann and Kindberg Distributed Systems: Concepts and Design Edn. 3  
© Addison-Wesley Publishers 2000

Figure 2.4 (CDK)  
Web proxy server



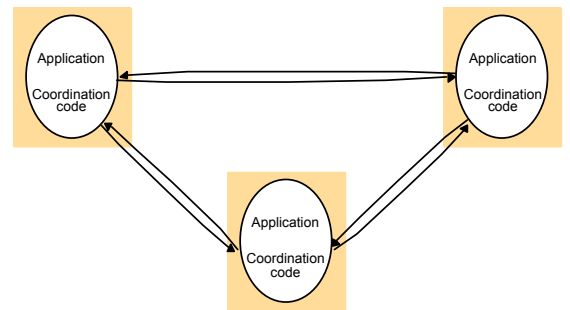
Instructor's Guide for Condonet, Delfmann and Kindberg Distributed Systems: Concepts and Design Edn. 3  
© Addison-Wesley Publishers 2000

Figure 2.2 (CDK)  
Clients invoke individual servers



Instructor's Guide for Condonet, Delfmann and Kindberg Distributed Systems: Concepts and Design Edn. 3  
© Addison-Wesley Publishers 2000

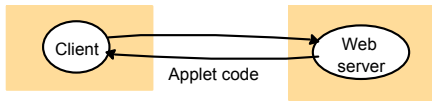
Figure 2.5 (CDK)  
A distributed application based on peer processes



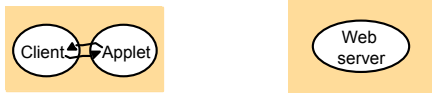
Instructor's Guide for Condonet, Delfmann and Kindberg Distributed Systems: Concepts and Design Edn. 3  
© Addison-Wesley Publishers 2000

Figure 2.6 (CDK)  
Web applets

a) client request results in the downloading of applet code



b) client interacts with the applet

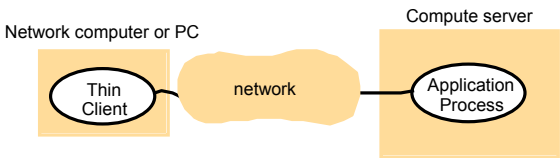


Instructor's Guide for Cimbura, Deffensee and Kuehling. Distributed Systems: Concepts and Design. Edn. 3  
© Addison-Wesley Publishers 2009

For next time:

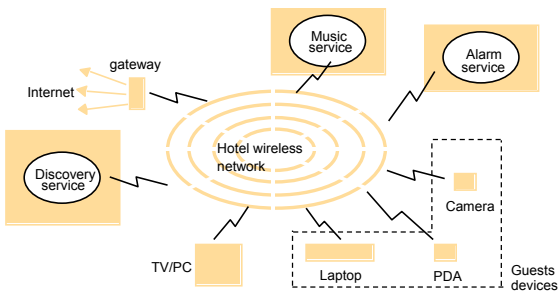
- Finish reading Chapter 2
- Start Laboratory 1 so that you can ask questions

Figure 2.7 (CDK)  
Thin clients and compute servers



Instructor's Guide for Cimbura, Deffensee and Kuehling. Distributed Systems: Concepts and Design. Edn. 3  
© Addison-Wesley Publishers 2009

Figure 2.8 (CDK)  
Spontaneous networking in a hotel



Instructor's Guide for Cimbura, Deffensee and Kuehling. Distributed Systems: Concepts and Design. Edn. 3  
© Addison-Wesley Publishers 2009