Lecture 4

S3 - Summary of Web/ServerSide Development in JEE

Section 3

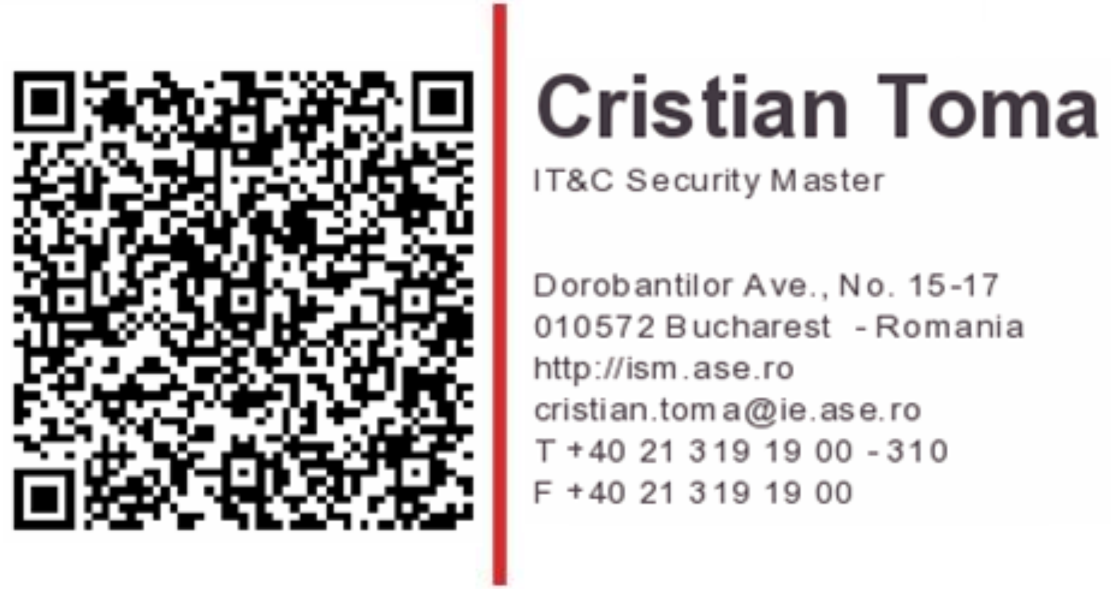
presentation

DAD – Distributed Applications Development

Cristian Toma

D.I.C.E/D.E.I.C – Department of Economic Informatics & Cybernetics
www.dice.ase.ro

Cristian Toma – Business Card



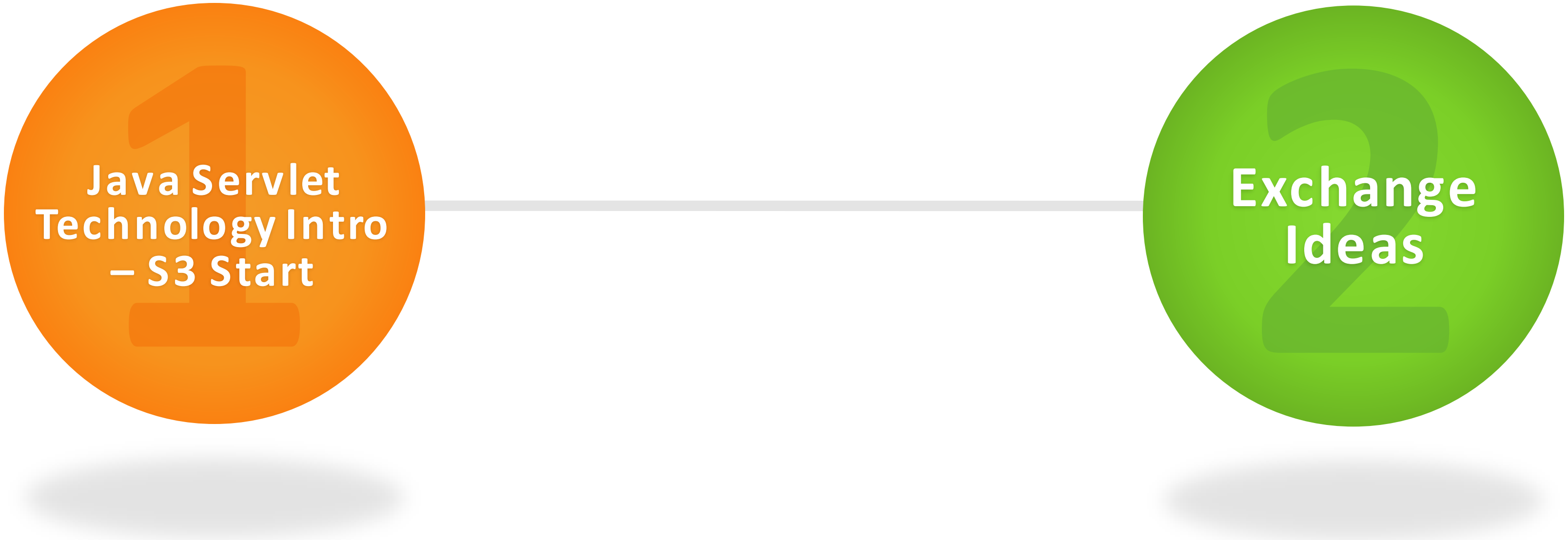
Cristian Toma

IT&C Security Master

Dorobantilor Ave., No. 15-17
010572 Bucharest - Romania
http://ism.ase.ro
cristian.toma@ie.ase.ro
T +40 21 319 19 00 - 310
F +40 21 319 19 00

IT&C SECURITY MASTER
ISM
2005
ism.ase.ro
Scientia potestas est

Agenda for Lecture 4



1 Java Servlet Technology Intro – S3 Start

2 Exchange Ideas

1

DAD Section 3 - Summary of Web Development in JEE, Servlet Lifecycle, Java Servlet Samples

Java Servlet Tech Recapitulation

1. Java Servlet Technology

What is Java Servlet?

Sun: Java Servlet technology provides Web developers with a simple, consistent mechanism for extending the functionality of a Web server and for accessing existing business systems.

WiKi: Servlets are Java programming language objects that dynamically process requests and construct responses. The Java Servlet API allows a software developer to add dynamic content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML.

The web server for the Java servlet simple tests is Apache Tomcat in Ubuntu Linux and Apache TomEE+ / Tom PluME
http://tomcat.apache.org
http://tomee.apache.org/index.html

1. Java Servlet Technology

What is Java Servlet?

* Java Servlets Intro & Development Cycle

- Java Servlet Structure
- Java Servlet sample that generates “Plain Text”
- Compiling and testing Java Servlet
- A Simple Servlet Generating HTML

* Processing the Request: Form Data

- Introduction (Format, URL-encoding, GET, POST)
- Example: Reading Specific Parameters
- Example: Making Table of All Parameters

1. Java Servlet Technology

What is Java Servlet?

* HTTP Request Headers

- Common Request Headers
- Sample: Java Servlet for displaying HTML table of the Request Headers

* HTTP Status Codes & HTTP Response Headers

- Overview: Status Codes & Response Headers
- Set Status Codes from Java Servlets
- Set Response Headers from Java Servlets
- Sample: Refresh at each 3 seconds based on Response Headers

* Handling Cookies

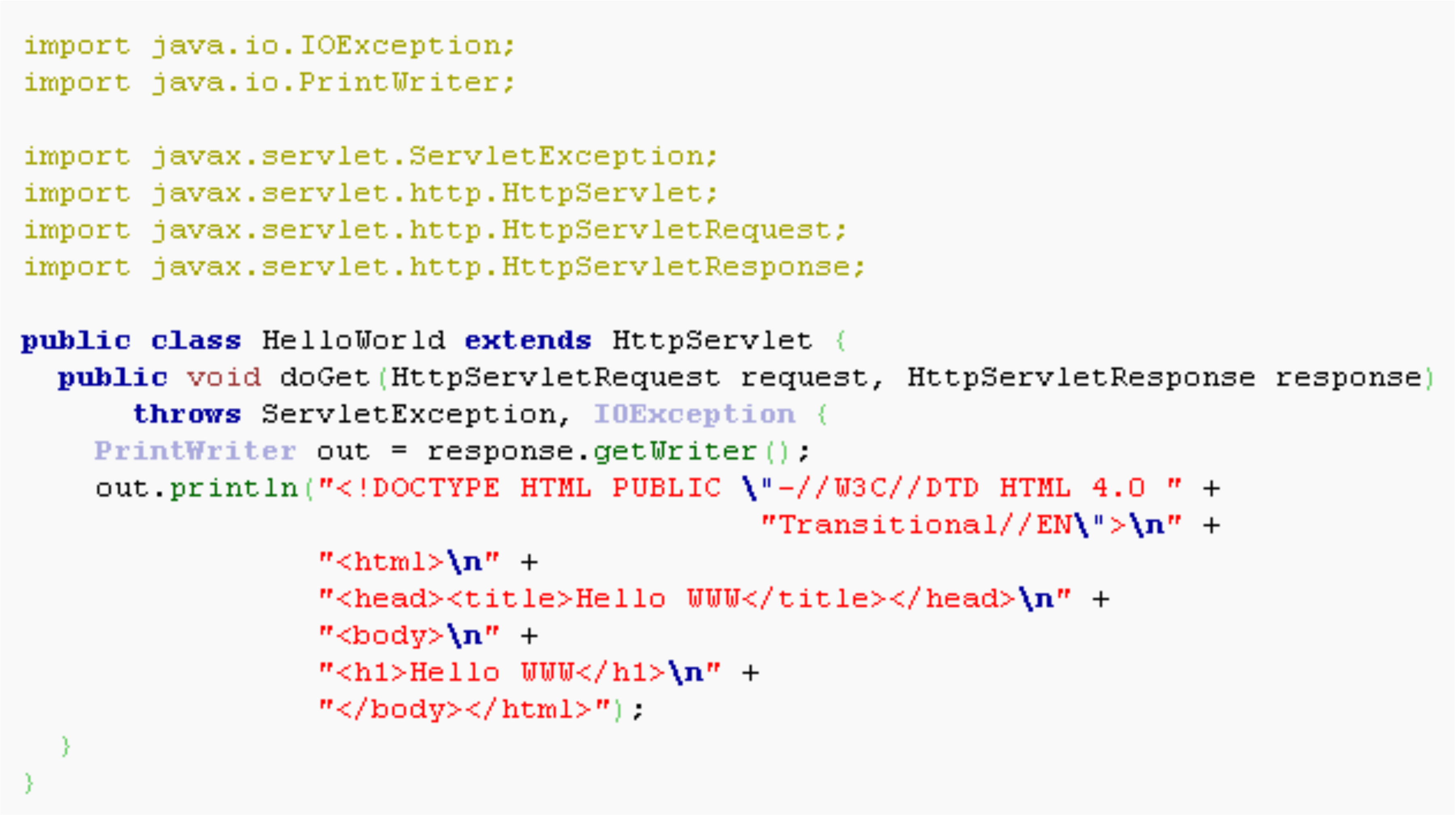
- Cookies Intro
- Java Servlet Cookie API
- Sample: Set/Get Cookie for Internet Explorer & Mozilla

* Session Tracking

- Session Tracking Overview
- Java Servlet Session Tracking API + Sample

1. Java Servlet Technology

Java Servlet API



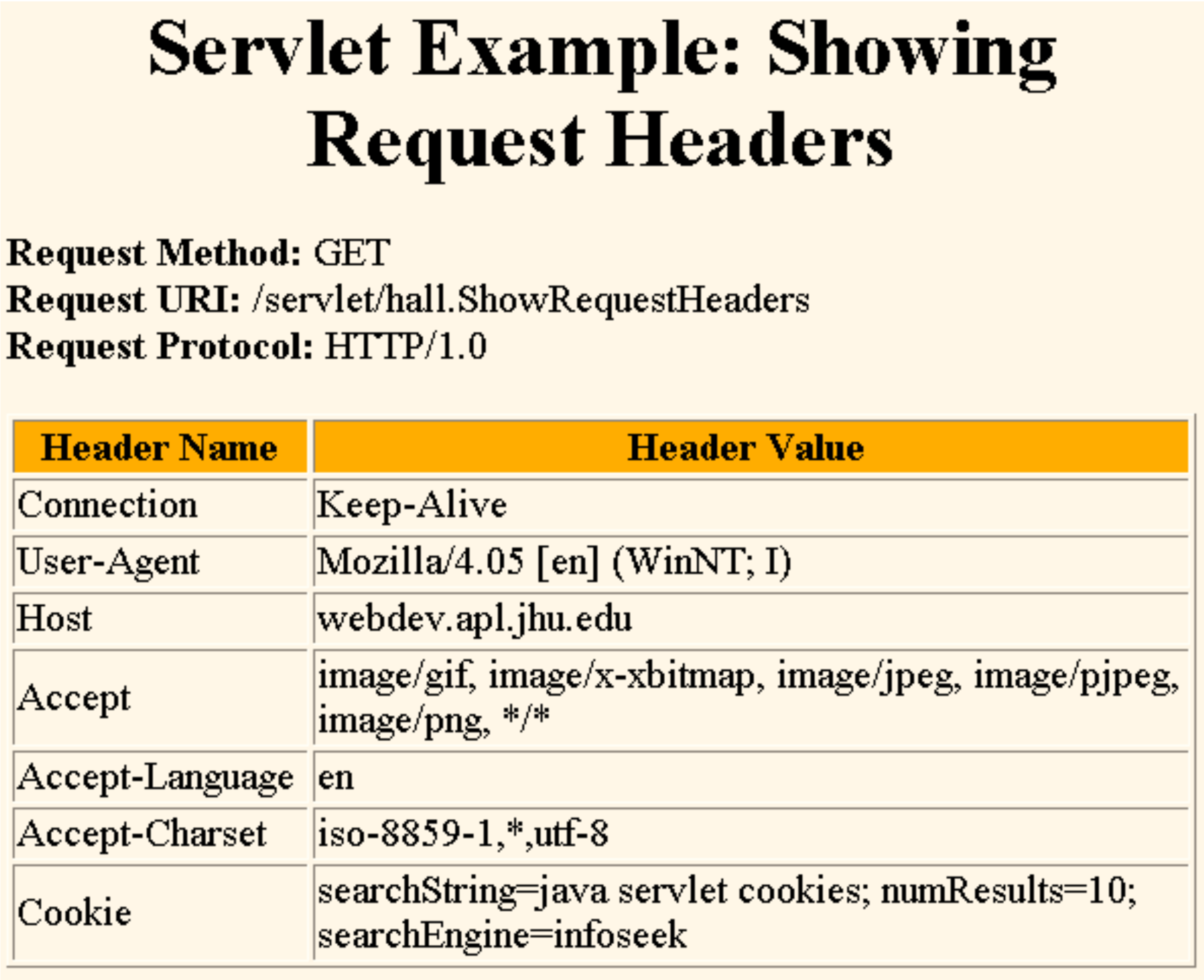
```
import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class HelloWorld extends HttpServlet {
  public void doGet(HttpServletRequest request, HttpServletResponse response)
      throws ServletException, IOException {
    PrintWriter out = response.getWriter();
    out.println("<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
                                        "Transitional//EN\">\n" +
                "<html>\n" +
                "<head><title>Hello WWW</title></head>\n" +
                "<body>\n" +
                "<h1>Hello WWW</h1>\n" +
                "</body></html>");
  }
}
```

1. Java Servlet Technology

Java Servlet Technology Intro



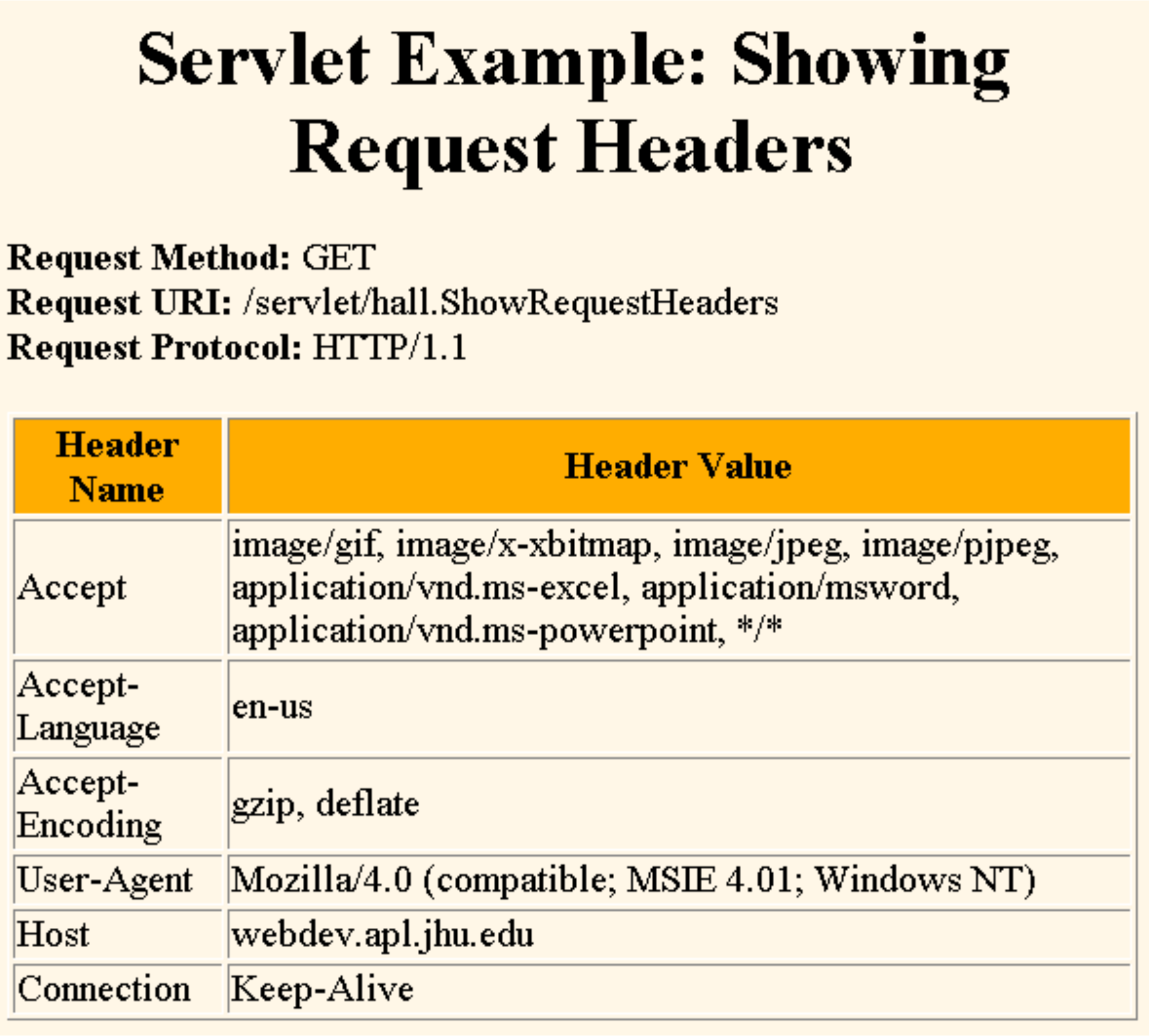
Servlet Example: Showing Request Headers

Request Method: GET
Request URI: /servlet/hall.ShowRequestHeaders
Request Protocol: HTTP/1.0

Header Name	Header Value
Connection	Keep-Alive
User-Agent	Mozilla/4.05 [en] (WinNT; I)
Host	webdev.apl.jhu.edu
Accept	image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
Accept-Language	en
Accept-Charset	iso-8859-1,*,utf-8
Cookie	searchString=java servlet cookies; numResults=10; searchEngine=infoseek

1. Java Servlet Technology

Java Servlet Technology Intro



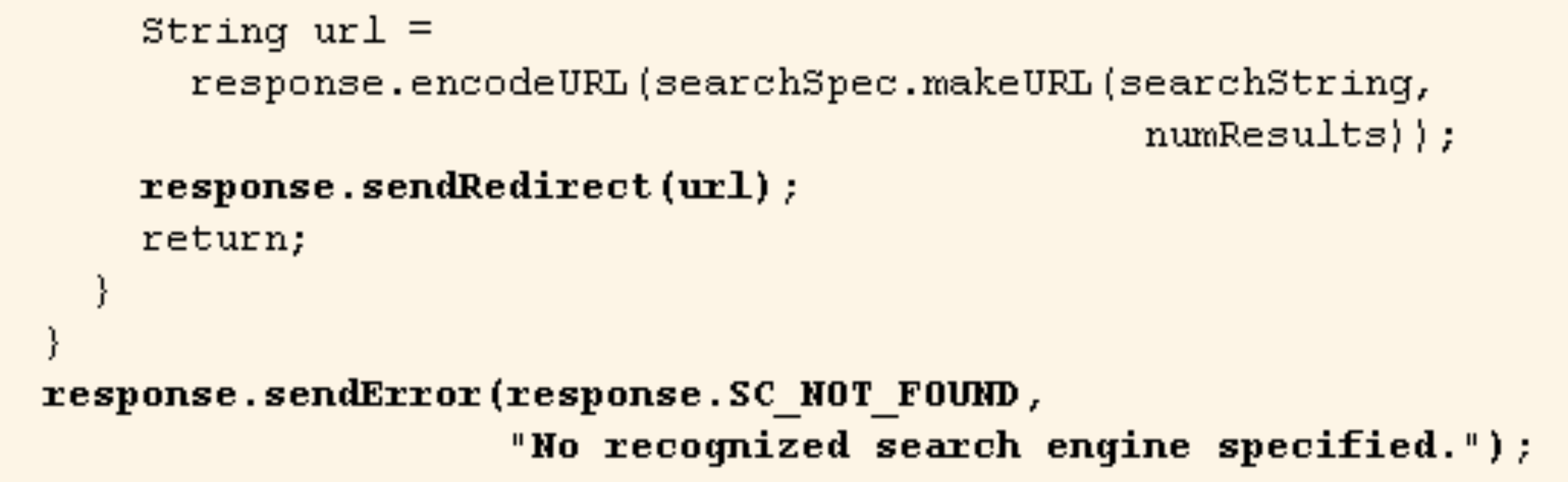
Servlet Example: Showing Request Headers

Request Method: GET
Request URI: /servlet/hall.ShowRequestHeaders
Request Protocol: HTTP/1.1

Header Name	Header Value
Accept	image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-excel, application/msword, application/vnd.ms-powerpoint, */*
Accept-Language	en-us
Accept-Encoding	gzip, deflate
User-Agent	Mozilla/4.0 (compatible; MSIE 4.01; Windows NT)
Host	webdev.apl.jhu.edu
Connection	Keep-Alive

1. Java Servlet Technology

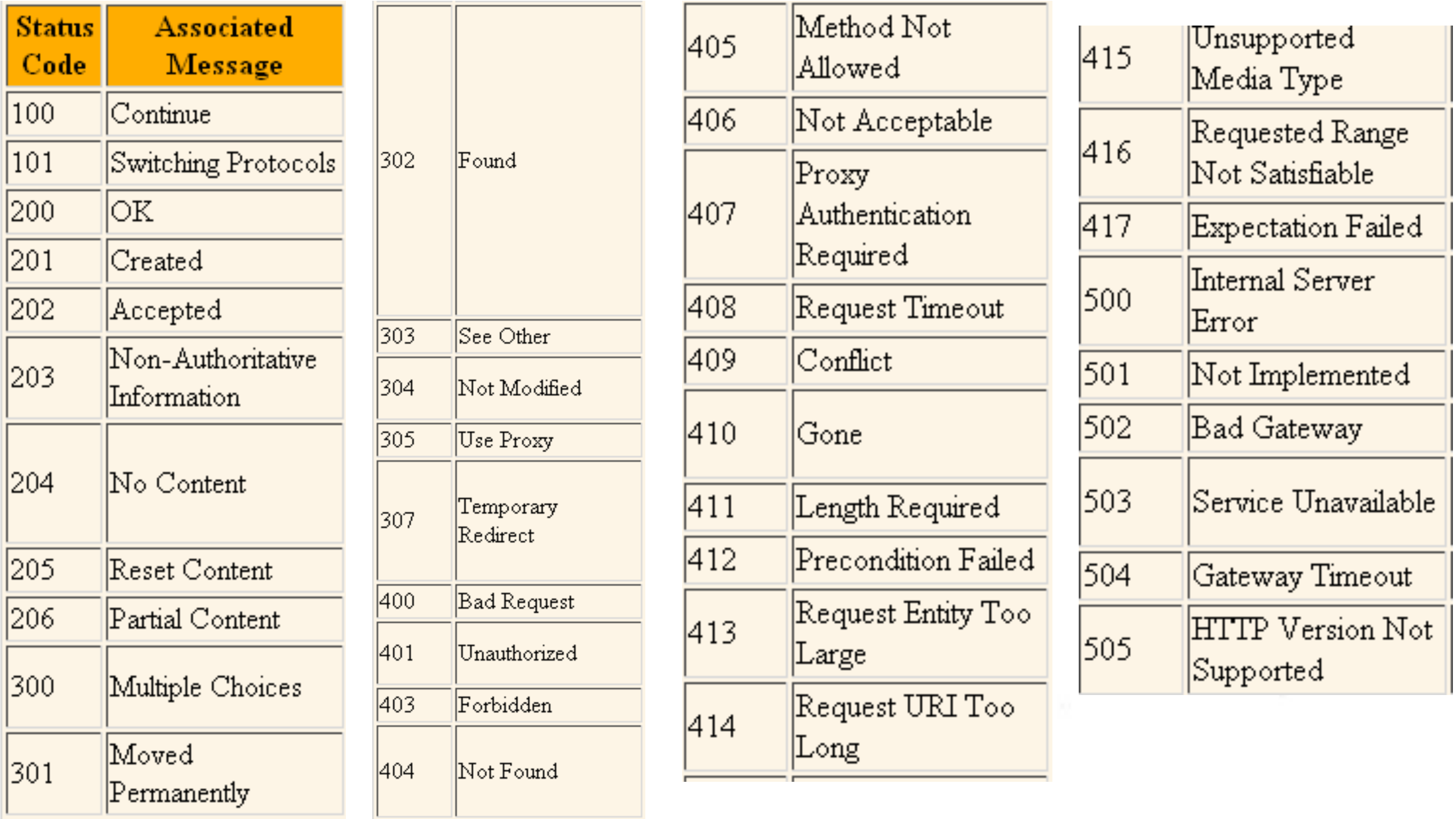
Java Servlet Technology Intro – Response Codes and Errors



```
    String url =
      response.encodeURL(searchSpec.makeURL(searchString,
                                             numResults));
    response.sendRedirect(url);
    return;
  }
}
response.sendError(response.SC_NOT_FOUND,
                   "No recognized search engine specified.");
```

1. Java Servlet Technology

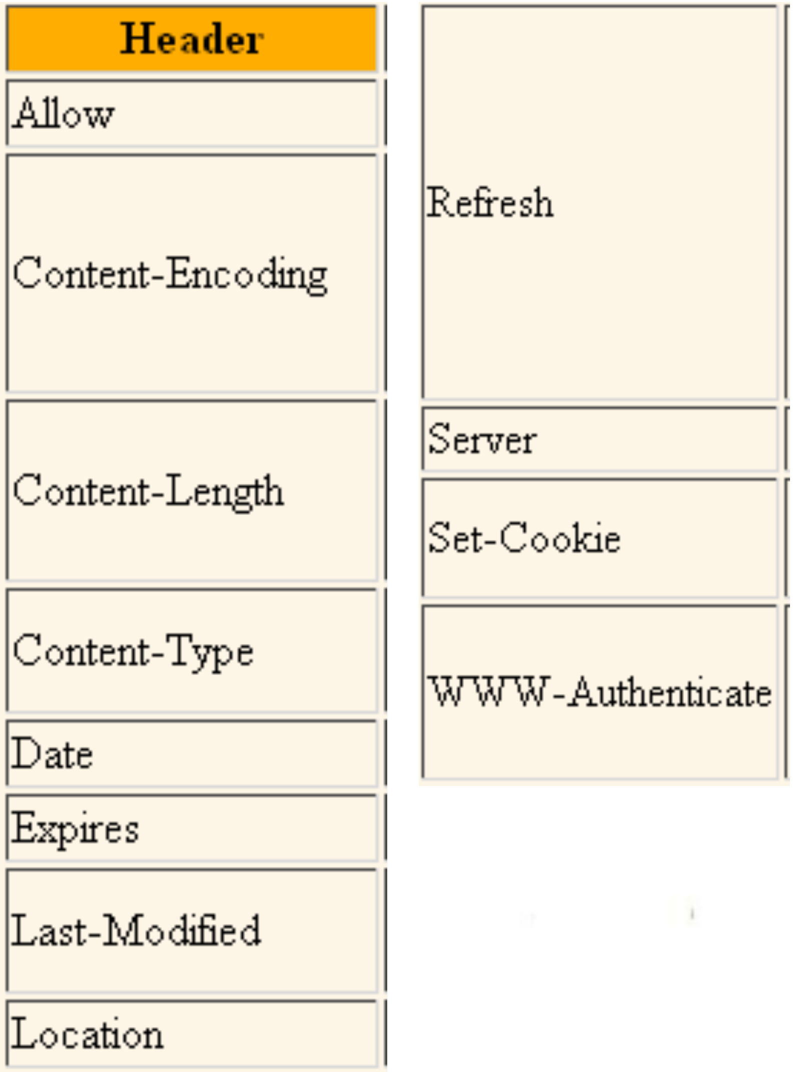
Java Servlet Technology Intro



Status Code	Associated Message
100	Continue
101	Switching Protocols
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request URI Too Long
415	Unsupported Media Type
416	Requested Range Not Satisfiable
417	Expectation Failed
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version Not Supported

1. Java Servlet Technology

Java Servlet Technology Intro – Response Headers



Header
Allow
Content-Encoding
Content-Length
Content-Type
Date
Expires
Last-Modified
Location
Refresh
Server
Set-Cookie
WWW-Authenticate

1. Java Servlet Technology

Java Servlet Technology Intro – Cookies and Session Tracking

There are tech issues with HTTP, because it is a ”stateless“ protocol.

Usually this may be solved as it follows:

1. Cookies. Most used way to transform HTTP from “stateless” to “state-full”. The objects associated to the cookie are NOT going through the network and are stored on the web server side.

2. URL Rewriting. For each HTTP request there is attached in the end or the URL an unique char string generated by the web server.

3. Hidden form fields. Are used HTML tags such as:

<INPUT TYPE="HIDDEN" NAME="session" VALUE="...">

1. Java Servlet Technology

Java Servlet Technology Intro – Cookies and Session Tracking

```
//create cookie 1 - implicit value in seconds of cookie is within the session
Cookie userCookie = new Cookie("CookieGigel", "CucuBau");
response.addCookie(userCookie);

//create cookie 2 - is per year
Cookie userCookie2 = new Cookie("CookieIon", "IONIONION");
userCookie2.setMaxAge(SECONDS_PER_YEAR);
response.addCookie(userCookie2);


…

Cookie[] cookies = request.getCookies();
if (cookies != null) {

     for(int i=0; i<cookies.length; i++) {
      Cookie cookie = cookies[i];
      if ("CookieGigel".equals(cookie.getName())) {…
```

1. Java Servlet Technology

Java Servlet Technology Intro – Cookies and Session Tracking

```
public void processRequest(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {

    HttpSession session = request.getSession(true);
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String title = "Show Session"; String heading;
    Integer accessCount = new Integer(0);

    if (session.isNew()) {
      heading = "Welcome, Newcomer";
    } else {
      heading = "Welcome Back";
      Integer oldAccessCount =(Integer)session.getAttribute("accessCount");
      if (oldAccessCount != null) {
        accessCount = new Integer(oldAccessCount.intValue() + 1);
      }
    }
    session.setAttribute("accessCount", ""+accessCount);
```

Section Conclusion

Fact: **DAD needs Web Programming**

In few **samples** it is simple to remember: Java Servlet Programming with HTTP protocol analysis in real time for request headers, responses’ codes and headers, session tracking – generates standards HTML pages as entering gate for distributed computing and systems.

2

Java Server Page – JSP & Java Servlet Technology Intro

Communicate & Exchange Ideas

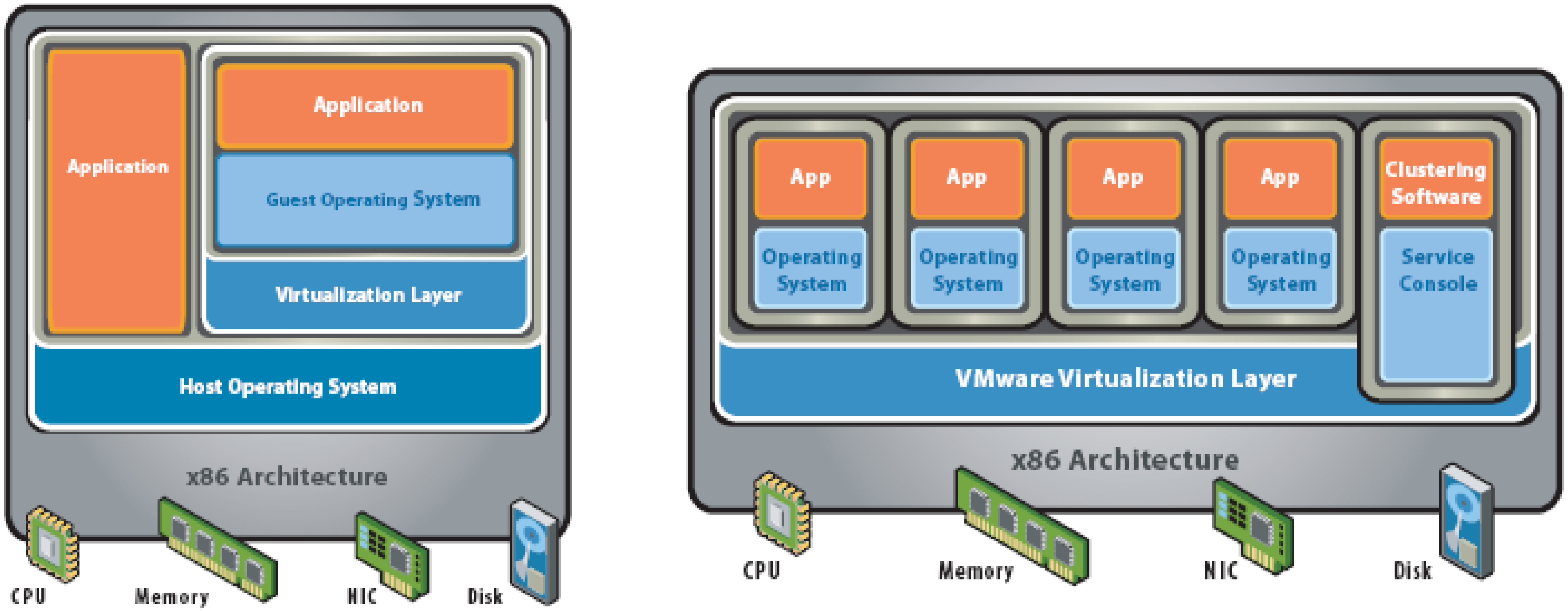
Questions & Answers!

But wait…

There’s More!

Distributed Systems: Cloud Architecture

Cloud Concepts – Intro – Virtualization Overview



Hosted Architecture

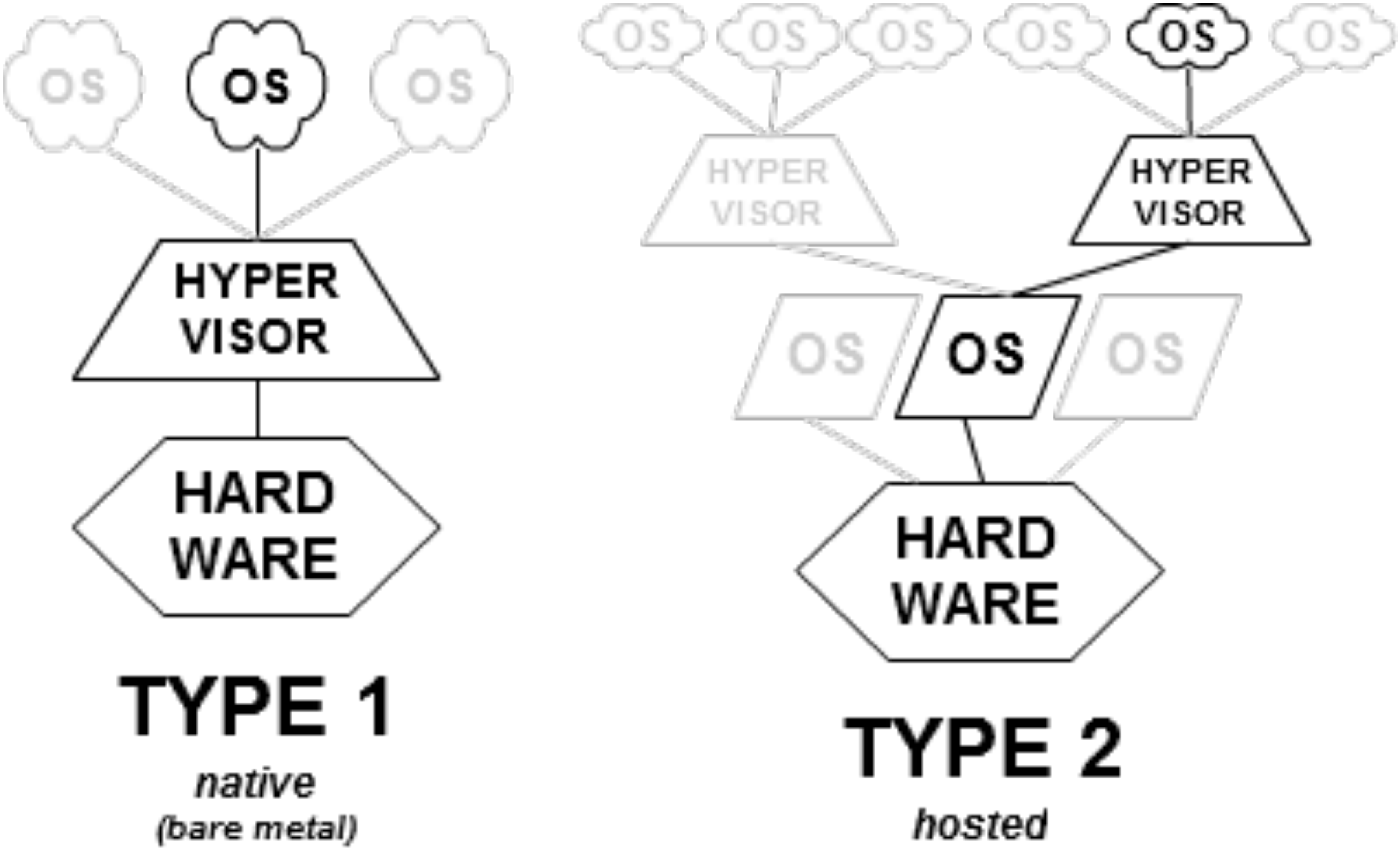
- Installs and runs as an application
- Relies on host OS for device support and physical resource management

Bare-Metal (Hypervisor) Architecture

- Lean virtualization-centric kernel
- Service Console for agents and helper applications

Figure 2: Virtualization Architectures

Cloud Concepts – Intro – Virtualization Overview

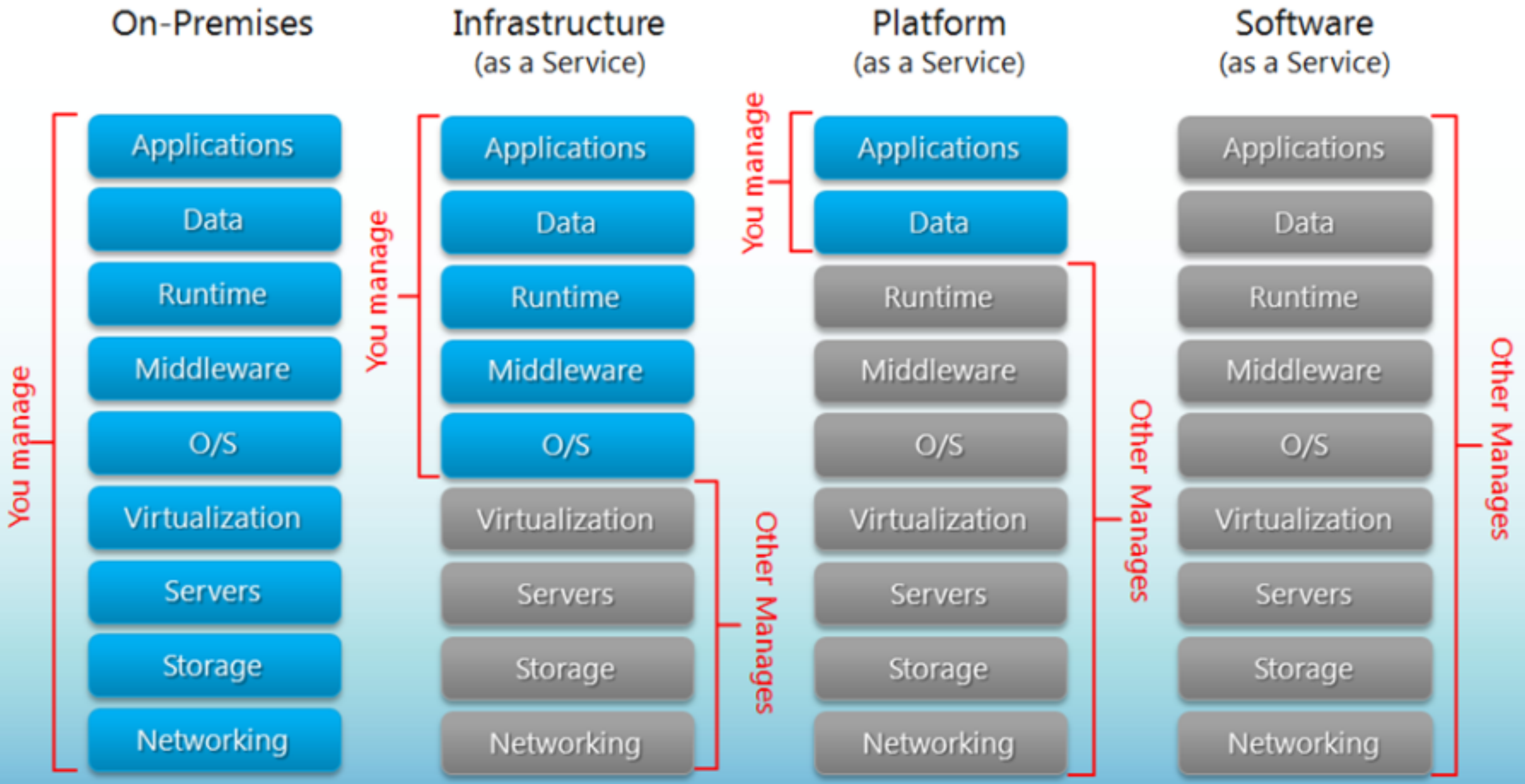


Copyright to *www.wikipedia.org*

Distributed Systems: Cloud

Separation of Responsibilities

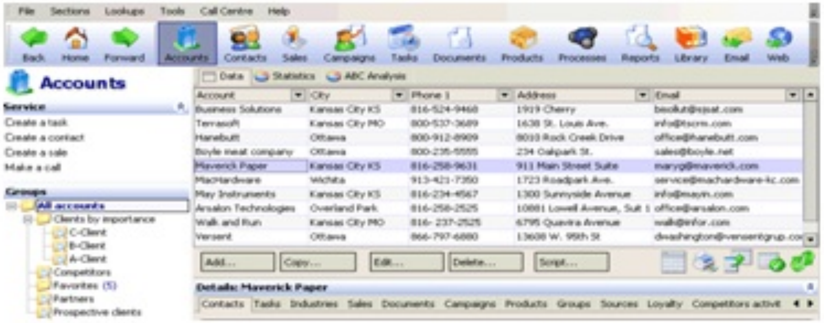
Cloud Concepts – Intro – IaaS, PaaS, SaaS



Copyright to
http://blogs.technet.com/b/kevinremde/archive/2011/04/03/saas-paas-and-iaas-oh-my-quot-cloudy-april-quot-part-3.aspx

Cloud Concepts – Intro – IaaS, PaaS, SaaS

SaaS



Accounts

Service

- Create a task
- Create a contact
- Create a sale
- Make a call

Groups

- All accounts
  - Clients by importance
    - C-Client
    - B-Client
    - A-Client
  - Competitors
  - Favorites (5)
  - Partners
  - Prospective clients

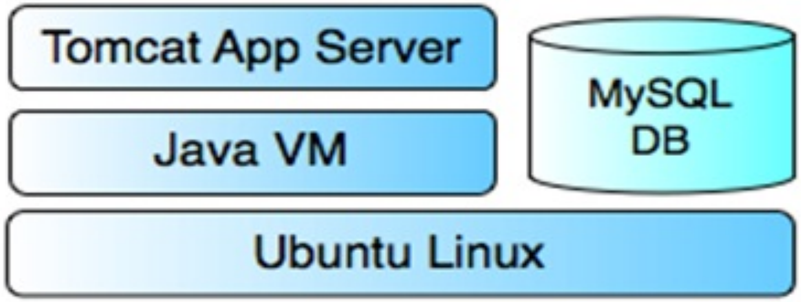
Data | Statistics | ABC Analysis

Account	City	Phone 1	Address	Email
Business Solutions	Kansas City KS	816-524-9460	1919 Cherry	bsolut@sspat.com
Terrasoft	Kansas City MO	800-537-3689	1638 St. Louis Ave.	info@tscrm.com
Hanebutt	Ottawa	800-912-8909	8010 Rock Creek Drive	office@hanebutt.com
Boyle meat company	Ottawa	800-235-5555	234 Oakpark St.	sales@boyle.net
Maverick Paper	Kansas City KS	816-258-9631	911 Main Street Suite	maryg@maverick.com
MacHardware	Wichita	913-421-7350	1723 Roadpark Ave.	service@machardware-kc.com
May Instruments	Kansas City KS	816-234-4567	1300 Sunnyside Avenue	info@mayin.com
Arsalon Technologies	Overland Park	816-258-2525	10881 Lowell Avenue, Suit 1	office@arsalon.com
Walk and Run	Kansas City MO	816- 237-2525	6795 Quavira Avenue	walk@infor.com
Versent	Ottawa	866-797-6880	13608 W. 95th St	dwashington@versentgrup.co

Details: Maverick Paper

SalesForce.com, Google Apps

PaaS



Tomcat App Server

Java VM

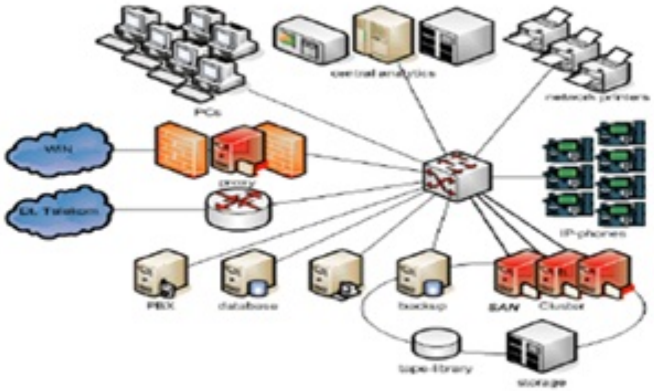
MySQL DB

Ubuntu Linux

Google App Engine for:
Java, Ruby, Python & GO

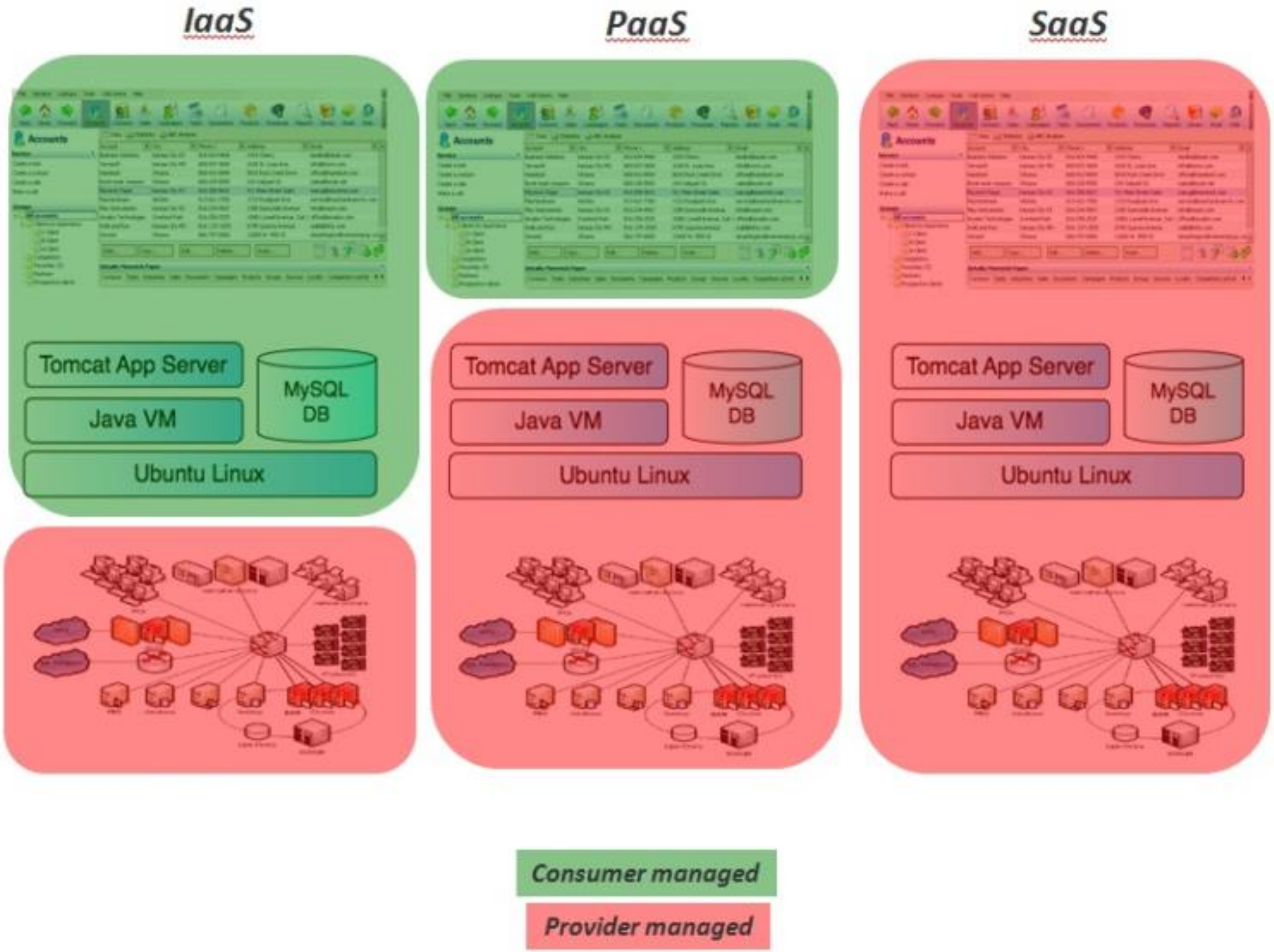
VMForce.com, MS Azure

IaaS



vCloud Express/Datacenter,
Amazon EC2

Cloud Concepts – Intro – IaaS, PaaS, SaaS



Copyright to
http://it20.info/2010/11/random-thoughts-and-blasphemies-around-iaas-paas-saas-and-the-cloud-contract/

Cloud Concepts – Intro – using Google App Engine Cloud PaaS

https://cloud.google.com/appengine/docs/standard/java/quickstart

https://ssh.cloud.google.com/cloudshell/editor

```
# secitc@cloudshell:~$ gcloud projects create wserv001
### it must be created from Web GUI:
```

https://console.cloud.google.com/projectcreate?previousPage=%2Fprojectselector2%2Fbilling%3Flang%3Djava%26st%3Dtrue%26_ga%3D2.187602464.925220531.1582115517-284239323.1582115517%26pli%3D1&project=&folder=&organizationId=0

```
secitc@cloudshell:~$ gcloud projects describe wserv0001

createTime: '2020-02-19T12:38:54.623Z’
lifecycleState: ACTIVE
name: wserv001
projectId: wserv001projectNumber: '163714956555'
```

Cloud Concepts – Intro – using Google App Engine Cloud PaaS

```
secitc@cloudshell:~$ gcloud app create --project=wserv0001
Please enter your numeric choice:  6
Please choose the region where you want your App Engine application located:
[1] asia-east2      (supports standard and flexible) [2] asia-northeast1 (supports standard and flexible) [3] asia-northeast2 (supports standard and flexible) [4] asia-south1    (supports standard and flexible) [5] australia-southeast1 (supports standard and flexible) [6] europe-west   (supports standard and flexible) …
Creating App Engine application in project [wserv0001] and region [europe-west]....done.Success! The app is now created. Please use `gcloud app deploy` to deploy your first app.

secitc@cloudshell:~$ git clone https://github.com/GoogleCloudPlatform/getting-started-java.git
secitc@cloudshell:~$ pwd
/home/secitc
secitc@cloudshell:~$ ls
getting-started-java  README-cloudshell.txt
```

Cloud Concepts – Intro – using Google App Engine Cloud PaaS

```
secitc@cloudshell:~$ cd getting-started-java/appengine-standard-java8/helloworld

# localhost
secitc@cloudshell:~$ mvn package appengine:run -DprojectId=wserv0001

# real instance
secitc@cloudshell:~$ gcloud config set project wserv0001
# when is finished:
secitc@cloudshell:~$ gcloud config unset project

# modify wserv0001 in pom.xml and then:
secitc@cloudshell:~$ mvn package appengine:deploy

# https://wserv0001.appspot.com
# https://wserv0001.appspot.com/hello
secitc@cloudshell:~$ gcloud app browse
```

Thanks!

DAD – Distributed Application Development
End of Lecture 4 – Section 3