



פיתוח מערכות תוכנה מבוססות Java

# Web Applications

אוהד ברזילי

# מה זה?

- לא ברור
- אנחנו נדון במקרים הבאים:
  - "אתר אינטרנט דינאמי"
  - הדפדפן הוא הלקוח
  - Front End לאפליקציה קיימת או האפליקציה כולה

# רלוונטיות

## ■ תחום בהתהוות

- ריבוי כלים וסטנדרטים

- היווצרות של תת תחומים (לדוגמא: רשתות חבריות)

## ■ תחום פעיל מאוד

- קהילת מפתחים

- קהילת מפתחי כלים

- בגלל בוסריות חלק מהכלים מפתחי האתרים גם מפתחים את כלי הפיתוח (קהילת משפצרים)

# Porting Desktop Applications to the Web - Motivation

## ■ נגישות מכל מקום

- דוגמאות: תיבת דוא"ל

- אלטרנטיבות: DiskOnKey, Laptop, Remote Login

## ■ שיתוף

- דוגמאות: אלבומי תמונות, כתיבת מסמכים משותפת

- אלטרנטיבות: שליחת קבצים בדואר

## ■ קהילתיות

- דוגמאות: Forums, Wiki, OpenSource Projects

- אלטרנטיבות?

# Porting Desktop Applications to the Web - Challenges

## חווית משתמש ■

- Right Click, Drag 'n Drop
- מגמות: Ajax, new standards

## בטיחות ■

- HTTP is Firewall Breaker
- <http://video.google.com/videoplay?docid=2575564563023304756&q=engedu+crime>
- <http://video.google.com/videoplay?docid=5159636580663884360&q=engedu%20crime&hl=en>

# Porting Applications to the Web - trends

■ גרסת Web המחליפה client שאינו HTTP

■ לדוגמא: MSN Messenger

■ לדוגמא: Skype vs. Jajah

■ לכולם יש כבר דפדפן מותקן וגלישה באינטרנט

■ הבעיה של תוכנות שיתוף

■ גרסת Web המשלימה גרסת Desktop

■ לדוגמא: Picasa + Picasa Web

■ הפרד ומשול

# Web Technologies Trends

---

## ■ Server Side

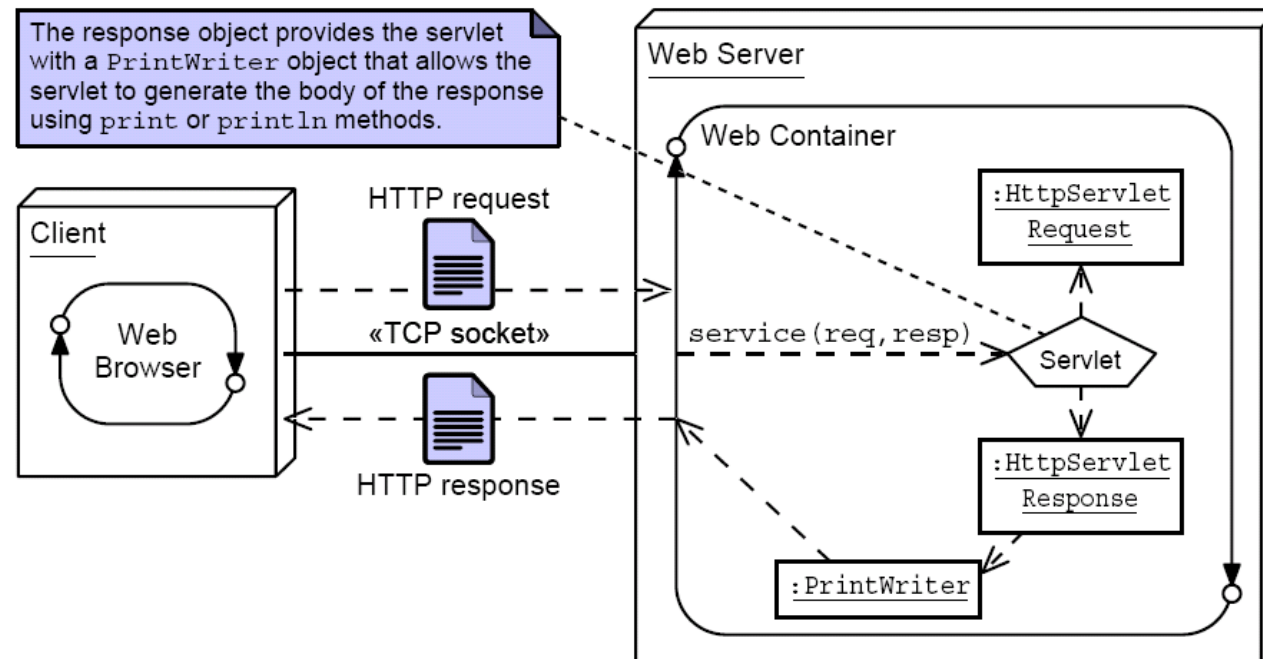
- CGI, Servlets
- ASP, JSP, PHP
- Frameworks

## ■ Client Side

- JavaScript, Ajax
- Google's GWT
- Adobe (Flash, ActionScript, Flex, AIR)
- Microsoft's Silverlight

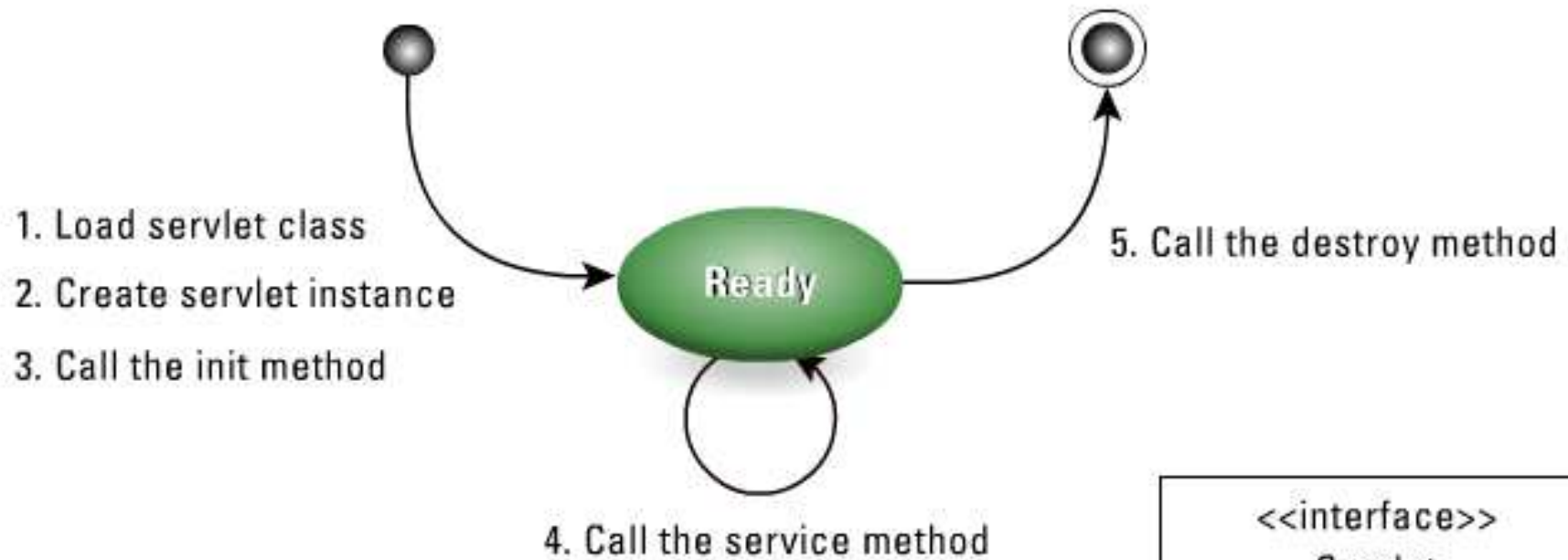
# Server Side Web Technologies

- הוספת לוגיקה דינאמית לשרת אינטרנט
- שרות doGet של מחלקת Servlet המותקנת ב J2EE Container מממש את הלוגיקה העסקית
- תקשורת, ניהול חוטים, הראשות, מימוש HTTP, תצורה ועוד מנהלים ע"י ה Container אשר מספק Java API להבטים אלו





# Servlet Life Cycle



<code>&lt;&lt;interface&gt;&gt;</code> <code>Servlet</code>
<code>init(ServletConfig)</code> <code>service(req,resp)</code> <code>destroy( )</code>

# Servlet API

---

- Parameter Handling
- Initialization and configuration
- Filters (aspects)
  - Authorization and Authentication
  - Profiling
- Session Management
  - Cookies
  - URL Rewriting

# Hello World Servlet

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloWorld extends HttpServlet {

    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws IOException, ServletException
    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        out.println("<head>");
        out.println("<title>Hello World!</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Hello World!</h1>");
        out.println("</body>");
        out.println("</html>");
    }
}
```

# Java / HTML

- על אף שרצינו שפלט התוכנית יהיה ב HTML איננו רוצים לערבב קוד Java וקוד HTML
  - Model/View Separation
  - לפעמים נכתבים ע"י מתכנתים שונים
- יש צורך בטכנולוגיה שתאפשר פיתוח אורתוגונלי של המודל ושל ההצגה (גם אם בפועל הקוד שירוצ ממזג את שניהם)
- טכנולוגיית JSP מתיימרת להשיג מטרה זו בדיוק:
  - JSP היא שפת תגיות (בדומה ל html) המאפשרת לשלב בתוכה תוכן דינאמי
  - בזמן ריצה קובץ ה JSP עובר קומפילציה לקובץ Servlet

# Servlet הוא JSP

```
jEdit
File Edit Search Markers Folding View Utilities Macros Plugins Help
10 public class HelloServlet extends HttpServlet {
11
12     private static final String DEFAULT_NAME = "World";
13
14     public void doGet(HttpServletRequest request,
15         HttpServletResponse response)
16         throws IOException {
17         generateResponse(request, response);
18     }
19
20     public void doPost(HttpServletRequest request,
21         HttpServletResponse response)
22         throws IOException {
23         generateResponse(request, response);
24     }
25
26     public void generateResponse(HttpServletRequest request,
27         HttpServletResponse response)
28         throws IOException {
29
30         String name = request.getParameter("name");
31         if ( (name == null) || (name.length() == 0) ) {
32             name = DEFAULT_NAME;
33         }
34
35         response.setContentType("text/html");
36         PrintWriter out = response.getWriter();
37
38         out.println("<HTML>");
39         out.println("<HEAD>");
40         out.println("<TITLE>Hello Servlet</TITLE>");
41         out.println("</HEAD>");
42         out.println("<BODY BGCOLOR='white'>");
43         out.println("<B>Hello, " + name + "</B>");
44         out.println("</BODY>");
45         out.println("</HTML>");
46
47         out.close();
48     }
}
```

```
jEdit
File Edit Search Markers Folding View Utilities Ma
1 <%! private static final String DEFAULT_NAME = "World"; %>
2
3 <html>
4
5 <head>
6 <title>Hello JavaServer Page</title>
7 </head>
8
9 <!-- Determine the specified name (or use default) -->
10 <%
11     String name = request.getParameter("name");
12     if ( (name == null) || (name.length() == 0) ) {
13         name = DEFAULT_NAME;
14     }
15 %>
16
17 <body bgcolor='white'>
18
19 <b>Hello, <%= name %></b>
20
21 </body>
22
23 </html>
24
```

# JSP / Java

## ■ בעיות:

- כעת במקום לזהם קוד Java ב HTML אנו מזהמים קוד JSP ב Java
- לתחביר הסטנדרטי של JSP יכולות מוגבלות (בעיקר שילוב קוד HTML ו Java)
- יש צורך בטכנולוגיה שתאפשר פיתוח לוגיקה ב Java ושיבוצה בתוך קובצי JSP ללא זיהום ה JSP

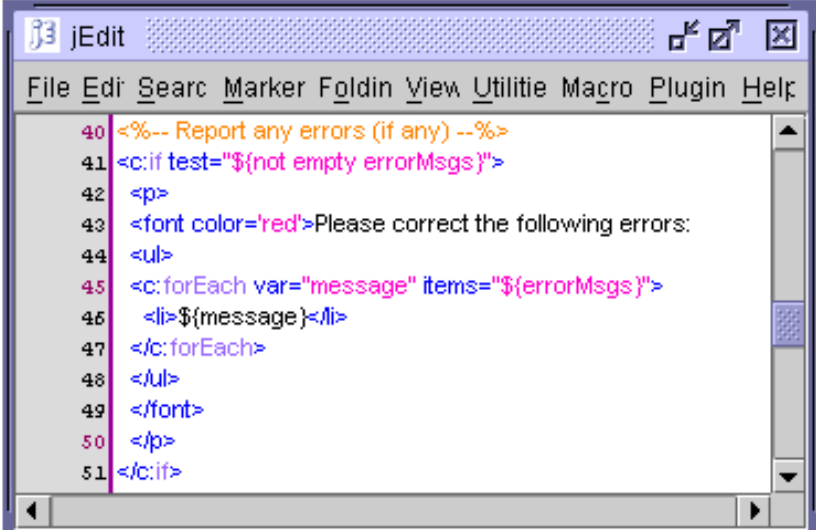
## ■ תגיות JSP מתיימרות להשיג מטרה זו בדיוק – המתכנתת יכולה להגדיר בעצמה תגיות JSP

- התגיות ימומשו בשפת Java בקובץ נפרד
- הדבר הופך את JSP לשפת תכנות לכל דבר עם מבני בקרה, שימוש ב Java Beans ועוד...

# JSP Custom Tags

```
<!-- Report any errors (if any) -->
<%
  // Retrieve the errorMsgs from the request-scope
  List errorMsgs = (List) request.getAttribute("errorMsgs");
  if ( (errorMsgs != null) && !errorMsgs.isEmpty() ) {
%>
<p>
<font color='red'>Please correct the following errors:
<ul>
<%
  Iterator items = errorMsgs.iterator();
  while ( items.hasNext() ) {
    String message = (String) items.next();
%>
    <li><%= message %></li>
<%
  } // END of while loop over errorMsgs
%>
</ul>
</font>
</p>
<%
  } // END of if errorMsgs is not empty
%>
```

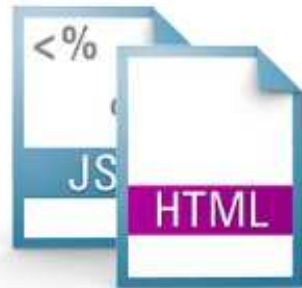
■ בעזרת תגיות משתמש  
ניתן להשתמש  
במעצבים ללא ידע  
בתכנות לצורך כתיבת  
עמודי JSP



```
jEdit
File Edit Search Marker Foldin View Utilitie Macro Plugin Help
40 <!-- Report any errors (if any) -->
41 <c:if test="{not empty errorMsgs}">
42 <p>
43 <font color='red'>Please correct the following errors:
44 <ul>
45 <c:forEach var="message" items="{errorMsgs}">
46 <li>${message}</li>
47 </c:forEach>
48 </ul>
49 </font>
50 </p>
51 </c:if>
```

# Job Roles

■ ארכיטקטורת Web Applications מחזקת את הצורך לחלק את משימות הפיתוח בין מתכנתים שונים



- Web Designer – Creates view elements (web pages and layouts)



- Web Component Developer – Creates controller elements



- Business Component Developer – Creates model elements



- Data Access Developer – Creates database access elements

Copyright 2005 Sun Microsystems, Inc. All Rights Reserved.



# JSP vs. ASP

- JSP אינו רעיון חדש
- קדם לו ASP של מייקרוסופט (1996) שהיה מבוסס על שפות תסריט כגון VBScript (בעיקר) ו-Jscript

```
<html>  
<body>  
<%= "Hello World!" %>  
</body>  
</html>
```

- החל מ-2002 מחליפה ASP.net את ASP עם תמיכה מלאה של כל שפות .NET.

```
<html>
<head>
  <title>PHP Test</title>
</head>
<body>
  <?php echo '<p>Hello World</p>'; ?>
</body>
</html>
```

# PHP

*PHP: Hypertext Preprocessor* ■

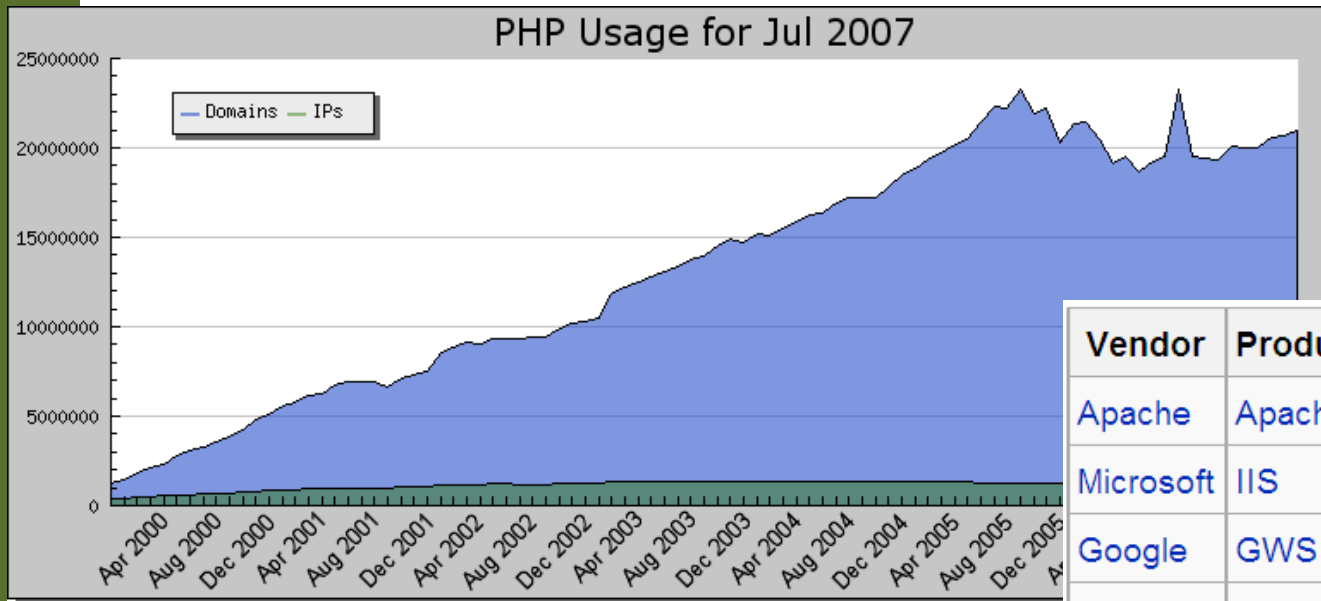
התחילה כשפת סקריפט פשטנית בשנת 1995 ■

כרגע פעילות גרסאות 4 ו-5 אשר מוסיפות לשפה יכולות OO (כמעט מלאות), תמיכה ב XML, ב web services ועוד... ■

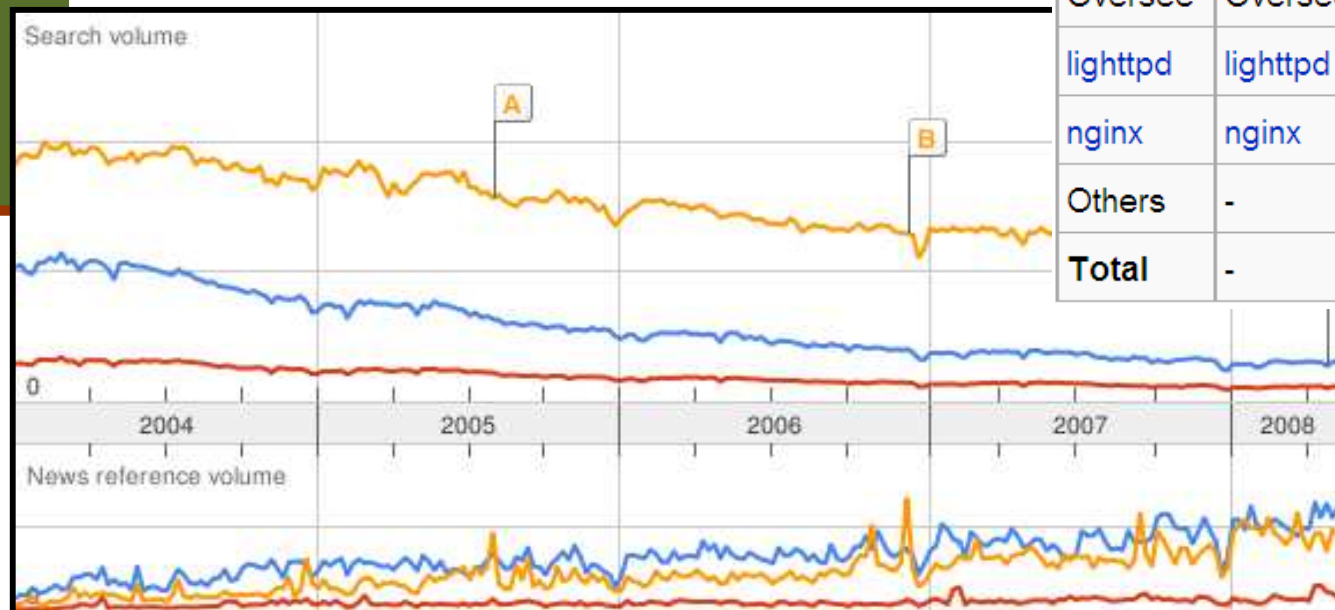
ניתנת להורדה כחלק מחבילת xAMP ■  
(**A**pache, **M**ySQL and **P**HP) ■

כנראה השפה הפופולרית ביותר לצד השרת (40% משרתי Apache: <http://www.appideas.com/technology.html>) ■

# PHP, ASP, JSP and Web Servers



Vendor	Product	Web Sites Hosted	Percent
Apache	Apache	83,206,564	50.22%
Microsoft	IIS	58,540,275	35.33%
Google	GWS	10,075,991	6.08%
Oversee	Oversee	1,926,812	1.16%
lighttpd	lighttpd	1,495,308	0.9%
nginx	nginx	1,018,503	0.61%
Others	-	9,432,775	5.69%
<b>Total</b>	-	<b>165,696,228</b>	<b>100.00%</b>



# Scripting languages and Server Side

■ בחלק מהמקרים הלוגיקה של השרת הדינאמי פשוטה למדי ומסתכמת בשליפה והצגה של מידע מתוך בסיס נתונים

■ במקרים כאלה ניתן לשקול שימוש בשפות תסריט שאינן יעודיות לשימוש בשרתים

■ דוגמאות:

■ Perl

■ Ruby

■ python

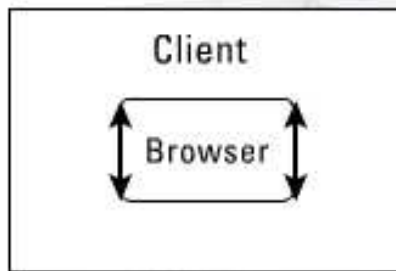


# Frameworks

# Model 2 Framework

## Popular Model 2 Frameworks:

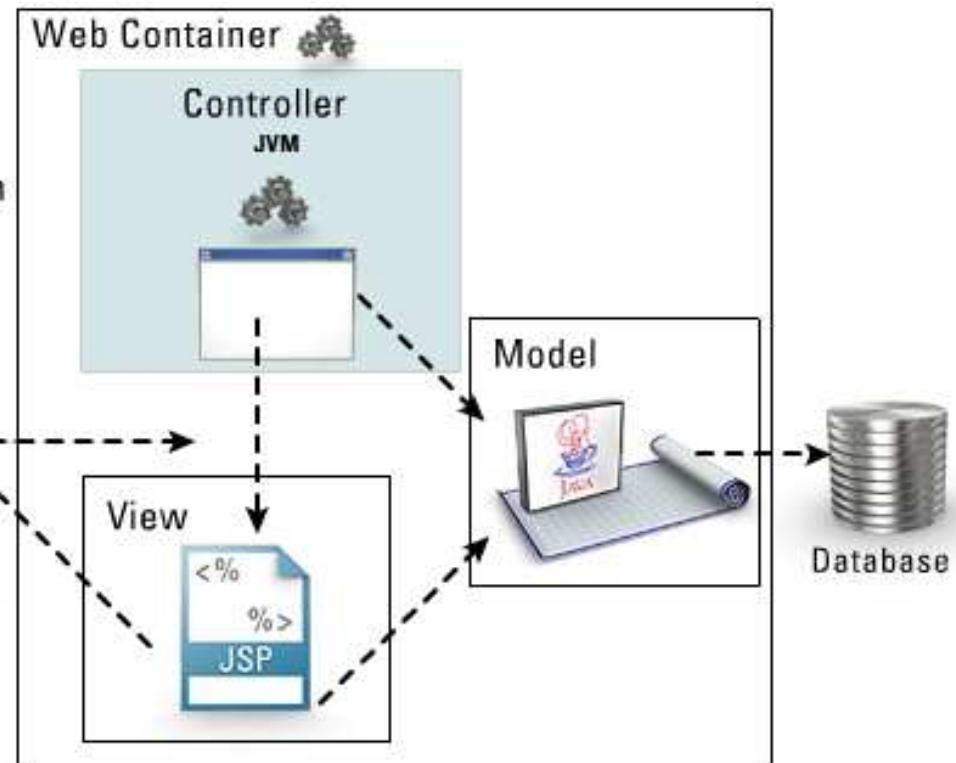
- Struts, part of the Jakarta project
- JavaServer™ Faces technology from Sun
- Velocity from Apache



<< HTTP >>

**This combination provides the most advantages:**

- Fast
- Powerful
- Easy to create
- Cross-platform
- Scalable
- Maintainable



# Types of View Components



## Data Presentation:

- Includes any data presented on a visible screen
- There are many forms of presentation including graphs, spreadsheets, listings, and so on



## Data Forms:

- Forms for data entry are also considered view components



## Navigational Aids:

- This type of view includes menus, hyperlinks, site maps, and so on



## Informational Screens or Pop-Ups:

- This type of view includes welcome text, instructions, help screen, error messages, confirmation dialogs, and so on

# Types of Controller Components



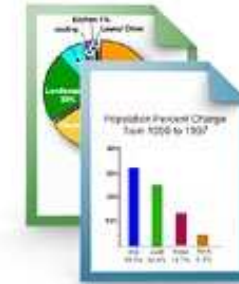
## Process input from a user

- Enter text data
- Select radio or checkbox buttons
- Drop-down lists
- Use a submit button to send the data



## Screen navigation

- Hyperlinks in web pages



## Preparing data for view components

- Reports present business data in a view

### Form-processing controller servlets usually perform the following tasks:

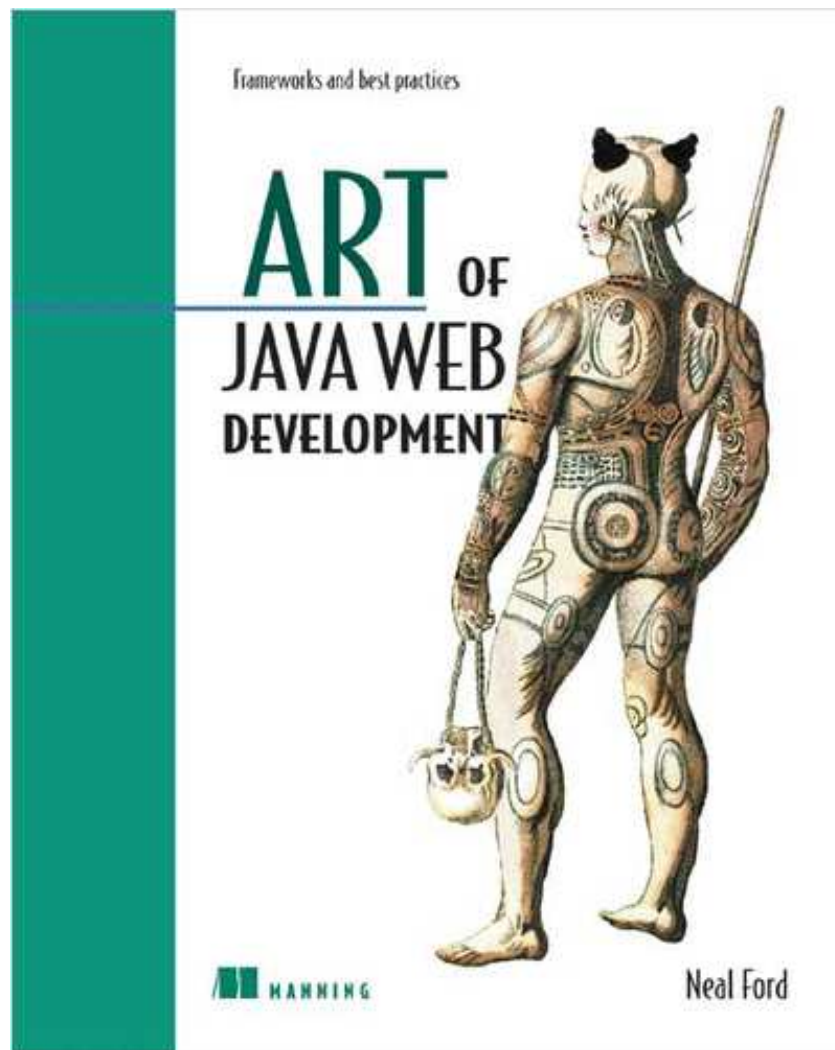
1. Retrieve form parameters from the HTTP request
2. Perform any data conversion on the form parameters
3. Verify the form parameters
4. Execute the business logic
5. Dispatch to the next view components based on the results of the previous steps



# MVC Using Struts

- מימוש רעיון ה MVC בסביבת Web Container מכיל פעולות שחוזרות על עצמן בכל יישום:
  - ה- Controller מבצע את בדיקת תקינות הנתונים
  - ה- Controller מפנה את הבקשה עם תוצאות הבדיקה לעמוד ההצגה המתאים
- היינו מעוניינים לבצע אוטומטיזציה של פעולות אלו, בדרך זו ה Controller יתרכז רק בלוגיקה העסקית וההפרדה בין View ובין Controller תאכף ע"י הסביבה
- Struts Framework מספקת את המחלקה Action אשר עוטפת את פעולת מחלקות ה- Controller
  - בדרך זו ה Controller אפילו לא יודע את שם המחלקה שאליה מופנית ההודעה לאחר ביצוע הלוגיקה העסקית

# Java Web Frameworks



קיימות כמה Frameworks המספקות תשתית לפיתוח יישומי אינטרנט מעל Servlets

ה Frameworks השונות מסייעות לאכוף הרגלי עבודה נכונים (כגון הפרדה בין מודל ותצוגה)

הספר

Art of Java Web Development  
*Struts, Tapestry, Commons, Velocity, JUnit, Axis, Cocoon, InternetBeans, WebWork*  
Neal Ford

סוקר חלק מהסביבות השונות, עומד על ההבדלים ביניהן ומשווה בין קוד "ידני" ובין הקוד שנכתב בעזרתן

# Web Frameworks

## Struts 2 (Struts + Webwork)

### ■ *Build!*

- **Easy startup** - Jumpstart new projects with our bootstrap tutorial and template application or Maven archetype.
- **Improved Design** - Code clean against HTTP-independant framework interfaces.
- **Enhanced Tags** - Code less with stylesheet-driven form tags that provide their own markup.
- **Stateful Checkboxes** - Avoid special handling with smart checkboxes that know when they are toggled.
- **Flexible Cancel Buttons** - Go directly to a different action on cancel.
- **First-class AJAX support** - Add interactivity and flexibility with AJAX tags that look and feel just like standard Struts tags.
- **Easy Spring integration** - Inject dependencies into Actions using Spring without glue code or red tape. (Plexus support also available.)
- **Enhanced Results** - Do more with speciality results for JasperReports, JFreeChart, Action chaining, and file downloading.
- **POJO forms** - No more ActionForms! Use any JavaBean to capture form input or put properties directly on an Action class. Use both binary and String properties!
- **POJO Actions** - Use any class as an Action class -- even the interface is optional!

### ■ *Deploy!*

- **Easy plugins** - Add framework extensions by dropping in a JAR. No manual configuration required! Bundled plugins add support for JavaServer Faces, JasperReports, JFreeChart, Tiles, and more ...
- **Integrated profiling** - Peek inside Struts2 to find where the cycles are going!
- **Precise Error Reporting** - Flip directly to the location and line of an error.

### ■ *Maintain!*

- **Easy-to-test Actions** - Test Struts2 Actions directly, without resorting to mock HTTP objects.
- **Intelligent Defaults** - Skip obvious and redundant settings. Most framework configuration elements have a default value that we can set and forget. Say it once!
- **Easy-to-customize controller** - Customize the request handling per action, if desired. Struts2 only does what you want it to do!
- **Integrating Debugging** - Research problem reports with built-in debugging tools.
- **Easy-to-tweak tags** - Customize tag markup by editing a FreeMarker template. No need to grok the taglib API! *JSP, FreeMarker, and Velocity tags are fully supported.*

<http://struts.apache.org/2.x/index.html>

# Battle of the Frameworks!

- A good framework makes web development easier, faster, and more consistent, but which one should you choose?
  - **Seam** or **Struts 2**?
  - **Zend Framework** or **CakePHP**?
  - **Laszlo** or **Flex 3**?
  - **jQuery** or **Prototype**?
  - **Wicket** or **GWT**?
  - **Rails 2** or **Merb**?
  - **Adobe AIR** or **Silverlight**?
  - Maybe you're even considering **Second Life**
- Not only you should choose a framework you should choose the underlying technology
- (<http://phillyemergingtech.com/abstracts.php#frameworks>)

# כלים וסביבות

- כאשר טכנולוגיה הופכת לפופולרית פועלים מספר כוחות:
  - כח אדם פחות מיומן צריך **כלים תומכים** כדי להשתלב במאמץ הפיתוח
  - מתגלים **שימושים חדשים** לטכנולוגיה אשר לא נתמכים בה היטב מכיוון שלא נלקחו בחשבון בתחילה
- כדי לענות על הדרישה מתפתחים כלים, סביבות ושפות ייעודיות אשר מנסים לפשט את משימות הפיתוח
- לאיזו טכנולוגיה עולה דרוש פיתוח של כלים חדשים היום?
- אילו יכולות על הכלים לספק?

# Higher Level Frameworks and tools

---

- <http://wordpress.org/>
- <http://www.joomla.org/>
- <http://drupal.org/>



# Client Side

# HTML

- שפת תגיות (logical, visual) להצגת עמודי אינטרנט בדפדפן
- XHTML מהווה גשר בינה (HTML 4.01) ובין XML
- עמוד HTML הוא עץ עם שורש יחיד אשר לצאצאיו יכולים להיות מאפיינים (attributes, כולל ארועים)
- מרכיבי העמוד
  - **head**: link, base, script, title, meta
  - **body**: כותרות, פסקאות, עיצוב טקסט, רשימות, טבלאות, תמונות, קישורים, טפסים
- עבודה עם מסגרות
- עבודה עם CSS
- עבודה עם JavaScript



# Dynamic Client

- שפת JavaScript (אין קשר ל Java) היא שפת תסריטים מוטמעת בקוד HTML אשר רצה בדפדפן
- זוהי **שפת תכנות מלאה** עם תמיכה בתנאים, לולאות, משתנים, מערכים, מחרוזות, פונקציות, מחלקות ועצמים, חריגים, ביטויים רגולרים וארועי דפדפן
- מחלקות ספרייה:
  - Object
    - String ,RegExp ,Number ,Function, Math ,Date ,Boolean ,Array ,Arguments
    - Error
      - EvalError,RangeError,URIError, ReferenceError,SyntaxError,TypeError
- התחביר שלה פשוט ומתירני יותר משפות עילויות. למשל אין צורך להצהיר ב JavaScript על טיפוס המשתנים
- בגלל המאבק בין הדפדפנים השונים קיימות מספר גרסאות של השפה:
  - Javascript 1.0, 1.1, ... (Netscape/Mozilla)
  - JScript 1.0, 2.0, ... (Microsoft)
  - ECMAScript v1, v2, ... (International Standard)

# *Hello World*

```
<html>
  <script>
    var x = 3;
    var y = 47;
    alert( " 3 x 47 = " + (x*y) );
  </script>
</html>
```

# שימושי JavaScript

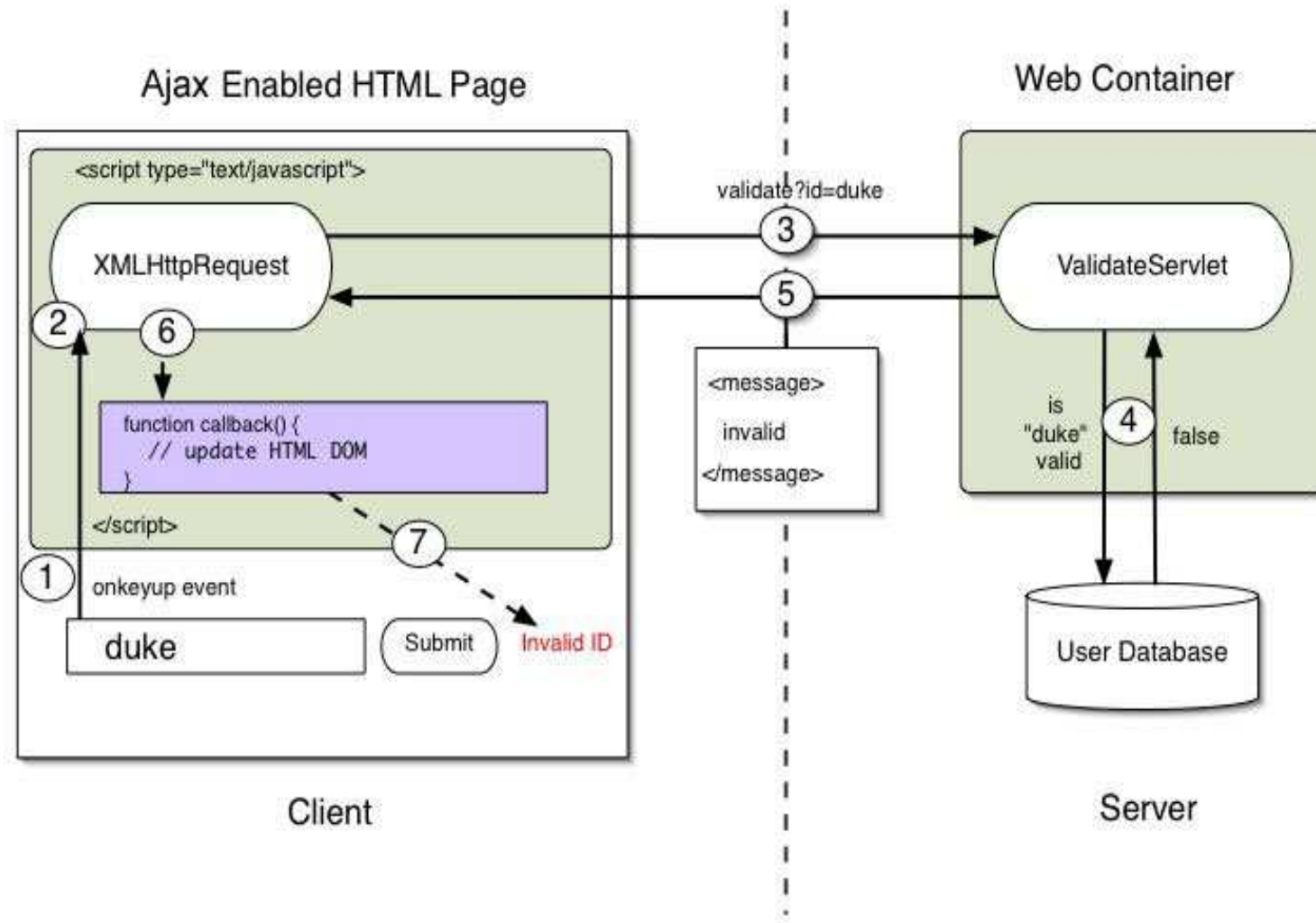
■ אימות טפסים

■ Dynamic GUI

■ האתגר: כאשר המידע נמצא בשרת



# XMLHttpRequest



# Asynchronous JavaScript and XML

■ הודעות ה HTTP הסטנדרטיות עובדות במודל של request-response

■ בעזרת טכנולוגיית Ajax ניתן להוסיף "תת הודעות" בפורמט XML אשר אינן דורשות את רענון העמוד

■ בשונה מהשרת הדינאמי (שמחולל תשובה בצורה דינאמית) ומהלקוח הדינאמי (קוד JavaScript שרץ בדפדפן) Ajax מוסיף עוד מימד של דינאמיות באינטראקציה בין שרת ולקוח

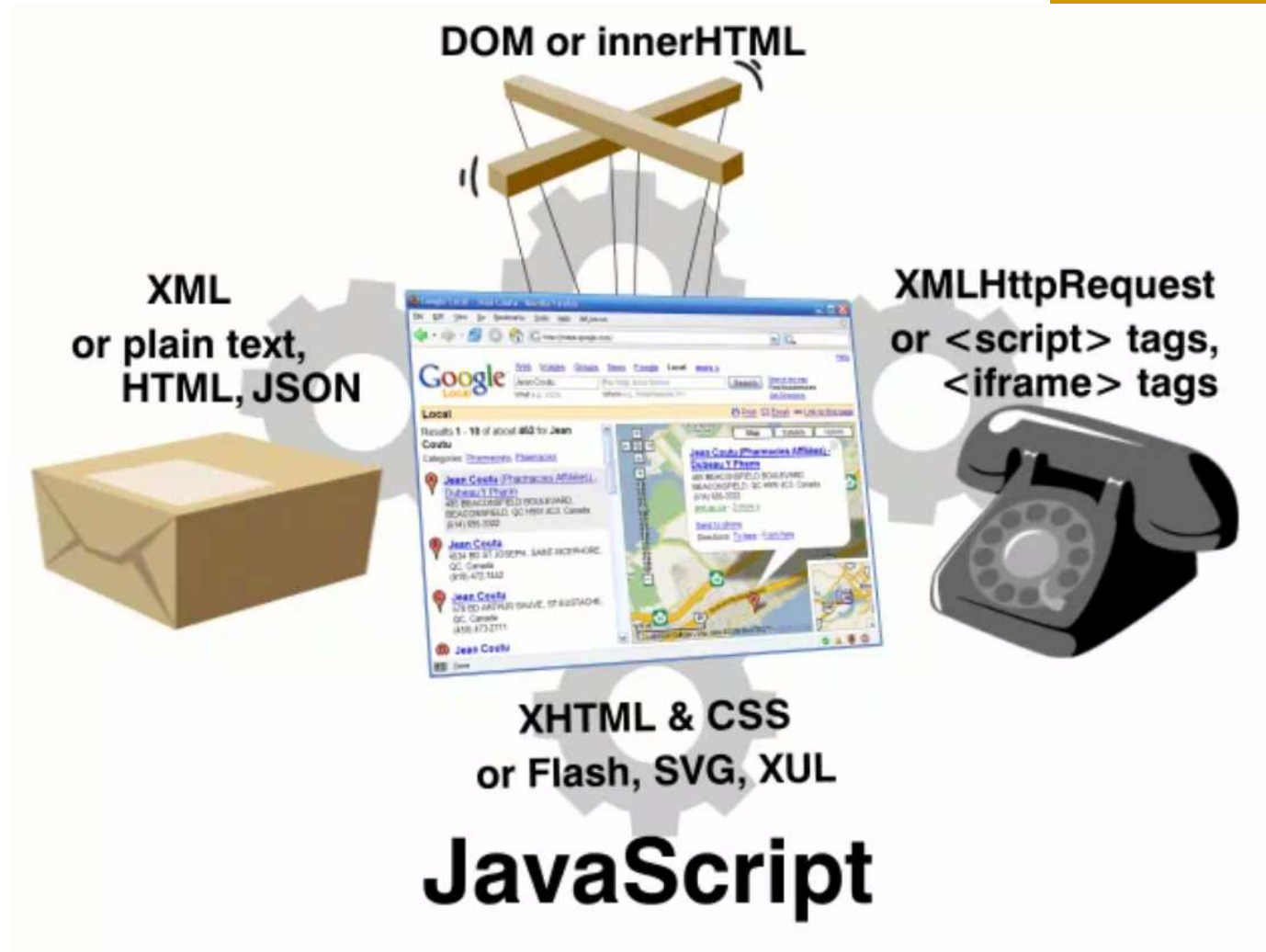
■ דוגמא טובה:

■ <http://www.nettravel.co.il/basic/flights/>

■ דוגמא לא טובה:

■ <http://msdn.microsoft.com/en-us/default.aspx>

# Ajax Fundamentals



# Google Web Toolkit (GWT)

- Google Web Toolkit makes it easier to write high-performance AJAX applications
- You write your front end in the **Java** programming language and **GWT compiles your source** into highly optimized JavaScript.
- Writing web apps today is a tedious and error-prone process. You spend 90% of your time working around **browser quirks**, and JavaScript's lack of modularity makes sharing, testing, and reusing AJAX components difficult and fragile

(<http://code.google.com/webtoolkit/>)

# Google Web Toolkit (GWT)

- **High performance JavaScript.** GWT produces AJAX apps that:
  - Load faster than hand-written JavaScript apps
  - Use smaller, more compact, cacheable code
  - Automatically support IE, Firefox, Mozilla, Safari, and Opera
  - Use the browser's "back" button correctly
- **Better development tools.** Since you're writing in Java, you can use:
  - IDEs that you love like Eclipse, IntelliJ, and NetBeans
  - Full-featured debugging, with variable watches and breakpoints
  - Unit tests (based on JUnit) both in a debugger and in a browser
- **Google APIs and reusable UI components.** GWT comes equipped with useful libraries:
  - Built-in UI components serve as cross-browser building blocks for your app
  - RPC helps with client-server interaction
  - JavaScript Native Interface (JSNI) simplifies integrating GWT code with existing JavaScript code



# Java Applets

■ תגית applet בעמוד HTML יוזמת הורדת של קובץ Java מהשרת והרצתו בדפדפן של הלקוח

■ יתרונות:

■ מימוש לוגיקה עשירה המשובצת בדפדפן

■ חסרונות:

■ תכנות בשפה עילית אינו מתאים למעצבים

■ השפה אינה מותאמת לעיצוב אתרים

■ מדיניות הרשאות בעייתית (תקשורת, משאבים)

# Adobe Technologies

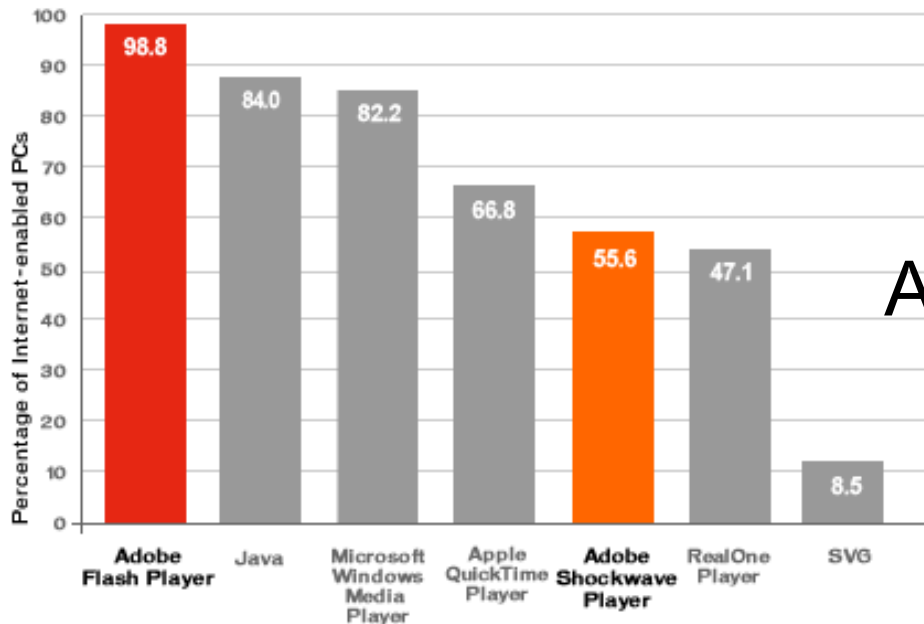
- חברת Adobe מציעה מבחר טכנולוגיות פרטיות (proprietary) Flash לצד הלקוח
- מאוד פופלרית
- הורחבה במספר כיוונים:

■ הוספת לוגיקה תכנותית: ActionScript

■ פיתוח יישומים: Flex

■ ספריות, כלי פיתוח

■ דואליות Web/Desktop: AIR



# Google's Dual Mode

- Applications: Picasa + Picasa Web
- Infrastructure Technology: Google Gears
  - LocalServer Cache and serve application resources (HTML, JavaScript, images, etc.) locally
  - Database Store data locally in a fully-searchable relational database
  - WorkerPool Make your web applications more responsive by performing resource-intensive operations asynchronously
- <http://code.google.com/apis/gears/>

# Microsoft Silverlight

- Microsoft Silverlight is a cross-browser, cross-platform, and cross-device plug-in for delivering the next generation of .NET based media experiences and rich interactive applications for the Web.

(<http://silverlight.net/>)