Bibhas Bhattacharya
CTO, Web Age Solutions
bibhas@webagesolutions.com

www.webagesolutions.com/training

Get Started With HTML5

Browser Support

- Your best bet to experiment with HTML5
  - Safari
  - Chrome
  - Beta: FireFox 4 and IE 9.
- Test your browser:
  - html5test.com
  - html5demos.com
  - www.findmebyip.com/litmus
- JavaScript library to test for HTML5 features
  - www.modernizr.com

As of
October
2010

- Dreamweaver CS5 11.0.3 update will install HTML5 compatibility pack.
- CS4 and CS3 users should download the pack from Dreamweaver Exchange.
- See a demo:
  - www.youtube.com/watch?v=soNIxy2sj0A

DreamWeaver Story

What's New in HTML5?

- New semantic elements
(header, section, footer etc.)
- New form input elements
(e-mail, date, time etc.)
- Audio and video
- CSS3

For web page designers

- The canvas element
- Custom audio & video player
- Geolocation
- Local data storage
- Web SQL & IndexedDB
- Offline apps
- Messaging
- Web worker
- Push using WebSocket

For developers

Basic HTML 5 Document

```html
<!DOCTYPE html>

<html>
<head>
<title>Page Title</title>
<meta charset="UTF-8">
</head>
<body>
<p>Hello World</p>
</body>
</html>
```

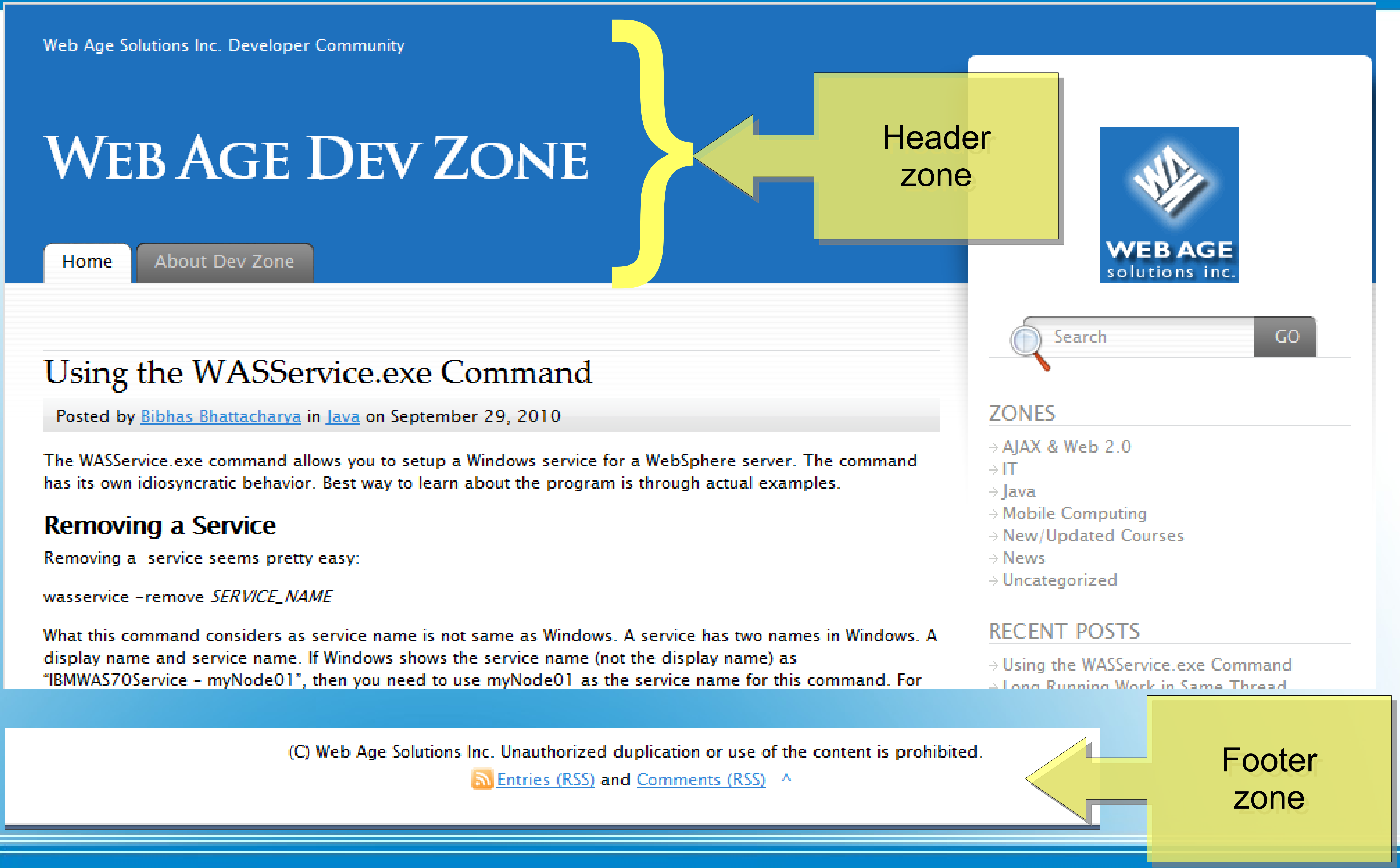
Simplified DOCTYPE

Simplified charset setting

- Elements that have no special visual styling but are used to provide structure and meaning to the content.
- HTML4 semantic elements
  - div (block)
  - span (in line)
- HTML5 semantic elements
  - header, footer, section, article, aside, time and others...
- In contrast, these have visual styling: h1, pre, b.

What are Semantic Elements?

Identify Header and Footer



Web Age Solutions Inc. Developer Community

WEB AGE DEV ZONE

Header zone

Using the WASService.exe Command

Posted by Bibhas Bhattacharya in Java on September 29, 2010

The WASService.exe command allows you to setup a Windows service for a WebSphere server. The command has its own idiosyncratic behavior. Best way to learn about the program is through actual examples.

Removing a Service

Removing a service seems pretty easy:

wasservice -remove *SERVICE_NAME*

What this command considers as service name is not same as Windows. A service has two names in Windows. A display name and service name. If Windows shows the service name (not the display name) as "IBMWAS70Service - myNode01", then you need to use myNode01 as the service name for this command. For

ZONES

- AJAX & Web 2.0
- IT
- Java
- Mobile Computing
- New/Updated Courses
- News
- Uncategorized

RECENT POSTS

- Using the WASService.exe Command
- Long Running Work in Same Thread

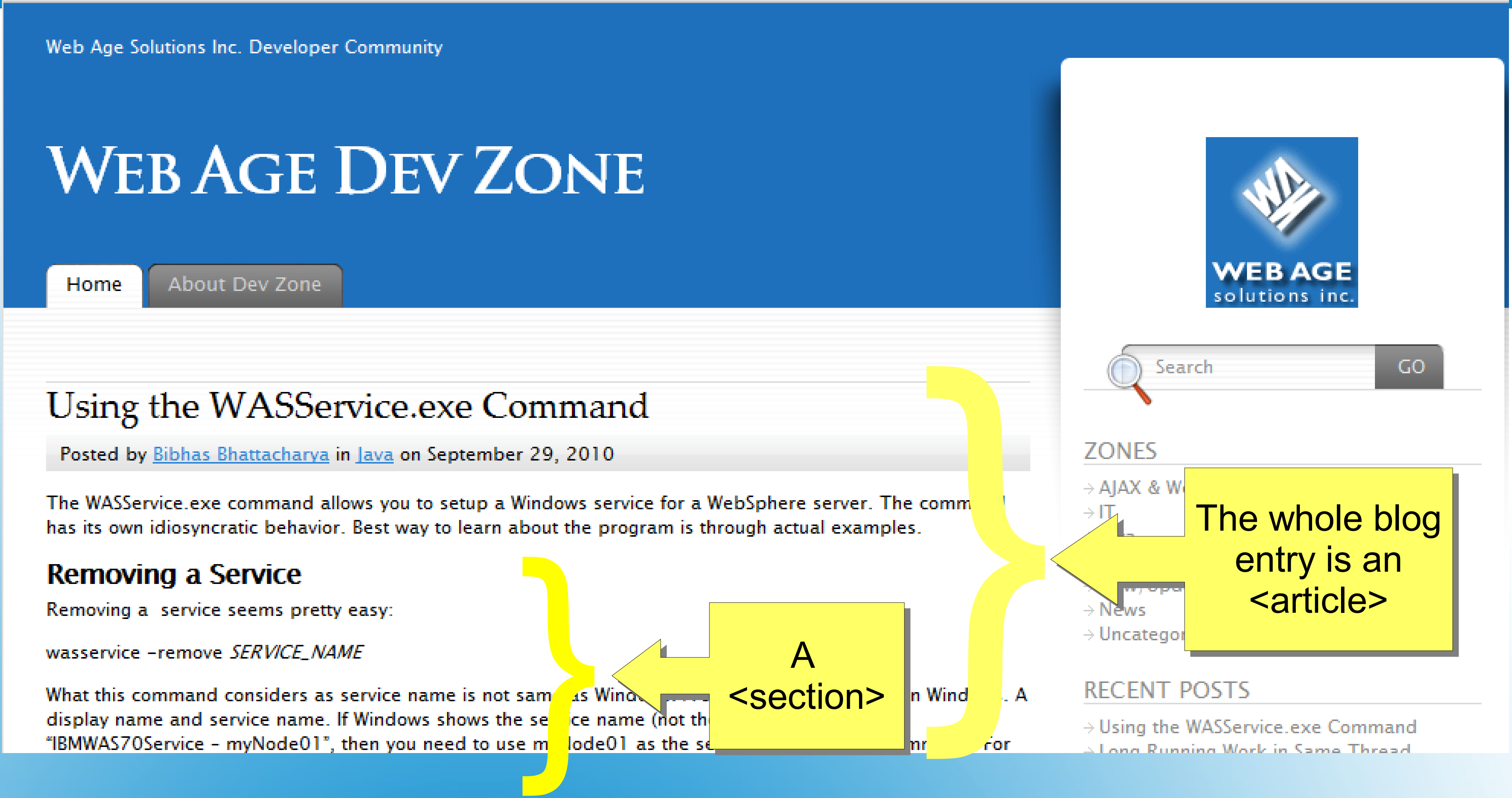
(C) Web Age Solutions Inc. Unauthorized duplication or use of the content is prohibited.

Entries (RSS) and Comments (RSS) ^

Footer zone

Define Header and Footer

```
<body>

<header>
<h1>WEB AGE DEV ZONE</h1>
</header>

<!-- Body content -->

<footer>
<p>(C) Copyright Web Age Solutions Inc.</p>
</footer>

</body>
```



Web Age Solutions Inc. Developer Community

WEB AGE DEV ZONE

Using the WASService.exe Command

Posted by Bibhas Bhattacharya in Java on September 29, 2010

The WASService.exe command allows you to setup a Windows service for a WebSphere server. The comm has its own idiosyncratic behavior. Best way to learn about the program is through actual examples.

Removing a Service

Removing a service seems pretty easy:

wasservice -remove *SERVICE_NAME*

What this command considers as service name is not sam s Wind n Wind . A display name and service name. If Windows shows the se ce name (not th "IBMWAS70Service - myNode01", then you need to use m lode01 as the se mr for

ZONES
AJAX & W
IT
News
Uncategor
RECENT POSTS
Using the WASService.exe Command
Long Running Work in Same Thread

A <section>

The whole blog entry is an <article>

Identify Section and Article

- A fully self contained chunk of content that can live on its own is an <article>
    - For example: a blog posting, product information and a complete news item.
- A collection of text elements is a <section>
    - Used often in place of a <div>
    - Used mainly to apply group styling or to hide/show/collapse content.
    - It is common for an <article> to have several sections. Although, sections can be used on their own to identify regions of a page.
- Do your best to replace <div> elements in an HTML4 page with either <article> or <section>.

Guidelines for Section and Article

Example Section and Article

```
<article>
<h1>Using the WASService.exe Command</h1>
...
<section>
<h3>Removing a Service</h3>
...
</section>
</article>
```

HTML5 Video

```
<video>

<source src="myvid.mp4" type='video/mp4;
   codecs="avc1.42E01E, mp4a.40.2" '>
<source src="myvid.ogv" type='video/ogg;
   codecs="theora, vorbis" '>

<object width="425" height="350">
     <param name="movie"
value="http://www.youtube.com/v/vvtdkxCIKC8"/>
     <embed src="http://www.youtube.com/v/vvtdkxCIKC8"
       type="application/x-shockwave-flash" width="425"
height="350">
     </embed>
</object>

</video>
```

Alternate
format support

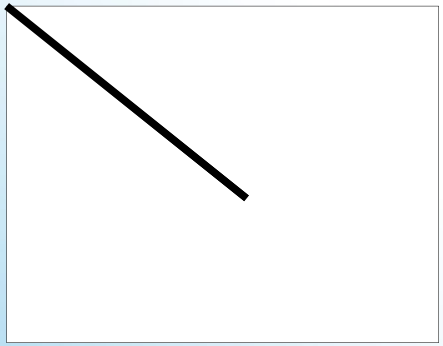
Flash
Fallback

What Can Canvas Do?

- 2D graphics. 3D coming soon.
- Hardware accelerated in most browsers.
- Example use cases:
  - Dynamic data driven charts. Lots of jQuery and Prototype plugins already out.
  - Interactive drawing
  - Custom, fancy and funny form input elements

Example Canvas

```
function drawLine() {

   var canvas = document.getElementById('c1');
   var ctx= canvas.getContext('2d');

   ctx.beginPath();
   ctx.moveTo(0, 0);
   ctx.lineTo(100, 100);
   ctx.stroke();
}

<canvas id="c1" width="500" height="500">

This text is displayed if your browser does not
support HTML5 Canvas.

</canvas>
```



Geolocation

```
function testGeoloc() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(
        geoSuccess, geoFailed);

   }
}

function geoSuccess(pos, timestamp) {

  alert("Lat: " + pos.coords.latitude +
    " Long: " + pos.coords.longitude);

}

function geoFailed(err) {
  alert(err.message);
}
```

Called only if user permits and browser finds location

Browser Side Data Storage

- Extensive data storage capability:
  - Web Storage – Small amounts of data using an easy name value pair based API.
  - IndexedDB – Store large amounts of structured (table & column oriented) data. No SQL.
  - Web SQL – SQL oriented. May lose out to IndexedDB.
- Example use cases:
  - Offline apps.
  - Caching Ajax pulled data to improve performance.

- Can store string data only but can be larger than a cookie.
- Two global objects are used:
  - localStorage – Stores data for the duration of browser's lifetime only. Scope is per window. Good for Ajax data caching.
  - sessionStorage – Data is permanently stored. Scope is per domain. Good for offline apps.
- Both objects expose the same methods:
  - setItem(name, value)
  - getItem(name)
  - removeItem(name)

Web Storage

Web Storage Example

Store structured data using JSON

```
function saveData() {
   var o = new Object();
   o.fullName = document.getElementById(
    "fullName").value;
   o.salary = document.getElementById(
    "salary").value;

   localStorage.setItem("employee", JSON.stringify(o));
}

function loadData() {
   var str = localStorage.getItem("employee");
   if (str != null) {
      var o = JSON.parse(str);
      document.getElementById("fullName").value =
         o.fullName;
     document.getElementById("salary").value =
         o.salary;
   }
}
```

- Allows user to be productive when not online. Example use cases:
  - E-mail program that allows sending e-mail while offline.
  - Employee contact app that works when browser is offline.
- Key strategies to build offline apps:
  - Use a data storage mechanism to retain data locally. Sync local data with server when online.
  - Cache static files (CSS, HTML, JavaScript etc.) in the browser.

Offline Apps

Caching Files in Browser Side

- Create a cache manifest file in your application.

```
CACHE MANIFEST
/index.html
/stylesheet.css
/images/masthead.png
/scripts/misc.js
NETWORK:
/api
FALLBACK:
images/dynamic.php static_image.png
```

- Set the MIME type of the manifest file to text/cache-manifest.
- Use the manifest file from the app's HTML file.

```
<!DOCTYPE HTML>
<html manifest="manifest.cache">
```

- Allows a window to post messages to another window.
  - Windows can be from different domains.
  - Any JavaScript object can be sent as message payload.
- Sending a message:

```
var popupWindow =
    window.open("http://xyz.com/message_receiver.html");
...

var o = new Object();
o.fullName = document.getElementById("fullName").value;
o.salary = document.getElementById("salary").value;

popupWindow.postMessage(o, "xyz.com");
```

Web Messaging

```
window.addEventListener("message", onMessage, false);

...

function onMessage(e) {

    o = e.data;

    document.getElementById("fullName").value = o.fullName;
    document.getElementById("salary").value = o.salary;
}
```

Receiving a Message

Web Worker

- Allows an app to perform background work in a different thread.
  - Takes advantage of multi-core machines.
- Main GUI thread and the background thread communicates through the messaging API.
- Example, starting background work:

```
var w = new Worker("my_worker.js");
w.onmessage = function(e) {
  document.getElementById("progress").innerText =
    "Progress: " + e.data;
}

function startWork() {
  w.postMessage(""); //Notify worker. Also starts work.
}
```

- The my_worker.js file.

```
onmessage = function (event) {
  var i = 0;
  for (i = 0; i < 9000000; ++i) {
    Math.sqrt(i); Math.cos(i); Math.sin(i);
    if (i % 100 == 0) {
        postMessage(i); //Notify main thread
    }
   }
};
```

Performing Background Work

- New semantic elements
- Built-in audio and video
- Canvas for 2D and 3D hardware accelerated drawing.
- New data storage options
- Offline applications
- Inter window messaging
- Multi-threading using Web Worker.

Summary