Browser Capabilities Component

The ASP Browser Capabilities component creates a BrowserType object that determines the type, capabilities and version number of a visitor's browser.

When a browser connects to a server, a User Agent header is also sent to the server. This header contains information about the browser.

The BrowserType object compares the information in the header with information in a file on the server called "Browscap.ini".

If there is a match between the browser type and version number in the header and the information in the "Browsercap.ini" file, the BrowserType object can be used to list the properties of the matching browser. If there is no match for the browser type and version number in the Browscap.ini file, it will set every property to "UNKNOWN".

Syntax

```
<%
Set MyBrow=Server.CreateObject("MSWC.BrowserType")
%>
```

ASP Browser Capabilities Example

The example below creates a BrowserType object in an ASP file, and displays some of the capabilities of your browser:

Example

```
<html>
<body>
<%
Set MyBrow=Server.CreateObject("MSWC.BrowserType")
%>

<table border="0" width="100%">
<tr>
<th>Client OS</th><th><%=MyBrow.platform%></th>
</tr><tr>
<td >Web Browser</td><td ><%=MyBrow.browser%></td>
</tr><tr>
<td>Browser version</td><td><%=MyBrow.version%></td>
</tr><tr>
<td>Frame support?</td><td><%=MyBrow.frames%></td>
</tr><tr>
<td>Table support?</td><td><%=MyBrow.tables%></td>
</tr><tr>
<td>Sound support?</td><td><%=MyBrow.backgroundsounds%></td>
</tr><tr>
```

```
<td>Cookies support?</td><td><%=MyBrow.cookies%></td>
</tr><tr>
<td>VBScript support?</td><td><%=MyBrow.vbscript%></td>
</tr><tr>
<td>JavaScript support?</td><td><%=MyBrow.javascript%></td>
</tr>
</table>

</body>
</html>
```

Output:

Client OS	**WinNT**
Web Browser	IE
Browser version	5.0
Frame support?	True
Table support?	True
Sound support?	True
Cookies support?	True
VBScript support?	True
JavaScript support?	True

The Browscap.ini File

The "Browsercap.ini" file is used to declare properties and to set default values for browsers.

This section is not a tutorial on how to maintain "Browsercap.ini" files, it only shows you the basics; so you get an idea what a "Browsercap.ini" file is all about.

The "Browsercap.ini" file can contain the following:

```
[;comments]
[HTTPUserAgentHeader]
[parent=browserDefinition]
[property1=value1]
[propertyN=valueN]
[Default Browser Capability Settings]
[defaultProperty1=defaultValue1]
[defaultPropertyN=defaultValueN]
```

Parameter	Description
comments	Optional. Any line that starts with a semicolon are ignored by the BrowserType object
HTTPUserAgentHeader	Optional. Specifies the HTTP User Agent header to associate with the browser-property value statements specified in propertyN. Wildcard characters are allowed
browserDefinition	Optional. Specifies the HTTP User Agent header-string of a browser to use as the

	parent browser. The current browser's definition will inherit all of the property values declared in the parent browser's definition
propertyN	Optional. Specifies the browser properties. The following table lists some possible properties: • Backgroundsounds - Support background sounds? • Cdf - Support Channel Definition Format for Webcasting? • Tables - Support tables? • Cookies - Support cookies? • Frames - Support frames? • Javaapplets - Support Java applets? • Javascript - Supports JScript? • Vbscript - Supports VBScript? • Browser - Specifies the name of the browser • Beta - Is the browser beta software? • Platform - Specifies the platform that the browser runs on • Version - Specifies the version number of the browser
valueN	Optional. Specifies the value of propertyN. Can be a string, an integer (prefix with #), or a Boolean value
defaultPropertyN	Optional. Specifies the name of the browser property to which to assign a default value if none of the defined HTTPUserAgentHeader values match the HTTP User Agent header sent by the browser
defaultValueN	Optional. Specifies the value of defaultPropertyN. Can be a string, an integer (prefix with #), or a Boolean value

A "Browsercap.ini" file might look something like this:

```
;IE 5.0
[IE 5.0]
browser=IE
Version=5.0
majorver=#5
minorver=#0
frames=TRUE
tables=TRUE
cookies=TRUE
backgroundsounds=TRUE
vbscript=TRUE
javascript=TRUE
javaapplets=TRUE
ActiveXControls=TRUE
```

```
beta=False

;DEFAULT BROWSER
[*]
browser=Default
frames=FALSE
tables=TRUE
cookies=FALSE
backgroundsounds=FALSE
vbscript=FALSE
javascript=FALSE
```

Email Handling Using ASP

CDOSYS(CDO) is a built-in component in ASP. This component is used to send e-mails with ASP.

CDO (Collaboration Data Objects) is a Microsoft technology that is designed to simplify the creation of messaging applications.CDOSYS is a built-in component in ASP.

Examples using CDOSYS

Sending a text e-mail:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.TextBody="This is a message."
myMail.Send
set myMail=nothing
%>
```

Sending a text e-mail with Bcc and CC fields:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.Bcc="someoneelse@somedomain.com"
myMail.Cc="someoneelse2@somedomain.com"
myMail.TextBody="This is a message."
myMail.Send
set myMail=nothing
%>
```

Sending an HTML e-mail:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.HTMLBody = "<h1>This is a message.</h1>"
myMail.Send
set myMail=nothing
%>
```

Sending an HTML e-mail that sends a webpage from a website:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.CreateMHTMLBody "http://www.w3schools.com/asp/"
myMail.Send
set myMail=nothing
%>
```

Sending an HTML e-mail that sends a webpage from a file on your computer:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.CreateMHTMLBody "file://c:/mydocuments/test.htm"
myMail.Send
set myMail=nothing
%>
```

Sending a text e-mail with an Attachment:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.TextBody="This is a message."
myMail.AddAttachment "c:\mydocuments\test.txt"
myMail.Send
set myMail=nothing
%>
```

Sending a text e-mail using a remote server:

```
<%
Set myMail=CreateObject("CDO.Message")
myMail.Subject="Sending email with CDO"
myMail.From="mymail@mydomain.com"
myMail.To="someone@somedomain.com"
myMail.TextBody="This is a message."
myMail.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/configuration/sendusing")=2
'Name or IP of remote SMTP server
myMail.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/configuration/smtpserver")="smtp.server.com"
'Server port
myMail.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/configuration/smtpserverport")=25
myMail.Configuration.Fields.Update
myMail.Send
set myMail=nothing
%>
```