



## Filen global.asa , en asp programmørs ven

**Artiklen beskriver opbygning, brug, muligheder og begrænsninger mm for global.asa**

**Undervejs bliver der givet et par eksempler på brug.**

**Artiklen kræver ikke en stor ingeniør uddannelse, men forudgående kendskab til Session og Application er en fordel**

Skrevet den **03. Feb 2009** af **busschou** | kategorien **Programmering / ASP** | ★★★★★

### Introduktion

Som tiden går har jeg stødt på mange brugere som ikke ved hvad global.asa er for noget. Derfor vil jeg med denne artikel gerne beskrive hvad global.asa er for en fil og hvad den kan bruges til.

Finder du undervejs en fejl/mangel eller måske en udvidelse jeg mangler at beskrive, så skriv endelig en kommentar.

### Inden du går igang bør du være opmærksom på følgende:

- 1) Jeg har ladet mig fortælle at en del gratis udbydere ikke understøtter brugen af global.asa, dette har jeg dog ikke testet. Vær blot opmærksom på det, hvis du står og vil have den billigste løsning.
- 2) Vær grundig når du retter i global.asa. Du kan risikere at din applikation kommer til at hænge så kun din udbyder kan sætte den igang igen!
- 3) Når du ændrer din global.asa fil vil serveren vente med at recompile den førend alle kørende processer er endte. Den vil samtidig nægte nye forespørgsler med meddelelsen "500-12 Application Restarting". Således vil hele din hjemmeside altså blive sat "offline" indtil den sidste bruger har forladt siden.

### Hvad er det

Global.asa er en fil der benyttes til at deklare globale variable på applikations og sessions niveau.

Filen er event baseret, og kan kun i begrænset omfang benyttes som en normal asp fil. Fx. kan den på nær eet tilfælde ikke benyttes til at skrive ud på en skærm mm. og er af samme årsag nogen gange lidt sværd at debugge.

Filen skal hedde "global.asa" og hører til i roden af ens domæne. Man kan kun have een global.asa fil pr domæne, med mindre man har subdomæner, hvorved hvert subdomæne kan have sin egen global.asa fil.

### Fast indhold

Global.asa har en rimelig streng opbygning og består af fire metoder som hver især reagerer på givne hændelser.

```
<SCRIPT LANGUAGE=VBScript RUNAT=Server>
Sub Application_OnStart
    //Kode der udføres når applikationen starter
End Sub

Sub Application_OnEnd
    //Kode der udføres når applikationen lukker
End Sub

Sub Session_OnStart
    //Kode der køres når en session starter
End Sub
```

```
Sub Session_OnEnd
  //Kode der køres når en session slutter
End Sub
</SCRIPT>
```

Jeg vælger her at kode i VBScript, men filen kan kodes frit i det script sprog som står en nærmest fx JScript.

Filen reagerer altså på fire hændelser men undlades en af metoderne i filen, svarer dette blot til at lade den stå tom som herover.

Brug af scripting udenfor <script> tag'ene resulterer i en fejl.

Brug af html kode, eller funktioner som ikke benyttes i nogen af de fire hændelser bliver ignoreret af serveren.

Det eneste kode man kan have uden for de fire standard metoder er andre metoder eller funktioner, altså ingen "løs" kode.

Bemærk at metoderne er event baseret, de kører altså kun når en given event indtræffer og de kan ikke kaldes fra andre asp sider.

Tilsvarende kan evt andre metoder i global.asa kun kaldes fra disse fire metoder og ikke fra andre asp sider.

Ud over metoderne herover har man mulighed for at deklarerer objekter og typebiblioteker.

Herunder vil jeg beskrive metoderne nærmere:

### **Application\_OnStart**

Hvornår?:

Metoden køres når ens applikation starter. Dette sker een gang, nemlig aller første gang en bruger går ind på ens site. Metoden kaldes som det første og køres således før Session\_OnStart

Når man har redigerer i sin global.asa fil og serveren har recompileret den, så genstartes ens applikation hvorved metoden igen køres een gang første gang en bruger går ind på ens site.

Hvad kan jeg?:

Du har adgang til Application og Server objektet

Hvad kan jeg ikke?:

Du har ikke adgang til Session, Request og Response objektet. Kald til disse vil give fejl

Bemærk:

Eftersom metoden kun køres een gang når den første session starter applikationen behøves man ikke at låse og frigive applicationen ved Application.Lock og Application.Unlock

### **Application\_OnEnd**

Hvornår?:

Metoden kører een gang når ens applikation slutter. Hvis du fx retter i din global.asa, så vil serveren først nægte adgang til nyankomne processer. Den vil herefter vente på at samtlige processer afsluttes. Når alle processer er afsluttet kalder den denne metode, genkompilerer global.asa og genstarter applikationen.

Metoden kører som det sidste dvs den køres først efter Session\_OnEnd er kaldt for sidste gang.

Hvad kan jeg?:

Du har adgang til Application og Server objektet nær Server.Mappath

Hvad kan jeg ikke?:

Du har ikke adgang til Session, Request og Response objektet samt Server.Mappath. Kald til disse vil

give fejl

Bemærk:

Metoden køres som den "anonyme bruger" på serveren. I tilfælde af der ikke er en sådan, eller den ikke kan logge på, da køres denne metode ikke

## Session\_OnStart

Hvornår?:

Metoden køres een gang, hver den første gang en bruger går ind på ens site. Hver gang en bruger åbner en browser og går ind på ens side så starter der en session som gør at serveren kan holde styr på hvem den kommunikerer med. Inden denne session startes kaldes denne her metode een gang.

Metoden køres således inden den først kaldte side køres

Hvad kan jeg?:

Du har adgang til alle objekter dvs, Application, Server, Session, Response, Request og ObjectContext

Bemærk:

Da denne metode køres inden nogen sider køres vil det være god kodestil at definere sine sessions variable her.

Hvis du i metoden angiver en redirect, da vil evt kode efter denne redirect ikke blive udført.

## Session\_OnEnd

Hvornår?:

Metoden køres een gang inden en brugers session dør. Metoden køres enten explicit hvis man kalder Session.Abandon eller når en session får timeout. Ved timeout kaldes først denne metode, dernæst frigiver serveren sine brugte resourcer.

Hvad kan jeg?:

Du har adgang til Application, Server og Session objektet på nær Server.MapPath

Hvad kan jeg ikke?:

Du har ikke adgang til Response og Request objektet samt Server.MapPath

Bemærk:

Metoden køres som den "anonyme bruger" på serveren. I tilfælde af der ikke er en sådan, eller den ikke kan logge på, da køres denne metode ikke

Det var faktisk det. Herover står hvad du kan med global.asa og hvad du ikke kan. For at finde en fornuftig nytte kræver det selvfølgelig at man ved hvad sessionen og applicationen kan bruges til. Måske endvidere at man ved hvad Object og TypeLibrary deklorationer er og kan bruges til. Men i bund og grund er det det. Jeg vil derfor slutte af med et par eksempler på forskellige anvendelses muligheder

## Redirect så en bruger altid starter på en given side:

```
' Bemærk at dette selvfølgelig kun virker hos bruger som har cookies slået til
' da der ellers ikke dannes en unik sessions id
<SCRIPT Language=VBScript RUNAT=Server>
  Sub Session_OnStart
    ' Definer en startside
    startPage = "/MyApp/StartHere.asp"
```

```

' Find den nuværende side
currentPage = Request.ServerVariables("SCRIPT_NAME")

' Lav en sammenligning og redirect hvis de er forskellige
If strcomp(currentPage,startPage,1) then
    Response.Redirect(startPage)
End If
End Sub
</SCRIPT>

```

### Tæl online brugere:

```

' Bemærk at applikationen låses hver gang der tilgås applikations variable
' Endvidere bemærkes her at der ikke er garanti for at brugerantallet tælles
ned hvis en bruger lukker sin browser
<SCRIPT LANGUAGE=VBScript RUNAT=Server>
    Sub Application_OnStart
        Application("intBrugereOnline") = 0
    End Sub

    Sub Session_OnStart
        Application.Lock
        Application("intBrugereOnline") = Application("intBrugereOnline") + 1
        Application.Unlock
    End Sub

    Sub Session_OnEnd
        Application.Lock
        Application("intBrugereOnline") = Application("intBrugereOnline") - 1
        Application.Unlock
    End Sub
</SCRIPT>

```

### Gem connectionstrings som globale variable

```

' Bemærk at på denne måde behøves man ikke gå og huske ens connectionstring
' Mange gemmer den i en side som de inkluder på alle sider, her gemmes den een
gang for alle
' Samtidig ses at server.mappath kan benyttes i metoden Application_OnStart
<SCRIPT LANGUAGE=VBScript RUNAT=Server>
    Sub Application_OnStart
        Application("db1") =
"DBQ=d:\home\domæne.dk\database\database.mdb;UID=;PWD=;"&_
            "Driver={Microsoft Access Driver (*.mdb)};"
        Application("db2") = "DBQ=" & Server.MapPath("database.mdb")
&" ;UID=;PWD=;"&_
            "Driver={Microsoft Access Driver (*.mdb)};"

    End Sub
</SCRIPT>

```

## Opdater database når brugere kommer online og når de går offline

Mange spørger hvordan de sætter deres brugere offline når brugerne ikke logger af, men når de lukker deres browser.

Hvis en logud funktion kalder Session.Abandon, da vil nedenstående tage højde for begge dele

Hvis brugeren lukker browseren vil Session\_OnEnd blive kaldt når sessionen timer ud efter 10-20min afhængig af hvad man har sat den til

Første gang der kommer en bruger ind på siden sættes en connectionstreng, herved slipper man for bøvl med at server.mappath ikke kan benyttes i Session\_OnEnd

Læg mærke til at der i Session\_OnStart sættes sessions variable ud fra det som er indtastet i en login form Dette kræver at ens login form er \*.html og ikke \*.asp. Hvis filen hed asp så ville metoden jo kaldes inden man indtaster i formen

Bemærk også hvordan man her kan tjekke på login og redirecte hvis det fejler med mere

```
<SCRIPT LANGUAGE=VBScript RUNAT=Server>
  Sub Application_OnStart
    Application("db") = "Data Source=" & Server.MapPath("database.mdb") &
      ";Provider=Microsoft.Jet.OLEDB.4.0;"
  End Sub

  Sub Session_OnStart
    Session("bruger") = Replace(Request.Form("bruger"), "'", "'")
    Session("kode") = Replace(Request.Form("kode"), "'", "'")
    Dim Conn, rs
    Set Conn = Server.CreateObject("ADODB.Connection")
    Conn.Open Application("db")
    Set rs = Conn.Execute("Select * From tabel Where bruger =
'" & Session("bruger") &
      "' And kode = '" & Session("kode") & "'")
    If not rs.eof Then
      Session("id") = rs("Id")
      Conn.Execute("Update tabel Set online = True Where Id = " & Session("id"))
    Else
      Session.Abandon
      Response.Redirect "login.html"
      Conn.Close
      Set Conn = Nothing
    End If
    Conn.Close
    Set Conn = Nothing
  End Sub

  Sub Session_OnEnd
    Dim Conn
    Set Conn = Server.CreateObject("ADODB.Connection")
    Conn.Open Application("db")
    Conn.Execute("Update tabel Set online = False Where Id = " & Session("id"))
    Conn.Close
    Set Conn = Nothing
  End Sub
</SCRIPT>
```

### **Kommentar af bongi d. 24. Oct 2005 | 1**

### **Kommentar af cult d. 06. Nov 2006 | 2**

Topklasse artikel med en god forklaring og kanon eksempler. Bør læses af enhver der endnu ikke har stiftet bekendskab med global.asa !

### **Kommentar af tjaz d. 03. Oct 2006 | 3**

Hvis man er fattig på point, så prøv at søg "global.asa" på google.dk, præcis samme artikel :P

### **Kommentar af ahjdp1 d. 08. Sep 2006 | 4**

### **Kommentar af amikk d. 05. Aug 2005 | 5**

Godt forklaret - og i særdeleshed velstruktureret.

### **Kommentar af mini-me d. 04. Aug 2005 | 6**

Fornuftig artikel, giver en god forståelse af hvad kan bruge global.asa til!

### **Kommentar af znuggle.com d. 07. Nov 2007 | 7**

En stjerne artikel

### **Kommentar af lund\_dk d. 16. Aug 2005 | 8**

### **Kommentar af pallenoc d. 25. Oct 2005 | 9**

Jeg er ret ny indenfor programering generelt og ASP i særdeleshed, så hvis dette fremover kunne være standard bliver indlæringen væsentligt nemmere;-)

### **Kommentar af thomas\_b d. 09. Feb 2006 | 10**

### **Kommentar af mathiasovesen d. 09. Feb 2006 | 11**

### **Kommentar af seb-boy d. 04. Aug 2005 | 12**

Smart.

Det er en ret go' artikle.

Den artikle, har jeg lært meget af ;D