The new HTTP Handler

Presenter: Harm Wibier

SYNERGY 2019

CRUISING TO NEW HORIZONS

What is HTTP?

- HyperText Transfer Protocol
  - Invented at CERN in 1989
  - Application layer protocol built on top of TCP
  - Client & server communicate via request and response messages

- The basis of the web..
  - Web sites run on HTTP
  - Web applications run on HTTP
  - SOAP runs on HTTP

HTTP GET Request

```
GET /products/dataflex-63 HTTP/1.1
Host: www.dataaccess.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/web
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9,nl-NL;q=0.8,nl;q=0.7
```

HTTP Response

```
HTTP/1.1 200 OK
Content-Type: text/html
Date: Mon, 11 Mar 2019 20:21:19 GMT
Content-Length: 23679
Server: Microsoft-IIS/8.0

<!DOCTYPE html>
<html lang="en">
<head>
<base href="https://www.dataaccess.com/">
     <title>Build business software faster at lower cost | DataFlex | D
     <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />
     <meta http-equiv="Content-Type" content="text/html; charset=iso-88
```

HTTP POST Request

```
POST /WebOrder_19_1/WebServiceDispatcher.wso/CallAction/JSON HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
Accept: application/json
Accept-Encoding: gzip, deflate, br
```

```
{"ActionRequest":{"Header":{"sFocus":"oOrder.oWebMainPanel.oOrderHeaderOrder_Number"
"aDDOViews":["oOrder"],"aSyncProps":[{"sO":"","aP":[{"sN":"peMode","sV":"10"},
{"sN":"pbIsMobile","sV":"0"},{"sN":"piScreenWidth","sV":"1920"},{"sN":"piScreenHeight",
"sV":"1200"},{"sN":"piWindowWidth","sV":"1920"},{"sN":"piWindowHeight","sV":"1089"},
{"sN":"psLocationHash","sV":"#Order"},{"sN":"psSPKey","sV":
"OLaC9X2XS9fhUWfyEbQ3eMbOBPEpF3O/mo+oZoL/XlQ59kEZr52kc+79oVnNMwW9"}]},
```

DataFlex WebApp Server & HTTP

- Microsoft IIS handles the network communication

- Classic ASP End-Point
  - Write ASP scripts that call into WebApp
- Web Services End-Point
  - Write functions called using SOAP
  - JSON Support was added later
  - Handling of requests happens in waswsvc.dll

Why a new handler?

- To support building REST services
- To better handle file steaming
  - Uploads & downloads
- To reduce the Classic ASP dependency

- The web services handler sometimes lacks low level control (or flexibility)

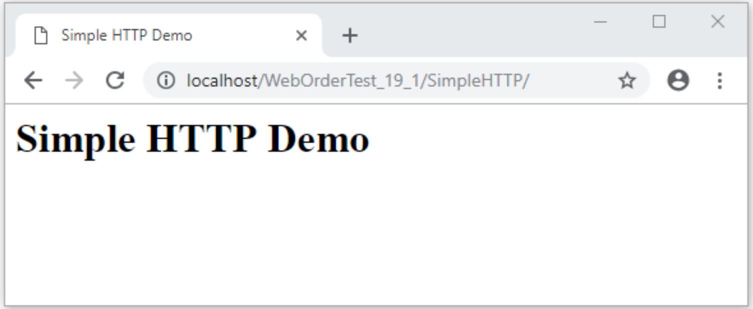
The new HTTP Request Handler

- New end-point for the DataFlex WebApp Server
  - Web Services via cWebService
  - Classic ASP via cWebAspClassicObject (or cWebBusinessProcess)
  - Plain HTTP via cWebHttpHandler

- Handle the entire HTTP request in DataFlex code
  - Full control over the response
  - Flexible URL handling
  - Handle binary data
    - Supports chunks
    - File upload & file download
  - Easily build powerful REST services

cWebHttpHandler

- Create a singleton object instance (like cWebService)
  - Configure which URL’s to handle using psPath & psVerbs
  - Implement OnHttpRequest… events



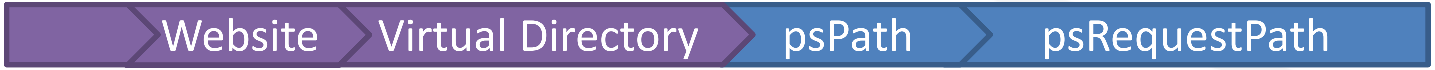
Simple HTTP Demo

```
Use cWebHttpHandler.pkg

Object oSimpleHTTPHandler is a cWebHttpHandler
    Set psPath to "SimpleHTTP"
    Set psVerbs to "GET"

    Procedure OnHttpGet String sPath String sAcceptType
        Send AddHttpResponseHeader "Content-Type" "text/html"
        Send OutputString "<html><head><title>Simple HTTP Demo</title></head><body><h1>Simple HTTP Demo</h1></bod
    End_Procedure

End_Object
```

Demo…

psPath

- Determines the virtual path for which requests will be handled
- Relative to the root of the WebApp
- Requests send to subfolders will be handled as well
    - psRequestPath (or sPath parameter) will receive the subpath
- No wildcards allowed
- No trailing backslash
- Deep paths (like "SimpleHttp/SubFolder“) are allowed



http://localhost/MyApplicationApp/**SimpleHTTP**/AnythingElse/AndMore

API

https://docs.dataaccess.com/dataflexhelp/#t=mergedProjects%2FVDFClassRef%2FcWebHttpHandler.htm

Implementation

- Native HTTP Module for IIS
  - WasHttpMod.dll
  - Registered globally during installation
  - Activated when creating virtual directory
    - By the Studio or WebApp Administrator
- Configuration part of IIS configuration scheme
  - Written by the studio in web.config in AppHtml

Binary data

- Work with UChar arrays
  - Direct_Input / Direct_Output
  - Get_Field_Value
  - Field_Current_UCAValue
- Work in chunks

Demo…

Error Handling

- You are responsible for error handling
  - ErrorQueueStart
  - ErrorQueueEnd
  - ErrorCount
  - ErrorMessage
- Errors will be written to the webapp log
  - Unless they are queued
- Errors are not sent to the client by default

HTTPS

- This all happens at IIS level
  - Configure your certificate in the IIS Manager
- No extra handling at DataFlex level needed

Web Framework

- File upload & download now use the new handler
  - cWebFileUploadButton / cWebFileUploadForm
  - DownloadUrl(cWebResourceManager)
- No code changes are needed in your application
  - The file access is now done by the webapp.exe
    - System Account instead of IUSR
  - The file access now happens on the Slave with SPLF

19.1 Goals

- Provide a solid basis
  - That can be used for REST services
  - That allows replacement of ASP file streaming
- Did not go into the dept of supplying special REST classes, wizards and such
  - Left for future revisions
  - Mike Peat already has a library

Thank you for your time!

I’ll be around for any questions you might have…