

Using DataFlex 2017's Server Web Properties

Harm Wibier

Data Access Worldwide

WebApp Framework

- Web Objects
 - DataFlex Classes <> Matching JavaScript Classes
- Definitions are sent to the client
 - Structure of objects with their properties and child objects
- JavaScript Engine renders the WebApp

What are web properties?

- Properties available on Web Objects
- Are sent to the client
- Invented to configure the User Interface
- Can change on the client
- JavaScript classes know how to handle them

```
{ WebProperty=True }  
Property String psValue ""
```

Flow

- View is loaded
 - Object definitions are loaded to the client
 - Web properties with their initial values are included
- User finds data
 - Event is sent to the server
 - Server changes web properties and sends new values to the client (WebSet)
- User enters data
 - Changes a web property on the client (psValue)
- User saves data
 - Event is sent to the server
 - Changed web properties are sent to inform the server of their state
 - Web properties are read from the request (WebGet) and are used to save the record

Application State

- There is no persistency on the server
- Web Properties do provide persistency
 - Turned out to be a convenient place
- Application state is stored on the client

```
{ WebProperty=True }  
Property String psFollowUpRowId ""
```

```
{ WebProperty=True }  
Property String psFilter ""
```

Security

- Web properties can be seen and edited using browser development tools!
- Encrypted properties have been requested
- RowId's are protected by a checksum

```
{ WebProperty=True }  
Property Integer piTennantId 1
```

Performance

- Web properties are sent back an forth with each request!
- Mobile devices can run on slow connections

●●●●●	O2-UK	E
●●●●○	O2-UK	GPRS
●●●○○	O2-UK	3G
●●●●●	O2-UK	4G
●●●●○	O2-UK	

Why send data to the client that is not used on the client?

Server side web properties

- Same syntax as existing web properties
- Handled by the framework

```
{ WebProperty=Server }  
Property String psFilter ""
```

```
Procedure OnConstrain  
String sFilter
```

```
WebGet psFilter of oView to sFilter
```

Flavors

- { WebProperty=Server }
 - Lifetime equal to client web properties
- { WebProperty=ServerSession }
 - Lifetime equal to sessions
 - Survives page reload!
 - Shared across browser tabs

Implementation

- Stored in a single database table
 - WebAppServerProps
- Serialized to JSON by the runtime
- Page lifetime or session lifetime

```
{ WebProperty=Server }  
Property String[] paCustomers ""
```

```
{ WebProperty=ServerSession }  
Property String[] paCustomers ""
```

Upgrade

- New projects will have the table
- Migration will add the table
- When converting to MS SQL the WebAppServerProps.Data will be converted to varchar(MAX)

Migration

- { WebProperty=True } continues to work as { WebProperty=Client }
- Basically all web properties within your application can be changed to { WebProperty= Server } unless they have custom JavaScript using them

A silver MacBook Air laptop is open on a wooden desk. To the left of the laptop is a glass of water. In the foreground, a smartphone lies on a piece of paper. The background is a blurred indoor setting with warm lighting.

Thank you for your time

Have a nice day!