

DYALOG

Elsinore 2019



What's New with HTMLRenderer

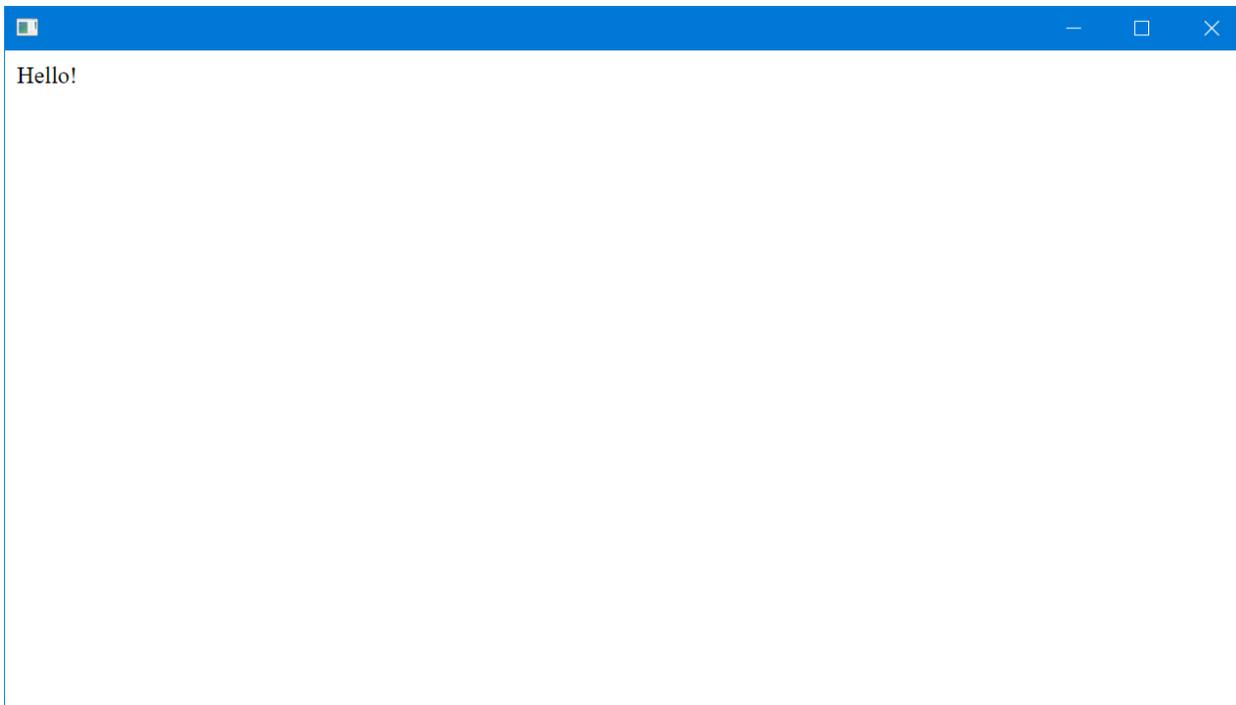
Brian Becker, Dyalog APL Tools Architect

HTMLRenderer

- ◆ Dyalog APL object
 - ◆ `'hr' ⍵WC 'HTMLRenderer' 'Hello!'`
 - ◆ `(hr←⍵NEW ,c'HTMLRenderer').HTML←'Hello!'`



HTMLRenderer



HTMLRenderer

- Dyalog APL object
 - `'hr' ⍵WC 'HTMLRenderer' 'Hello!'`
 - `(hr←NEW ,c'HTMLRenderer').HTML←'Hello!'`
- Interface to the Chromium Embedded Framework (CEF)
 - Chromium Web Browser – the foundation for Google Chrome
- Renders HTML5/CSS/JavaScript
- Adobe Acrobat
- Amazon Music
- Evernote
- Google Web Designer
- MATLAB
- Sling TV
- Steam



History

- Born in 2017 (Dyalog v16.0)
 - What is this?
 - Sometimes makes a mess (SYSERROR)



History

- Born in 2017 (Dyalog v16.0)
 - What is this?
 - Sometimes makes a mess (SYSERROR)
- Grows up a bit in 2018 (Dyalog v17.0)
 - Falls over less often



History

- ◆ Born in 2017 (Dyalog v16.0)
 - ◆ What is this?
 - ◆ Sometimes makes a mess (SYSERROR)
- ◆ Grows up a bit in 2018 (Dyalog v17.0)
 - ◆ Falls over less often
 - ◆ Interesting to play with



History

- Born in 2017 (Dyalog v16.0)
 - What is this?
 - Sometimes makes a mess (SYSERROR)
- Grows up a bit in 2018 (Dyalog v17.0)
 - Falls over less often
 - Interesting to play with
- Standing up (Dyalog v17.1)
 - Started communicating more (WebSockets)
 - Going places (Windows, macOS and Linux)



Cross platform

- ◆ Windows / macOS / Linux
- ◆ Write once, deliver on any platform
- ◆ TamStat



New and changed properties

- Properties have been implemented on platforms that support them

HTMLRenderer Properties as implemented as of 5 Sept 2019

Property	Windows	Mac OS	Linux
Type	+	+	+
HTML	+	+	+
Posn	+	+	+
Size	+	+	+
URL	+	+	+
Coord	+	+	+
Border *	-	-	-
Visible	+	+	+
Event	+	+	+
Sizeable *	+	+	-
Moveable *	-	-	-
SysMenu *	+	-	-
MaxButton *	+	-	-
MinButton *	+	+	-
IconObj	+	!	!
Data	+	+	+
Attach	+	!	!
KeepOnClose	+	+	+
AsChild	+	!	!
InterceptedURLs	+	+	+
CEFVersion	+	+	!
Caption	+	+	+
MethodList	+	+	+
ChildList	+	+	+
EventList	+	+	+
PropList	+	+	+

* indicates settable at creation time only

+ indicates it works

- indicates it has no effect

! Indicates it produces an error



New and changed properties

- Properties have been implemented on platforms that support them
- IconObj added
- CEFVersion added
- Caption added

```
<title>My Caption</title>
Caption←'My Caption'
```
- InterceptedURLs didn't change... (sort of)



InterceptedURLs

- Introduced in v17.0
- Determines whether a request for a resource is passed to the net, or back to APL

dyalog_root	1
*	0

A `dyalog_root` is the "default" URL that would be prepended to requests for "local" resources

`<script src="https://somecdn.com/scripts/utils.js"/>`

(via the net)

`<script src="/scripts/utils.js"/>`

(via APL callback)

- Probably made obsolete in v17.1

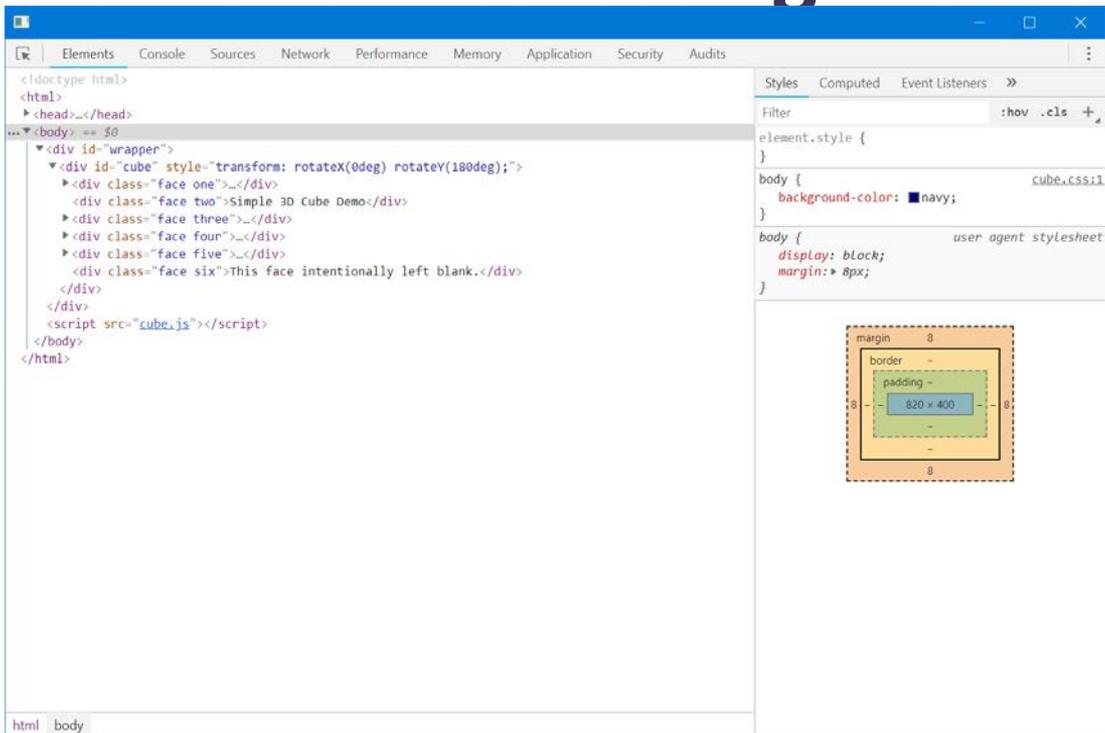


New and interesting methods

- ShowDevTools
 - `Renderer.ShowDevTools 1`



New and interesting methods



The screenshot shows a browser's developer tools interface. The left pane displays the HTML structure of a page, including a `<div id="cube" style="transform: rotateX(0deg) rotateY(180deg);">` element. The right pane shows the 'Styles' panel with a filter set to `:hov .cls`. The 'element.style' section is empty, while the 'body' section shows a `background-color: navy;` rule. Below the styles, a box model diagram is displayed, showing a blue content box (820 x 400) with a green padding area, a yellow border, and an orange margin area.

```
<!doctype html>
<html>
  <head></head>
  <body> == $0
    <div id="wrapper">
      <div id="cube" style="transform: rotateX(0deg) rotateY(180deg);">
        <div class="face one">...</div>
        <div class="face two">Simple 3D Cube Demo</div>
        <div class="face three">...</div>
        <div class="face four">...</div>
        <div class="face five">...</div>
        <div class="face six">This face intentionally left blank.</div>
      </div>
    </div>
    <script src="cube.js"></script>
  </body>
</html>
```

Styles

Filter: :hov .cls

element.style { }

body { background-color: navy; }

body { display: block; margin: 8px; }

margin: 8px

border: 8px

padding: 8px

820 x 400



New and interesting methods

- ◆ ShowDevTools
 - ◆ `Renderer.ShowDevTools 1`
- ◆ PrintToPDF
 - ◆ Prints rendering to PDF file
 - ◆ `Renderer.PrintToPDF '/tmp/myfile.pdf'`

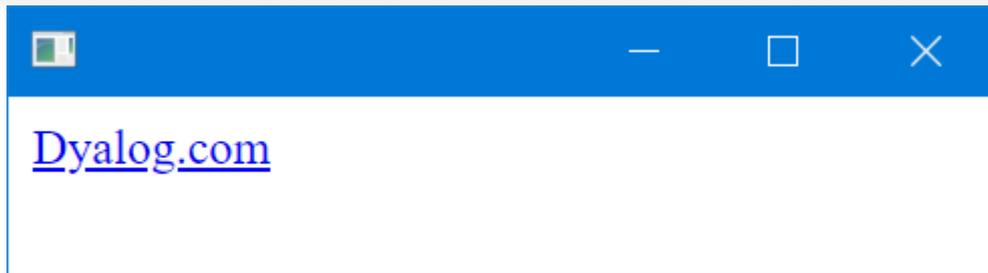


New Events

- ◆ `SelectCertificate`
 - ◆ Used when requesting a resource that requires a certificate. A dialog box opens to allow you to select the certificate to use.
- ◆ `DoPopup`
 - ◆ When the client requests that a resource be loaded in a new window, a `DoPopup` event is triggered.



New Events



```

▽ {r}←DoPopupDemo2 args;prop
[1]   :If 0εargs
[2]     prop←('Size'(10 20))('Event'('onClose' (⇒SI)))('Event'('onDoPopup' (⇒SI)))
[3]     Renderers←,NEW'HTMLRenderer'prop A create a new renderer
[4]     Renderers[1].HTML←'<a href="http://www.dyalog.com" target="_blank">Dyalog.com</a>'
[5]   :Else
[6]     :Select 2>args A select based on event
[7]     :Case 'DoPopup' A DoPopup event?
[8]       Renderers←,NEW'HTMLRenderer'(',<'URL'(3>args)) A create a new renderer
[9]     :Case 'Close' A Close window event?
[10]      EX'Renderers'
[11]   :EndSelect
[12] :EndIf
▽
  
```



WebSockets

- ◆ What's a WebSocket?
 - ◆ A bi-directional, asynchronous, communications channel between client and server
 - ◆ In standard HTTP, all requests are initiated from the client. The server can only respond to a request.
 - ◆ The client can request to "upgrade" the HTTP connection to a WebSocket, which the server can accept or reject.
- ◆ Once a WebSocket is established, either side can send a message. No response is required.
- ◆ Either side can close the WebSocket.
- ◆ You can have multiple WebSockets to any combination of local or remote servers.



WebSockets

- ◆ Uses
 - ◆ Pushing information
 - ◆ "financial ticker", chat services, notifications, education
 - ◆ Databinding
 - ◆ Keeping data in your workspace in sync with the data displayed in the GUI
 - ◆ Changes made in the GUI are updated in the workspace
 - ◆ Changes in the workspace are displayed on the client

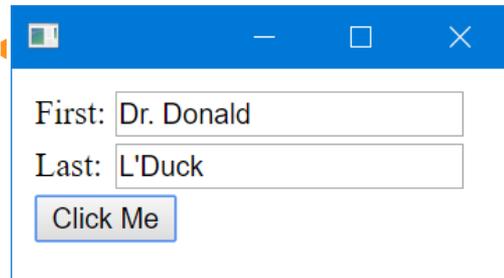


Utilities and Frameworks

- ◆ HttpUtils
 - ◆ Utilities to convert HTTPRequest event arguments and results



Utilities and Frameworks



A screenshot of a web browser window showing a form. The form has two input fields: "First:" with the value "Dr. Donald" and "Last:" with the value "L'Duck". Below the fields is a button labeled "Click Me".

```
10>args  
first=Dr.+Donald&last=L%27Duck
```

```
req←NEW HttpUtils.Request args
```

```
req.FormData
```

first	Dr. Donald
last	L'Duck



Utilities and Frameworks

- ◆ HttpUtils
 - ◆ Utilities to convert HTTPRequest event arguments and results
- ◆ EasyGUI
 - ◆ A collection of interactions that can be used stand-alone or in combination
 - ◆ Josh will be presenting EasyGUI in the next presentation
- ◆ DUI – Dyalog User Interface <https://github.com/Dyalog/DUI>
 - ◆ Library for Web Content Creation
 - ◆ Server
 - ◆ MiServer for running over the net
 - ◆ HRServer (HTMLRenderer server) for running as a desktop app



DUI

- ◆ Page class
 - ◆ Self-contained MiPage using internal HTMLRenderer
- ◆ MiSite

`DUI.Run '/path/to/misite'` A run locally using HTMLRenderer

`DUI.Run '/path/to/misite' port` A run on net using MiServer



In the pipeline

- ◆ A couple minor tweaks to HTMLRenderer
 - ◆ Extend default root (`http[s]://dyalog_root`) to allow other protocols (`ws[s]://` for WebSockets)
 - ◆ Support references to `file://` to read files directly rather than pass request back to the workspace
- ◆ DUI
 - ◆ DoPopup support
 - ◆ Databinding
 - ◆ Documentation, Tutorials



