

Cross-platform User Interfaces

Morten Kromberg, Dyalog Ltd.



(Big) Bets We Are Placing



1. Parallel Processing is getting Critical
2. New Platforms will soon be Important
3. Most new User Interfaces will use either HTML5/JS or WPF/XAML



2. New Platforms



- UNIX variants
 - ARM Linux (Raspberry Pi), Apple OSX, Android
- Windows variants without Win32
 - **UWP**, New Servers, Windows for ARM
- Our new tools are designed to be cross-platform
 - RIDE, MiServer, SAWS, R-Connect, DyaCrypt
 - Some old ones too: Conga, SQAPL, ...
- Our goal is to allow future apps to be developed on (any) one platform and deployed on (any) other



3. User Interfaces




- **Cross-platform apps will use HTML5 and Javascript to define the user interface (for at least 5 years)**
- **Some New Microsoft Desktop and UWP apps will use WPF. UWP is also XAML-based.**
 - WPF and "UWP" will morph into a single toolset
- We will support data binding and "MVVM" style application building with both these tools
- We are looking at embedded HTML5/JS rendering engines to allow the same UI to service Web and Desktop or "Phablet" applications.



Dyalog-Supported UI Technologies

All of the following are being actively enhanced:

- Win32 /  WC
 - Occasional addition of features on demand, like support for touch & high DPI screens - underlying platform is now "stable"
- Windows Presentation Foundation => UWP
 - DataBinding and other .NET interface enhancements
 - Syncfusion Controls bundled with Dyalog APL
 - Working with Michael Hughes on tools and training materials
- HTML5/JavaScript
 - MiServer
 - Syncfusion Controls bundled with Dyalog APL



MiServer Growing Up

- Has been coming for some time
 - Stefanos WildServer
 - Mortens MildServer 1.0
 - Brians MiServer 2.0
 - Adám, Dan, Morten - but still mostly Brians MiServer 3.0
- Each was usable to a larger group than before
- With MiServer 3.0 we think everyone can play



What is MiServer?



Well... First - What is `□WC`?

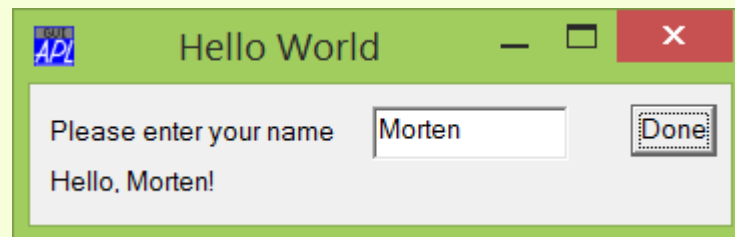
- A set of classes for creating GUI

```
'F1'□WC'Form' 'Hello World'('Coord' 'Pixel')('Size'(70 300))
'F1.label'□WC'Label' 'Please enter your name'(10 10)
'F1.name'□WC'Edit' ''(10 170)(θ 100)
'F1.done'□WC'Button' 'Done'(10 300)
'F1.done'□WS'Event' 'Select' 'CallbackFn'
'F1.result'□WC'Text' ''(40 10)
```

▽ CallbackFn

```
F1.result.Text←'Hello, ',(F1.name.Text),'!'
```

▽

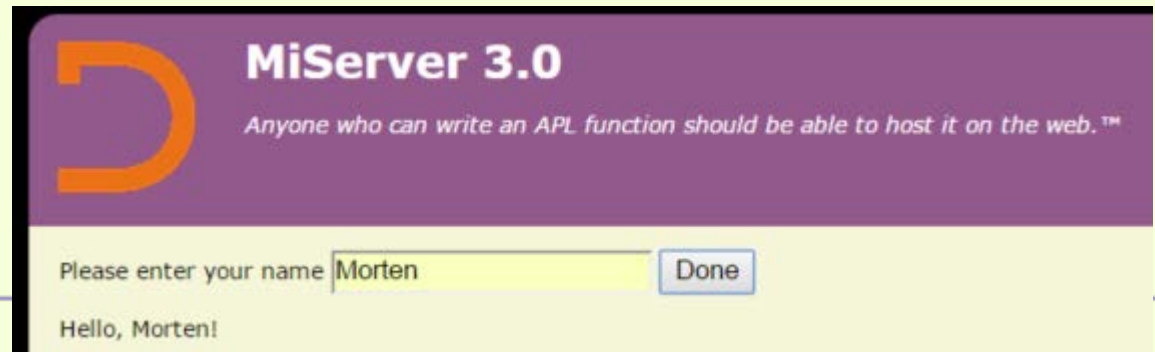


What is Mi Server?

- A set of classes for creating GUI

```
F1←Add _.Form ◇ Add _.Title 'Hello World'
('for' 'name') F1.Add _.label 'Please enter your name'
'name' F1.Add _.EditField
done←'done' F1.Add _.Button 'Done'
done.On 'click' 'CallbackFn'
'result' F1.Add _.div
```

```
▽ r←CallbackFn
:Access Public
r←'#result' Replace _.p ('Hello, ',(Get'name'),'!')
▽
```



Rosetta Code – ⚡WC vs MiServer

```

'F1'⚡WC'Form' 'Hello World'('Coord' 'Pixel')('Size'(70 300))
F1←Add _.Form ⋄ Add _.Title 'Hello World'

'F1.label'⚡WC'Label' 'Please enter your name'(10 10)
('for' 'name') F1.Add _.label 'Please enter your name'

'F1.name'⚡WC'Edit' ''(10 170)(θ 100)
'name' F1.Add _.EditField

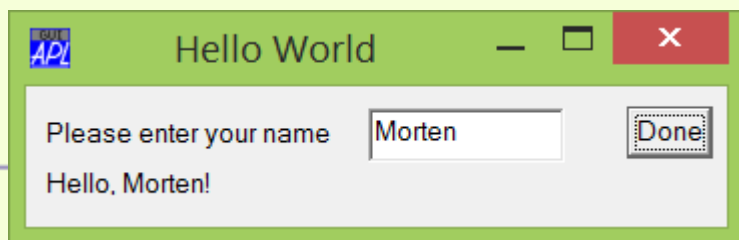
'F1.done'⚡WC'Button' 'Done'(10 300)
done←'done' F1.Add _.Button 'Done'

'F1.done'⚡WS'Event' 'Select' 'CallbackFn '
done.On 'click' 'CallbackFn'

'F1.result'⚡WC'Text' ''(40 10)
'result' F1.Add _.div

▽ CallbackFn
F1.result.Text←'Hello, ',(F1.name.Text), '!'
r←'#result' Replace _.p ('Hello, ',(Get'name'), '!')

```



Let's Take a Look

- The MiServer Home Page
<http://miserver.dyalog.com>



Uses of MiServer

At Dyalog'15

- **(U06) Ants, AJAX and APL**
Ray Cannon
- **(U13) TamStat, a Statistical Package**
Steve Mansour & Jerry Brennan
- A demonstration UI build for a new insurance application:
<http://127.0.0.1:8081>



Automated Testing with Selenium

- Selenium is a cross-platform tool for web ui testing
- [c:\devt\Selenium\selenium.dws](#)



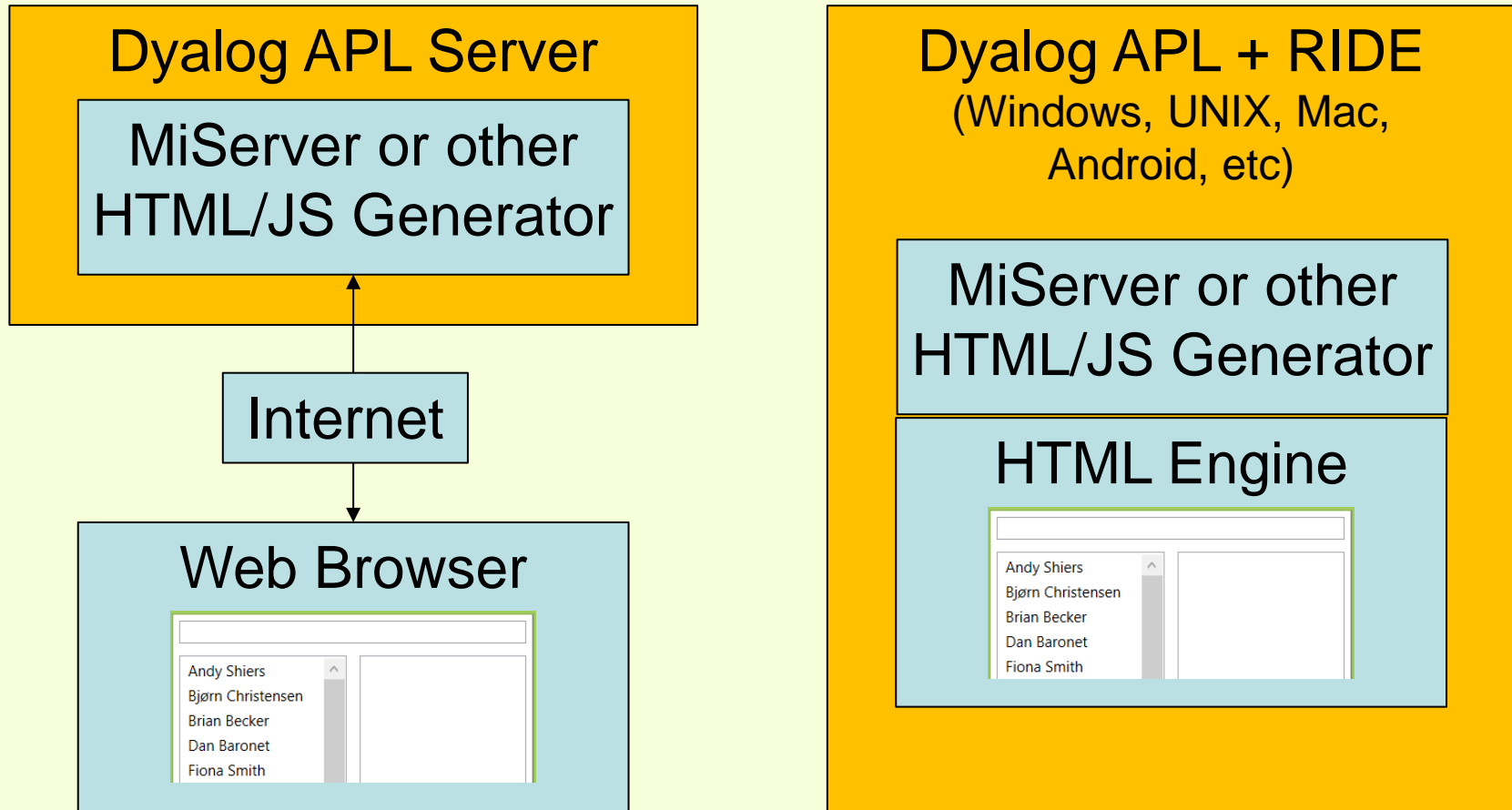
How to Get Started?

- Get it from <https://github.com/Dyalog/MiServer>

```
)LOAD miserver.dws  
Start 'MS3' @ The sample site  
]open http://127.0.0.1:8080
```



Not Only Web Applications



<c:\dropbox\Dyalog15\cefsimple\Release\loadurl.dws>



What else is Mi Server?

- A set of classes for creating GUI plus...
- A Web Server to deliver content to Web Browsers
 - And receive input from browsers
- Classes for delivering *Web Services* to application clients which render own GUI
 - "SAWS" is being integrated with MiServer and will also support RESTful web services
- A Framework for multi-user application deployment
- <C:\devt\miserver\ms3>



Summary – Cross Platform UI

- With version 3.0, MiServer is ready for use by “anyone who can write an APL function”
- Dan & Adám will present the growing library of supported Controls or “Widgets” tomorrow
- Easy to use by non-technical developers but easy for experts to extend using Javascript etc
- You can start writing apps that will render UI’s from Windows, Linux, AIX & OSX now!
- Android, UWP and iOS “soon”



Cross Platform

- From Cross Platform UI
to
Cross Platform File Functions

... Next up is Richard Smith

