

# *Web Server Primer*



Bill Keech

Sunbelt Computer Systems

# ***What Is PWS? OVERVIEW***



- **Client/Server PL/B runtime environment**
- **Like plbserve but with light clients**
- **Clients are standard web browsers or PlbWebCli**
- **Server contains more logic than plbserve**

# ***What Is PWS? OVERVIEW***



- **Communications uses HTTP and is connectionless**
- **Provides graphic user interface on any OS with a web browser**
- **Enhanced UNICODE support**
- **Same administration features as plbserve including WATCH**

# *Technologies Used*



- HTML 5 <http://www.w3schools.com/html/default.asp>
- JavaScript <http://www.w3schools.com/js/default.asp>
- CSS 3 <http://www.w3schools.com/css/default.asp>
- JQuery <http://jquery.com/>

# *Technologies Used*



- JQuery UI <http://jqueryui.com/>
- Jquery Mobile (.plm) <https://jquerymobile.com/>
- JSON <http://www.json.org/>
- REST <https://www.restapitutorial.com/>

# *Connectionless Communication*



- Runtime is not permanently connected to client
- Interaction performed through integrated HTML web server
- Allows client to be suspended and resumed
- iOS requires this feature with only one task running

# *Integrated HTML Web Server*



- Provides HTTP protocol engine
- Multi-threaded
- Works with both basic sockets and SSL
- CGI interface is only PL/B
- REST support
- .plc and .plm as mime type

# *Security/Performance*



- **PL/B compiler 'R' option prevents programs from being run directly**
- **Support for SSL (Secure Sockets Layer) to provide secure communications**
- **Programs can exist outside integrated HTML web server root directory**
- **Data is kept at the server**



# *Security/Performance*



- All messages are cached
- Flushing occurs only when user interaction is necessary
- Server-side runtime re-written to retain object information
- Support for an IP blacklist

# *Program Startup*



- .plc or .plm request is converted to send bootstrap page
- Page loaded is <program>.html
- If not found, then default plbwebstart.html or plbmobstart.html
- The bootstrap also loads plbwebuser.js

# *Program Startup*



- Page is processed to have program id
- Browser loads page and PL/B runtime integration JavaScript
- PlbWebCli performs these operations in a self-contained manner
- PLBWEB\_APP\_SUPPORT can prevent PlbWebCli access

# *Program Startup*



- **PL/B runtime task is created**
- **Communications takes place using integrated HTML web server**
- **Use runopt= to specify runtime options**
- **Use cmdline= for S\$CMDLIN support**

# *Program Startup*



- **PLBWEB\_USER\_CSS** is used to load user css
- **PLBWEB\_APP\_IMAGE\_URL** for PlbWebCli images
- **PLBWEB\_UPLOADS** for PlbWebCli uploads

# *Advanced Licensing*



- **2000 max child tasks**
- **PLBWEB\_MAX\_TASKS** can be used to reduce this number
- **Child tasks can be active, idle or suspended**
- **Idle or suspended tasks need no license**

# *Advanced Licensing*

- Idle when ALERT, ERROR dialog, KEYIN, MENU, or EVENTWAIT
- PLBWEB\_IDLE\_TIMEOUT sets the idle detection timeout
- PLBWEB\_IDLE\_MAX\_TIMEOUT sets the idle termination timeout
- PLBWEB\_MSG\_TIMEOUT sets the suspend detection timeout

# *Advanced Licensing*



- **PLBWEB\_SUSPEND\_MAX\_TIMEOUT** sets the suspend termination timeout
- **Child tasks that become active again must be able to get a license**



# *Basic Configuration*



- **Simplified configuration based upon keywords**
- **Setup like plbserve**
- **Use PLBWEB\_HOSTNAME and PLBWEB\_LOCALIP for command interaction**
- **Use PLBWEB\_ADDRESS keyword to assign an IP address and optional port**

# *Basic Configuration*



- **Port 80** can be used
- **PLBWEB\_PROGRAM\_ROOT** keyword specifies a root directory where PL/B programs are located
- **PLBWEB\_ROOT** specifies the integrated HTML web server root directory
- **PLBWEB\_DEFAULT** keyword specifies the default PLB program name or web page

# *Basic Configuration*

- **PLBWEB\_LOGNAME** keyword specifies the logfile name

- **Keyword(s) to setup SSL**

- **PLBWEB\_SSL\_CERTFILE**

- **PLBWEB\_SSL\_KEYFILE**

- **PLBWEB\_SSL\_PROTOCOL**

- **PLBWEB\_SSL\_SETCIPHER**

# *Advanced Configuration*

---

- **Use PLBWEB\_CGI\_INFODIR to capture CGI (Common Gateway Interface) data**
- **Use PLBWEB\_LOG\_HTTP to log HTTP information**
- **PLBWEB\_IMAGES keyword specifies a directory for extracted images**
- **PLBWEB\_SAVETOCLIENT keyword specifies a directory for download of a file**

# *Runtime Customization*



- **Error conditions are in html pages that can be customized**
- **Use PLBWEB\_USER\_CSS to specify a custom CSS file**
- **PLBWEB\_ERROR keyword defines behavior on an un-trapped error**
- **PLBWEB\_MENU keyword defines the behavior on runtime termination**

# *Runtime Customization*



- **SETMODEs** can change both **PLBWEB\_ERROR** and **PLBWEB\_MENU**
- **PLBWEB\_MSGBUFSIZE** keyword allows the message size to be increased

*Questions?*

