



The PHP Company

# PHP: Toolkit Examples

Mike Pavlak  
Solutions Consultant  
[mike.p@zend.com](mailto:mike.p@zend.com)  
(815) 722 3454



# Audience

---

- New to PHP
- Looking to leverage IBM i resources in web applications
- Seeking business value of PHP

# Agenda

---

- **Review Zend Server: Home of the Toolkit**
- **Overview of toolkit functions**
  - ▶ Studio Integration
- **Examples**
  - ▶ Data Area
  - ▶ Data Queue
  - ▶ Program Call
- **Explore 5250 bridge**
  - ▶ Bridge functions
  - ▶ Updated emulator
- **ITK - A new way to do IBM i functions**

# Questions?

---

- Let's keep it interactive!

- Follow us!



▶ <http://bit.ly/cjueZg> (Zend Technologies or search for Zend)



▶ <http://twitter.com/zend>

# PHP: Toolkit Examples

[www.zend.com](http://www.zend.com)

Review Zend Server

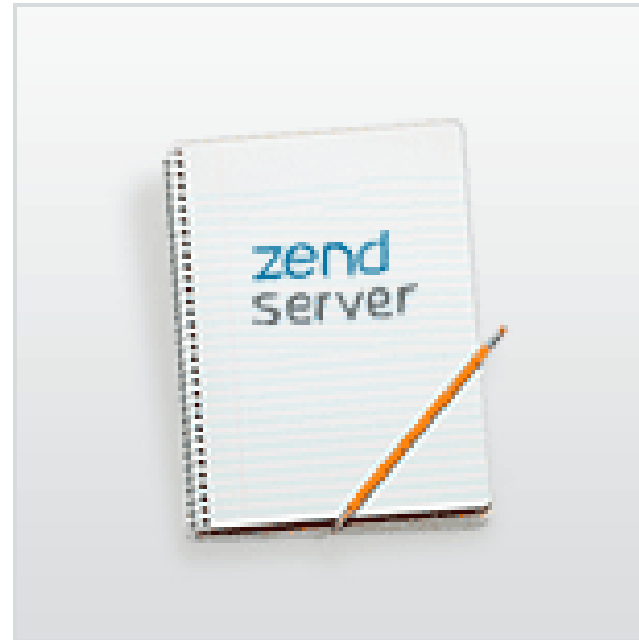
# Review of Zend Server

---

- Variables & Data Types
- Functions
- Copybooks

# Zend Server 5.0 GA on 04/12/2010!

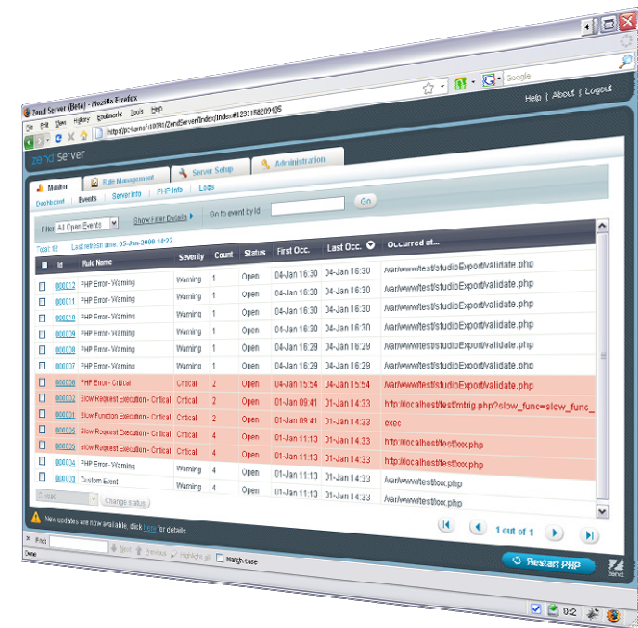
---



- The Linux and Windows distributions have been GA since Feb
- Code base consistency is making this gap smaller
- Training class available...see [zend.com/store](http://zend.com/store) for more info

# What Is Zend Server?

- Production quality PHP stack
  - ▶ PHP, ZF, DB connectivity, debugging extension, and more
- Two Editions - Free “Community Edition” provided by IBM & Full commercial edition
  - ▶ Both are production ready
- Application monitoring and diagnostics (integrated with Zend Studio)
- Multi-level performance enhancement capabilities
- Software updates and security hot fixes
- Easy and quick installation
- Zend Framework Integration





# Zend Server for IBM i

---

- Next generation of PHP stack for IBM i
- Best of both Zend Core and Zend Platform
- Single Licensed Program Installation
- Two products
  - ▶ Zend Server for IBM i Community Edition
    - Available at no charge per IBM partnership
  - ▶ Zend Server for IBM i
    - Subscription available from Zend
    - High value extra features
    - Higher Support SLAs

# Why a new generation?

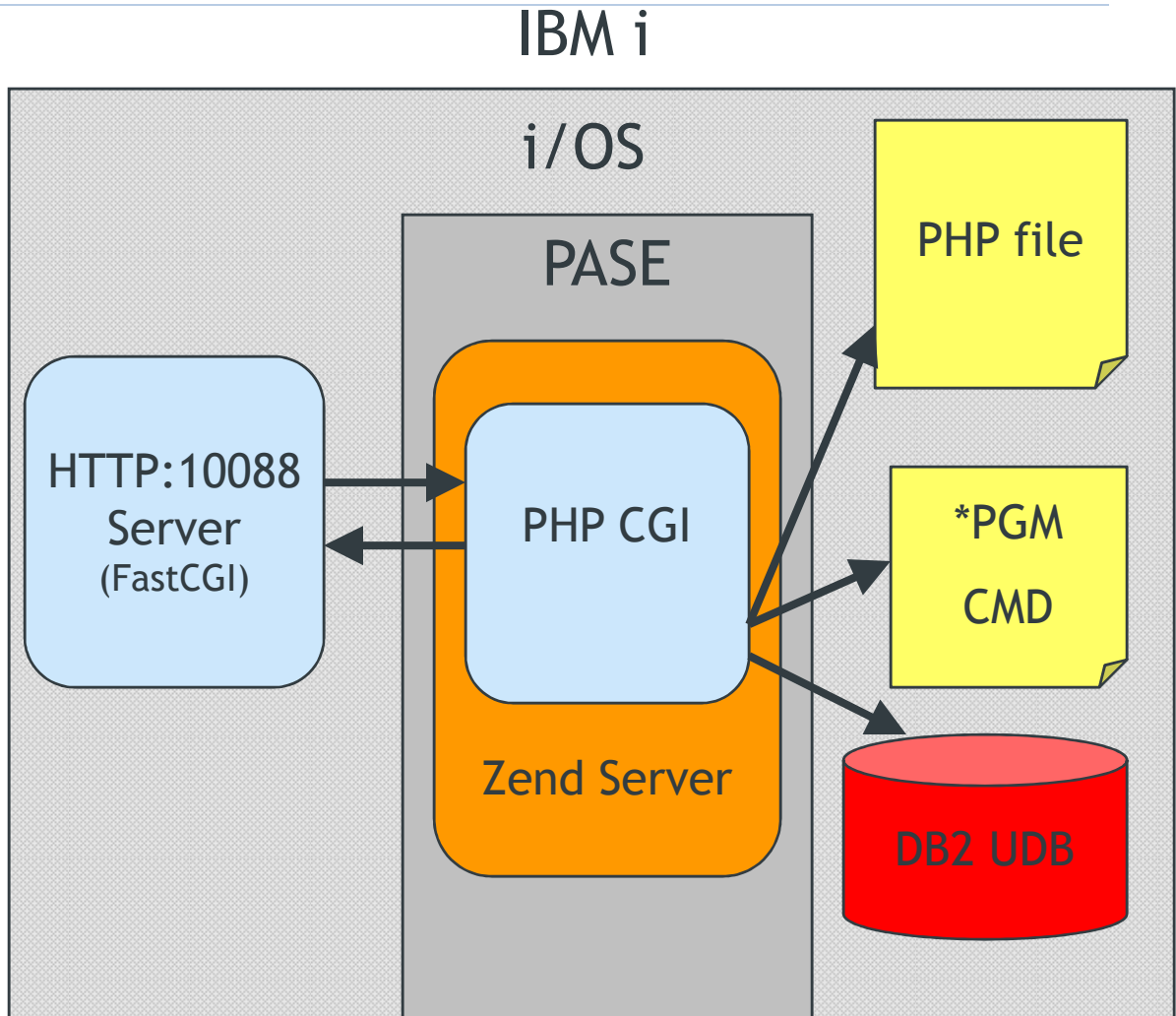
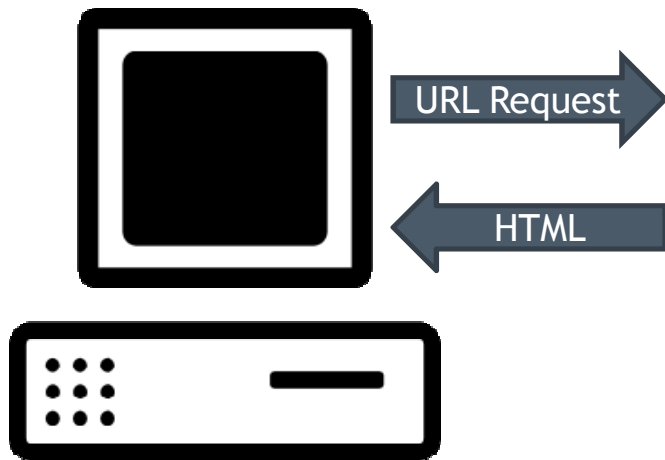
---

- **Based on feedback of key areas for Improvement**
- **Improved basic PHP Performance**
  - ▶ 30% to 600% depending on the application
- **New capabilities**
  - ▶ Code Tracing - Faster problem resolution
  - ▶ Job Queue - Performance and scalability
  - ▶ Page Caching - Performance and Scalability
  - ▶ Support for PHP 5.3
- **Easier installation, administration & maintenance**
  - ▶ Single apache server

# Zend Server Under the Covers

ILE Apache:10088

- Default configuration FastCGI



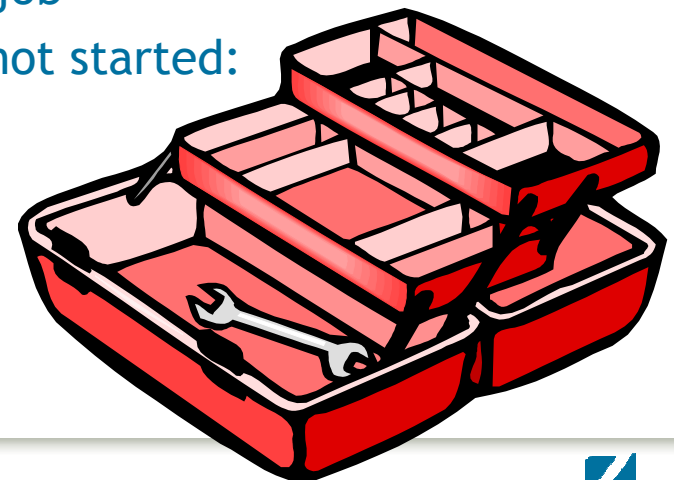
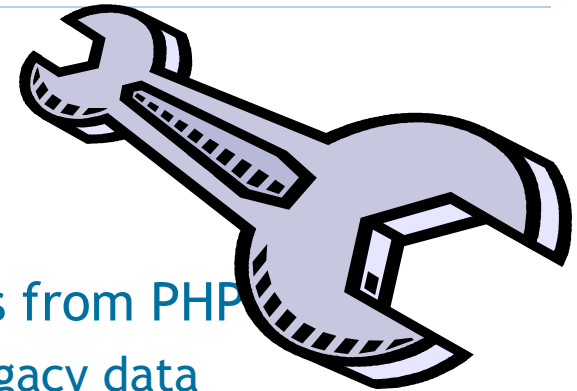
# PHP: Toolkit Examples

[www.zend.com](http://www.zend.com)

## Overview of Toolkit Functions

# i5 Toolkit APIs

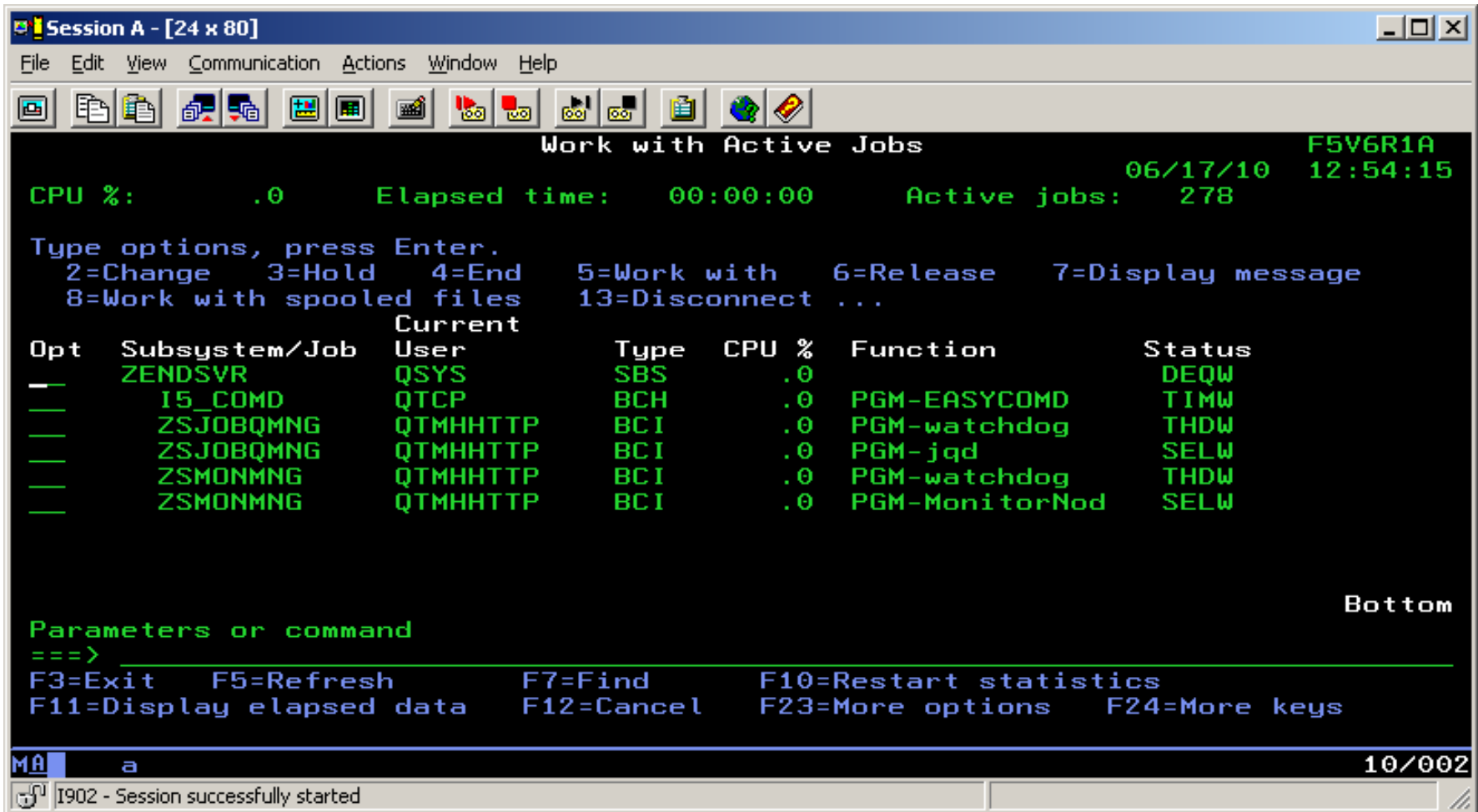
- Are shipped with Zend products
  - Zend Server for IBM i
  - Documented in the Zend Server User Guide
- Geared for accessing DB2 data & IBM I resources from PHP
  - Simplifies modern application integration with legacy data and applications
- Note: The I5\_COMD job must be running in the ZENDSVR subsystem
  - Use WRKACTJOB SBS(ZENDSVR) to see i5\_COMD job
  - Use the Zend menu to start the i5\_COMD job if not started:
    - GO ZENDSVR/ZSMENU
    - Option 5 (Service Management menu)
    - Option 8 (Start I5\_COMD service)



# Zend Server Menu

```
Session A - [24 x 80]
File Edit View Communication Actions Window Help
ZSMENU                               Zend Server for IBM i Setup Menu           System:  F5V6R1A
Select one of the following:
    1. Change password for Web Administration Console
    2. Update using Zend Server PTFs menu
    3. Run Support Tool
    5. Service Management menu
    6. MySQL Management menu
    7. 5250 Bridge Management Menu
    9. Reset Zend Server environment
   90. Signoff
Selection or command
===> _
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F23=WRKUSRJOB
MA a                                           21/007
I902 - Session successfully started
```

# Zend Server Subsystem



The screenshot shows a terminal window titled "Session A - [24 x 80]" with a menu bar (File, Edit, View, Communication, Actions, Window, Help) and a toolbar. The main display area shows the following information:

Work with Active Jobs F5V6R1A  
06/17/10 12:54:15

CPU %: .0      Elapsed time: 00:00:00      Active jobs: 278

Type options, press Enter.  
2=Change    3=Hold    4=End    5=Work with    6=Release    7=Display message  
8=Work with spooled files    13=Disconnect ...

| Opt | Subsystem/Job | User     | Type | CPU % | Function       | Status |
|-----|---------------|----------|------|-------|----------------|--------|
| —   | ZENDSVR       | QSYS     | SBS  | .0    |                | DEQW   |
| —   | I5_COMD       | QTCP     | BCH  | .0    | PGM-EASYCOMD   | TIMW   |
| —   | ZSJOBQMNG     | QTMHHTTP | BCI  | .0    | PGM-watchdog   | THDW   |
| —   | ZSJOBQMNG     | QTMHHTTP | BCI  | .0    | PGM-jqd        | SELW   |
| —   | ZSMONMNG      | QTMHHTTP | BCI  | .0    | PGM-watchdog   | THDW   |
| —   | ZSMONMNG      | QTMHHTTP | BCI  | .0    | PGM-MonitorNod | SELW   |

Bottom

Parameters or command  
==>

F3=Exit    F5=Refresh    F7=Find    F10=Restart statistics  
F11=Display elapsed data    F12=Cancel    F23=More options    F24=More keys

MA a 10/002

I902 - Session successfully started

# i5 Toolkit APIs

---

- Connection management
  - i5\_connect
  - i5\_close
  - i5\_adopt\_authority
  - i5\_error
  - i5\_errno
  - i5\_errormsg
- Command calls
  - i5\_command
- Program calls
  - i5\_program\_prepare
  - i5\_program\_prepare\_PCML
  - i5\_program\_call
  - i5\_program\_close
- Data retrieval
  - i5\_fetch\_array
  - i5\_fetch\_assoc
  - i5\_fetch\_object
  - i5\_fetch\_row
  - i5\_info
  - i5\_field\_len
  - i5\_field\_name
  - i5\_field\_scale
  - i5\_field\_type
  - i5\_list\_fields
  - i5\_num\_fields
  - i5\_result
- Native file access
  - i5\_open
  - i5\_addnew
  - i5\_edit
  - i5\_delete
  - i5\_cancel\_edit
  - i5\_setvalue
  - i5\_update
  - i5\_range\_from
  - i5\_range\_to
  - i5\_range\_clear
  - i5\_data\_seek
  - i5\_seek
  - i5\_bookmark
  - i5\_free\_file
  - i5\_new\_record
  - i5\_update\_record



# i5 Toolkit APIs

---

- System values
  - i5\_get\_system\_value
- Data areas
  - i5\_data\_area\_prepare
  - i5\_data\_area\_receive
  - i5\_data\_area\_send
  - i5\_data\_area\_close
- Print/Get spooled file
  - i5\_spool\_list
  - i5\_spool\_list\_read
  - i5\_spool\_list\_close
  - i5\_spool\_get\_data
  - i5\_spool\_from\_file
- Job logs
  - i5\_jobLog\_list
  - i5\_jobLog\_list\_read
  - i5\_jobLog\_list\_close
- Active jobs
  - i5\_job\_list
  - i5\_job\_list\_read
  - i5\_job\_list\_close
- Objects list
  - i5\_object\_list
  - i5\_object\_list\_read
  - i5\_object\_list\_close
- User space
  - i5\_userspace\_create
  - i5\_userspace\_prepare
  - i5\_userspace\_get
  - i5\_userspace\_put

# Connection

---

- Must connect to IBM i system to use any of the toolkit APIs
  - Connecting (i5 for i5 functions, db2 for DB2 functions)
    - i5\_connect
      - Need system, user profile name, and password as parameters
      - Can set library list here
    - i5\_close
      - Always close a connection you've opened
    - i5\_adopt\_authority
      - Can adopt other authorities while running
  - Error handling
    - i5\_error
      - Get data about an error
    - i5\_errno
      - Error number
    - i5\_errormsg
      - Error message

# PHP: Toolkit Examples

[www.zend.com](http://www.zend.com)

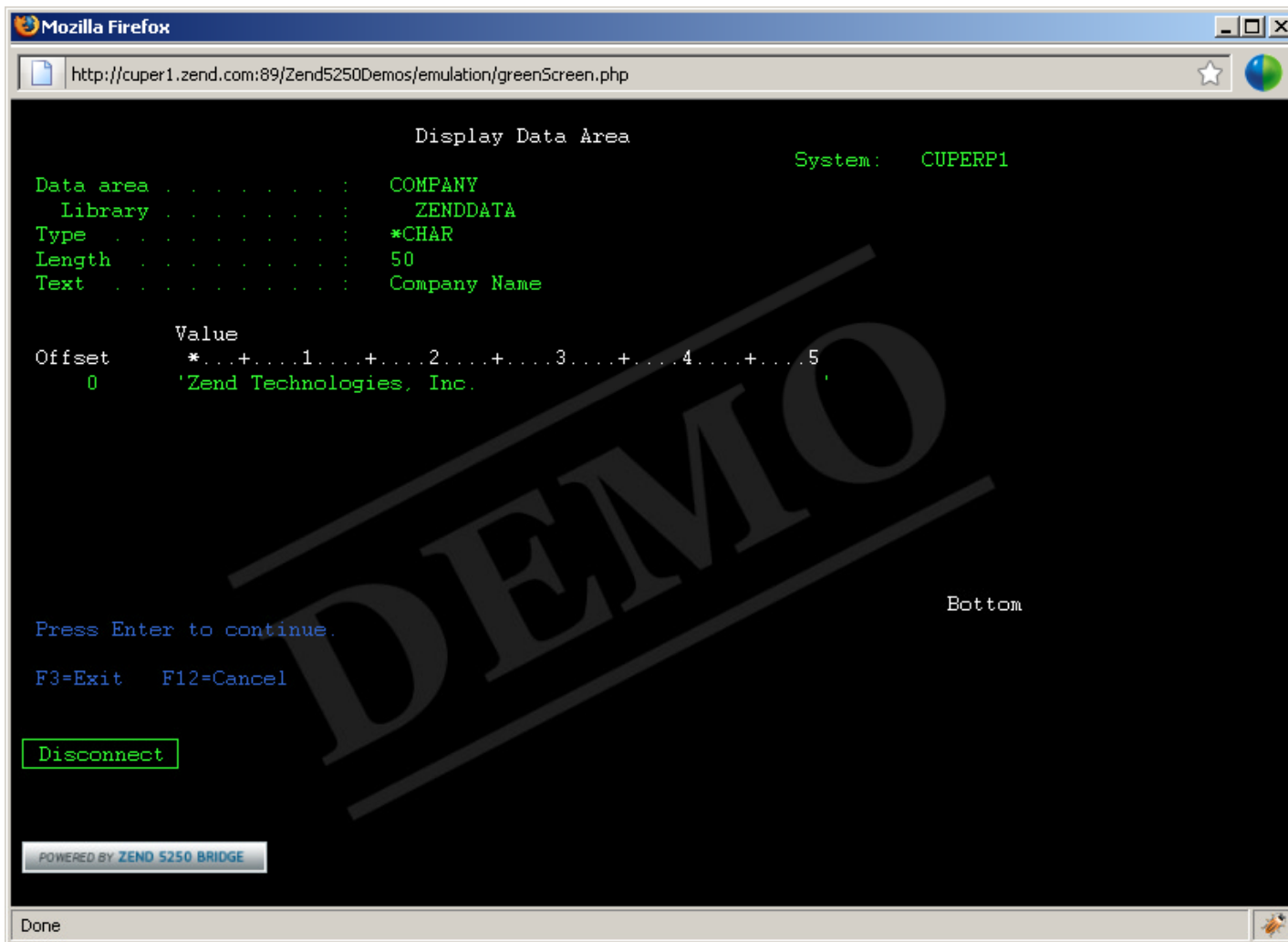
## Examples

# Two toolkit examples

---

- Data Area
- System Values
- Program call
- Spooled File Access

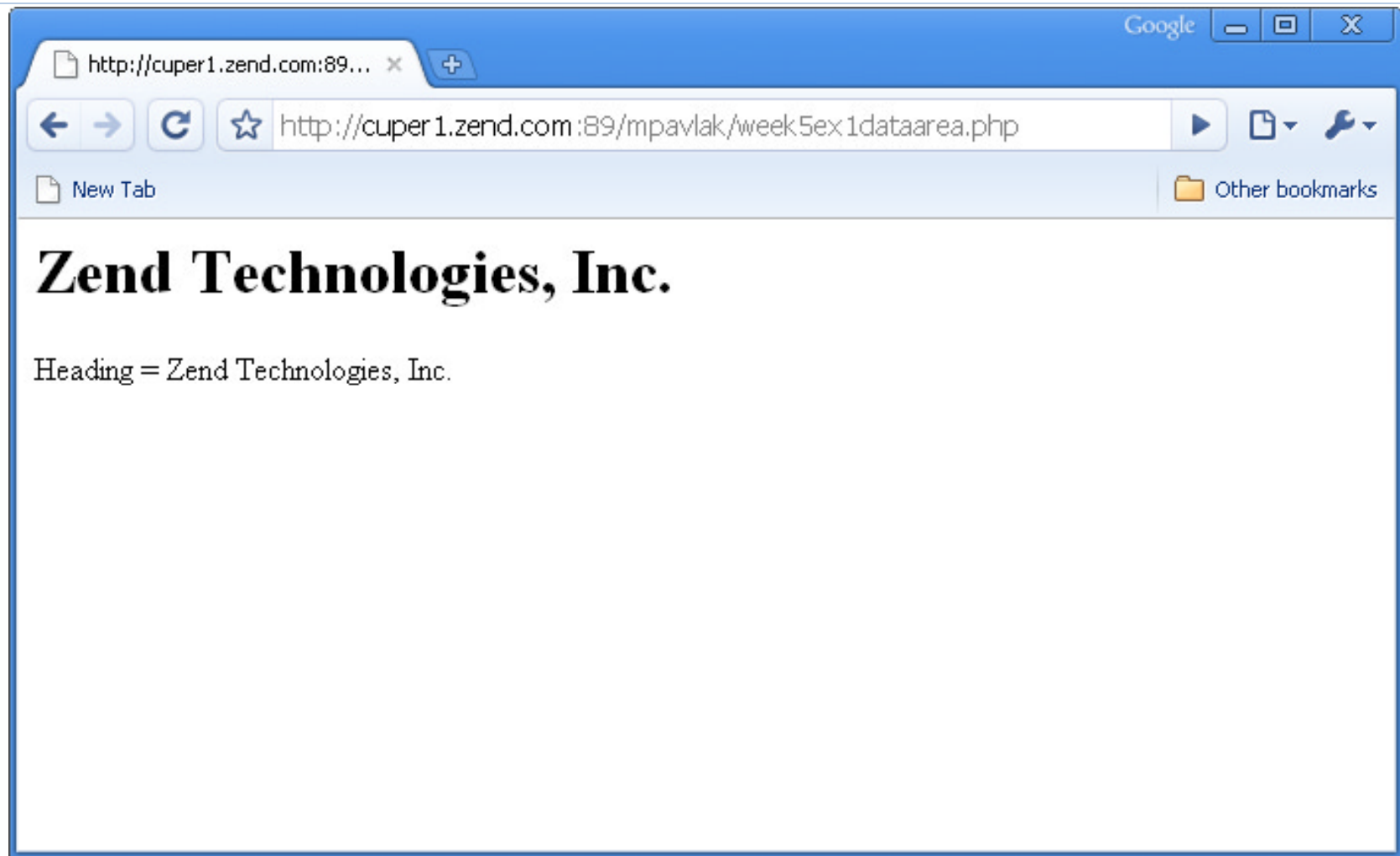
# Data Area Contents



# Data Area Code

```
1 <?php
2
3 include("i5db2connectonly.php");
4 $heading='test';
5
6 $heading = i5_data_area_read("ZENDDATA/COMPANY");
7 if (!$heading)
8 die("<br>data area read failed.");
9
10 echo "<h1>" . $heading . "</h1>";
11
12 echo "Heading = $heading";
13 i5_close($conn);
14
15 ?>
```

# Data Area Output



# System Values Code Part 1

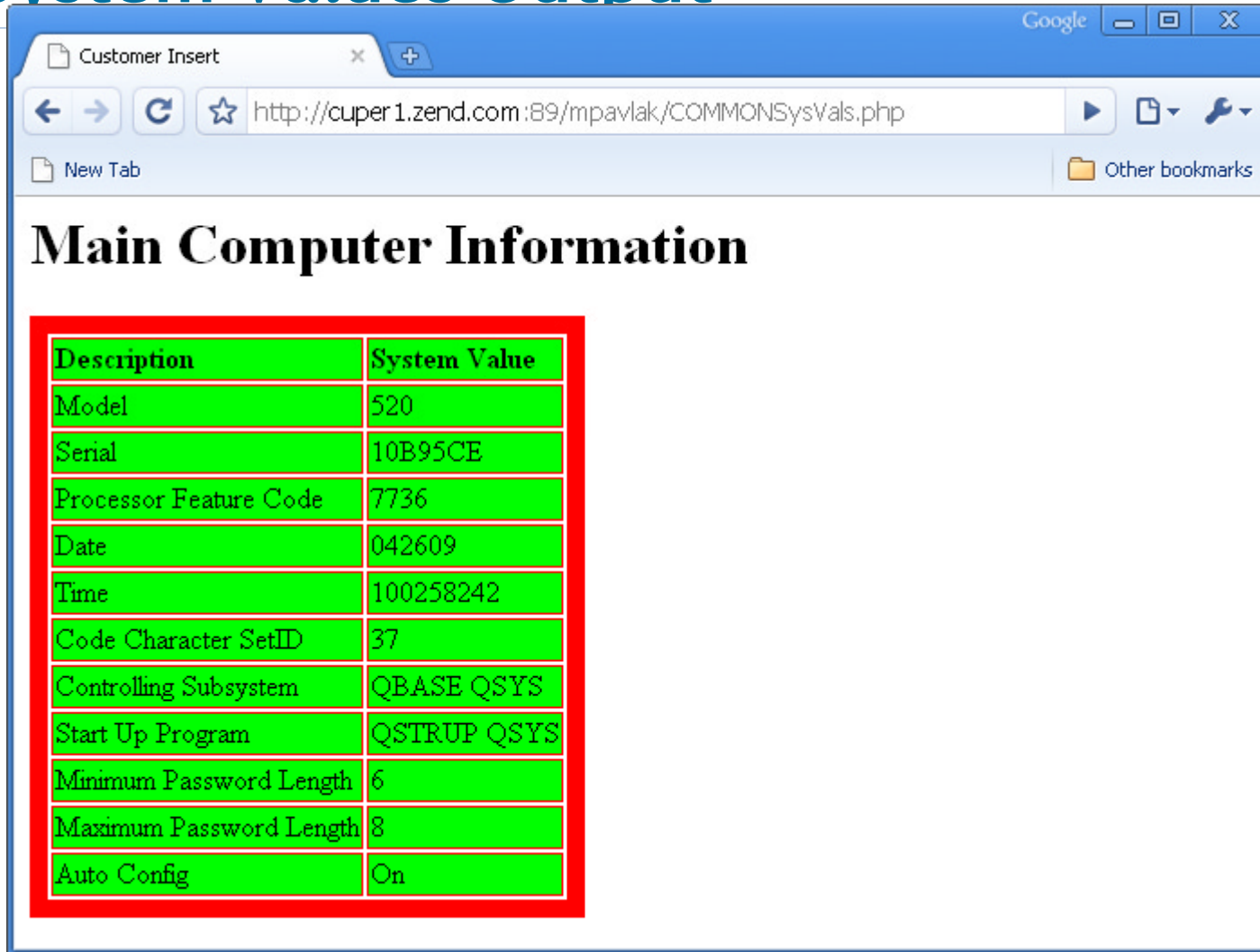
```
1 <html> <head><title>Customer Insert</title></head><body>
2
3 <h1>Main Computer Information</h1>
4
5 <?php
6
7 include("i5db2connectonly.php");
8
9 // $conn = i5_connect("localhost", "PHPDEV", "phpdev1");
10
11 //if (!$conn)
12 //die("<br>Connection failed. Error number = " . i5_erro() . " msg = " . i5_errormsg());
13
14 //Define table and start filling rows...
15 echo '<TABLE BORDER="10" BORDERCOLOR="red"><TBODY BGCOLOR="00FF00">';
16 echo '<TR><TD><B>Description</TD><TD><B>System Value</TD></TR>';
17
18 print "<TR><TD>Model</TD><TD>" . i5_get_system_value("QMODEL") . "</TD></TR>";
19
20 print "<TR><TD>Serial</TD><TD>" . i5_get_system_value("QSRLNBR") . "</TD></TR>";
```



## System Value Code Part 2

```
34 print "<TR><TD>Minimum Password Length</TD><TD>" . i5_get_system_value("QPWDMINLEN") . "</TD></TR>";
35
36 print "<TR><TD>Maximum Password Length</TD><TD>" . i5_get_system_value("QPWDMAXLEN") . "</TD></TR>";
37
38 print "<TR><TD>Auto Config</TD><TD>";
39
40 if(i5_get_system_value("QAUTOCFG") == 1) {
41     echo "On";
42 }
43 else {
44     echo "Off";
45 }
46
47 print "</TD></TR>";
48
49
50 echo '</table>';
51 i5_close($conn);
52 ?>
```

# System Values Output



The screenshot shows a web browser window with the following details:

- Tab: Customer Insert
- Address Bar: <http://cuper1.zend.com:89/mpavlak/COMMONSysVals.php>
- Page Title: Main Computer Information

The main content is a table with 12 rows, each with a description and a system value. The table is highlighted with a red border.

| Description             | System Value |
|-------------------------|--------------|
| Model                   | 520          |
| Serial                  | 10B95CE      |
| Processor Feature Code  | 7736         |
| Date                    | 042609       |
| Time                    | 100258242    |
| Code Character SetID    | 37           |
| Controlling Subsystem   | QBASE QSYS   |
| Start Up Program        | QSTRUP QSYS  |
| Minimum Password Length | 6            |
| Maximum Password Length | 8            |
| Auto Config             | On           |

# Program Call...

---

- Most popular use of the API Toolkit
- Can call ANY IBM i program
- Our example will show CL
- Three parameters
- Concatenate a string
- Steps:
  - Parameter setup
  - Prepare the program
  - Load the i/o array of parameters & map return values
  - Call the program
  - Retrieve values

# Program Call, script...Parameters

---

```
<h1>Village Water Billing System</h1>
<h2>Program Call</h2>
<?php

include("i5db2connectlib.php");

// Setup parameters in associative array...
$desc = array (
    array ("name"=>"string1", "io"=>I5_INOUT, "type" => I5_TYPE_CHAR,
        "length"=> "10"),
    array ("name"=>"string2", "io"=>I5_INOUT, "type" => I5_TYPE_CHAR,
        "length"=> "10"),
    array ("name"=>"string3", "io"=>I5_INOUT, "type" => I5_TYPE_CHAR,
        "length"=> "20"),
);
```

# Program Call, script...Prepare

---

```
// Prepare the program, similar to prototype in ILE...
$prog = i5_program_prepare("clp1", $desc);
if ($prog === FALSE)
{
    $errorTab = i5_error();
    echo "Program prepare failed <br>";
    var_dump($errorTab);
    die();
}
```

# Program Call, script...Load parms

---

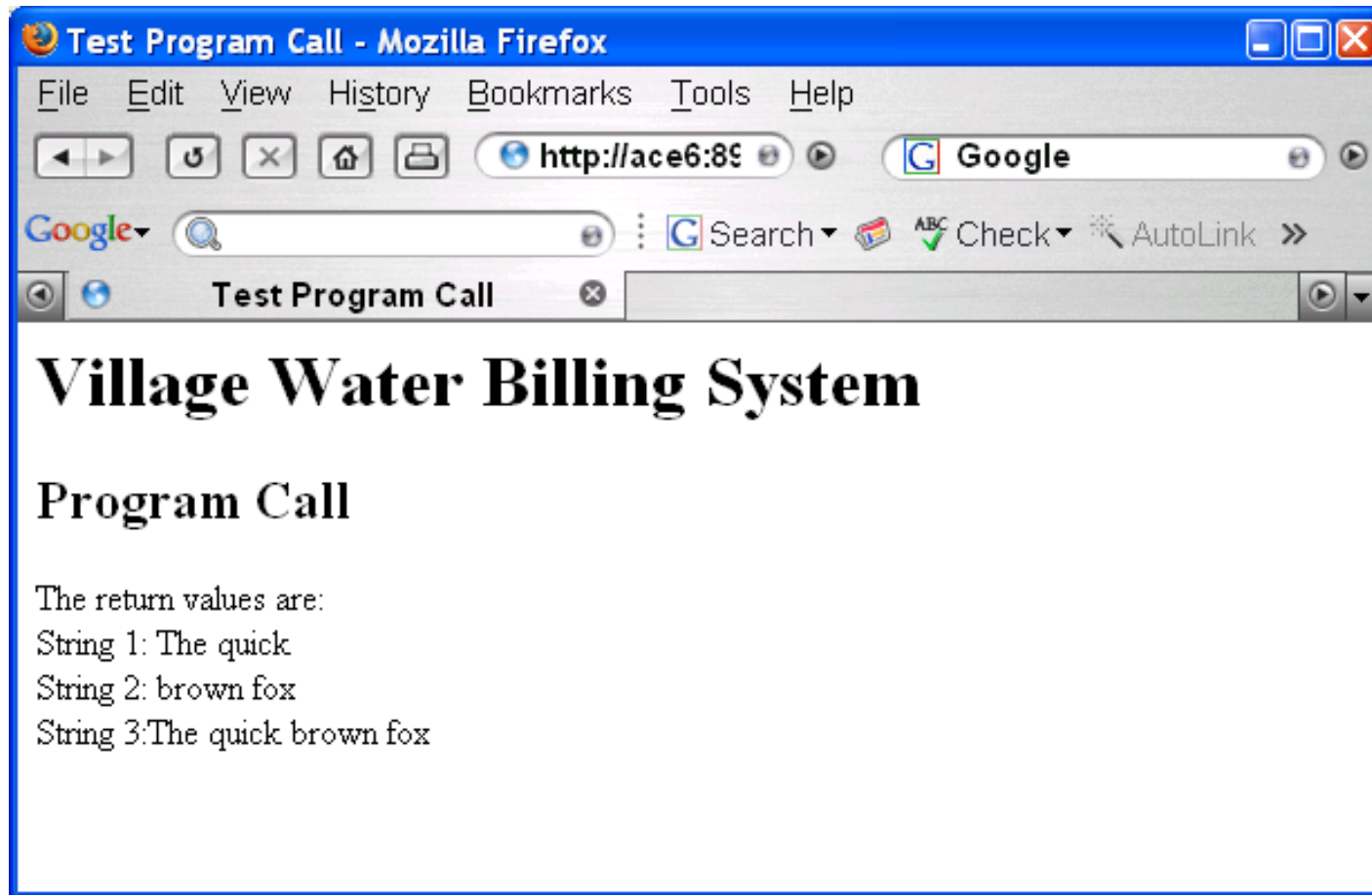
```
// Load parameters...
$params = array ("string1"=>"The quick ", "string2"=>"brown fox ", "string3"=>"");

// Map parameters to variables...
$retvals =
array("string1"=>"string1", "string2"=>"string2", "string3"=>"string3");
```

## Program Call, script...Execute!

```
$ret = i5_program_call($prog, $params, $retvals);  
if ($ret === FALSE)  
{  
    $errorTab = i5_error();  
    echo "FAIL : i5_program_call failure code <br>";  
    var_dump($errorTab);  
    die();  
}
```

# Program Call, example...



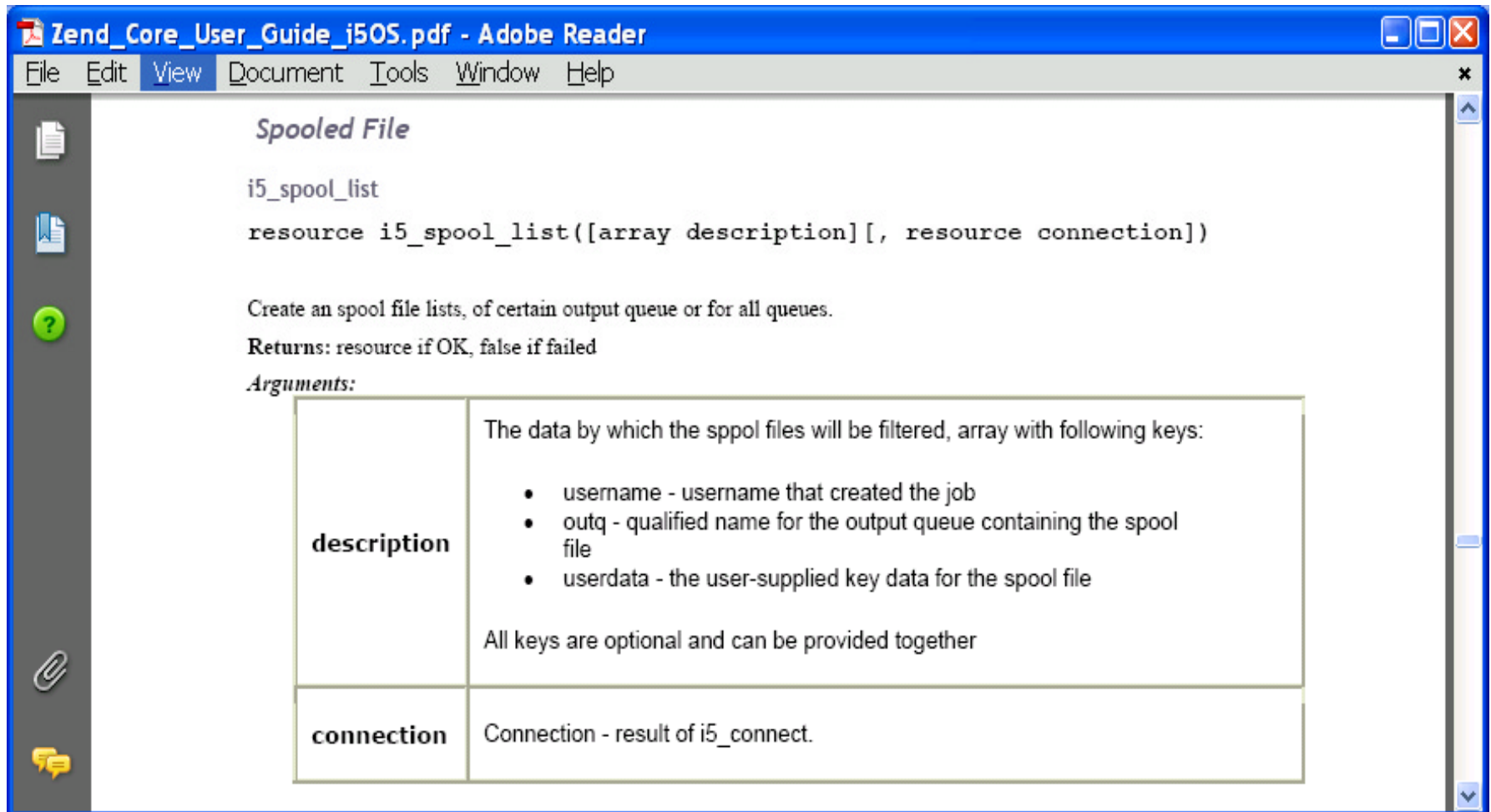


## Spooled file listing...

---

- Everybody has them
- Used to print miles of standard reports
- Many utilities to make them “pretty”
- List them
- Maybe do something more

# Spooled file listing...



**Spooled File**

```
i5_spool_list  
resource i5_spool_list([array description][, resource connection])
```

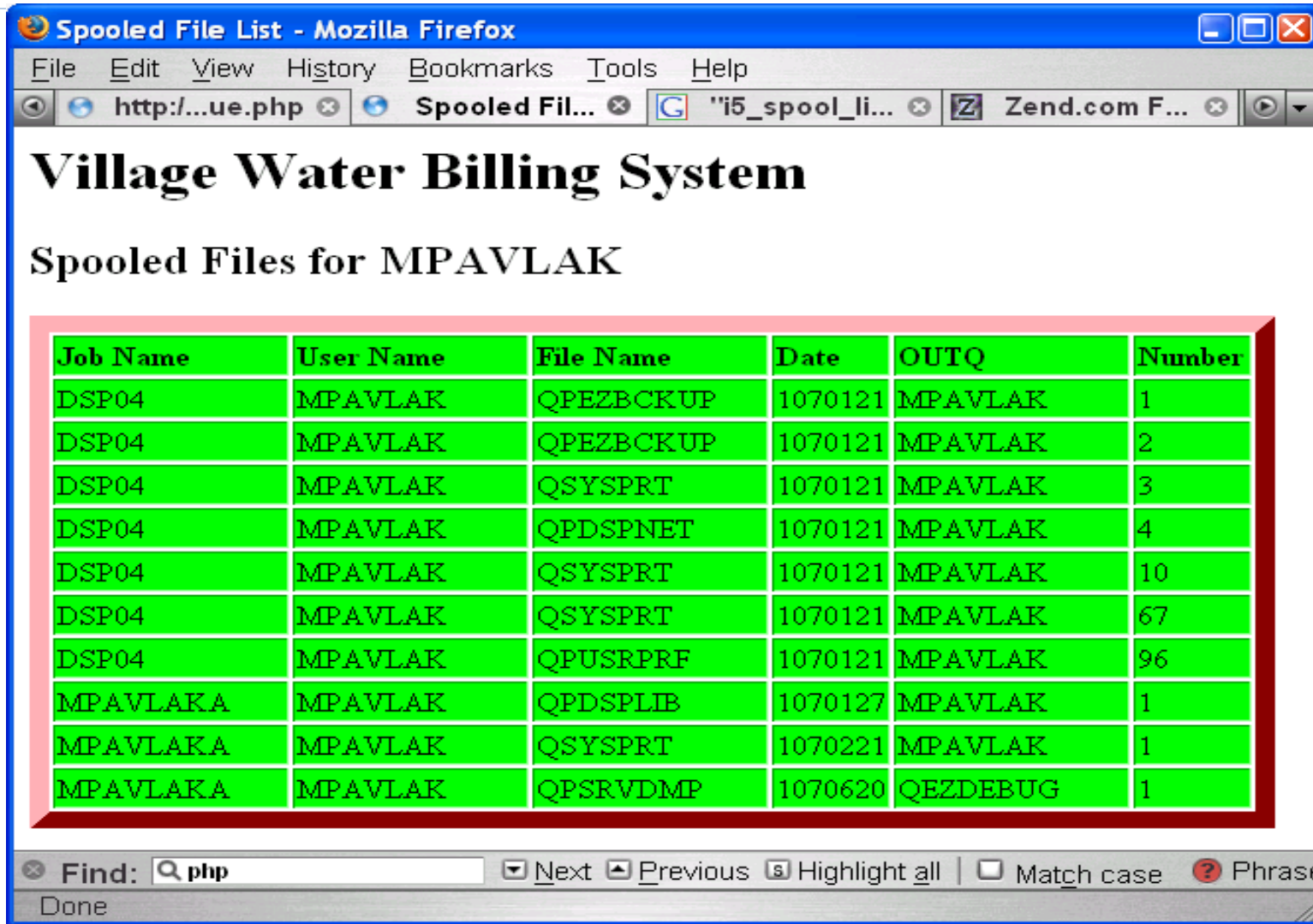
Create an spool file lists, of certain output queue or for all queues.

**Returns:** resource if OK, false if failed

**Arguments:**

|                    |  |
|--------------------|--|
| <b>description</b> | The data by which the spool files will be filtered, array with following keys: <ul style="list-style-type: none"><li>• username - username that created the job</li><li>• outq - qualified name for the output queue containing the spool file</li><li>• userdata - the user-supplied key data for the spool file</li></ul> All keys are optional and can be provided together |
| <b>connection</b>  | Connection - result of i5_connect.   |

# Spooled file listing...



**Village Water Billing System**

**Spooled Files for MPAVLAK**

| Job Name | User Name | File Name | Date    | OUTQ     | Number |
|----------|-----------|-----------|---------|----------|--------|
| DSP04    | MPAVLAK   | QPEZBCKUP | 1070121 | MPAVLAK  | 1      |
| DSP04    | MPAVLAK   | QPEZBCKUP | 1070121 | MPAVLAK  | 2      |
| DSP04    | MPAVLAK   | QSYSRPT   | 1070121 | MPAVLAK  | 3      |
| DSP04    | MPAVLAK   | QPDSNET   | 1070121 | MPAVLAK  | 4      |
| DSP04    | MPAVLAK   | QSYSRPT   | 1070121 | MPAVLAK  | 10     |
| DSP04    | MPAVLAK   | QSYSRPT   | 1070121 | MPAVLAK  | 67     |
| DSP04    | MPAVLAK   | QPUSRPRF  | 1070121 | MPAVLAK  | 96     |
| MPAVLAKA | MPAVLAK   | QPDSPLIB  | 1070127 | MPAVLAK  | 1      |
| MPAVLAKA | MPAVLAK   | QSYSRPT   | 1070221 | MPAVLAK  | 1      |
| MPAVLAKA | MPAVLAK   | QPSRVDMP  | 1070620 | QEZDEBUG | 1      |

Find:   Next  Previous  Highlight all  Match case  Phrase

Done

# Spooled file listing...

---

- **Need some particulars**
  - User
  - Output Queue
  - User data
- **Break code and conquer!**
  - Create the handle
  - Read through the table

# Spooled file listing...

```
<html> <head><title>Spooled File List</title></head><body>

<h1>Village Water Billing System</h1>

<?php
$userName="MPAVLAK";

echo "<h2>Spooled Files for " . $userName . " </h2>";
include("i5db2connectonly.php");

$HdlSpl = i5_spool_list(array(I5_USERNAME=>$userName));
if (!$HdlSpl){
    $ret = i5_errno();
    print_r($ret);
}

echo '<TABLE BORDER="10" BORDERCOLOR="red"><TBODY BGCOLOR="00FF00">';
echo '<TR><TD><B>Job Name</TD><TD><B>User Name</TD><TD><B>File Name</TD>
      <TD><B>Date</TD><TD><B>OUTQ</TD><TD><B>Number</TD></TR>';
```

# Spooled file listing...

```
$continue=true;
while ($continue){
    $ret = i5_spool_list_read($HdlSpl);
    if (!$ret){
        $erreur = i5_error();
        if ($erreur["num"] != 14){
            // error code here...
        }
        $continue = false;
        break;
    }
}
?>
<tr><td width="20%"><?php echo $ret["JOBNAME"]; ?> </td>
    <td width="20%"><?php echo $ret["USERNAME"]; ?> </td>
    <td width="20%"><?php echo $ret["SPLFNAME"]; ?> </td>
    <td width="10%"><?php echo $ret["DATEOPEN"]; ?> </td>
    <td width="20%"><?php echo $ret["OUTQNAME"]; ?> </td>
    <td width="20%"><?php echo $ret["SPLFNBR"]; ?> </td></tr>
<?php
}
```

# PHP: Toolkit Examples

[www.zend.com](http://www.zend.com)

## 5250 Bridge & Emulator

# What is the "Bridge"

---

- **Three sets of APIs that allow a PHP script to drive a 5250 session**
  - ▶ Green Screen Simulator
  - ▶ Object Oriented APIs
  - ▶ Procedural APIs
    - During the session these are the ones we will be focusing on
- **Multiple 5250 sessions can be connected to a single script**
  - ▶ So you could combine the data from several green screen applications in a single browser window!
- **Bridge is available as part of Zend Platform App Server**



# Product Definition

- ▶ The 5250 Bridge is a PHP based solution for running interactive applications in the i5/OS environment

```
Date: 12/17/07      User: Admin@ztd.com      Zend Technologies
Time: 00:59:40      5250 Bridge Version 1.0

Position to. . . . .

Type options, press Enter.
5-Display

Opr  User ID      Serial Number  Application Name  Status
---  ---
---  BENJAMIN      1848142       BRIDGE            1 ACTIVE
---  BEVERLY       6532972       MEDICAL           1 ACTIVE
---  DIANA         6532972       MEDICAL           1 ACTIVE
---  JAMES        1848142       BRIDGE            1 ACTIVE
---  JIM          2114748       STAFF             1 ACTIVE
---  KATE         1848142       BRIDGE            1 ACTIVE
---  NICHOLAS     2114748       STAFF             1 ACTIVE
---  WILLIAM      7845152       TACTICAL          1 ACTIVE

Bottom

F3-Exit
Response

© Copyright Zend Technologies 2008.
```



PHP Application for i5/OS 29 Jan 2008, Tuesday 09:27AM Discarded

User Maintenance - demo program ZEND5250/ZMI001R

| User id  | Serial no. | Details                 |
|----------|------------|-------------------------|
| BENJAMIN | 1848142    | <a href="#">Details</a> |
| BEVERLY  | 6532972    | <a href="#">Details</a> |
| DIANA    | 6532972    | <a href="#">Details</a> |
| JAMES    | 1848142    | <a href="#">Details</a> |
| JIM      | 2114748    | <a href="#">Details</a> |
| KATE     | 1848142    | <a href="#">Details</a> |
| NICHOLAS | 2114748    | <a href="#">Details</a> |
| WILLIAM  | 7845152    | <a href="#">Details</a> |



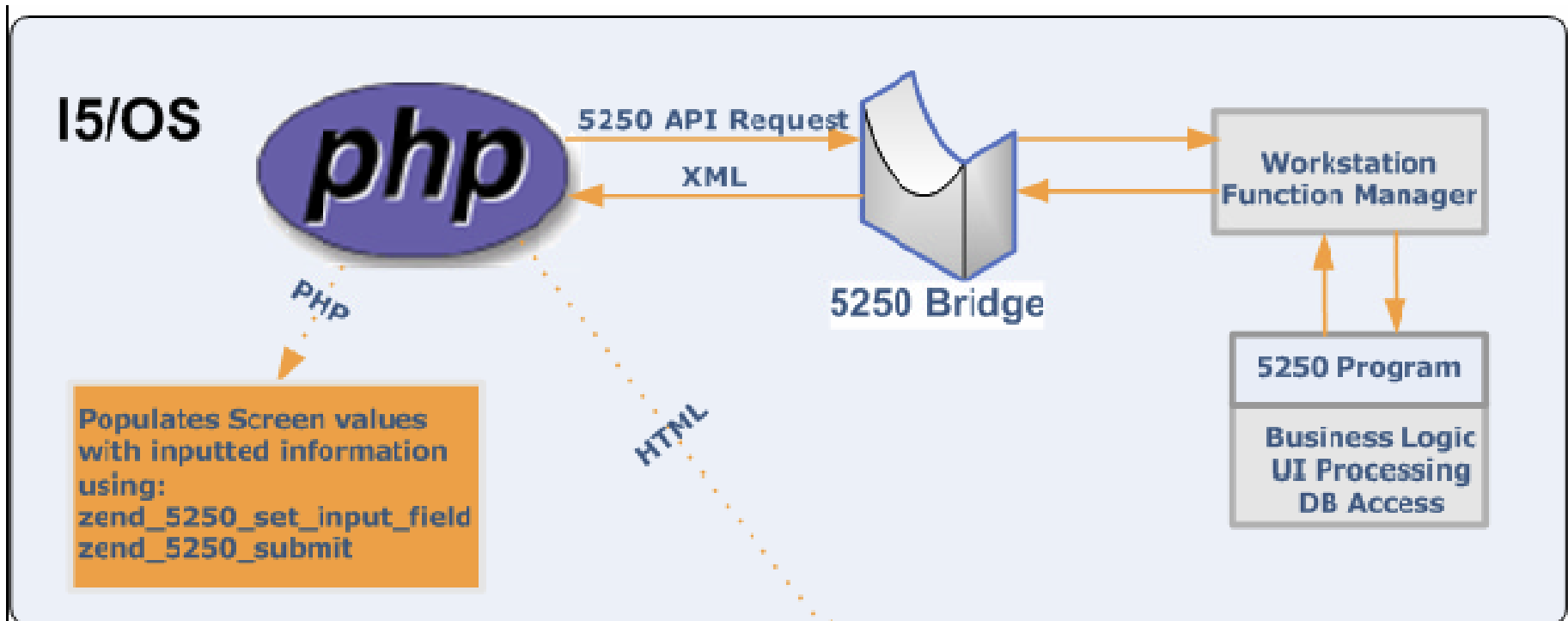
**BENJAMIN** 

Serial: #1848142  
Application: BRIDGE  
Status: Active  
Number of Sign On: 3

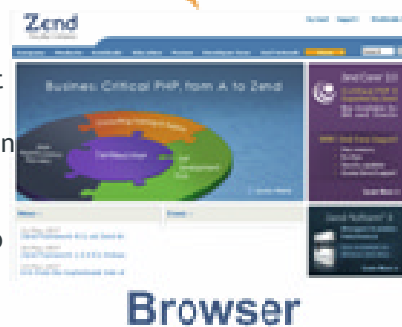
© 2008 Zend Technologies

LICENSED BY ZEND 5250 BRIDGE

# 5250 Bridge Diagram



1. PHP sends a 'start session' request to the 5250 Program using the 5250 Bridge.
2. The 5250 Bridge returns a confirmation to PHP that the session has started.
3. The 5250 Bridge sends current screen information in XML format to PHP (e.g. format name, input and output fields and current cursor position).
4. The input values can be automatically entered into the program or sent on in an html format for user input.



# ARRAY DUMP OF RETURN SET

```
1. array(5)
2. { [0]= array(8) { ["id"]= int(0) ["row"]= int(6)
3. ["column"]= int(53) ["length"]= int(10) ["value"]=
4. string(10) " " ["type"]= string(11) "Alpha shift" ["font"]=
5. string(11) "Not defined" ["format"]= string(11) "Not Defined" }

6. [1]= array(8){ ["id"]= int(1) ["row"]= int(7)
7. ["column"]= int(53) ["length"]= int(10) ["value"]=
8. string(10) " " ["type"]= string(11) "Alpha shift" ["font"]=
9. string(11) "Not defined" ["format"]= string(11) "Not Defined" }

10. [2]= array(8) { ["id"]= int(2) ["row"]= int(8)
11. ["column"]= int(53) ["length"]= int(10) ["value"]= string(10) " " ["type"]= string(11) "Alpha shift"
    ["font"]=
12. string(11) "Not defined" ["format"]= string(11) "Not Defined" }

13. [3]= array(8) { ["id"]= int(3) ["row"]= int(9)
14. ["column"]= int(53) ["length"]= int(10) ["value"]=
15. string(10) " " ["type"]= string(11) "Alpha shift" ["font"]=
16. string(11) "Not defined" ["format"]= string(11) "Not Defined" }

17. [4]= array(8) { ["id"]= int(4) ["row"]= int(10)
18. ["column"]= int(53) ["length"]= int(10) ["value"]= string(10) " " ["type"]= string(11) "Alpha shift"
    ["font"]=
19. string(11) "Not defined" ["format"]= string(11) "Not Defined" } }
```

# Function Junction

ITK Class Library - Kevin Schroeder

# A bit about me

---

- Kevin Schroeder
  - ▶ Technology Evangelist - Zend Technologies
  - ▶ Blog: <http://www.eschrade.com>
  - ▶ Email: [kevin@zend.com](mailto:kevin@zend.com)

# ITK

---

- **What is it?**
  - ▶ An easy to use interface into the i5 PHP Toolkit
  - ▶ Fully unit tested
- **Why use it?**
  - ▶ It drastically simplifies accessing RPG functionality in PHP
- **What's it cost?**
  - ▶ Nothing
- **Where do I get it?**
  - ▶ <http://www.github.com/...>

---

# How does one currently access RPG functionality in PHP?

---

# How can one now access RPG functionality in PHP?



# Concepts

---

- **Adapters**
  - ▶ Represents the interface to the functionality be accessed
- **Programs**
  - ▶ Represents the program definition
- **Unit Tests**
  - ▶ Mechanisms for testing whether or not program functions are properly handling RPG-based data

# Adapters

---

- **Two adapters are included**
  - ▶ **Live**
    - Represents an individual connection to the PHP Toolkit implementation
    - Used in production scenarios
  - ▶ **Mock**
    - Used for Unit Testing
    - Will return specific results for specific calls
    - Allows you to test that your class is properly handling data
- **A Program class can be called regardless of the adapter being used**

# Programs

---

- Based on the `Itk_PgmAbstract` class
- Definitions required

- ▶ Program Name

- The name of the RPG program being called

```
protected $_programName = 'ZENDSVR/COMMONPGM';
```

- ▶ Description

- Describes the input/output parameters

```
protected $_description = array(  
    'CODE'=> array(  
        self::DESC_IO=> I5_INOUT,  
        self::DESC_TYPE=> I5_TYPE_CHAR,  
        self::DESC_LENGTH=> "10"  
    )  
);
```

# Unit Testing

---

- **The entire library is designed to be easily testable**
  - ▶ Repeatability in testing is KEY to not losing your hair
  - ▶ Well written tests ensure that changes to not break compatibility
- **Unit Tests can be run using**
  - ▶ Live adapter on i Series machines
  - ▶ Mock adapter on either
    - i Series machines (if you don't want to change data)
    - Linux/Windows workstations
- **Unit Tests are run using PHPUnit**

# Recommendations

---

- Use a bootstrap to define the adapter at the front of the request
- Keep as much code out of your HTML as possible
- If you are not familiar with OOP this is a good place to start
  - ▶ Small example
  - ▶ Follows good practices
  - ▶ Demonstrates the benefits on a small scale

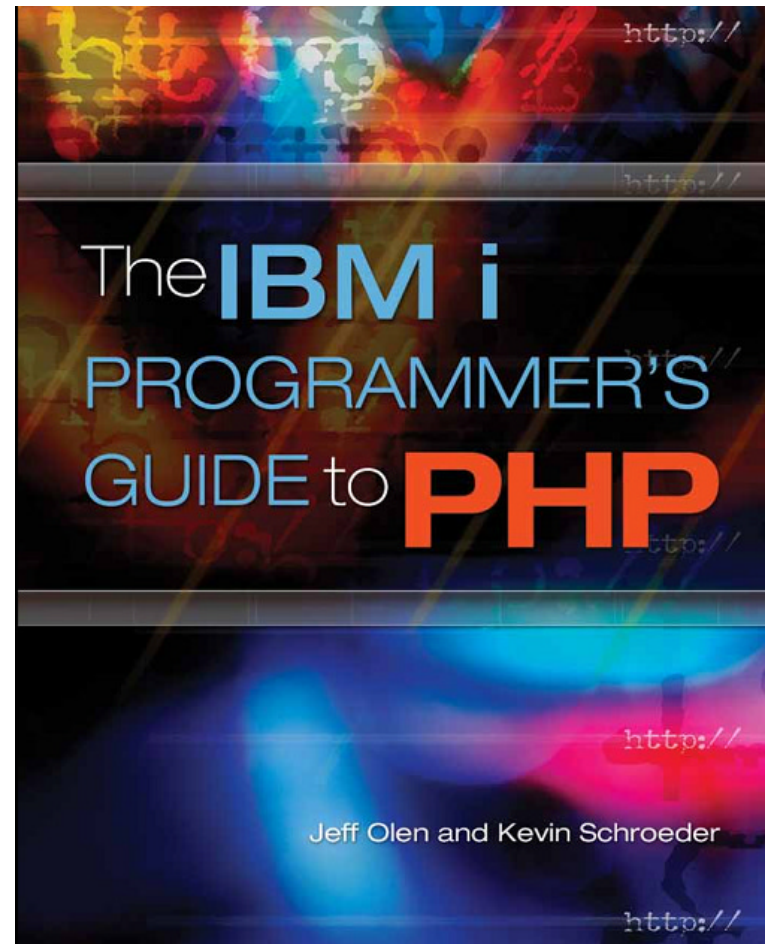
# New book, new printing, same great stuff!

Kevin Schroeder from Zend's  
Global Services Group

with

Jeff Olen, co-author of...

Get yours at [MCPressonline](http://MCPressonline)  
or at fine bookstores everywhere



# Resources

---

- Recorded Webinars

- ▶ <http://www.zend.com/en/resources/webinars/i5-os>

- Zend Server for IBM i main page, link to downloads

- ▶ <http://www.zend.com/en/products/server/zend-server-ibm-i>

- Zend Server manual:

- ▶ PDF: <http://www.zend.com/topics/Zend-Server-5-for-IBMi-Reference-Manual.pdf>

- ▶ Online: [http://files.zend.com/help/Zend-Server-5/zend-server.htm#installation\\_guide.htm](http://files.zend.com/help/Zend-Server-5/zend-server.htm#installation_guide.htm)

# Thank you & Q&A

---

- Follow us!



▶ <http://bit.ly/cjueZg> (Zend Technologies or search for Zend)



▶ <http://twitter.com/zend>



# Q&A

[www.zend.com](http://www.zend.com)

[mike.p@zend.com](mailto:mike.p@zend.com)

*Please fill out your  
Session Evaluation!*

A small thumbnail image of a session evaluation form. The form is titled "Zend Business & Marketing Evaluation" and "Professional Development Evaluation". It contains several sections with checkboxes and a table for ratings. The table has columns for "Rating" and "Comments". The form is partially filled out with handwritten text.