



The PHP Company

Improving Application Performance by Submitting Scripts to Batch using Zend Server for IBM i

Mike Pavlak

Solution Consultant

mike.p@zend.com



Agenda

- Overview of Zend Server
- Advantages of asynchronous processing
- Zend Server CE for IBM i Batch Process
- Zend Server for IBM i
 - ▶ Scheduled jobs
 - ▶ Jobs spawned from other jobs
- Monitoring jobs
- Compare CE and full solution
- Wrap-up

Questions?

- If you have a question you can
 - ▶ Type in the Chat panel

- Follow us!

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

 ▶ <http://twitter.com/zend>

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

On the road with Zend



- MITEC - Livonia, Michigan

June 8, 2010

▶ www.gomitec.com

- OCEAN - Orange County, CA

July 16, 2010

▶ www.ocean400.org

- COMMON - San Antonio, TX

October 4-6, 2010

▶ www.common.org



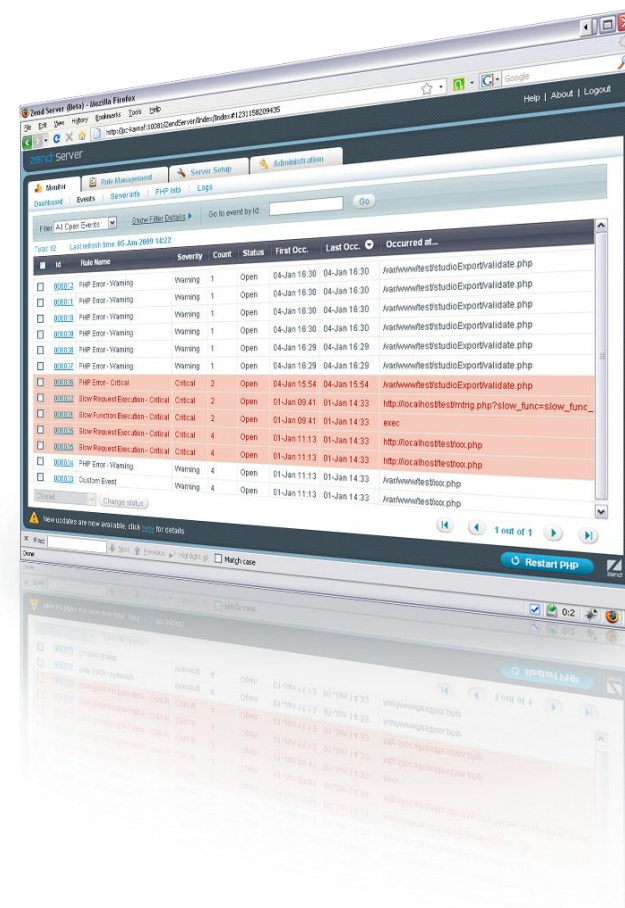
Zend Server 5 for IBM i

www.zend.com

Technical overview

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**

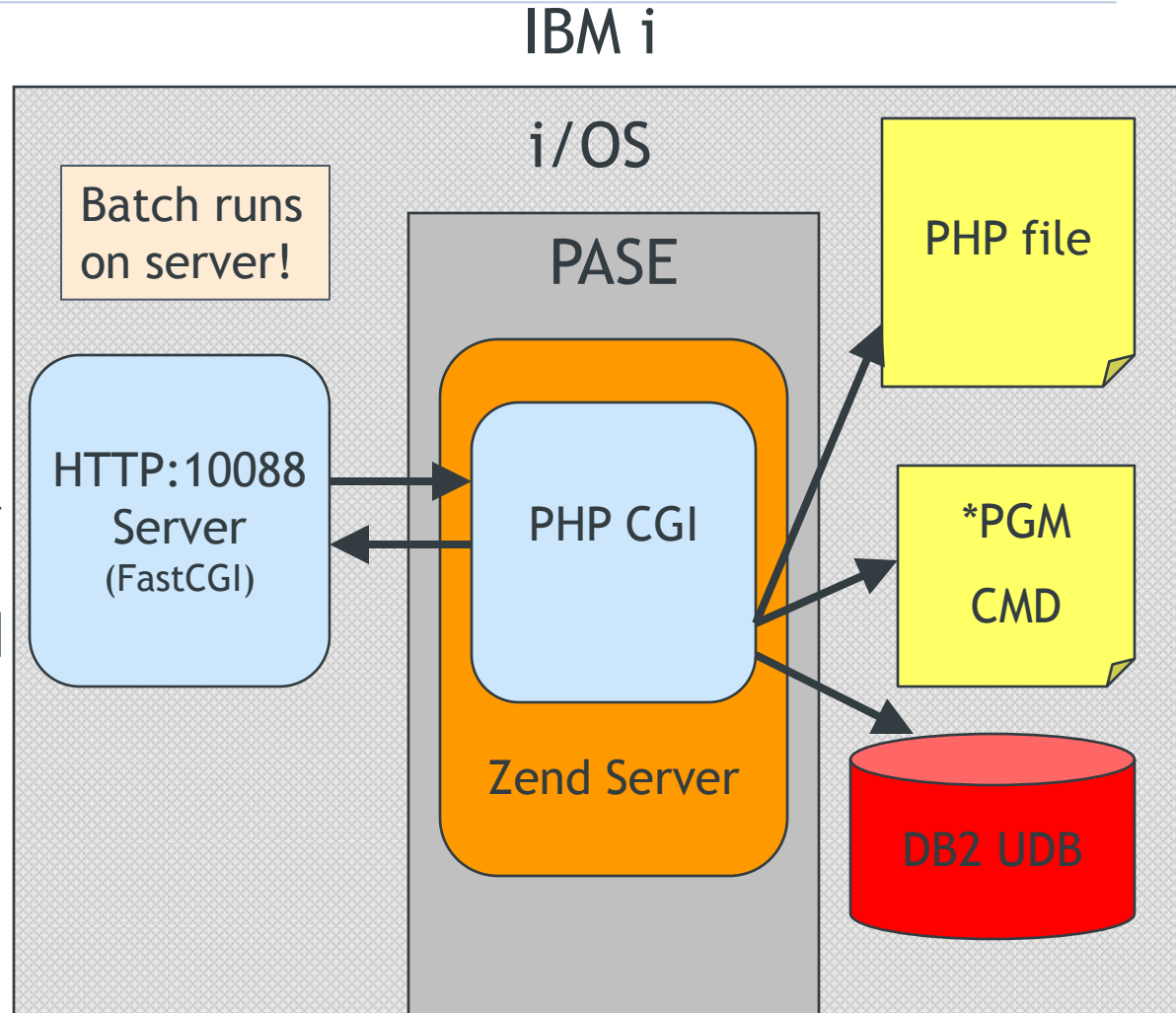
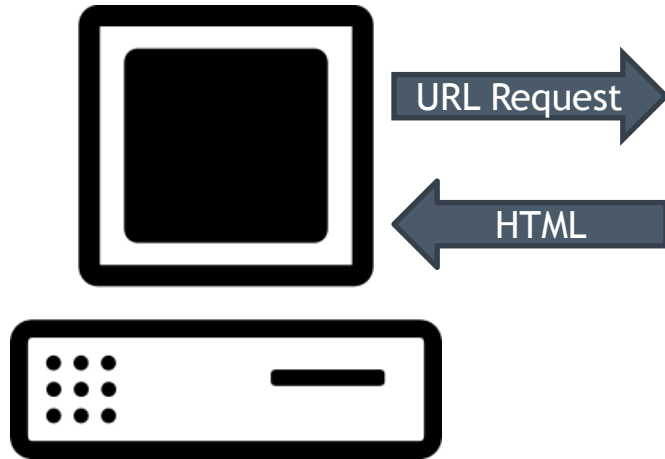
Zend Server vs. CE comparison

Features	Zend Server	Zend Server Community Edition
Native installation	✓	✓
Certified PHP	✓	✓
Zend Framework	✓	✓
Apache or IIS integration	✓	✓
MySQL (on Windows and Mac)	✓	✓
Built-in DB connectivity (Oracle, DB2, MySQL,...)	✓	✓
Java connector	✓	✓
Web-based administrator console	✓	✓
Debugger interface	✓	✓
Bytecode acceleration	✓	✓
Caching API	✓	✓
Page caching	✓	
Application monitoring (alerting)	✓	
Application problem diagnostics	✓	
Zend Download Server (Linux only)	✓	
Software updates and hot fixes	✓	
Technical support	✓	

Zend Server Under the Covers

ILE Apache:10088

- Default configuration FastCGI



IBM Relationship FAQ's

- **What do we get?**
 - ▶ Zend Server CE
 - ▶ Zend Studio
 - ▶ 1 year Silver Level Support (email/web 72 hour response)

- **What happens after 1 year?**
 - ▶ Product will still run
 - ▶ No updates to product or PHP
 - ▶ Forum support only

IBM Relationship FAQ's

- **What do I do when support runs out?**
 - ▶ Roll the dice
 - ▶ Zend account representative can discuss support options

- **I downloaded Core and it's been a year. Can I download Server and get support for another year?**
 - ▶ No, the support entitlement is by server, not Zend Product
 - ▶ The product will still run, however.
 - ▶ Forum support is always available

Batch Based PHP

www.zend.com

Advantages of Batch Processing

Slow is bad

- Have you ever been to a slow website?
- Wondered why the pages took so long to change?
- So bad it mad you want to get out and help push?

- Reasons:
 - ▶ Slow data access
 - ▶ Too much data
 - ▶ Superfluous activities

Zend Server to the rescue

- Many performance enhancing features w/o the steroids!!!
 - ▶ Caching, Optimizer, single Apache, etc.
- **Today's focus: Asynchronous processing (a.k.a.: Batch jobs)**
 - ▶ Explore batch work in both CE and full Zend Server
 - ▶ Scheduled jobs
 - ▶ Ad-hoc jobs

Advantages of batch processing

- Nothing new to IBM i folks!
- We've been preserving the 5250 experience for years
- RPG batch techniques:
 - ▶ CL programs
 - ▶ Batch RPG
 - ▶ Data Queue's
 - ▶ Message Queue's
 - ▶ SBMJOB

Advantages of batch processing...cont.

- With PHP and Zend Server there are similar opportunities
- Interactive experience is in the browser
- PHP code CAN be run asynchronously
- Take mundane tasks out of the users way

PHP Batch Processing

www.zend.com

Community Edition

Zend Server CE

- No access to jobq or recurring jobs
- But you can still run PHP jobs in batch!
- CL Programs can lead the way...
- From there you can call the PHP-CLI job
 - ▶ PHP Command Line Interface
 - ▶ Use Get method to pass parameters, etc.

First you need the CLI-Program

```
Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Object Links
Directory . . . . . : /usr/local/zendsvr/bin
Type options, press Enter.
  2=Edit  3=Copy  4=Remove  5=Display  7=Rename  8=Display attributes
 11=Change current directory ...

Opt  Object link      Type      Attribute  Text
---  -
_    php              STMF
_    php-cgi          STMF
_    php-cgi.bin      STMF
_    php-cli          F
_    php-config       STMF
_    phpize           STMF
_    setup_jb.sh     STMF
_    shell_functions.rc STMF
_    watchdog        STMF
More...

Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F9=Retrieve  F12=Cancel  F17=Position to
F22=Display entire field  F23=More options

MA a MW 10/002
I902 - Session successfully started
```



Create a CL Program to process it..

```
call PGM(QP2SHELL)      +  
      PARM( '/usr/local/zendsvr/bin/php-cli' +  
            '/www/zendsvr/htdocs/Samples/MikeBatch.php' )
```

Process...

- Before the update...

```
.....+.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+..
CUSTOMER_NUMBER  CUSTOMER_USAGE  CUSTOMER_EXTEND
          1          5,000  0.0000000000000000E+000
          2          7,000  0.0000000000000000E+000
          3         10,000  0.0000000000000000E+000
***** End of data *****
```

- Call the CL

```
Parameters or command
==> call batchphp
```

- After the update

```
Position to line .....+.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+..
CUSTOMER_NUMBER  CUSTOMER_USAGE  CUSTOMER_EXTEND
          1          5,000  3.5000000000000000E+002
          2          7,000  4.2000000000000000E+002
          3         10,000  5.0000000000000000E+002
***** End of data *****
```

More information...

- Alan Seiden...
 - ▶ Does a fantastic presentation on batch based PHP
 - ▶ More focus on Community Edition and options
 - <http://www.alanseiden.com/>
 - ▶ Great member of the community
 - ▶ Speaker at ZendCon 2009

PHP Batch Processing

www.zend.com

Zend Server Job Scheduler

Job Queue allows you to...

- **Put certain tasks into a separate execution queue**
 - ▶ Off-load to new process (or defer to a later time)
- **Execute certain tasks at a specified time**
 - ▶ Distribute processing load to off-hours
- **Execute certain tasks periodically**
- **While..**
 - ▶ Maximizing reuse of existing infrastructure & code
 - ▶ Making sure nothing falls between the cracks
 - ▶ Doing it all from a PHP API

Why would I schedule a job?

- Many Open Source applications have updates
- Updates are smartly distributed via a PHP script call
- Let's look at Drupal
- Cron job - batch based update, needs no interactive

Drupal cron Documentation

The screenshot shows a web browser window displaying the Drupal documentation page for "Running cron manually". The browser's address bar shows the URL `http://drupal.org/node/158922`. The page features the Drupal logo and a navigation menu with links for Documentation, Download, Support, Forum, Contribute, and Contact. A search bar is located in the top right corner. The main content area is titled "Running cron manually" and includes a "User login" form and a "Contributor links" section.

Installation guide

- System requirements
- Basic installation
 - Quick install for experts
 - Before you begin
 - Step 1: Download and unzip Drupal
 - Step 2: Create the database
 - Step 3: settings.php
 - Step 4: Run the installation script
 - After Installation
 - Set up cron
 - Configuring cron jobs
 - Configuring cron jobs on DreamHost
 - Configuring cron jobs on Media Temple Grid Servers (gs)
 - Configuring cron jobs on Windows

Running cron manually

Last modified: January 4, 2010 - 21:14 Drupal 5.x · Drupal 6.x

To run a cron job manually, simply visit your own site's url at <http://example.com/cron.php> with your web browser, and cron will run and complete in the background. You'll be left with a blank white page, which is normal. Use the browser's back button to return to the previous page and continue where you left off. It doesn't matter how long you wait while you're seeing the blank white page, and returning to the previous page at any time won't stop the cron process from running.

Core provides a convenience link for running cron manually too. Look at the Status page's link: `/admin/logs/status/run-cron?destination=admin%2Flogs%2Fstatus`. You can paste that into your own site's address bar any time you want to run a cron.

In addition, if you have the [update_status.module](#) installed, and you should since it's simply glorious, you'll be presented with a custom link on your `admin/logs/updates` page.

For Drupal 6.x, see in your site: *Administer > Reports > Status report > Cron maintenance tasks > Run cron manually.*

[< Simple wget multisite cron, up](#) [Setting up cron on Hostmonster through the cPanel interface >](#)

» [Login](#) or [register](#) to post comments

User login

Username: *

Password: *

- [Create new account](#)
- [Request new password](#)

Contributor links

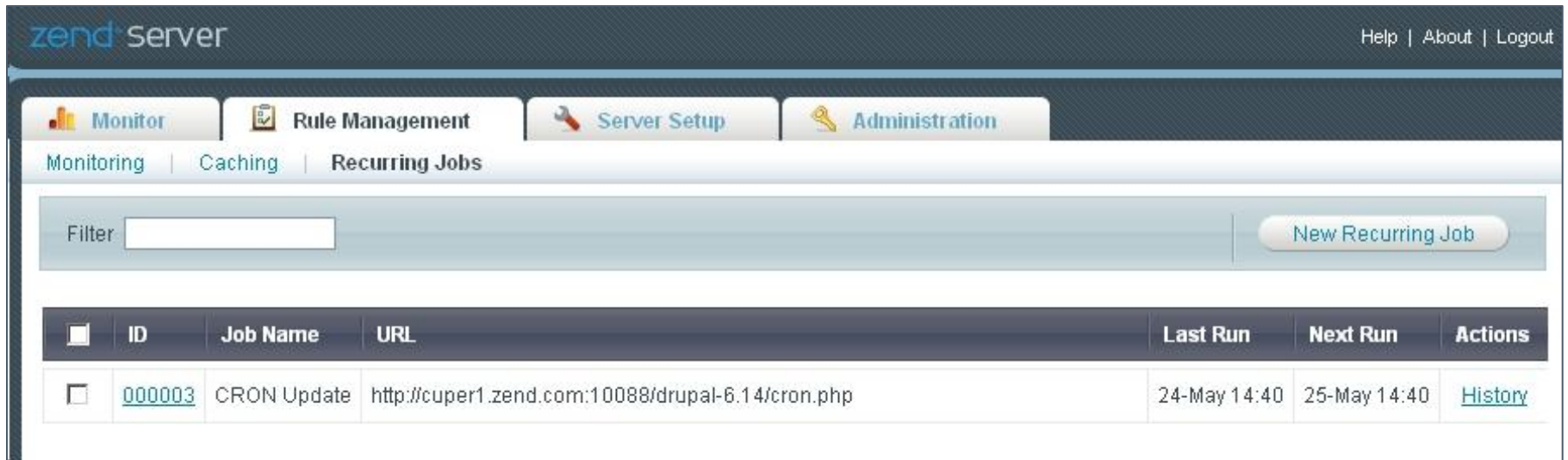
- [Community initiatives](#)
- [Queues](#)
 - [My issues](#)
 - [471 Pending bugs \(D7\)](#)
 - [79 Critical issues \(D7\)](#)
 - [1519 Patch queue \(D7\)](#)
 - [404 Patches to review \(D7\)](#)
 - [Performance issues \(D7\)](#)
 - [Usability issues \(D7\)](#)
 - [Fields in Core issues](#)

Zend Server Job Scheduler

- Can take a URL
- Also application and description
- Set a particular date/time
 - ▶ For a single event
 - ▶ For a recurring event
- Very powerful schedule process.

Zend Server Job Scheduler

- To see job scheduler
 - ▶ Navigate to Zend Server admin interface
 - ▶ Rule Management → Recurring jobs



The screenshot displays the Zend Server administration interface. At the top left, the 'zend server' logo is visible. On the right side of the top bar, there are links for 'Help | About | Logout'. Below the top bar, there are four main navigation tabs: 'Monitor', 'Rule Management', 'Server Setup', and 'Administration'. Under the 'Rule Management' tab, there are sub-tabs for 'Monitoring', 'Caching', and 'Recurring Jobs'. A 'Filter' input field is present on the left, and a 'New Recurring Job' button is on the right. Below these elements is a table listing recurring jobs.

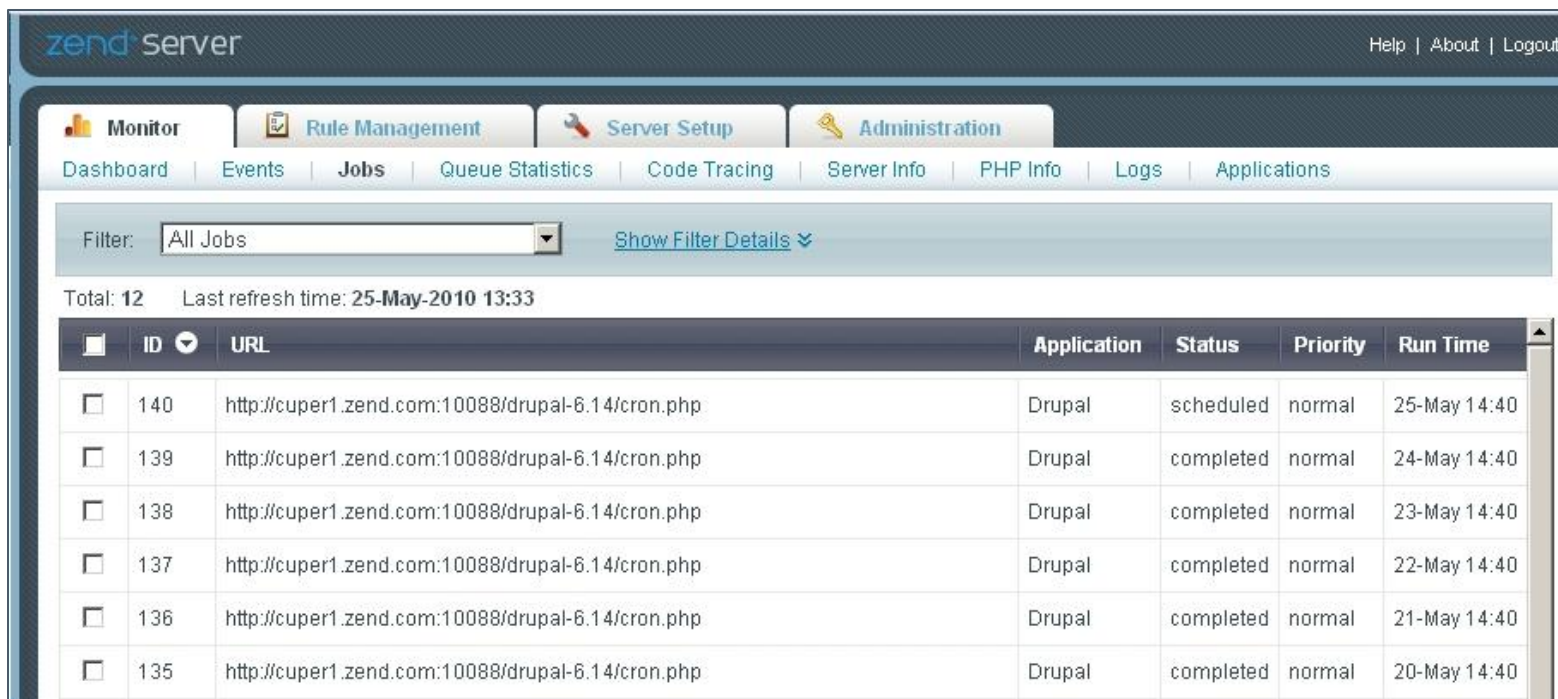
<input type="checkbox"/>	ID	Job Name	URL	Last Run	Next Run	Actions
<input type="checkbox"/>	000003	CRON Update	http://cuper1.zend.com:10088/drupal-6.14/cron.php	24-May 14:40	25-May 14:40	History

Job Details

The screenshot displays the Zend Server web interface. At the top, the 'zend server' logo is visible. Below it, a navigation bar contains four main sections: 'Monitor' (with a bar chart icon), 'Rule Management' (with a document icon), 'Server Setup' (with a wrench icon), and 'Administration' (with a key icon). Under 'Rule Management', there are sub-tabs for 'Monitoring', 'Caching', and 'Recurring Jobs'. The 'Recurring Jobs' tab is active, showing a form to 'Edit Scheduling Rule'. At the top of this form are two buttons: 'Update Rule' and 'Cancel'. The form fields are: 'URL' with the value 'http://cuper1.zend.com:10088/drupal-6.1.4/cron.php', 'Name' with 'CRON Update', and 'Application' with 'Drupal'. Below these is a 'Schedule' section with a dropdown menu for 'Execute Job' set to 'Daily...' and a time field set to 'at 14:40'.

How did the job do?

- Monitoring system will track batch jobs
- Can review output
- Navigate to Monitor → Jobs



The screenshot displays the Zend Server Monitor interface. At the top, there's a navigation bar with 'zend Server' and links for 'Help | About | Logout'. Below this is a secondary navigation bar with tabs for 'Monitor', 'Rule Management', 'Server Setup', and 'Administration'. A third navigation bar contains links for 'Dashboard', 'Events', 'Jobs', 'Queue Statistics', 'Code Tracing', 'Server Info', 'PHP Info', 'Logs', and 'Applications'. The main content area shows a filter dropdown set to 'All Jobs' and a 'Show Filter Details' link. Below the filter, it indicates 'Total: 12' and 'Last refresh time: 25-May-2010 13:33'. A table lists the jobs with columns for ID, URL, Application, Status, Priority, and Run Time.

ID	URL	Application	Status	Priority	Run Time
140	http://cuper1.zend.com:10088/drupal-6.14/cron.php	Drupal	scheduled	normal	25-May 14:40
139	http://cuper1.zend.com:10088/drupal-6.14/cron.php	Drupal	completed	normal	24-May 14:40
138	http://cuper1.zend.com:10088/drupal-6.14/cron.php	Drupal	completed	normal	23-May 14:40
137	http://cuper1.zend.com:10088/drupal-6.14/cron.php	Drupal	completed	normal	22-May 14:40
136	http://cuper1.zend.com:10088/drupal-6.14/cron.php	Drupal	completed	normal	21-May 14:40
135	http://cuper1.zend.com:10088/drupal-6.14/cron.php	Drupal	completed	normal	20-May 14:40

Spawning Jobs

- Jobs are created using the *createHttpJob()* method

```
$queue = new ZendJobQueue ();  
$queue->createHttpJob (  
    'http://backend.local/jobs/somejob.php' );
```

- Pass parameters
 - ▶ Simple ones, via query string *\$_GET*
 - ▶ Complex ones, as an associative array of key => value pairs
- Set job options
 - ▶ Name, priority, schedule, etc.
 - ▶ Create deferred or recurring jobs
- Or... create from the UI

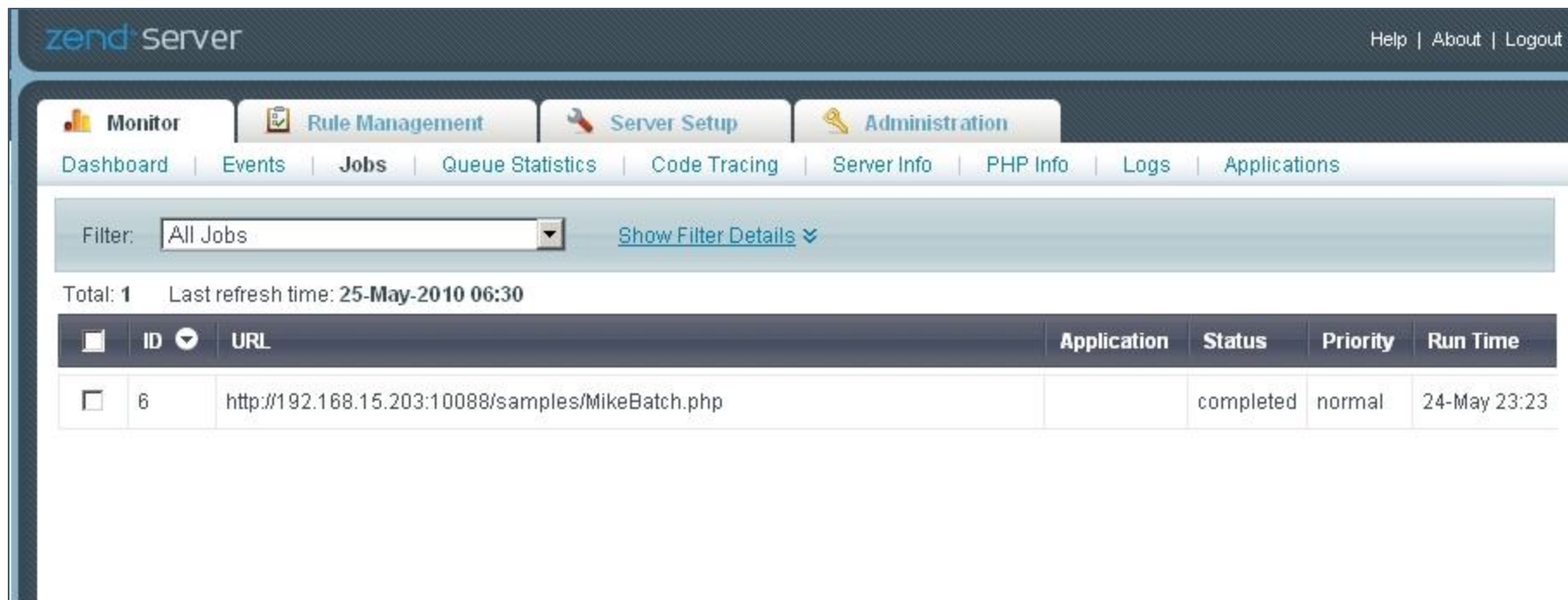
Let's go back to our original example...

- To perform a batch update.
- But can be used for so much more!
- Used SQL to clear out the totals.
- Then wrote a short PHP script that I invoke from the browser with the following code:

```
10 $queue = new ZendJobQueue();
11 $queue->createHttpJob('http://192.168.15.203:10088/samples/MikeBatch.php');
12
13 echo "job submitted";
```

Spawned job in Zend Server

- Navigate to Monitor → Jobs
- Select All jobs from the filter
- Click the jobs to see details

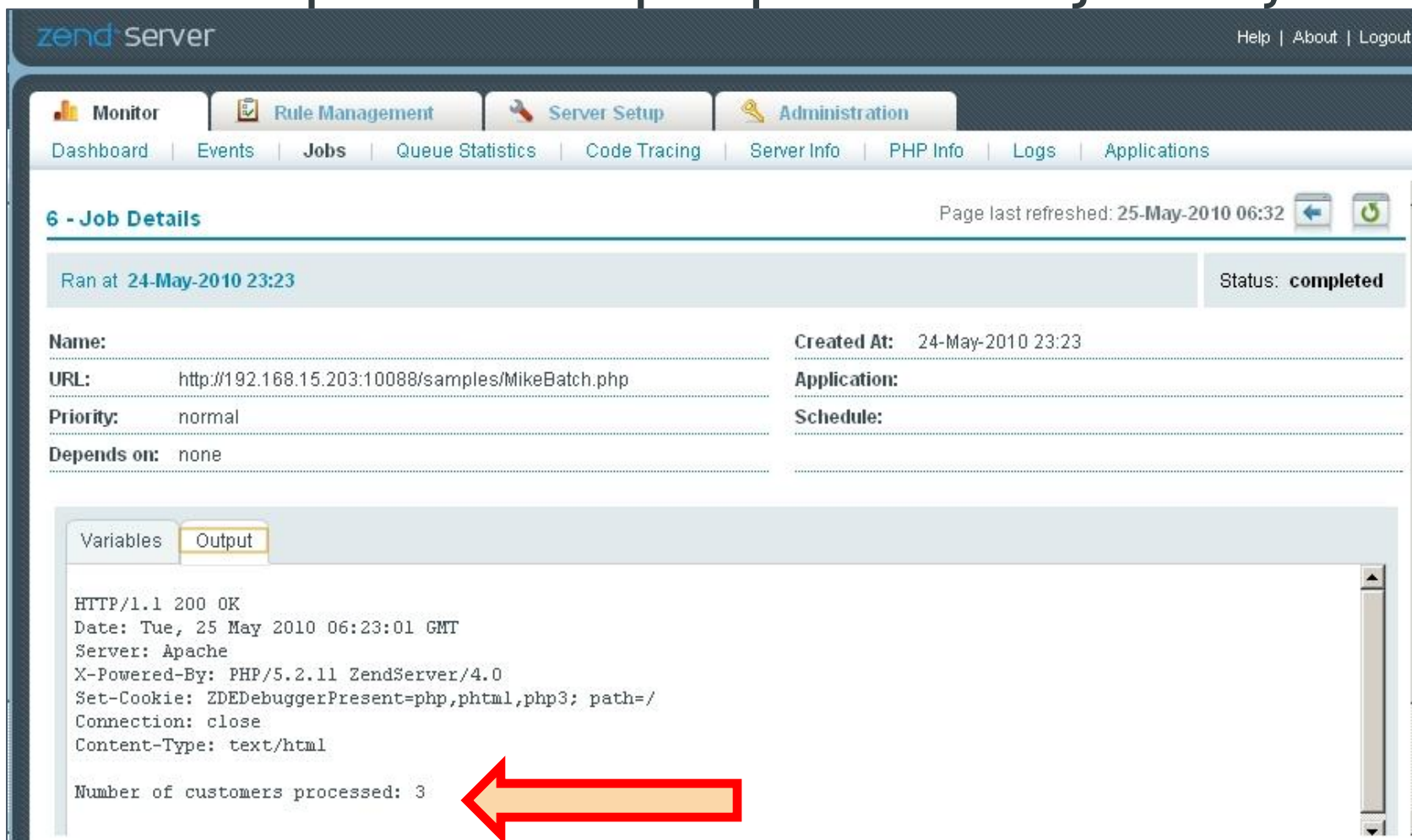


The screenshot displays the Zend Server Monitor interface. At the top, the 'zend server' logo is on the left, and 'Help | About | Logout' is on the right. Below the logo, there are four main navigation tabs: 'Monitor' (selected), 'Rule Management', 'Server Setup', and 'Administration'. Underneath these are sub-navigation links: 'Dashboard', 'Events', 'Jobs' (selected), 'Queue Statistics', 'Code Tracing', 'Server Info', 'PHP Info', 'Logs', and 'Applications'. A filter dropdown menu is set to 'All Jobs', with a 'Show Filter Details' link next to it. Below the filter, it shows 'Total: 1' and 'Last refresh time: 25-May-2010 06:30'. A table lists the job details:

<input type="checkbox"/>	ID	URL	Application	Status	Priority	Run Time
<input type="checkbox"/>	6	http://192.168.15.203:10088/samples/MikeBatch.php		completed	normal	24-May 23:23

Details of spawned jobs

- See the output of the script captured in the job entry



The screenshot displays the Zend Server web interface. At the top, the 'zend server' logo is on the left, and 'Help | About | Logout' is on the right. Below the logo is a navigation bar with tabs for 'Monitor', 'Rule Management', 'Server Setup', and 'Administration'. Underneath this is a secondary navigation bar with links for 'Dashboard', 'Events', 'Jobs', 'Queue Statistics', 'Code Tracing', 'Server Info', 'PHP Info', 'Logs', and 'Applications'. The main content area is titled '6 - Job Details' and shows a job that ran at '24-May-2010 23:23' with a status of 'completed'. The job details include: Name (empty), URL ('http://192.168.15.203:10088/samples/MikeBatch.php'), Priority ('normal'), and Depends on ('none'). To the right, it shows 'Created At: 24-May-2010 23:23', 'Application:' (empty), and 'Schedule:' (empty). Below the details are two tabs: 'Variables' and 'Output'. The 'Output' tab is active, showing the following text: 'HTTP/1.1 200 OK', 'Date: Tue, 25 May 2010 06:23:01 GMT', 'Server: Apache', 'X-Powered-By: PHP/5.2.11 ZendServer/4.0', 'Set-Cookie: ZDEDebuggerPresent=php,phtml,php3; path=/', 'Connection: close', and 'Content-Type: text/html'. At the bottom of the output, it says 'Number of customers processed: 3', which is highlighted by a red arrow pointing from the right.

Pro's & Con's

- **Community edition:**

- ▶ It's free
- ▶ Can run batch
- ▶ Pass simple parameters

- **Full Zend Server**

- ▶ Not free
- ▶ More powerful job submission options
- ▶ Job tracking
- ▶ Production ready solution for Enterprise!
- ▶ Jobq is just one of the many productivity features

Thank you & Q&A

- If you have a question you can

- ▶ Type in the Q&A panel
- ▶ Tweet #zendwebinar

- Follow us!

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

 ▶ <http://twitter.com/zend>

Q&A

www.zend.com

mike.p@zend.com