

most significant bits

News and Information For High-Tech Professionals

Published by Pocket Protector Press™

A division of Stout Systems



Spring 2008

In This Issue

Handling Adversity in the IT Business

Considering a job change? What do you need to know about a prospective employer to feel confident in the company's longrange prospects?

An Introduction to PHP

What can this cross platform, open source dynamic scripting language do for you?

Recent News

Stout Systems would like to welcome new employees Jose Mojica, Rodney Lambert, Jim Morgenstern, John Fohrman and Lee Downey.

Job Openings

Check out current permanent and contract openings on StoutSystems.com and on our new mobile site StoutSystems.MOBI.

Subscribe To Our Newsletter

If you would like to receive this newsletter directly at your desk or in-box, send a note to info@stoutsystems.com providing your subscription info (e-mail or ground mail address).

Call us:

(734) 663-0877

An Important Consideration in Choosing an IT Employer Handling Adversity in the IT Business

by John W. Stout

I was recently invited to speak on the topic of IT employment at a local technology group. During the course of the evening, we discussed the future of IT employment. A member of the audience—who had been seeking a new job for some time—asked me which characteristics were most important to find out about a potential employer.

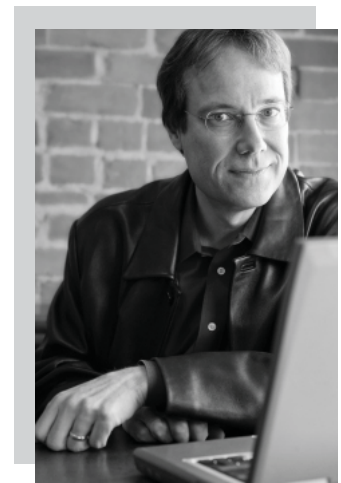
Of course, classic advice would be to review, if possible, the company's financial position, its management philosophy, the state of its internal technology usage and future technology plans, its growth projections—all sound advice. But there is one other area that is mostly overlooked but very worthy of investigation, and one that will tell a prospective employee a lot about the viability and philosophy of a company. And that is how does the company handle adversity?

We all know the story of the *Titanic* disaster. This was a bad situation made far worse by lack of preparedness and ineffective response. The captain had never been in a wreck during his entire career. The crew were unprepared to manage the emergency. There were insufficient lifeboats due to lax industry regulations of the time. The owners fostered an arrogance that nothing disastrous would ever happen anyway.

As I have seen it, there are three general categories of adversity:

A) Adversity that is internally created. This category would include bad business decisions, failure to predict the course of the industry that the company is in, bad investment, disastrous hiring decisions.

B) Adversity that comes from external factors in the business environment. This includes



such things as a rough economy, domestic and overseas competition, new government regulations, and, in IT, a new sweeping technology that makes a broadly used older technology rapidly obsolete.

C) A combination of the above factors.

Category A is the most challenging in that it requires the courage to admit a mistake, the wisdom not to repeat it, the continued forward energy devoted to repairing any damage and then solidifying a base from which to grow again. If you are a prospective employee investigating a company, this is definitely an area to look at. Find out about past challenges or failures by the company and its management. What did they do to recover from them? Specific to IT, find out about the company's security policies and how strenuously these are enforced. If a public company, check out the company's cash position. And if the company never has had any serious setbacks, find out what prevention methods they have in place. Remember the *Titanic* captain example: never having been in a wreck might not always be a good thing.

continued on page 4

An Introduction to PHP

By Elizabeth Naramore

PHP is a cross platform, open source dynamic scripting language primarily (but not exclusively) used for creating Web applications. It works seamlessly with HTML for controlling dynamically created Web sites. In other words, you can store your information in a database, and PHP will use that information to create your Web site on the fly. PHP also has a multitude of built-in functions for uses other than database interaction, making it a popular choice in Web applications. It has been around since 1995 and continues to mature with each release.

PHP is Free

PHP is open source, it is free to use, and integrates well with other open source technologies such as Linux, Apache and MySQL. In fact, this open source combination is so popular, it is sometimes referred to as the “LAMP” stack. (The “P” can also refer to Python or Perl, but most times it refers to PHP.)

It is licensed under its own “PHP License” which basically means you are free to create whatever application you like, as long as you refer back to the fact that PHP is freely available at <http://www.php.net> and include a copy of the PHP license copyright statement. You are also prohibited from using “PHP” in the name of your application without obtaining permission from the PHP Group. Other than that, you can do whatever you like with your PHP application. (For a full copy of the PHP

License, you may view it in its entirety at http://us3.php.net/license/3_01.txt.)

PHP is Easy

One of PHP’s strengths comes from how easy it is to use and learn. For an example of how easy it is to write a PHP program, you can open an instance of Notepad and type:

```
<?php
    echo "Hello world!";
?>
```

If you save the file to your Web server, then refer to it in your browser, it will output:

```
Hello world!
```

And there’s your first PHP program.

PHP can also be embedded directly into an HTML file (see Code Sample 1 below).

You could then control the title and the content of your HTML page by altering the value stored in the \$title variable.



PHP is Powerful

PHP contains many built-in functions for uses other than database interaction, such as manipulating strings, arrays, objects, images and dates, file management, mathematical equations, sending emails, interacting with third party APIs, regular expressions, and much more. Besides the multitude

of default functions, PHP’s functionality is extended even further with libraries such as PEAR (PHP Extension and Application Repository, available at <http://pear.php.net>) and PECL (PHP Extension Community Library, available at <http://pecl.php.net>). PEAR contains over 450 ready-made packages of code, and PECL contains over 180. If there is a function you need your Web site to perform, chances are someone has already written a piece of that for you.

There are numerous frameworks available for PHP, such

Code Sample 1

```
<html>
<head>
  <meta>
    <title><?php echo $title ?></title>
  </meta>
</head>
```

```
<body>
```

The title of this page is

```
<?php echo $title ?>
</body>
```

as CakePHP and Symfony (MVC frameworks), Zend Framework, Solar, CodeIgniter, and many others. PHP programmers definitely appreciate flexibility and the ability to use the tools best suited for their specific needs, which is why there isn't any "one" PHP framework.

There are a multitude of database abstraction layers which facilitate integration with just about every database available. Packages such as PDO (PHP Data Objects) make it easy for users to switch from one database system to another with minimal coding required.

PHP is Popular

It's estimated that PHP is currently in use by 33% of all Web sites on the internet (source: <http://www.nexen.net>.) The TIOBE index for January 2008 (<http://www.tiobe.com>) also lists PHP as the fourth most popular programming language, behind Java, C and Visual Basic.

There are many open source PHP applications that can be used free of charge for common Web functionalities. Examples include blogging software (such as Wordpress), Content Management Systems (such as Joomla! or Drupal), shopping carts (such as osCommerce, Magento, and Zen Cart), and forums (such as phpBB). If you're not familiar with the Web site OpenSourceCMS (<http://www.opensourcecms.com>), it is a site that offers demo installations of many of these types of programs for you to play around with before you install them on your own system.

The PHP community is also one of the strengths of PHP. Generally speaking, the PHP community is helpful, friendly and very active. The PHP documentation is outstanding and being improved on a daily basis. There are hundreds of Web sites providing tutorials, forums, articles, and other helpful resources for PHP users of all levels. Some of those Web sites include:

<http://www.php.net> - the official documentation of the PHP project. Also includes information on PHP conferences, events and archives of individual mailing lists.

<http://www.phpbuilder.com> - a source of PHP articles, as well as an active and helpful forum.

<http://www.phpdeveloper.org> - a wonderful news source for the goings-on in the PHP world.

<http://www.phparch.com> - home of php|architect magazine and also the home of the PHP:Pro Podcast. Also a great article repository.

<http://www.planet-php.org> - an aggregate of blogs from those in the PHP community.

<http://devzone.zend.com> - the Developer's Zone at Zend, which provides news, articles, and other items of general PHP interest.

There are also numerous PHP-centric conferences and hundreds of PHP user groups across the globe. For a complete list, you can go to the PHP home page (<http://www.php.net>). These are great places to go if you want to learn more about PHP from the developers who use it on a daily basis.

A Final Word

PHP has been criticized by some as being an "insecure language" and not "enterprise ready." While most of this is FUD (fear, uncertainty and doubt) spread by a handful of uninformed individuals, it can also be said that because PHP's barrier to entry is fairly low, the chances

“The PHP community is also one of the strengths of PHP. Generally speaking, the PHP community is helpful, friendly and very active.”

of someone writing a simple PHP program and calling themselves a "PHP Programmer" is higher than in a language such as Java. As well, sometimes the flexibility offered inherently in PHP can be exploited by less experienced programmers. An insecure program can be written in any language, but you should make sure that the PHP programmer working for you is truly qualified and

has employed the best programming practices.

The PHP community is working hard to educate PHP coders on security pitfalls and the importance of writing secure code. Zend has also created the Zend Certified Engineer (ZCE) program which certifies that a PHP programmer has an excellent working knowledge of basic PHP functionality and best programming practices.

With regard to scalability and its use in enterprise, PHP has proven itself to be scalable, stable, and definitely "enterprise ready" as can be seen in cases such as Facebook, Wikipedia and Flickr, all of which are PHP-powered.

Elizabeth Naramore works in e-commerce and in PHP, and has since 2001. She is an author, speaker, blogger, editor, freelance consultant and is very active in the PHP Community. She blogs at <http://www.naramore.net/blog> and can be reached at elizabeth@naramore.net.

continued from page 1

Category B has its own challenges. Here you would want to find out whether management felt it could take on and overcome the environmental factors and if so, how they intended to do that. In a difficult economy, I know companies that take the low profile approach, in that they shut down their marketing, thinking “it won’t work because of the economy” and try to “ride out the storm.” If they do succeed in riding it out, one often finds two things when business picks up again: 1) the competitors who did promote effectively during the rough times now own a bigger market share and 2) the company itself is smaller and has fewer resources to deliver its products to a potentially increased demand. That doesn’t indicate a bright future for the company.

Of course a Category A decision can follow as a reaction to a Category B circumstance resulting in a Category C situation. If you have found a company whose management team came back from a Category C disaster, you probably have a winner of an employer.

John W. Stout is the founder and president of Stout Systems Development. He has nearly thirty years’ experience in the software industry. He is also sought after as a technology speaker, presenting sessions at developer conferences and user groups. E-mail john@stoutsystems.com.

Stout Systems: TechTown’s Newest Tenant
Stout Systems has recently acquired tenant space in TechTown, which is a state-of-the-art business center in Detroit. This gives our company presence in the growing downtown Detroit business community and additional facilities from which to continue to expand. You can read more about TechTown and see our article in the latest newsletter at:
http://www.techtownwsu.org/may_newsletter/stout.html.



ADDRESS SERVICE REQUESTED

Presorted Standard
US Postage Paid
Permit #187
Ann Arbor MI

In This Issue:

- Handling Adversity in the IT Business
- An Introduction to PHP