PHP and Open Source

May 7, 2003. San Francisco

Rasmus Lerdorf <rasmus@php.net>

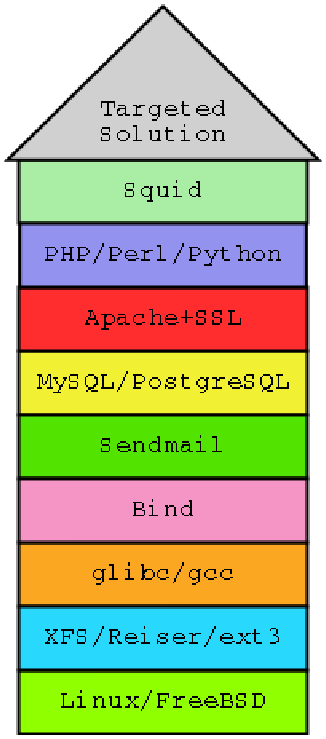
http://lerdorf.com/sfsu.pdf

Where is the money?

Verticals!

The money is where it has always been, in the verticals!

- Customers care about the solution, not implementation details
- Vendor has more control and flexibility with Open Source horizontal components
- Vendor avoids paying per-seat licensing fees which directly improves the bottom-line



Targeted Solution

Squid

PHP/Perl/Python

Apache+SSL

MySQL/PostgreSQL

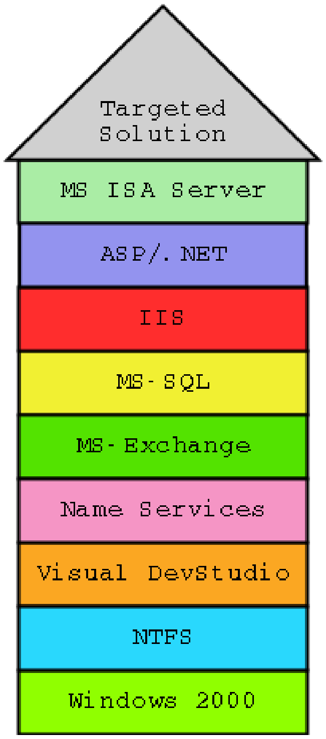
Sendmail

Bind

glibc/gcc

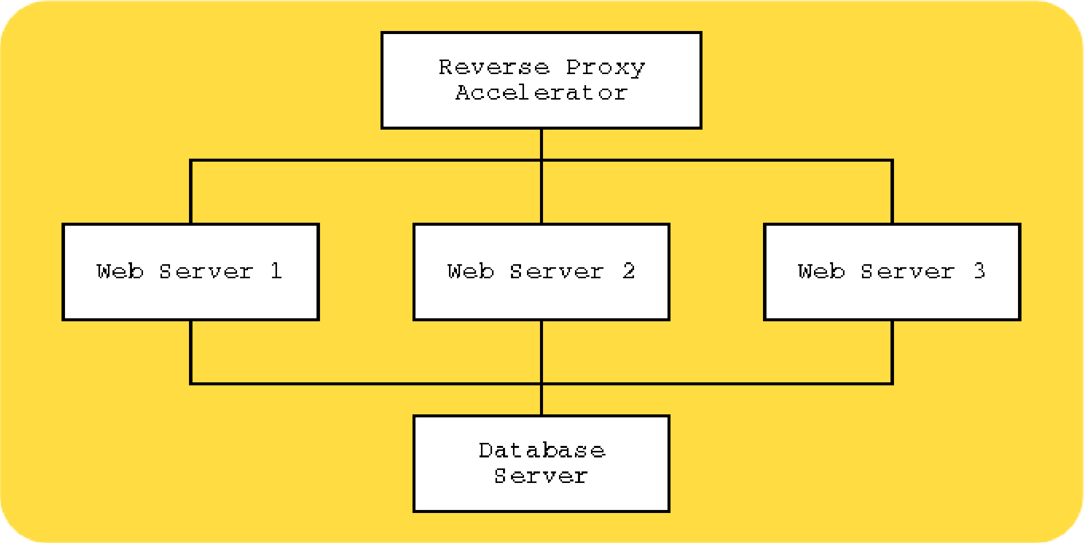
XFS/Reiser/ext3

Linux/FreeBSD


Targeted Solution
MS ISA Server
ASP/.NET
IIS
MS-SQL
MS-Exchange
Name Services
Visual DevStudio
NTFS
Windows 2000

Is it Worth It?

Architecture



Reverse Proxy Accelerator

Web Server 1

Web Server 2

Web Server 3

Database Server

Hardware

5 Dual Processor machines with a good amount of disk and RAM. $3,000 per machine. A good quality Switch. $1,000. Total: $16,000.

Software

Microsoft® Windows® Server Enterprise 2003 25 Client x 5	. . $3,919
Internet Security and Acceleration Server (per cpu) x 2	. . $5,999
SQL Server 2000 Enterprise Edition (per cpu) x 2	. $19,999
Exchange Enterprise Server (25 CALs)	. . $6,999
Commerce Server (per cpu) x 6	. $6,999
Application Server (per cpu) x 6	. . $2,999
Content Management Server 2002 (per cpu) x 2	. $39,101
MSDN Universal Subscription per developer x 2	. . $2,799
Total:	$266,174

The Alternative

Hardware

5 Dual Processor machines with a good amount of disk and RAM. $3,000 per machine. A good quality Switch. $1,000. Total: $16,000.

Software

Software costs for this architecture is $0.

What don't you get?

- (-) Shrinkwrapped packaged solution - more work to configure
- (-) Someone to call for technical support
- (+) Ongoing yearly license renewal fees
- (+) No vendor lock-in
- (+) Nimda Virus
- (+) Complete flexibility for both hardware and software

The Web Problem

- The Web Problem is simple!
- The Web Solution is often overly complex.

The Good Old Days

Handling simple data coming from a form took something like this to do in C:

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <string.h>

#define ishex(x) (((x) >= '0' && (x) <= '9') || ((x) >= 'a' &&
                   (x) <= 'f') || ((x) >= 'A' && (x) <= 'F'))

int htoi(char *s) {
    int     value;
    char    c;

    c = s[0];
    if(isupper(c)) c = tolower(c);
    value=(c >= '0' && c <= '9' ? c - '0' : c - 'a' + 10) * 16;

    c = s[1];
    if(isupper(c)) c = tolower(c);
    value += c >= '0' && c <= '9' ? c - '0' : c - 'a' + 10;

    return(value);
}

void main(int argc, char *argv[]) {
    char params, data, dest, s, *tmp;
    char name, age;

    puts("Content-type: text/html\r\n");
    puts("<html><head><title>Form Example</title></head>");
    puts("<body><h1>My Example Form</h1>");
    puts("<form action=\"form.cgi\" method=\"GET\">");
    puts("Name: <input type=\"text\" name=\"name\">");
    puts("Age: <input type=\"text\" name=\"age\">");
    puts("<br><input type=\"submit\">");
    puts("</form>");

    data = getenv("QUERY_STRING");
    if(data && *data) {
        params = data; dest = data;
        while(*data) {
            if(data=='+') dest=' ';
            else if(data == '%' && ishex((data+1))&&ishex(*(data+2))) {
                *dest = (char) htoi(data + 1);
                data+=2;
            } else dest = data;
            data++;
            dest++;
        }
        *dest = '\0';
        s = strtok(params,"&");
        do {
            tmp = strchr(s,'=');
            if(tmp) {
                *tmp = '\0';
                if(!strcmp(s,"name")) name = tmp+1;
                else if(!strcmp(s,"age")) age = tmp+1;
            }
        } while(s=strtok(NULL,"&"));

        printf("Hi s, you are s years old\n",name,age);
    }
    puts("</body></html>");
}
```

The Perl alternative

Perl became an obvious choice because it was made for text processing. The same thing in Perl using CGI.pm:

```
use CGI qw(:standard);
print header;
print start_html('Form Example'),
    h1('My Example Form'),
    start_form,
    "Name: ", textfield('name'),
    p,
    "Age: ", textfield('age'),
    p,
    submit,
    end_form;
if(param()) {
    print "Hi ",em(param('name')),
        "You are ",em(param('age')),
        " years old";
}
print end_html;
```

Much easier both to read and to write, at least to people with a bit of a programming background.

The PHP Approach

PHP has an HTML-centric approach. The same script in PHP became:

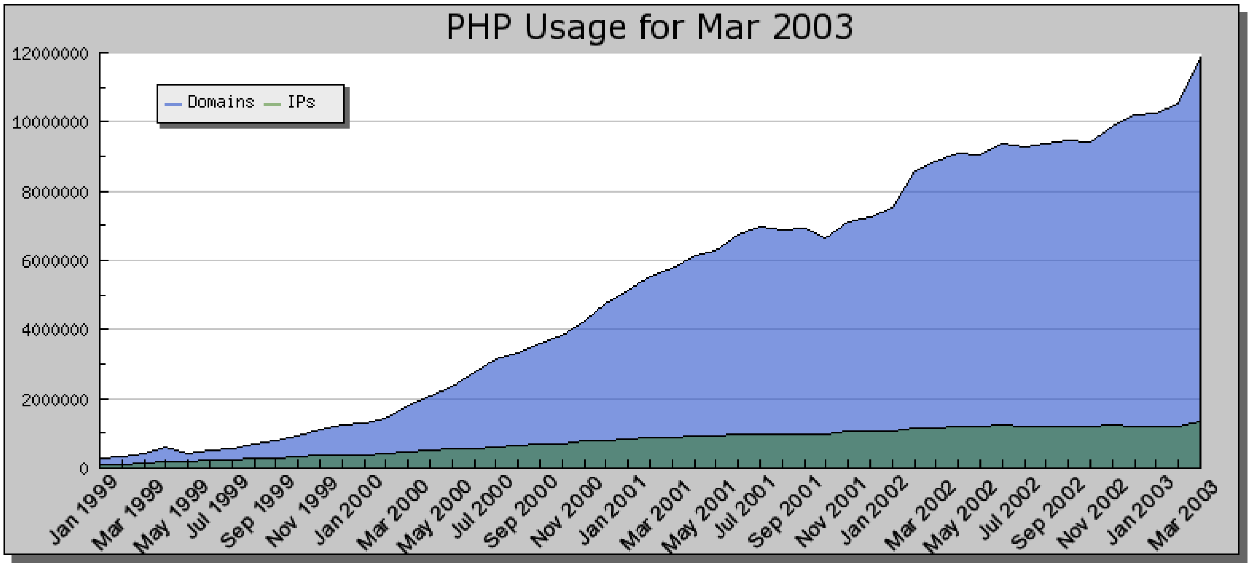
```
<html><head><title>Form Example</title></head>
<body><h1>My Example Form</h1>
<form action="form.phtml" method="POST">
Name: <input type="text" name="name">
Age: <input type="text" name="age">
<br><input type="submit">
</form>
<?if($name):?>
Hi <?echo $name?>, you are <?echo $age?> years old
<?endif?>
</body></html>
```

A block of raw HTML followed by the minimum amount of logic possible.

PHP Usage Growth

March 2003 Netcraft Report

- 40,100,739 Domains queried
- 11,860,645 Domains. 1,316,288 IP addresses
- PHP installed on 29.58% of all domains



Source: Netcraft

March 2003 Apache Module Report

- 6,134,492 Apache Servers surveyed
- 3,107,624 (50.66%) PHP
- 1,877,707 (30.61%) OpenSSL
- 1,813,509 (29.56%) mod_ssl
- 1,345,589 (21.93%) Frontpage
- 1,297,683 (21.15%) mod_perl
- 370,550 (6.04%) DAV
- 321,928 (5.25%) mod_throttle
- 175,924 (2.87%) AuthMySQL
- 172,517 (2.81%) mod_jk
- 170,581 (2.78%) mod_auth_pam

Source: SecuritySpace.com

Solution

A good solution should

- Have a shallow learning curve
- Instant gratification
- Build on what you know
- Great documentation
- Solve the simple problem easily
- Eliminate tedium
- Be able to solve even the most complex problem
- Be secure
- Use/borrow existing technology
- Work everywhere

Bonus

- Be Free
- Teach the basics by not hiding the problem

Form Handling

Traditional Form Handling

```
<form action="<?=$PHP_SELF?>" method="POST">
Your name: <input type=text name=name><br>
You age: <input type=text name=age><br>
<input type=submit>
</form>
```

Output:

```
Your name:
You age:
```

Receiving Script

```
Hi <?echo $name?>.
You are <?echo $age?> years old.
```

GD 1/2

Creating a PNG with a TrueType font

```
<?
Header("Content-type: image/png");
$im = ImageCreate(630,80);
$blue = ImageColorAllocate($im,0x5B,0x69,0xA6);
$white = ImageColorAllocate($im,255,255,255);
$black = ImageColorAllocate($im,0,0,0);
ImageTTFText($im, 45, 0, 10, 57,$black,"CANDY",$text);
ImageTTFText($im, 45, 0, 6, 54,$white,"CANDY",$text);
ImagePNG($im);
?>

<IMG src="txt.php?text=<?echo urlencode($text)?>">
```

ImageColorAt

Using ImageColorAt()

```
<? $image = "presentations/slides/intro/php-tiny.jpg"; ?>
<img src="<?=$image?>"><br />
<?
    $im = imagecreatefromjpeg($image);
    $dx = imagesx($im);
    $dy = imagesy($im);
    for($y = 0; $y < $dy; $y++) {
        for($x=0; $x < $dx; $x++) {
            $col = imagecolorat($im, $x, $y);
            $rgb = imagecolorsforindex($im,$col);
            printf("<font color=#02x02x%02x>#</font>",
              $rgb['red'],$rgb['green'],$rgb['blue']);
        }
        echo "<br />\n";
    }
    imagedestroy($im);
?>
```

Output:

```
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
################################################################
```

GD 1/2

CreateFrom and Bounding Box Math

```
<?
Header("Content-type: image/png");
$font = 'phpi';
if(!$si) $si = 66;
$im = ImageCreateFromPNG('php-blank.png');
$tsize = imagettfbbox($si,0,$font,$text);
$dx = abs($tsize[2]-$tsize[0]);
$dy = abs($tsize[5]-$tsize[3]);
$x = ( imagesx($im) - $dx ) / 2;
$y = ( imagesy($im) - $dy ) / 2 + 3*$dy/4;
$blue = ImageColorAllocate($im,0x5B,0x69,0xA6);
$white = ImageColorAllocate($im,255,255,255);
$black = ImageColorAllocate($im,0,0,0);
ImageAlphaBlending($im,true);
ImageTTFText($im, $si, 0, $x, $y, $white, $font, $text);
ImageTTFText($im, $si, 0, $x+2, $y, $white, $font, $text);
ImageTTFText($im, $si, 0, $x, $y+2, $white, $font, $text);
ImageTTFText($im, $si, 0, $x+2, $y+2, $white, $font, $text);
ImageTTFText($im, $si, 0, $x+1, $y+1, $black, $font, $text);
ImagePNG($im);
?>
```

```
<IMG src="txt2.php?text=<?echo urlencode($text)?>&si=<?echo $si?>">
```

```
Text:  Size:
```

TTF Text

TrueType Fonts

You can use any TrueType Font that includes a Unicode mapping table. Fonts such as Wingdings will not work.

```
<?
$im  = ImageCreate(600,7150);
$white = ImageColorAllocate($im,255,255,255);
$black = ImageColorAllocate($im,0,0,0);
$dir = opendir('/usr/share/fonts/truetype');
$y=30;
while($file = readdir($dir)) {
    if(substr($file,strrpos($file,'.'))=='.ttf') {
        ImageString($im,5,5,$y-20,substr($file,0,-4),$black);
        ImageTTFText($im,30,0,100,$y,$black, substr($file,0,-4),"ABCdéf123");
        $y+=40;
    }
}
Header('Content-Type: image/png');
ImagePNG($im);
?>
```

Output:

abalc	ABCdéf123
a_d_mono	ABCDÉF123
Adresack	ABCD□F123
aggstock	ABCd f123
ariblk	ABCdéf123
arnari	*ABCdéf123*
arnar	ABCdéf123
bkant	ABCdéf123
Bookosbi	***ABCdéf123***
bookosb	**ABCdéf123**
Bookosi	*ABCdéf123*
Bookos	ABCdéf123
calist	ABCdéf123
comicbd	**ABCdéf123**
comic	ABCdéf123
coprgtb	**ABCDÉF123**
coprgtl	ABCDÉF123
courbd	**ABCdéf123**
courbi	***ABCdéf123***
couri	*ABCdéf123*
cour	ABCdéf123
dc_sans	ABC'D□'F□□□
dc_serif	ABC'D□'F□□□
dilate	abcdéf123
dirtydoz	ABCD F123
Dotmatrx	ABCdéf123
dronecat	abcd□f□□□
earth	abcd
Eklekti0	ABCDÉF123
electroh	ABCD□F123
Eroded2020	ABCD□F123

Reading EXIF Headers from a JPEG

```
<?php
$data = exif_read_data('presentations/slides/intro/img_resize.jpg');
foreach($data as $key=>$val) {
    if(is_array($val)) {
        foreach($val as $k=>$v) {
            echo $key."[$k]: $v<br />\n";
        }
    } else
        echo "$key: ".@substr($val,0,40)."<br />\n";
}
?>
```

Output:

```
FileName: img_resize.jpg
FileDateTime: 1027351588
FileSize: 669158
FileType: 2
MimeType: image/jpeg
SectionsFound: ANY_TAG, IFD0, THUMBNAIL, EXIF
COMPUTED[html]: width="1536" height="1024"
COMPUTED[Height]: 1024
COMPUTED[Width]: 1536
COMPUTED[IsColor]: 1
COMPUTED[ByteOrderMotorola]: 0
COMPUTED[ApertureFNumber]: f/4.0
COMPUTED[FocusDistance]: 1.07m
COMPUTED[Thumbnail.FileType]: 8
COMPUTED[Thumbnail.MimeType]: image/tiff
COMPUTED[Thumbnail.Height]: 64
COMPUTED[Thumbnail.Width]: 96
Make: Eastman Kodak Company
Model: KODAK DC265 ZOOM DIGITAL CAMERA (V01.00)
Orientation: 1
XResolution: 150/1
YResolution: 150/1
ResolutionUnit: 2
YCbCrPositioning: 1
Exif_IFD_Pointer: 190
THUMBNAIL[ImageWidth]: 96
THUMBNAIL[ImageLength]: 64
THUMBNAIL[BitsPerSample]: Array
THUMBNAIL[Compression]: 1
THUMBNAIL[PhotometricInterpretation]: 2
THUMBNAIL[StripOffsets]: 1748
THUMBNAIL[Orientation]: 1
THUMBNAIL[SamplesPerPixel]: 3
THUMBNAIL[RowsPerStrip]: 64
THUMBNAIL[StripByteCounts]: 18432
THUMBNAIL[XResolution]: 72/1
THUMBNAIL[YResolution]: 72/1
THUMBNAIL[PlanarConfiguration]: 1
THUMBNAIL[ResolutionUnit]: 2
ExposureTime: 1/250
FNumber: 400/100
ExifVersion: 0200
DateTimeOriginal: 1999:01:31 04:17:59
ComponentsConfiguration: *
CompressedBitsPerPixel: 24/10
ShutterSpeedValue: 800/100
ApertureValue: 400/100
ExposureBiasValue: 0/100
MaxApertureValue: 300/100
```

```
SubjectDistance: 107/100
MeteringMode: 2
LightSource: 0
Flash: 1
FocalLength: 80000/10000
MakerNote: *       Eastman Kodak Company
FlashPixVersion: 0100
ColorSpace: 1
ExifImageWidth: 1536
ExifImageLength: 1024
```

Fetching an embedded thumbnail

```
<?
Header('Content-type: image/tiff');
echo exif_thumbnail('p0004557.jpg');
?>
```

A PDF Invoice

```
<?php
$pdf = pdf_new();
pdf_open_file($pdf);
pdf_set_info($pdf, "Author","Rasmus Lerdorf");
pdf_set_info($pdf, "Title","Sample Invoice");
pdf_set_info($pdf, "Creator", "See Author");
pdf_set_info($pdf, "Subject", "Sample Invoice");

$sizes = array('a4'=>'595x842', 'letter'=>'612x792', 'legal'=>'612x1008');

if(!isset($type)) $type='letter';
list($x,$y) = explode('x',$sizes[$type]);

$items = array(array('Our special low-cost widget that does everything','299.99'),
               array('Our special high-cost widget that does more','1899'),
               array('A blue widget','29.95'),
               array('And a red widget','49.95'),
               array('A yellow widget that makes noise','49.9'),
               array('And one that doesn\'t','999.95'),
              );

pdf_begin_page($pdf, $x, $y);

$im = pdf_open_jpeg($pdf, "php-big.jpg");
pdf_place_image($pdf, $im, 5, $y-72, 0.5);
pdf_close_image ($pdf,$im);

pdf_set_value($pdf, 'textrendering', 0); // fill

pdf_set_font($pdf, "Helvetica" , 12, winansi);
pdf_show_xy($pdf, 'Generic Evil Company Inc.',145,$y-20);
pdf_continue_text($pdf, '123 Main Street');
pdf_continue_text($pdf, 'Dark City, CA 98765');

pdf_set_font($pdf, "Helvetica" , 10, winansi);
pdf_show_xy($pdf, 'Helpless Customer Ltd.',20,$y-100);
pdf_continue_text($pdf, '2 Small Street');
pdf_continue_text($pdf, 'Little Town, ID 56789');

pdf_set_font($pdf, "Helvetica" , 10, winansi);
pdf_show_xy($pdf, 'Terms: Net 30',150,$y-100);
pdf_continue_text($pdf, 'PO #:  12345');

pdf_set_font($pdf, "Helvetica-Bold" , 30, winansi);
pdf_show_xy($pdf, " I N V O I C E ",$x-250,$y-112);

pdf_setcolor($pdf,'fill','gray',0.9,0,0,0);
pdf_rect($pdf,20,80,$x-40,$y-212);
pdf_fill_stroke($pdf);

$offset = 184; $i=0;
while($y-$offset > 80) {
    pdf_setcolor($pdf,'fill','gray',($i%2)?0.8:1,0,0,0);
    pdf_setcolor($pdf,'stroke','gray',($i%2)?0.8:1,0,0,0);
    pdf_rect($pdf,21,$y-$offset,$x-42,24);
    pdf_fill_stroke($pdf);
    $i++; $offset+=24;
}

pdf_setcolor($pdf,'fill','gray',0,0,0,0);
pdf_setcolor($pdf,'stroke','gray',0,0,0,0);
pdf_moveto($pdf, 20,$y-160);
pdf_lineto($pdf, $x-20,$y-160);
```

```
pdf_stroke($pdf);

pdf_moveto($pdf, $x-140,$y-160);
pdf_lineto($pdf, $x-140,80);
pdf_stroke($pdf);

pdf_set_font($pdf, "Times-Bold" , 18, winansi);
pdf_show_xy($pdf, "Item",30,$y-150);
pdf_show_xy($pdf, "Price",$x-100,$y-150);


pdf_set_font($pdf, "Times-Italic" , 15, winansi);

$offset = 177;
foreach($items as $item) {
    pdf_show_xy($pdf, $item[0],30,$y-$offset);
    pdf_show_boxed($pdf, '$'.number_format($item[1],2), $x-55, $y-$offset, 0, 0,
'right');
    $offset+=24;
    $total += $item[1];
}

pdf_set_font($pdf, "Times-Bold" , 17, winansi);
$offset+=24;
pdf_show_xy($pdf, 'Total',30,$y-$offset);
pdf_show_boxed($pdf, '$'.number_format($total,2), $x-55, $y-$offset, 0, 0,
'right');

pdf_end_page($pdf);
pdf_close($pdf);

$data = pdf_get_buffer($pdf);
header('Content-type: application/pdf');
header("Content-disposition: inline; filename=invoice.pdf");
header("Content-length: " . strlen($data));
echo $data;
?>
```

Ming-Flash

See http://www.opaque.net/ming/

```
<?
    $s = new SWFShape();
    $fp = fopen('php-big.jpg','r');
    $jpg = new SWFBitmap($fp);
    $w = $jpg->getWidth(); $h = $jpg->getHeight();

    $f = $s->addFill($jpg);
    $f->moveTo(-$w/2, -$h/2);
    $s->setRightFill($f);

    $s->movePenTo(-$w/2, -$h/2);
    $s->drawLine($w, 0);
    $s->drawLine(0, $h);
    $s->drawLine(-$w, 0);
    $s->drawLine(0, -$h);

    $p = new SWFSprite();
    $i = $p->add($s);

    for($step=0; $step<360; $step+=2) {
        $p->nextFrame();
        $i->rotate(-2);
    }

    $m = new SWFMovie();
    $i = $m->add($p);
    $i->moveTo(230,120);
    $m->setRate(100);
    $m->setDimension($w1.8, $h1.8);

    header('Content-type: application/x-shockwave-flash');
    $m->output();
?>
```

Output:

More Ming

Flash + RSS/XML

```
<?php
require 'XML/RSS.php';

$r =& new XML_RSS('slashdot.rdf');
$r->parse();

$allItems = $r->getItems();
$itemCount = count($allItems);
$width = 1000;
$m = new SWFMovie();
$m->setDimension($width, 70);
$m->setBackground(0xcf, 0xcf, 0xcf);

$f = new SWFFont("../../../fonts/Techno.fdb");

$hit = new SWFShape();
$hit->setRightFill($hit->addFill(0,0,0));
$hit->movePenTo(-($width/2), -30);
$hit->drawLine($width, 0);
$hit->drawLine(0, 60);
$hit->drawLine(-$width, 0);
$hit->drawLine(0, -60);
$x = 0;

// build the buttons
foreach($allItems as $Item) {

    $title = $Item['title'];
    $link = $Item['link'];

    // get the text
    $t = new SWFText();
    $t->setFont($f);
    $t->setHeight(50);
    $t->setColor(0,0,0);
    $t->moveTo(-$f->getWidth($title)/2, 25);
    $t->addString($title);

    // make a button
    $b[$x] = new SWFButton();
    $b[$x]->addShape($hit, SWFBUTTON_HIT);
    $b[$x]->addShape($t, SWFBUTTON_OVER | SWFBUTTON_UP | SWFBUTTON_DOWN);
    $b[$x++]->addAction(new SWFAction("getURL('$link','_new');"), SWFBUTTON_MOUSEUP);
}


// display them
for($x=0; $x<$itemCount; $x++) {

    $i = $m->add($b[$x]);
    $i->moveTo($width/2,30);

    for($j=0; $j<=30; ++$j) {
        $i->scaleTo(sqrt(sqrt($j/30)));
        $i->multColor(1.0, 1.0, 1.0, $j/30);
        $m->nextFrame();
    }

    for($j=0; $j<=30; ++$j) {
        $i->scaleTo(sqrt(sqrt(1+($j/30))));
        $i->multColor(1.0, 1.0, 1.0, (30-$j)/30);
        $m->nextFrame();
    }
    $m->remove($i);

}

header('Content-type: application/x-shockwave-flash');
$m->output();
?>
```

Output:

Resources

Home Page: http://www.php.net

Manual: http://php.net/manual

Tutorial: http://php.net/tut.php

Books: http://php.net/books.php

Index