



PHP mail();
Marcus Bointon
PHP London '08

PAR COURRIEL
BY EMAIL

Who's Marcus?

- Programming since 1982
- Years of multimedia production for CD-ROM
- In PHP since 2001
- Technical Director of Synchromedia Ltd
- Sole architect and coder of Smartmessages.net
- Delivering around 1 million messages/day
– NOT SPAM!
- This talk very loosely based on Wez Furlong's

Contents

- Anatomy of an email message
- MIME
- PHP email libraries
- Addressing
- Beyond sending

Anatomy of an email message

Header

```
From: Marcus Bointon <marcus@smartmessages.net>  
To: Marcus Bointon <marcus@synchronmedia.co.uk>  
Subject: Hello London!  
Date: Fri, 29 Feb 2008 14:00:33 +0000
```

Body

Is this thing on?

- Deceptively simple plain-text format
- a.k.a. RFC 2822, the tip of a veritable iceberg of RFCs...
- Many people get this wrong

MIME

- Multipurpose Internet Mail Extensions
- RFCs 1521, 2045, 2046
- Allows all kinds of interesting and useful structures, encodings and data formats
- Ubiquitous support
- Used all over the place outside email, especially on the web, all sorts of microformats

MIME Message

From: Marcus Bointon <marcus@smartmessages.net>
To: Marcus Bointon <marcus@synchronmedia.co.uk>
Subject: Hello London!
Date: Fri, 29 Feb 2008 14:00:33 +0000
Mime-Version: 1.0
Content-type: text/plain; charset="us-ascii"

Is this thing on?

- Just a simple extension to RFC 2822
- Now we have much more power and control
- Remember we're stuffing all kinds of media through a system built for ASCII

MIME multipart/mixed

- Probably the most common structure
- Perfect for message + attachments
- Notice that all elements have the same status
- Most clients will simply display all parts in order

```
From: Marcus Bointon <marcus@smartmessages.net>  
To: Marcus Bointon <marcus@synchronmedia.co.uk>  
Subject: Hello London!  
Date: Fri, 29 Feb 2008 14:00:33 +0000  
Mime-Version: 1.0  
Content-type: multipart/mixed; boundary="simple  
boundary"
```

```
--simple boundary  
Content-type: text/plain; charset=us-ascii
```

```
Nice piccy  
--simple boundary  
Content-type: image/gif  
Content-Disposition: attachment  
Content-Transfer-Encoding: base64
```

```
R0IGODIhAQABAIAAAP///wAAACH5BAEAAAAALAAAAABAAEAAAIC  
RAEAOw==  
--simple boundary--
```

MIME

multipart/alternative

- Presents alternative renderings of the same content with increasing quality
- Usually only one part displayed to user
- Classic text + HTML
- Text + image

```
From: Marcus Bointon <marcus@smartmessages.net>  
To: Marcus Bointon <marcus@synchronmedia.co.uk>  
Subject: Hello London!  
Date: Fri, 29 Feb 2008 14:00:33 +0000  
Mime-Version: 1.0  
Content-type: multipart/alternative; boundary="simple  
boundary"
```

```
--simple boundary  
Content-type: text/plain; charset=us-ascii
```

```
Plain text  
--simple boundary  
Content-type: text/html; charset=utf-8
```

```
<p><strong>HTML Text</strong></p>  
--simple boundary--
```

Nesting MIME

- A text/HTML message with attachments might follow this structure:
 - ▶ `multipart/mixed`
 - ▶ `multipart/alternative`
 - ▶ `text/plain`
 - ▶ `text/html`
 - ▶ `image/jpeg`
 - ▶ `application/msword`

Other MIME Types

- multipart/digest
 - Put multiple messages into one – often used by mailing lists daily digests
- message/rfc822
 - Forward as attachment
- message/delivery-status
 - Used for bounces
- Plenty more in RFC 2046

Constructing email messages

- PHP has the `mail()` function to submit a message to either to a local MTA (sendmail etc) or a remote SMTP server (set in `php.ini`)
- But it doesn't build messages
- We need a library to do that, and there are lots to choose from
- This gives us our first rule of email in PHP...

DON'T USE MAIL()!

- Even if your messages are pathetically trivial, pretend that the mail() function doesn't exist.
- While there's nothing wrong with mail() itself, it implies you're doing the rest yourself too!
- The email and MIME RFCs have many subtleties that are *extremely* easy to get wrong
- Much example code using mail() is wrong
- Use a library to do it right – there are plenty, and it's much easier than not

PHP Email Libraries

- PHPMailer
 - Used everywhere. Was stagnating, now has a new maintainer, PHP5 version coming soon
- Zend_Mail
 - Part of the Zend framework, well supported
- ezComponents
- Alexandria
 - OmniTI's library has some unique tricks
- More: SwiftMailer, PEAR etc

PHPMailer

```
$mail = new PHPMailer;  
$mail->AddAddress('marcus@smartmessages.net', 'me');  
$mail->From = 'marcus@synchronomedia.co.uk';  
$mail->Subject = 'Hello London!';  
$mail->IsHTML();  
$mail->Body = '<p><em>Pretty version</em></p>';  
$mail->AltBody = 'Plain version';  
$mail->send();
```

- Simple, sufficiently capable, very popular
- Showing its age, not super fast
- Has a new maintainer, and I do lots of support!
- <http://phpmailer.sourceforge.net/>

Zend_Mail

```
$mail = new Zend_Mail;  
$mail->addTo('marcus@smartmessages.net', 'me');  
$mail->setFrom('marcus@synchronmedia.co.uk', 'me');  
$mail->setSubject('Hello London!');  
$mail->setBodyHtml('<p><em>Pretty version</em></p>');  
$mail->setBodyText('Plain version');  
$mail->send();
```

- Part of Zend Framework
- Pluggable transports (sendmail and smtp)
- Experimental MIME parsing class too

ezcMail

- Part of eZ Components (<http://ezcomponents.org/>) - talk to Derick Rethans!
- Tries to do everything!
- Has familiar simple interface via ezcMailComposer class
- Has a much deeper notion of MIME, can build arbitrary structures
- Support for POP3, IMAP, SSL, MIME parsing too

Alexandria

- Part of OmniTi's open-source libraries
- Doesn't try to do everything, keeps it simple
- Has a trick others can't do – asynchronous SMTP queueing via a PHP extension
 - Message generation CPU bound, submission network bound – so do both at once
 - Requires you think about your sending process a bit harder as return values are async too.
- <http://labs.omniti.com/>

More to consider

- So far we've just sent messages blindly without much regard for what happens to them
- How do we know they got there?
 - Return receipts, but nobody uses them now
 - Opening tracker images, but images off now
- What happens if they don't?
 - They bounce, or get eaten by a Grue*.

* a.k.a. Hotmail

Addressing Recap

- To
 - Who the message is going to, also cc, bcc
- From
 - Who the message is from
- Reply-to
 - When you want replies to go elsewhere
- Envelope Sender
 - Vital, but you probably never knew!

Valid email addresses

- Defined in RFC 2822
- Validate every address at the point of entry into your system - easier than getting bounces later
- A bit more open than you might expect
 - blimey_o'reilly@example.com
 - add+subtract@example.com
 - [!#\\$&=?+{|}@example.com](mailto:!#$&=?+{|}@example.com)
- Most regexes will reject some valid addresses
 - <http://www.linuxjournal.com/article/9585>
 - <http://mythic-beasts.com/~pdw/cgi-bin/emailval>

Who are you sending to?

- 1 message : 1 recipient
 - Use the To field
 - NO MORE THAN ONE!
- 1 message : many recipients
 - Use BCC
 - But still set To to something reasonable
 - No personalisation, no security

The Return Path

- You can provide an envelope sender to your MTA, which will convert it into a return-path
- **That's where bounces go**
- As we'll see later, bounces can be hard work
- Because most users never see this address, we can fiddle with it
- The return-path is used as the to address for a bounce message
 - Its own return-path is “<>”

VERP

- Variable Envelope Return Path
 - <http://cr.yp.to/proto/verp.txt>
- Custom return-path for every address, or even every message - make up the pattern yourself
- Allows you to positively identify bounces even if the message you get back is corrupt
- Works *really, really well*
- Alexandria has built-in VERP generation
`bounces-marcus=
smartmessages.net@example.com`

Implementing VERP

- PHPMailer

- `$mail->Sender = 'bounces-a=b.com@example.com';`

- Zend_Mail

- `$tr = new Zend_Mail_Transport_Sendmail('-f bounces-a=b.com@example.com');`
- `Zend_Mail::setDefaultTransport($tr);`

- Alexandria

- `$mail->setVerpPattern("bounces-%TO%@example.com");`

PHP mail() – Summary

- Anatomy of an email message
- MIME
- PHP email libraries
- Addressing and VERP
- Setting yourself up for easier bounce handling

- In the next talk, we'll see what happens next
- Q&A