Five Simple Steps

A Pocket Guide

HTML email

by Andy Croll

A Pocket Guide to HTML Email
by Andy Croll

Published in 2013 by Five Simple Steps
Studio Two, The Coach House
Stanwell Road
Penarth
CF64 3EU
United Kingdom

On the web: www.fivesimplesteps.com
and: http://andycroll.com/
Please send errors to errata@fivesimplesteps.com

Publisher: Five Simple Steps
Editor: Owen Gregory
Tech Editor: Nathan Ford
Production Manager: Jo Brewer
Art Director: Nick Boulton
Design: Nick Boulton, Colin Kersley

ISBN: 978-1-907828-13-3

A catalogue record of this book is available from the British Library.

1

HTML email

The personal prejudices of many designers and developers are such that HTML email is the internet's dirty little secret. Unloved, under-discussed, sidelined and forgotten: the effective design and use of email is ignored by the majority of our industry. I'm here to tell you different. HTML email can be awesome in the right hands. My mission is to make those hands yours.

I'll talk in general about visual design, email in the age of the smartphone, writing, and how to keep your emails out of the dreaded spam folder. I'll even talk a little about the law, and what shady practices to avoid. I'll also show you techniques (not all of them elegant, I'm afraid) to make sure your emails are as beautiful as they can be for your readers.

Who is this for?

Designers, developers, business owners. You can all learn something, I hope, despite the short length of this little book. While the technical stuff in chapters 3 and 4 might dive deep into the guts of how to code an email, the principles and layouts discussed there are still worth consideration by the more business-minded individual.

Why send email?

We're all familiar with receiving and reading a vast quantity of email, but how often do we think about how we're being perceived when we send it?

An open secret of lots of successful businesses is the deployment of gorgeous, functional emails with cracking content. This leads to more engaged users and customers, which leads to more sales, sign-ups and happy smiles, back-slapping, party hats and the sense of a job well done.

The web is a passive medium: you can write insightful articles, create a beautiful website or application, but it's true that if you build it they won't necessarily come. Any business needs marketing, to tell its customers and potential customers about the cool new widget it just released. Email is fabulous for this. Email is pretty much the only way to reach out directly to your customers and remind them you're still around. This applies to one-man consulting shops as much as to department stores and internet behemoths.

Speed and simplicity

Once you have all your email templates set up you'll need very little time to launch a marketing campaign and the emails are sent straight away. No waiting for the printers for leaflets, no renting a venue, no marching around town in a sandwich board – just your message reaching your customer's inbox within minutes.

Measurable

One of the great marvels of the internet is the ability to measure and find out almost exactly what your users are interested in. This is also true for email. Most marketing email clients will track the opens and clicks within your emails, meaning you'll know whether people really do want to read the latest message from the CEO, or whether they'd much rather simply buy the latest Stephen King novel.

Kinds of email

I break emails down into three broad types, with differing ways to reach out to your customers.

Marketing email

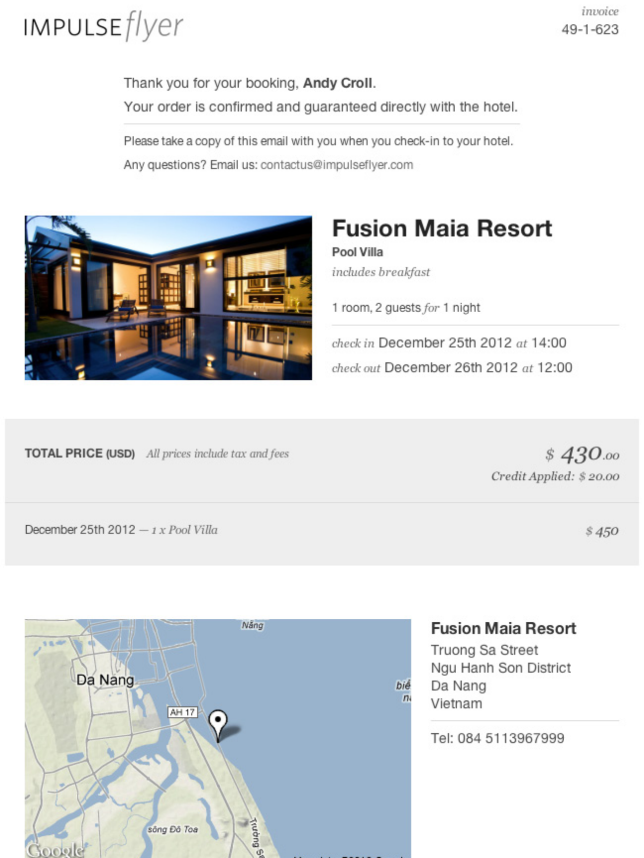
When you think of emails from companies and organisations, these are the emails you think about. Sent most often to the entire user-base of a website or service, they're used to announce new products, sales or launches. If the company creates a lot of supplementary content in its area of expertise, the messages are likely to include a few links to a blog or online reports. These are the emails that are most analogous to a TV or print advertisement, except they're delivered directly to your users, and you can see how engaging they find your missives. Useful.

Transactional email

Transactional emails are the nuts and bolts of your website: reset passwords; invoices; or changes to legal terms. The most important of these are receipts and invoices, the results of your customers buying your goods or services.

Do not underestimate the importance of transactional email. If you take one thing away from this book, let it be that your transactional and marketing emails should both receive the same amount of care and attention in design, content and copywriting.

At *ImpulseFlyer* our invoice also confirms a hotel booking. We therefore include all the details of the trip, information about the hotel, a map and a number to call us on if there are any problems.



IMPULSE*flyer*

invoice
49-1-623

Thank you for your booking, **Andy Croll**.
Your order is confirmed and guaranteed directly with the hotel.

Please take a copy of this email with you when you check-in to your hotel.
Any questions? Email us: contactus@impulseflyer.com

Fusion Maia Resort
Pool Villa
includes breakfast

1 room, 2 guests *for* 1 night

check in December 25th 2012 *at* 14:00
check out December 26th 2012 *at* 12:00

TOTAL PRICE (USD) *All prices include tax and fees* — *$ 430.00*
Credit Applied: $ 20.00

December 25th 2012 — *1 x Pool Villa* — *$ 450*

Fusion Maia Resort
Truong Sa Street
Ngu Hanh Son District
Da Nang
Vietnam

Tel: 084 5113967999

While this is a feature of businesses where the email is used as a proof of purchase, it's a good practice for all emails where you've taken money from your users! Why not write a little thank-you letter rather than send a dry, by the numbers note? Make your users feel good and they'll reward you with their business again.

Life-cycle email

There is a whole other book waiting to be written that deals with the when, how and why of life-cycle emails.

Life-cycle emails are generated by your application or by specific events triggered by your customers. They are typically attempts to re-engage customers, or to encourage a new visit to your website based on their behaviour in the past.

Examples of these emails you'll most likely be familiar with include:

- Facebook emailing you to tell you you've been tagged in a photo.
- Amazon letting you know that the new book by your favourite author is coming out.
- Twitter informing you of a new follower.
- A project management tool emailing you a couple of days before your free trial ends.

There's a crossover between your marketing and life-cycle emails. It's easy to recognise the continuum between an email sent to everyone announcing a new feature, through a new feature being available to users in a certain country, to “Hey, your free trial is about to expire.”

An aside on social media

This might be controversial. Hold tight.

We live now in the age of social media: Twitter, Facebook, Pinterest, MySpace. While these sites are great avenues for interacting with your users and should definitely be on your marketing radar, it’s worth considering a strong email presence before them.

Pew Internet suggests that *92% of online adults use email and 67% use it every day (PDF)*. About two-thirds of that number use social media. The audience (even for Facebook) is still a fraction of the total number of people online.

A problem with focusing on increasingly noisy social platforms for your marketing is whether you’ll be heard at all. You’re at the mercy of the algorithms of the social networks and the only way to guarantee your message is seen is to pay for an advert.

You might also have noticed that these very social networks are incredibly active in your inbox. There’s a reason for this: they know that telling you your friend just tagged you in a photo from the weekend is a great way to get you back on their site. This is definitely not to say you shouldn’t use social media; in fact, it may be a great way to build awareness and recruit users. But email is (or at least can be) personal and intimate and incredibly effective.

2

Email basics

If there's one rule about building an email list, it is: always be collecting. Always give your customers and users an option to hear from you.

Treasure the email addresses you are given. Treat these people's privacy and attention with respect and they'll think more of you and of your product. Seth Godin coined the term "*permission marketing*" and it's this you should keep in mind when you use the email addresses.

> *"Permission marketing is the privilege (not the right) of delivering anticipated, personal and relevant messages to people who actually want to get them.*
>
> *It recognizes the new power of the best consumers to ignore marketing. It realizes that treating people with respect is the best way to earn their attention."*

Buying email addresses… Just don't

While it's important to actively gather people's interest in what you're doing, beware the siren call of buying email lists or simply emailing anyone who's ever been in touch with you. Buying email lists is dubious at best and at worst is illegal. It does happen, but don't be that guy.

To double opt-in or not?

Some services send you a confirmation message when you first provide them with your email address. This email will contain a link that you have to follow to confirm your wish to receive future mailings. This is known as a double opt-in. Some email providers set this by default as it makes signing up to a list an explicit request:

the user cannot simply be signed up for emails by accident or by someone else.

Asking people to check their email and follow a link makes it much more likely they will accept future messages, but there will also be a significant number of people who'll never confirm. For that reason, my usual preference is to not operate a double opt-in, but to have every email contain a prominent unsubscribe link. However, it's up to you – do what feels right.

Let users unsubscribe

You need to have a clear, instant, one-click unsubscribe procedure on all your marketing communications and life-cycle emails. No excuses.

Do not be tempted to merely allow subscribers to change their communication preferences. If you do, many people will think you're just being awkward and weaselly, and your message is likely to end up in the spam folder – and you'll deserve it. No one wants to go through the rigmarole of turning off marketing emails, and they'll simply ditch it altogether, so do the right thing for your users.

And don't bother to send an email confirming that you have removed someone's email address from your database. That's just silly.

Sending emails

This is going to come as a shock to some of you, but you shouldn't send mass email from your personal account. Seriously.

Your personal email address should be used for just that: personal email. The last thing you want to do is use your desktop email application to send marketing communications. There are several reasons for this:

- Your email account probably has a limit on the number of emails you can send in one day.
- You cannot properly design or measure the deliverability of anything you send from a personal email account.
- You do not want to have your personal email address end up on a list of spammers.

Alongside the standard problems of using your email account to send email, having loads of addresses in the BCC field of your message is an obvious spam trigger. So, what should you do? Well, handling email properly, like so many things in life, means you have to either set things up yourself or pay somebody money to do it for you. I recommend the second option.

Time was that you'd drive to a data centre, plug in your expensive server to a rack, install an operating system, set up an incredibly complex mail server, and then wait, fretting, by your pager in case anything went wrong. Thankfully, those days are long behind us. And I'm showing my age by even mentioning them.

There is now a large number of services that let you send your emails, and they have teams of people dedicated to ensuring your emails get where you want them to go. The service providers worry about IP address blacklists, list management, monitoring engagement, and the infrastructure required to send emails to thousands of people all at once.

I'm going to talk briefly about a couple of email service providers that I've used in the past and then list a few others I've heard about.

Just to make it clear, I have no financial interest in any of the companies described in the next section. Any recommendations are based on excellent service that I have personally received.

Full service

A full service email provider deals with everything you might need. It worries about the hardware requirements, networking and reputation management, and often provides you with lovely default templates and a slick web interface.

They'll also help you be a responsible email-sending citizen. They don't want any of their architecture to be blacklisted, so they'll make sure (as far as they can) that you don't accidentally break the rules.

Campaign Monitor

Based in Sydney, *Campaign Monitor* has provided email list software with a lovely web interface since 2004. It allows agencies to white-label its web interface and provides an easy to use system to clients, and even makes a little money on top. It was also behind the formation of an *Email Standards Project* to put pressure on email software vendors to improve the CSS support of their email clients. A noble cause.

MailChimp

Another long-term email champion, *MailChimp* offers an extremely generous free plan and an irreverent monkey, Freddie, as a key part of their web interface.

Others

Both these providers try to outdo each other in support for the web design community. I've used them both for different projects and can happily recommend either. There are many alternative providers, some, like Campaign Monitor and MailChimp, catering to a small business crowd, and others moving much more in the enterprise direction.

- *MadMimi*
- *ConstantContact*
- *iConnect*
- *ReturnPath*

...and many more.

Nuts-and-bolts providers

The full service providers came first with their shiny web interfaces and generally excellent hand-holding customer service. In recent years, however, an alternative has emerged: email as a service. These providers require you to do a lot of the software heavy lifting before you plug into their systems to actually send the emails. They often involve sending messages programmatically through an API, and you don't get all the bells and whistles of a friendly web interface. Instead, you get incredibly good pricing, even compared to the good rates offered by the full service providers.

There are several companies that provide emailing services for applications, more targeted at the life-cycle and transactional email market:

- *Mailgun* – now owned by Rackspace
- *Sendgrid*
- *Mandrill* – from MailChimp
- *Simple Email Service* – from Amazon Web Services

These services are not really suited to the layman and you'll need a level of developer help to set the systems up. I mention them so you'll know they exist rather than recommending them for sending marketing email.

Real life

At ImpulseFlyer we use a combination of both approaches. Our in-application transactional and life-cycle emails are sent using an email as a service provider, and our weekly newsletter is sent from a full service provider. The technology team set up the templates to be flexible enough that the marketing team can use the web interface to send beautiful, branded emails with very little help from the designers and developers.

Spam

Your emails can end up in users' spam folders for many interconnected reasons: content; wording; punctuation; and even the mechanics of how your email is coded.

You need to run your subject lines and content through some of the standard spam tests. Fortunately, most of the emailing services discussed above can provide that service. There's also a *good web-based tool at Programmer's Heaven* which runs your email content through two well-known and widely used open source anti-spam systems: *SpamAssassin* and *SpamBayes*. You can also send a test email to *ISnotSPAM*, another clever service that tests your email against a few standard triggers that might cause it to be flagged as spam.

Wording and language

There are some lists around (from *Web Marketing Today*; *Mannix Marketing*; and *HubSpot*) that purport to be exhaustive collections of words you shouldn't use, but email deliverability is a lot more nuanced than that. However, making sure you're not using spammy language is a good start!

Some spam gotchas:

- Do not overpunctuate. You are not a fourteen-year-old flirting via SMS – no exclamation marks, and definitely not more than one.
- Using all uppercase is also a bad idea.
- Don't mention Viagra.

Legality

It's important to know that there are legal frameworks in place in most countries for sending digital marketing content – they're worth abiding by: no one likes fines. Or prison.

If you read the legislation across all major countries you'll find a large amount of commonality. A great deal of it uses similar terms and seeks to prevent the most major abuses of the email system we all depend on. None of the legislation can prevent spam entirely, but it is used to prosecute perpetrators of the worst, industrial-scale spam. It's worth understanding the legal requirements regarding email marketing, but if you behave politely and consider your users, you shouldn't find yourself in any bother.

United States

The Unites States has the CAN-SPAM Act (2003). Its major provisions are straightforward:

- An unsubscribe mechanism is present in all emails and should be honoured within 10 days.
- Accurate From addresses and relevant subject lines. If the content is adult in nature it should be labelled as such.
- A legitimate physical address of the publisher or advertiser is present.
- Email addresses should not be harvested, although if you have received a query or the email belongs to a customer (even without explicit permission) you are within the rules of CAN-SPAM.

For more detail, *read the Act itself* (PDF) or, more approachable than the original, *its Wikipedia page*.

Both the European Union and Australia have similar legislation. A European directive was passed in 2002 to encourage member states to pass their own legislation to protect consumers from spam. The *guidance for the UK* can be seen on the Information Commissioner's Office website. It provides a *PDF guide for organisations undertaking electronic marketing*.

The Australian Communications and Media Authority has *a very detailed website to help you comply with their Spam Act 2003*.

What does success look like?

So what defines a successful email campaign? It very much depends on what you are trying to achieve. Do your users re-engage with your product? Do they buy from you again? Do they simply visit your website or store?

Open rates

There is a fair bit of research into open rates and click rates that seems to indicate that a *20% open rate with a click rate of 3–7%* is a pretty reasonable barometer as an industry standard. MailChimp offers a detailed analysis of emails sent across its entire platform in *its research section*. As well as showing when most emails are opened (mid-afternoon laziness is a good driver of email checking!), they also include very detailed figures on open rates by industry, as well as examples and analysis of effective subject lines.

Don’t be a slave

However, being a slave to other people’s numbers is a mistake. You may have a small group of super-engaged, fanatical supporters, or you may have huge group of more casual users who don’t feel the need to devour your every word, but the ones who do are very keen. The most important things to track are how your emails perform over time. Your business will find its level for open rates, unsubscriptions and customer engagement. Measure against your own past performance as it’s the only reliable barometer you have. Don’t panic either. If one email doesn’t go down well it might be affected by seasonal variation, day of the week or even the time of day you send it.

For me one of the nicest things our customers at ImpulseFlyer say is that they look forward to receiving our weekly email. If we stand out as a high point in our customer’s weekly email barrage, that’s good enough for me.

3

Design and content

Before we get down to the business of coding, let's discuss the content and visuals.

Writing

Above all else, just like on the web, email is a writer's medium. Your email can be the prettiest thing in the world, but if the writing isn't good you won't get your message across.

The best summary of the importance of writing comes from *Rework* by Jason Fried and David Heinemeier Hansson.

> *"...[B]eing a good writer is about more than writing. Clear writing is a sign of clear thinking. Great writers know how to communicate. They make things easy to understand. They can put themselves in someone else's shoes[...]*
>
> *Writing is making a comeback all over our society. Look at how much people e-mail and text-message now rather than talk on the phone[...] Writing is today's currency for good ideas."*

If you are not a good writer, pay someone who can turn your words into something more readable. Some quick guidelines I use:

- Be clear, concise and say what you mean. Just like writing for the web, if you can say it in fewer words, do.
- Know your audience. Lose the industry-specific jargon, unless you are targeting a particular audience. In general, write like you're trying to make your grandmother understand.
- Be aware of your tone. Let your copy reflect the personality of you and your company.

Subject lines

Think about your subject line. This is the only chance you have to get a user to open your email. Get to the point. The right subject can have a large effect on your open rate. And there is a second issue at work here. The titles of emails are scanned very aggressively for keywords and spam wording, more even than the body text of your emails.

Typography

There are many greater typographers than me working on the web today, but suffice to say the same rules for websites apply to email.

Give your text room to breathe and so aid comprehension. Be sure to examine the actual legibility of your final email. Richard Rutter's "The Elements of Typographic Style Applied to the Web" translates some principles from the classic *The Elements of Typographic* Style by Robert Bringhurst into the online realm.

> "*Text on the web almost always benefits from an increase in line height, and figures upwards of 1.3 are common*"

It is also good to stick to approximately 45–75 characters per line for the main body text. This is a widely recommended range that makes your content easy to digest.

Visual design

When designing your emails it's worth thinking about how much design you need. Your job, with most emails, is to get the user to do or understand one thing. Like:

- You have made a purchase.
- You need to confirm your account.
- You should come and check out the cool stuff on our site.

Don't overcomplicate things. Restraint and focus can lead to very powerful and effective marketing messages. Here are some inspirational, successful examples of restrained email campaigns.

- *37signals*
- *Deus Ex Machina*
- *Made.com*
- *Enigma*
- *NetVibes*
- *Hot Rum Cow*
- *TimeOut*

Playing devil's advocate for a moment, there is a school of thought that suggests making your emails resemble a personal email can increase engagement. I prefer to use this technique when sending life-cycle emails, as it fits the conversational style of that use case.

Coding your email

As web designers we are in a maturing age of our medium. Competing browsers deliver new technology seemingly by the month and all the browsers from the major vendors are doing their best to one-up each other by improving performance and capability. **Forget that**. HTML email languishes in the web standards dark ages. It's time to put away grand visions of content and visual separation and reach for your pragmatic hat.

The main reason for this situation is the dire state of a lot of email software, both applications and websites. Worst among these offenders is Outlook 2007, because in that generation of Office, Microsoft chose to remove the decent-enough-for-the-time rendering engine from IE and replace it with the HTML renderer from Word. It has been that way ever since. While other email clients, notably Apple's Mail on iOS and Mac OS X, continue to improve their CSS support, Outlook continues to fall behind. So, we have a situation where the email client on your phone is hugely more capable than the one in a piece of enterprise software costing hundreds of dollars.

We can only play the hand we've been dealt, so let's stop our grumbling (at least until we talk about testing) and get going. It's time to party like it's 1999.

Tables

Oh, yes. Nested tables, even. This design technique from earlier, less enlightened days of the web is pretty much the only solution to ship attractively laid out email.

In simple terms, you need, even for the simplest emails, to nest one table inside another. One table is set to be 100% wide and its single `<td>` cell should be aligned to the centre. This is your background table.

```
<table width="100%" bgcolor="000000">
  <tr>
    <td width="100%" align="center">
      <table width="600" bgcolor="FFFFFF">
        ... everything goes in here ...
      </table>
    </td>
  </tr>
</table>
```

You'll note that I've set a `bgcolor` on both tables. This is because some clients strip background colours from the `<body>` element (where you also should be setting the same background colour using CSS).

You’ll need to:

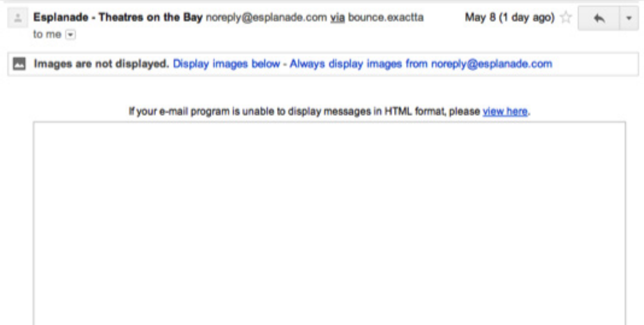
- Include `cellspacing="0" cellpadding="0"` and `border="0"` in every `<table>` element, mostly to make your maths easier.
- Set widths on every `<td>` tag – some email clients make terrible guesses.

Images

If there’s one thing that drives me nuts it’s the constant use of one massive image with HTML email.

Do. Not. Do. This.

Let me state it clearly: this is a *terrible* idea. Many clients ship with a default behaviour of blocking images. Therefore, having all of your message buried inside a huge JPEG renders your message invisible. An email containing a single image is also a flag for spam in many anti-spam algorithms.



Esplanade - Theatres on the Bay noreply@esplanade.com via bounce.exactta May 8 (1 day ago)
to me

Images are not displayed. Display images below - Always display images from noreply@esplanade.com

If your e-mail program is unable to display messages in HTML format, please view here.

Using images, however, should be a key part of making your content appealing. It's also a good idea to make your images links. People like to click on pictures so you may as well ensure that you are encouraging interaction.

As a back-up to the blocked images problem, ensure that you have useful (and click-worthy) `alt` attributes. As a bonus you can actually style the `alt` text with inline CSS on the `<img>` element in some email clients. Note that if you have any significant Lotus Notes usage, don't use PNGs – you'll need to use JPEGs or GIFs.

Width

In most situations you'll need to keep your emails between 550 and 650 pixels wide. The email experts at Campaign Monitor found in 2011 that *just under three-quarters of their surveyed users were producing emails in that range*. They concluded:

> *"The width limitation is largely set by the small, bottom-right aligned preview panes that feature in desktop clients such as Outlook and Apple Mail."*

This maximum width constraint is also a good thing for readability. Having emails constrained to around 600 pixels means that *the number of characters per line is approximately the optimum level for readability of 45–75 characters* if you use a sensible 14–16px for your body text.

CSS

Ignore everything you learned about web standards. Email is not a good place to expect all your CSS to work. The short story for the best compatibility:

- Colour, font size, height and margin are mostly OK.
- Use system-installed CSS font-stacks.
- Do not use anything other than tables for layout, as floating is inconsistent.
- Background colours need to be set on table cells (for example, `bgcolor="ffffff"`).

The gang at Campaign Monitor maintain *an epic CSS compatibility chart*.

Why inline?

Gmail strips `<style>` elements from the HTML so you need all your styles to be inline to the elements. All other clients leave the `<style>` block well alone. A good solution here is to code locally, separating your CSS into a `<style>` element in the `<head>` of the template and then pass the CSS through a service like *Premailer*. This will automatically inline your styles. You'll find that if you use a service the inlining will be done for you, meaning you can maintain at least some readability in your development template files.

Buttons

Buttons are a great click target in emails. We've been training our users to scan webpages for buttons (and to click them!) for the entire history of the web.

I did an awful lot of research into what techniques you can use to design a flexible button for consistent use across the wide variety of email clients. That's three days I'll never get back, so I feel I should share.

The simplest way to code up buttons that look like buttons is to simply create a button in your preferred image editor and include the image in your email. While you're guaranteed visual fidelity, the drawbacks to this approach should be obvious. Not only do you have the image blocking issues that plague HTML email (and this time on your main call to action), but if you want to change the text of your button, you'll need to fire up your image editor again.

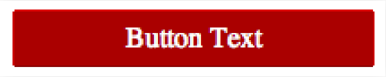
My solution is to once again return to the trusty HTML table used for layout. Sorry. For each button, create a 3×3 table, and use `bgcolor` on the table cells to fake a 3-D effect. Make the all the external cells 1px thick – these will be the button's bevelled edge. The centre cell is then filled with a clickable link. You can also make the corner cells have a background colour that matches the background behind the button.

This example creates a red button.

```
<table width="202" cellpadding="0" cellspacing="0" border="0">
  <tr>
    <td width="1" height="1" bgcolor="FFFFFF"></td>
    <td height="1" bgcolor="DD0000"></td>
    <td width="1" height="1" bgcolor="FFFFFF"></td>
  </tr>
  <tr>
    <td width="1" height="30" bgcolor="CC0000"></td>
    <td width="200" height="30" align="center" style="line-
       height: 30px;" bgcolor="AA0000"><a href="#" width="200"
       style="color:FFF;text-decoration:none;"> Button Text </a>
    </td>
    <td width="1" height="30" bgcolor="CC0000"></td>
  </tr>
  <tr>
    <td width="1" height="1" bgcolor="FFFFFF"></td>
    <td height="1" bgcolor="880000"></td>
    <td width="1" height="1" bgcolor="FFFFFF"></td>
  </tr>
</table>
```

You can see examples of both a fixed- and flexible-width button in *chapter-3-redbutton.html*.

Here is how the final button will look.



If the text is going to remain relatively short, you can set a fixed width on the central cell as I have in the example above. For a button that works at multiple widths, you'll need to add a few non-breaking spaces (`&nbsp;`) to either side of the button text.

Video

Embedding video files in your email is a bad idea. It wasn't always so, but owing to the increased security of email clients, as well as abuse of video by spammers, these days client support is virtually nonexistent.

However, people love videos and they are often a great way to get your message across. You've probably found on the websites you've worked on that a good video tutorial or explanation can work wonders. The simplest technique is to slap a big play button overlay on a linked image, designed to look like a video player.

Anecdotally, we've found at ImpulseFlyer that this technique does not confuse users and also increases clicks on the video image, particularly if the image is the main focus of the email. A fun enhancement might be to use an animated GIF to illustrate a few frames of your video. If you really want to get fancy, take a look at this *in-depth blog post from Ros Hodgekiss at Campaign Monitor* that discusses the use of HTML5 `<video>` markup in email.

CSS3

While the latest CSS techniques shouldn't be used to layout your emails, you should feel free to enhance elements of your email design with (inline) CSS3.

Judicious use of text shadow, gradients or RGBa colours can really add a little something extra. While these effects will not display on all clients, and may be stripped in others, more advanced clients will see the fruits of your labour. This is really just the progressive enhancement approach to web design applied to email.

> "*Progressive Enhancement (PE) is the principle of starting with a rock-solid foundation and then adding enhancements to it if you know certain visiting user-agents can handle the improved experience.*"

We're more limited when working with email clients rather than web browsers, but there is no reason not to provide the best experience to those using more capable applications.

Testing

Once you've created your masterpiece, you need to make sure that you see what your readers see. There are several approaches to take.

DIY

Get an email from each of the major webmail providers. Right now that means a core of:

- Gmail
- Yahoo! US
- Yahoo! (somewhere else, US behaves differently!)
- Hotmail/Outlook.com

Other web email clients should be selected based on the email addresses in use by the recipients of your mails. Sending to a lot of aol.com addresses? Probably a good idea to test on that!

For software you probably need to use:

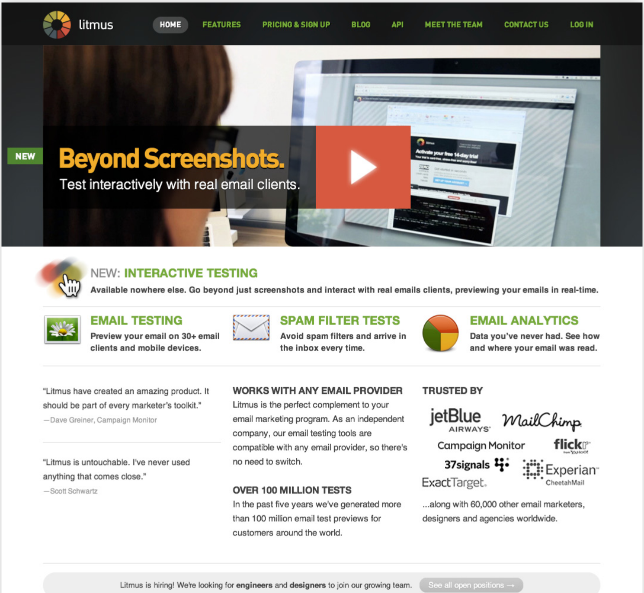
- Outlook 2007/2010 (and 2013 coming soon at the time of writing)
- Apple Mail
- iOS Mail
- Android Mail/GMail app

You can then use the service to send tests to your various email addresses and see how things look.

There are some apps for that

Of course, there are a number of online services which will help you test your emails. The main player in this arena is *Litmus*. You can even interact with a huge variety of clients directly through your browser. It's really quite clever stuff.

Both Campaign Monitor and MailChimp include functional testing tools. MailChimp has the *Inbox Inspector* and Campaign Monitor has *Inbox Preview*.



NEW: INTERACTIVE TESTING

Available nowhere else. Go beyond just screenshots and interact with real emails clients, previewing your emails in real-time.

EMAIL TESTING

Preview your email on 30+ email clients and mobile devices.

SPAM FILTER TESTS

Avoid spam filters and arrive in the inbox every time.

EMAIL ANALYTICS

Data you've never had. See how and where your email was read.

"Litmus have created an amazing product. It should be part of every marketer's toolkit."
—Dave Greiner, Campaign Monitor

"Litmus is untouchable. I've never used anything that comes close."
—Scott Schwartz

WORKS WITH ANY EMAIL PROVIDER

Litmus is the perfect complement to your email marketing program. As an independent company, our email testing tools are compatible with any email provider, so there's no need to switch.

OVER 100 MILLION TESTS

In the past five years we've generated more than 100 million email test previews for customers around the world.

TRUSTED BY

jetBlue AIRWAYS, MailChimp, Campaign Monitor, flickr, 37signals, Experian CheetahMail, ExactTarget

...along with 60,000 other email marketers, designers and agencies worldwide.

Litmus is hiring! We're looking for **engineers** and **designers** to join our growing team.

A/B testing

A lot of tools allow you to A/B test the subject lines, layout or content of your emails. They send two versions to an initial group of your customers and select the better performing version to send to the rest of your recipients. This can be a useful way to learn which copy resonates best with your user base. Used over time it can help you refine what your users are looking for.

Conclusion

So that's the basics. It's not pretty, but it can be very effective. In the next chapter we'll talk a little about the ways you can (and probably should) adapt this approach for the modern world, but mostly for that little slab of glass, metal and plastic in your pocket or bag.

Templates

If all the HTML in this book has put you to sleep or terrified you, feel free to use templates. You can buy them from an online marketplace such as *ThemeForest* and simply swap out a few things like the logo and colour schemes. Alternatively, most of the email management providers offer template generation tools, which will get you most of the way to an awesome-looking email.

HTML email boilerplate

Just as there is a set of best practices for a simple HTML5 page at html5boilerplate.com, there is a also a similar project for HTML email. Take a look at *http://htmlemailboilerplate.com*. It includes a lot of extra guidance and a few CSS and HTML tweaks that'll help you overcome even the strangest email client bugs.

4

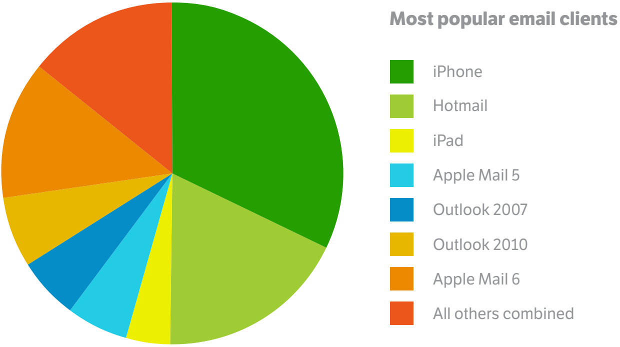
Email on mobile

Luke Wroblewski pulled together some stats for email showing a third of email globally was opened on a mobile device. *Some recent statistics compiled by the guys at Viget*, from various online surveys, make enlightening reading.

> *“There are more iPhones sold every day than there are babies being born (378,000 vs. 371,000).”*

> *“Android: 1.3 million device activations per day and climbing.”*

Campaign Monitor’s most recent research into over seven billion email opens shows the continuing rise of mobile, and that 2012 is the year when mobile email finally overtook the desktop, with over 80% of mobile opens on iOS and most of the remaining fifth on Android. This research no doubt mirrors your own experience if you own a smartphone. You checked your email on it today right? Thought so. It certainly matches our experience at ImpulseFlyer. Here’s a chart showing our percentage open rate by device.



I think I've made my point that making email look good on an iPhone (or similar) is incredibly important.

Responsive email (or, more properly, adaptive layouts)

Enter adaptive layouts for email.

You've probably heard and read a lot about responsive web design, a technique that has swept across our industry. If not, immediately buy *Responsive Web Design* by Ethan Marcotte, I'll be right here when you get back.

If you can't spare nine dollars, you should at the least *read his seminal article on responsive web design on A List Apart*. Responsive web design is defined by three techniques:

1. Flexible, grid-based layouts.
2. Flexible images and media.
3. Media queries, a module from the CSS3 specification.

We're working in the badlands of HTML email, nesting tables like naughty girls and boys. As we've already discussed, flexible width layouts and sexy CSS is not a good idea in this arena, which is why it is better to use the term adaptive layouts to describe the following techniques that modify your email designs for smartphones.

These methods are by no means universally supported (something of a theme in HTML email development) as Windows Phone 7 does not support CSS media queries, and it's best not to even think about the horrors of BlackBerry HTML rendering. However, Android and iOS devices manage very well. Given that the majority of smartphones are sold by Apple and the partners of Google, it seems like a good idea to progressively enhance our emails.

Technique 1: Changing the width of cells

This is the simplest technique. Its best use is simply to modify the width of the email so that you do not end up with really tiny text on small smartphone screens or, worse, horizontal scrolling due to badly scaled images. This involves giving each `<td>` a width in the code as normal, but adding a class to the cell to alter its width on a mobile device, using media queries.

As in the previous chapter, the regular styles are inline to the elements. This ensures that clients such as Gmail, that strip all `<style>` elements, get the email as intended. We then use a combination of a `<style>` element in the HTML `<body>` containing media queries and the new CSS rules with an `!important` declaration to change the appearance of elements on small screen devices.

```
<html>
  <head>
    ...
  </head>
  <body>
    <style>
      @media only screen and (max-width: 480px) {
        table[class=w280], td[class=w280], img[class=w280] {
          width: 280px !important; }
        table[class=w10],  td[class=w10],  img[class=w10]  {
          width:  10px !important; }
      }
    </style>

    <table>
      ... email content as before ...
    </table>
  </body>
</html>
```

This example merely indicates the structure of the email. For a full example see *chapter-4-adaptive-simple.html*. This technique works best when content (other than thumbnails) is kept in a single column.

Technique 2: Hiding supplementary content

If you'd like your desktop client emails to have more exciting layouts than a single column, you'll want to use a slightly modified technique.

Simply design the content of your email so that your primary content is in one cell for each of your content tables. Then, if you have secondary or non-essential content in your email, let that go in other cells. Just as in the initial adaptive technique, use media queries in the `<body>` to modify the styles applied in mobile browsers. In this case we use `display:none;` to hide supplementary information.

```
<td>
  <table width="600" class="w320">
    <tr>
      <td width="20" class="w10">&nbsp;</td>
      <td width="370" class="w300">Main Content</td>
      <td width="20" class="hideonphone">&nbsp;</td>
      <td width="170" class="hideonphone">Non-essential stuff</td>
      <td width="20" class="w10">&nbsp;</td>
    </tr>
  </table>
<td>
```

See *[chapter-4-adaptive-hiding.html]* email to see it in action.

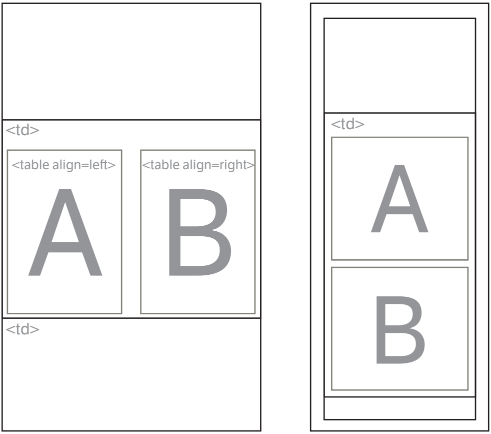
We use this technique on ImpulseFlyer to display extra images that add some visual embellishment to the presentation of a hotel. It could also be used to hide a table of contents, if you use those in your mail. You can even use the same technique to hide extra content, wrapping content to be hidden on mobile in a `<span>` element with the correct class. If you end up doing that a lot, perhaps you shouldn't include such content in the full email either.

This might go without saying, but make sure your call to action isn't in the hidden column!

Technique 3: Two-column tables

The final technique involves using two independent tables next to each other, while nested inside a single parent `<td>` from the structural table. The tables need to be under half the width of the parent cell and each needs to be given either `align="left"` or `align="right"` in the markup.

Within the responsive styles, you then set the width of both tables to be equal to the parent table, and the tables appear one on top of the other.



The simplest code to describe this technique looks like:

```
<td>
  <table width="280" class="w320" align="left">
    <tr>
      <td width="0" class="w10">&nbsp;</td>
      <td width="280" class="w300">left and on top on mobile</td>
      <td width="0" class="w10">&nbsp;</td>
    </tr>
  </table>
  <table width="280" class="w320" align="right">
    <tr>
      <td width="0" class="w10">&nbsp;</td>
      <td width="280" class="w300">table on right and below on
        mobile</td>
      <td width="0" class="w10">&nbsp;</td>
    </tr>
  </table>
<td>
```

You can see an example of this technique in *chapter-4-adaptive-two-column.html*. Note: this works in WebKit browsers (if you open the example on your desktop) but it seems not to work in Firefox. At the moment, however, there is no mobile Firefox-based email client, so this technique is still useful.

All together

The best thing about these techniques is that if you use the approach of one background table, one structural table, and then nest your content tables within, you can use all three techniques depending on your needs.

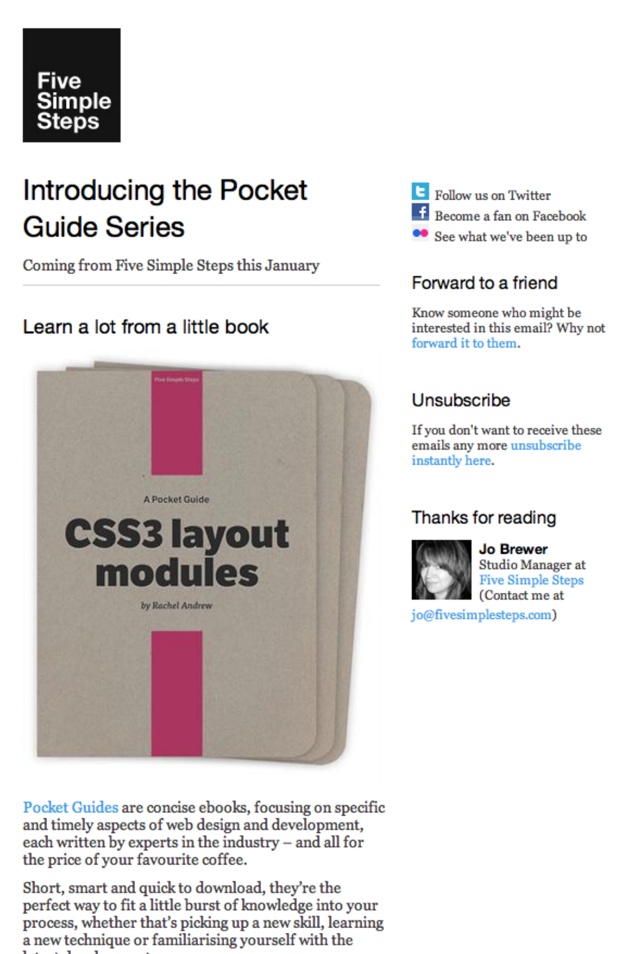
If you want to see a real life application of these techniques, look no further than the Five Simple Steps email newsletter.

5

A real example

As part of this project I agreed to adapt the Five Simple Steps email newsletter to mobile, hopefully to show how little effort it takes to make a huge difference to your customers' experience of your email.

Here is how the email looked before I started.



Introducing the Pocket Guide Series

Coming from Five Simple Steps this January

Learn a lot from a little book

Pocket Guides are concise ebooks, focusing on specific and timely aspects of web design and development, each written by experts in the industry – and all for the price of your favourite coffee.

Short, smart and quick to download, they're the perfect way to fit a little burst of knowledge into your process, whether that's picking up a new skill, learning a new technique or familiarising yourself with the

Follow us on Twitter
Become a fan on Facebook
See what we've been up to

Forward to a friend

Know someone who might be interested in this email? Why not forward it to them.

Unsubscribe

If you don't want to receive these emails any more unsubscribe instantly here.

Thanks for reading

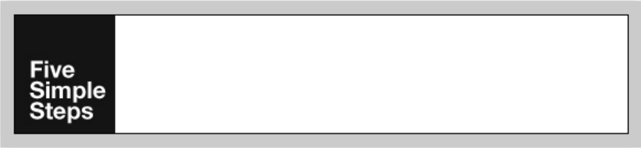
Jo Brewer
Studio Manager at Five Simple Steps
(Contact me at jo@fivesimplesteps.com)

As you can see, the basic layout is pretty simple: a logo at the top, followed by a primary content column on the left and a secondary content column on the right. The secondary column contains useful information, but it is clearly secondary to the main content. Were we to simplify the design to a single column, this would appear below the main body of the email.

This makes the body of the email a good candidate for the two-column approach.

An aside: the header

The header of the email was a full-width image with the Five Simple Steps logo on the left corner and the rest of the image had a white background.



However, if you just shrink the wide image using our first adaptive technique, the logo will become small and unreadable. My solution to the problem involves two simple techniques.

1. Just serve the logo, as there’s no need for the white padding area of the image. We can then use our adaptive width technique to make sure the image is correctly sized on smaller screens.

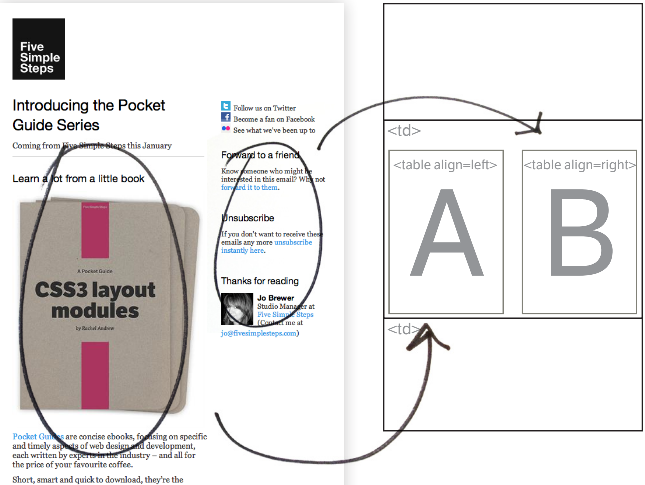
2. I also made the image twice as large as required, as we set the width of the image in the HTML. This technique lets high-density (Retina) devices scale the larger image so you end up with a sharp image.

   There are *other, more thoughtful, techniques you can use on the web*, but I've found the simple double-sized image to work best with email.

   I considered using a background colour on a cell with white text rather than including an image, which is a feasible solution if your logo is simple and uses web-safe fonts. In this case, though, I decided to replace the image.

The email layout

So, my proposed layout for the email looks like this:



Five Simple Steps

Introducing the Pocket Guide Series

Coming from Five Simple Steps this January

Learn a lot from a little book

Five Simple Steps
A Pocket Guide
CSS3 layout modules
by Rachel Andrew

Pocket Guides are concise ebooks, focusing on specific and timely aspects of web design and development, each written by experts in the industry – and all for the price of your favourite coffee.

Short, smart and quick to download, they're the

Follow us on Twitter
Become a fan on Facebook
See what we've been up to

Forward to a friend

Know someone who might be interested in this email? Why not forward it to them.

Unsubscribe

If you don't want to receive these emails any more unsubscribe instantly here.

Thanks for reading

Jo Brewer
Studio Manager at
Five Simple Steps
(Contact me at
jo@fivesimplesteps.com)

There is a single-celled background table, with an inner table (at a width of 580px) used to centre the email content, and then within the cells of this inner table we add the header, the two tables for the body content, and the footer.

The inner table header `<td>` contains a table containing just the logo, sized using HTML.

The next inner table row contains two tables. All the main markup goes into the first table, which has its width set to 360px and also set to `align="left"`. The supplementary content goes in the second table; it also has its width set (to 180px this time) and is set to `align="right"`.

The footer `<td>` contains a table containing some text and links to unsubscribe and view the email on the web.

Adaptive modifications

This is a relatively simple layout so all we need to do is add a few classes to elements whose appearance we want to change. The images and the tables all get a class of `w300` which I then set, inside a media query, to `width: 300px !important;`.

I've also used an `h21` class where I wanted to make a particular spacer row shorter on mobile.

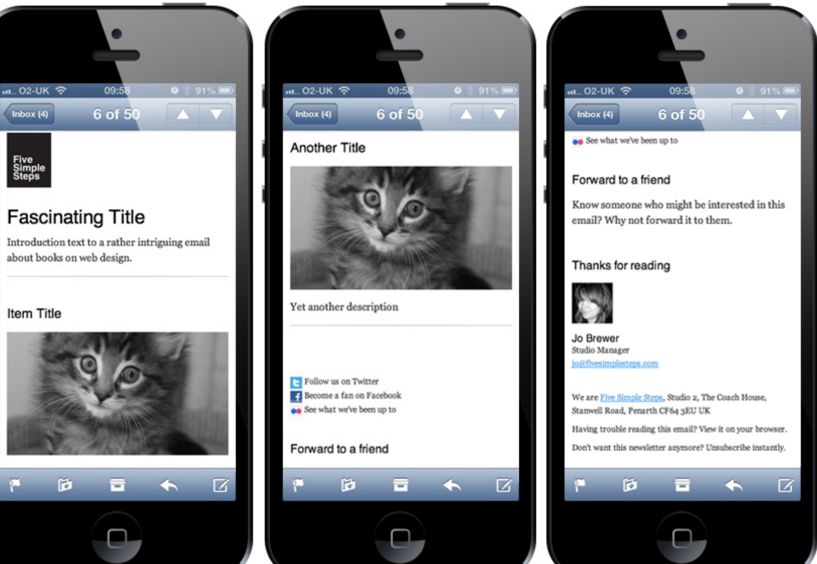
The Five Simple Steps logo takes up quite a lot of vertical space on a small screen, so I also applied a small width and height to the logo image, but kept it all in proportion.

You'll also notice the sidebar message about unsubscribing from the newsletter is gone from the mobile version of the email.

I've simply added a `hideonmobile` class to the table cell. In the single column mobile layout, it doesn't make sense to include the same messages so close to each other.

You'll see in the email footer I've used the `hideonmobile` class again to hide a spacer cell, so that the footer message is only as wide as the main content on desktop, but in the single page view it is the full width of the email.

When the display window is narrower than 480 pixels and the media query is engaged, all of the widths and heights are overridden and the single-column layout emerges.



Fascinating Title

Introduction text to a rather intriguing email about books on web design.

Item Title

Another Title

Yet another description

Follow us on Twitter

Become a fan on Facebook

See what we've been up to

Forward to a friend

See what we've been up to

Forward to a friend

Know someone who might be interested in this email? Why not forward it to them.

Thanks for reading

Jo Brewer
Studio Manager
jo@fivesimplesteps.com

We are Five Simple Steps, Studio 2, The Coach House, Stanwell Road, Penarth CF64 3EU UK

Having trouble reading this email? View it on your browser.

Don't want this newsletter anymore? Unsubscribe instantly.

I’ve included the basic layout template for the Five Simple Steps newsletter. It hasn’t had any CSS inlined (this needs to happen before sending) but it also makes it easier to look at the source.

chapter-5/fss-newsletter/example.html

You now have the tools...

So, go fix your email now.

I'm serious. Your users will thank you. Your finance team will probably thank you as well. Remember, be nice, respect your users' inboxes and drop a little well-written (and designed) joy into your users' lives every now and then.