



The Essential Guide to HTML Email Design

**Emailmovers Limited, Pindar House, Thornburgh Road
Scarborough, North Yorkshire, YO11 3UY**

**Tel: 0845 226 7181 Fax: 0845 226 7183
Email: enquiries@emailmovers.com**

All contents copyright Emailmovers Limited 2013

Index

	Page
Introduction	3
Images	4
Layout	5
CSS (Cascading Style Sheets)	6
Animation and Scripting	7
Best Practice HTML Email Example	8
How Spam Filters Think	9

Introduction

HTML email is a new form of media that is growing rapidly with the popularity of the Internet.

Email marketing is an industry still in its infancy, and is expanding in all directions - from small businesses, to big international companies - the benefits are obvious - reliability, specific sector targeting and cost effectiveness are just some of the advantages compared to standard advertising methods.

Styled messages have lots of appeal for marketers, designers and consumers - but for coders, it's a real headache to create them properly.

There is a lot of misguided information regarding what is involved in sending out an email campaign. One of the major factors which is misunderstood is how to construct a HTML email.

Many people fall foul of not knowing how to construct an HTML email. Unfortunately, email isn't the web. Designs that look fantastic in one email program, might not look the same in a different email program.

Creating an HTML email that looks right to every reader is a job that makes cross-browser testing complicated. After reaching its final destination, an email could be read using any of a number of stand-alone email clients (such as Outlook, AOL, and Eudora), and even more Web-based clients (Yahoo, Hotmail, and Gmail). Each of these email clients displays HTML email messages a little bit differently, so ensuring that your message displays correctly can be a daunting task.

Here is a simple guide to help you through the construction of an HTML email, offering you advice as to best practices.

Images

Using images and photos is the main reason people send HTML emails. The proper way to handle images in HTML email is to host them on a web server, then “pull them in” to your HTML email, using “absolute paths” in your code (see opposite). Graphics can’t be sent along with a message.

Use at least 70:30 text to image ratio. Email providers often block images and URLs in images. So more text ensures your message gets read – and also reduces spam scores.

Always optimise the image file size. Make sure you use GIF and JPEG correctly to get the lowest file size. Use JPEG only for photos and images containing gradient colors. GIF - for plain sharp images.

Don’t use images for important content such as call to action, headlines and links. Outlook, Gmail and others turn images off until allowed by the viewer. If your entire email is graphical, all your recipients are going to see is a lot of broken images.

Declare both the width and height parameters for images. Some email programs will default to “0” for undeclared width or height.

Provide alt text for all images. That way the reader will see some content in place of the disabled images. This works particularly well for logos. But never use spam messages in the alt tags, such as “Call now to claim your FREE DVD!!”

The correct way



Slice large images into sections and place them in their own table.

Code image tags using absolute paths...

Example:

```

```

Including the following code within the image tag can help remove table spacing problems with certain emailclients...

```
style="display: block;"
```

Never use background images. Gmail, among other, will ignore any `url()` attribute in an inline style, and the simple `background=` tag attribute.

You can use background colours (`bgcolor` tag attribute)

Layout

Use tables for layout as they are more consistently supported. Avoid using <div> tags for positioning and layout.

Avoid complicated layouts. You've got to keep your layouts very simple, if you want it to work across all the major email applications.

Do not use fancy image borders and complicated structures, as this will predominately show incorrectly in most inboxes.

Never assume that the email will look exactly the same on all email programs even though you did all the necessary tests. There will always be someone using an old email program that will distort your email.

Declare width, cellpadding, and cellspacing for all tables. Doing so will result in a fixed width for the template. Don't leave widths to chance because many email clients like to "fill in the blank" which can break the design.

Max Page Width 600 pixels

Main Email Title

Sub heading

[Call to action link](#)



Create focal points to guide the viewer to the areas of most importance. In this example the copy is set to the bottom - this allows other areas to breathe with effective use of space.

It is not necessary to display large amounts of text; this can appear too overwhelming. Provide only what is required. Excite the viewer with snippets of information to wet their appetite, and a more tantalising appeal can be achieved.

[Call to action link](#)

Phone
Email address
Web address
Company address

- Bullet point
- Bullet point
- Bullet point
- Bullet point
- Bullet point

Golden tips



Do Not Use White Text

Email providers read white text as invisible text – raising your email's spam score. Also, should the background not appear for any reason, then it will render all white text unreadable. An alternative font colour could be a light grey, such as "#CCCCCC" which works well with dark backgrounds.

Avoid using the "COLSPAN" attribute should the table layout be complex. You might think about chopping it up into separate tables instead. Better to go with more "simple" tables, than one big "complex" table.

5.

CSS (Cascading Style Sheets)

CSS (Cascading Style Sheets) is not widely supported for HTML email. For instance, you shouldn't expect DIVs to work. And definitely stay away from CSS positioning.

Some email programs are moving away from full CSS support. As of early 2007, Gmail is the most restrictive as it will ignore all external and embedded style sheets, but it will support inline styles.

You'll have to rely on old-fashioned <TABLES> for your layouts in HTML email, and only use CSS for simple font formatting and colors. Always design CSS to "fail gracefully." That means if someone took away your CSS, your design and content would still display well. Before you send your HTML email, delete the CSS and see what it looks like.

CSS is especially a problem if you're using an HTML Editor to design your HTML email, because HTML Editors insert all kinds of over-written CSS and DIVs by default. You might as well code by hand.

Using Microsoft Word to generate HTML is even worse than using WYSIWYGs. Word adds so much code behind the scenes, it's unbelievable. Don't do it! Learn to code HTML.

If your email design uses lots of CSS, be sure to check it in Google's Gmail. Gmail strips out virtually all CSS, and therefore demonstrates a good guide to final rendering.

Golden tips



- Go **easy on CSS!** There is no need to use CSS excessively. There is a thin line between an email that reaches the recipient and the email that 'junks'. Many leading email clients have poor CSS support. For example Outlook 2007 is using the MS Word HTML rendering engine which means basic CSS support, no background images etc.
- Make sure all your CSS is within the <body> tag. Everything outside <body> will be ignored. Using **Inline** styles ensures that TEXT fonts, colours and sizes do not conflict with the styles of our headers and footers.
- Avoid using CSS-positioning.
- Use Inline styles liberally in tables.

The correct way



Inline CSS Example:

```
<font style="color: #000000;  
font-family: arial, san-serif;  
font-size 12px"></font>
```

Animation and Scripting

Like so many other aspects of HTML Email, animation is considered problematic. Most Email Programs simply will not be able to render it.

Also, most anti-virus applications block the code used to embed Flash movies, and they often block JavaScript, ActiveX, and even background music files. This has been used in the past to spread viruses, so it's commonly blocked now.

You can deliver HTML emails with Flash embedded, but most recipients won't be able to view them - unless they're using an email program that uses a browser to render their HTML email.

If you must use animation, then consider creating a landing page - which will be linked through email.

Unless you know for a fact that every single one of your recipients uses Outlook 2000, and you know their anti-virus applications won't block your embedded media from running, don't send Flash, JavaScript, ActiveX, or movies in HTML email.

Golden tips



If you've got a great animation or movie to show to your recipients, just send a simple, GIF or JPEG graphic in your HTML email, then link it to a "landing page" on your website with the animation in it.

The same goes for ActiveX, and movie files. Anti-virus applications block them from running. So, fancy rollover or pop-up navigation and streaming videos just aren't going to work either.

Avoid using JavaScript. Most email software will disable it anyway.




Avoid common mistakes

Do not use the following:

- JavaScript
- Flash
- Animated GIFs

This type of content will not render in certain Email inboxes.

Best Practice HTML Email Example



1
Main Email Title
Sub heading

[Call to action link](#) **2**

3

4
Create focal points to guide the viewer to the areas of most importance. In this example the copy is set to the bottom - this allows other areas to breathe with affective use of space.

It is not necessary to display large amounts of text; this can appear too overwhelming. Provide only what is required. Excite the viewer with snippets of information to wet their appetite, and a more tantalising appeal can be achieved.

[Call to action link](#)

Phone
Email address **5**
Web address

Company address

- **Bullet point**
- **Bullet point**
- **Bullet point**
- **Bullet point**
- **Bullet point**
- **Bullet point**

6

1 HTML text headlines are the first thing the viewer will see.

2 Have a recognisable call to action.

3 Keep images to a minimum and slice up any large ones.

4 HTML Text used predominantly throughout in a clear and simple structure.

5 Always include your company's contact details.

6 Two columns is easier to read, rather than one continuous scroll.

How Spam Filters Think

How to Avoid Getting Your Campaigns Accidentally Junked.

If you send email campaigns long enough, you will inevitably run into spam filter issues. We've found that on average, you can expect anything up to 15% of your emails to just get lost in cyberspace, mostly due to overzealous spam filters. You don't even have to be a spammer to be spam-filtered. Innocent email marketers who send permission-based emails to people who requested them get spam filtered all the time.

Unfortunately, there is no quick fix. The only way to avoid spam filters is to understand how they work.

Spam filters look at a long list of criteria to judge whether or not your email is junk. For example, they might look for spammy phrases such as "**CLICK HERE!**" or "**FREE! BUY NOW!**" They'll assign points each time they see one of those phrases. Certain criteria get more points than others.

Here's a sample of criteria from Spam Assassin, one of the most popular spam filters out there:

- Talks about lots of money (.193 points)
- Describes some sort of breakthrough (.232 points)
- Looks like mortgage pitch (.297 points)
- Contains urgent matter (.288 points)
- Money back guarantee (2.051 points)
- Why Pay More? (1.249 points)

9.

Golden tips



Avoid Spam words as they will result in your email being blocked, such as...

"Win, No Obligation, Cash, Money, Special Offer, Free, Click here, Financial"



Avoid common mistakes

Using spam phrases, such as "Click here!", "Free" or "Once in a lifetime opportunity!"

Going crazy with exclamation points!!!!!!

USING ALL CAPS, WHICH IS LIKE YELLING IN EMAIL.

Coding sloppy HTML.

Creating an HTML email that's nothing but one big image and no text.

Using spam words and symbols in the subject line.