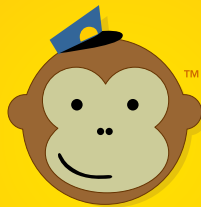


# Designing, Coding & Delivering HTML Email

.....  
*A beginners survival guide from*



MailChimp

# Table of Contents

<b>Introduction.....</b>	<b>4</b>
About the Second Edition .....	4
But First: “What The Heck is MailChimp?” .....	5
<b>How HTML Email Works .....</b>	<b>6</b>
The Multipart/Alternative MIME Format .....	6
Image Files in HTML Email.....	7
And don’t use free image hosting services! .....	8
Delivering HTML Email.....	8
<b>Designing and Coding .....</b>	<b>9</b>
Tools of the Trade .....	9
Design Guidelines .....	10
Set your width to about 500-600 pixels.....	10
Simple Layouts and Tables Work Best.....	11
Special Considerations for Browser-based Email Applications .....	11
CSS in HTML email.....	11
Flash, (and JavaScript, ActiveX, movies, and other stuff that won’t work) in HTML email.....	14
Point emails to landing pages .....	14
Background Colors .....	14
Anatomy of a Good, Healthy HTML Email Newsletter .....	15
Plain-text Email.....	16
The Art of Plain-text email.....	16
<b>Common Mistakes to Avoid .....</b>	<b>19</b>
<b>Designing around spam filters .....</b>	<b>22</b>
How anti-spam systems work.....	22
Bayesian Filtering .....	22
Blackhole Lists.....	22
Email Firewalls.....	23
Challenge/Response Filters. ....	23
Tactics for Avoiding Spam Filters.....	24
<b>Testing &amp; Troubleshooting Your Email Designs .....</b>	<b>26</b>
Web Designers: emails aren’t the same as web pages!.....	26
Testing in Different Email Applications .....	26
AOL.....	26
Testing Browser-based Email Services.....	29

Yahoo!Mail.....	29
Yahoo!Mail Beta.....	29
Hotmail.....	29
Gmail.....	29
Check Different ISPs, too.....	30
Send tests to friends and co-workers.....	30
<b>Email Marketing Basics &amp; Best Practices .....</b>	<b>32</b>
E-marketing Etiquette (How <i>Not</i> To Be A Jerk) .....	32
The CAN-SPAM Act of 2003.....	33
Double Opt-in .....	34
Pros & Cons .....	34
Segmenting & Experimenting.....	35
Segment Your Lists .....	35
Experiment Like Crazy.....	36
Personalizing Emails with MailChimp * MERGE_TAGS *.....	36
<b>Measuring Performance .....</b>	<b>38</b>
Open Rates.....	38
Click Rates.....	38
Unsubscribe Rate .....	39
Bouncebacks.....	39
MailChimp Helps Manage Your Bounces .....	40
Traffic to your website.....	41
Signups since last campaign .....	41
<b>Wrapping Up .....</b>	<b>43</b>
“My First Email Campaign” Checklist for Beginners.....	43
<b>Appendix .....</b>	<b>46</b>
<b>Useful Resources .....</b>	<b>47</b>
Email Marketing Sites.....	47
Email & Marketing Related Blogs .....	47
Products & Tools .....	47
Graphic design programs.....	48
Miscellaneous .....	48
<b>Email Applications &amp; Known Issues .....</b>	<b>49</b>

# Introduction

We wrote this “Beginner’s Guide” for people who are new to designing, coding, and delivering HTML email newsletters, and for people who are new to email marketing in general. It’s meant to be a plain-English, no-nonsense guide “for newbies,” so our apologies in advance if you were looking for something more complicated and harder to understand.

You don’t have to be a professional web designer to use this guide, but a little HTML knowledge will definitely help. First, we’ll cover all the **basics**, like how HTML emails work. Then, we’ll get into **technical stuff**, like how to design and code them. Finally, we’ll run through email marketing **best practices**, such as for campaign management, deliverability and measuring performance.

## About the Second Edition

This is the 2<sup>nd</sup> edition of this guide. The first edition was published on our website in 2001, and has been downloaded over 50,000 times now. Since it’s been nearly 5 years, a lot’s changed in the world of email marketing, so we decided this guide needed a huge re-write. What’s different? For one, email applications support HTML email a lot better now. As long as we keep designs nice and simple, we don’t have to worry so much about our emails breaking catastrophically all the time (all of our old, “don’t try this in AOL5!!!!” warnings have been removed). Also, “double opt-in” isn’t such a radical idea anymore, so there’s no need to really “sell” it so hard, or compare it to single opt-in.

**But new challenges have popped up.** For instance, spam, scams, and phishing attacks threaten deliverability for legitimate email marketers, because major ISPs and software makers are *constantly* coming up with new counter-measures to protect their customers. And if we want our customers to read our emails, we have to design with those counter-measures in mind. Designers and marketers now have to “learn to think like a spam filter” when creating newsletters (just like learning to think like a search engine when coding websites). Also, the CAN-SPAM Act became law on January 1, 2004. Email marketers need to understand the rules, in order to comply (or potentially face hefty fines of \$11,000 *per offense*).

With all that in mind, let’s get started!

## But First: “What The Heck is MailChimp?”

[MailChimp](#) is a product of [The Rocket Science Group](#), a web development company in Atlanta, Georgia. Our one, driving motivation is to “build tools that are **Useful**, **Simple**, **Efficient**, and that people *love* to use.” **U.S.E.** is the formula and philosophy for everything we do (incidentally, it started out as s.u.e, but that sounded way too litigious).

Way back in 2001, we noticed some of our clients were trying to send out HTML email campaigns and monthly newsletters to their customer lists, but stuff kept breaking. They were rigging Microsoft Outlook “stationery”, or trying to send ginormous PDF attachments. Images were missing, tables were broken, formatting was off. Their customers were complaining. They were exceeding their ISP bandwidth quotas. Bounces were out of control. Sheesh! It was a *mess*.

They could handle *design* just fine. Even a little HTML coding. But it was the **delivery** part that they needed help with. See, to send HTML email *properly*, you have to set up a server to deliver in “Multipart-Alternative MIME format,” then track bounces with “VERPs” and other geeky stuff like that (we’ll go over that soon).

So we built a simple little tool where they could log in, copy paste their customer list, copy paste their email code, and hit “send.” And since *they* took care of the *creative* stuff, and we just handled the *boring* tasks that “only a helper chimp should do,” we called it “MailChimp.”

Since then, thousands of customers from all over the world have discovered the simplicity of MailChimp. Maybe if you find this guide useful, you’ll give MailChimp a try, too (trust us, you’ll love it).

Learn more at: <http://www.mailchimp.com/>

# How HTML Email Works

Before you can start designing, coding, and sending HTML emails, you should know how it works, and what tools you'll need. Here's some background information every email designer and marketer should know...

## The Multipart/Alternative MIME Format

The most important thing you need to know about HTML email is that you **can't just attach** an HTML file and a bunch of images to a message and hit "send." Most of the time, your recipients' email applications will break all the paths to your image files. And you can't just paste all your code into your email application, either. Most email apps send messages in "plain-text" format by default, so the HTML won't work. That means your recipients would see all that raw source code, instead of the pretty email it's supposed to render.

You need to send HTML email from your server in "**Multipart-Alternative MIME format.**" Basically, that means your mail transfer agent bundles your HTML *code*, PLUS a plain-text version of the message, **together into one email.** That way, if a recipient can't view your beautiful HTML email, the good-old-fashioned plain-text version of your message is auto-magically displayed. It's kind of a nerdy gobbledy-geek thing, which is why a lot of people mess things up when they try to send HTML email themselves. You either need to program a script to send email in multipart/alternative MIME format, or just use an outside vendor (ahem, like [MailChimp](http://www.MailChimp.com)) to deliver things for you.

In MailChimp, you simply paste both versions of your email into our interface (see below), and we handle all that "multipart stuff" for you...

**3. Plain Text Email**

Paste your plain-text message below. You can insert \*[MERGETAGS]\* to personalize the message. bigger | smaller

Track Clicks [\[?\]](#) Pop-up Preview

**4. HTML Email**

Paste your HTML email's code below. You can insert \*[MERGETAGS]\* to personalize the message. bigger | smaller

Track Clicks  Track Opens [\[?\]](#) Pop-up Preview

*Note: Red callouts in the image point to the text input fields with the text: "Paste your plain-text message here..." and "...then your HTML code here".*

## Image Files in HTML Email

Embedding images and photos into messages is the number one reason people want to send HTML email. The proper way to handle images in HTML email is to *host them on your web server*, then “pull them in” to your HTML email, using “absolute paths” in your code. If you’re not a web designer, and you use “WYSIWYG” programs such as Microsoft Front Page, look up “relative paths” vs. “absolute paths” in your help docs for a complete explanation. Basically, you can’t send the graphics along with your message. You host the graphics on a web server, then the code in your HTML email downloads them whenever the message is opened. Incidentally, this is how “open tracking” works. You place a tiny, invisible graphic into the email, and then track when it’s downloaded. This is why open tracking only works in HTML email, not plain-text, and why the new email applications that block images by default (to protect your privacy) can screw up your open rate stats.

### When coding Image Tags in HTML email...

#### Do this:

```

```

#### Instead of this:

```

```

## **And don't use free image hosting services!**

When it comes to hosting the images for your HTML email, you really need your own server to do it. Don't try hosting images on a free "image hosting service," because those websites often put scripts in place to prevent you from linking to them in emails (they can't handle all the traffic). And since you really do get what you pay for, free image hosting services tend to be pretty unreliable under heavy traffic conditions. Also, spammers use free image hosting services all the time, to "cover their tracks." If you don't want to look like a spammer, use your own web server.

## **Delivering HTML Email**

Many newbies make the mistake of setting up forwarding lists, or "CC'ing" copies of a message to all their customers. This causes all sorts of problems, like when a customer hits, "reply to all." Plus, there's no way to do any kind of individual tracking or personalization when they CC: a big group like that. Finally, it just looks so unprofessional and impersonal.

That's why when an email marketing system (like MailChimp) sends your campaign, we take your message and send it one at a time to each recipient on your list (really, really fast). Unlike your work computer linked to your local ISP (which probably has a standard monthly bandwidth limit), email marketing vendors like us use dedicated mail servers that are capable of sending hundreds of thousands of emails (even millions, for larger vendors) per hour.

# Designing and Coding

Now that you have a basic understanding of how HTML email campaigns work, and how you should deliver them, we can start talking about designing and coding them (the fun part!).

## Tools of the Trade

First, here are the tools you'll need to do the work...

1. **A design application**, like Adobe Photoshop, Fireworks, Dreamweaver, or Illustrator. Use these tools to layout your template, and slice out your graphics (like your company logo, and product photos). We're really partial to Fireworks, but to each his own.
2. **A good, "pure" HTML/Text editor**, like BBEdit and TextWrangler for the Mac, or HomeSite, and NoteTab Pro for the PC. You *could* use "WYSIWYG" style tools to generate HTML, like Microsoft FrontPage, or maybe Adobe GoLive or Macromedia Dreamweaver. But they have their disadvantages. WYSIWYGs very often throw in gobs of junk into your code (FYI, Microsoft says the extra code is to help with "round tripping" which means you can export from Word to Powerpoint to HTML, then back again, without losing anything). All this extra code usually ends up just breaking things, or setting off spam filters. There are some WYSIWYGs out there that generate "clean code" but we've found those are a bit "*too perfect*," because they're designed for web pages, not HTML email (so they don't know all the things you have to "rig" in your code.). If you want to do HTML email *correctly*, you should learn to code HTML "from scratch" with a good text editor.

### Free HTML Email Templates:

If you haven't already found them, we offer free HTML email templates at:

<http://www.mailchimp.com/resources/templates/>

They include graphic source files, HTML code, and they've already been tested in all the major email applications.

3. **Your own web server**, to host all your images and archives. You should create a folder on your website for email newsletters. This is where you'll store images, then point to them with absolute paths in your HTML code. You might also want a folder on your website to store archived newsletters. We created a folder on our website called, /monkeywrench/ and then keep campaigns under /campaigns/ with subfolders named by /date/ to store each campaign. Everybody has their own organization style, though. Just be sure to come up with a system and process for archiving, so that each newsletter isn't a huge chore to publish.
4. **An FTP program**, like Fetch for the Mac, or CuteFTP for PCs. When you design your email and slice out all your graphics, you'll use FTP to move those files from your computer to the server.
5. **A test machine (or two)**, loaded with as many different email applications and accounts as possible. You'll want to check your HTML email designs under lots of different scenarios. Unlike web design, where you only have to check your work in a small handful of browsers, there are tons of email applications out there, and they all handle HTML email differently.

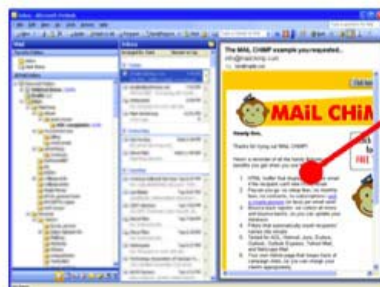
## Design Guidelines

Here are some pointers for designing your email. Early warning: If you're a seasoned web designer, you're not going to be happy with all the compromises and hacks you'll have to make when designing for email applications.

### Set your width to about 500-600 pixels.

If you've ever worked on a website, you're probably used to designing pages to fit in 800x600 pixel resolution screens, or maybe even 1024x768. But that won't work when you design HTML email. Most recipients will be looking at your email through their "preview pane" which is usually a small portion of their available screen (see screenshot below).

### Outlook 2003 on 1024x768 monitor



Preview pane  
is about 440  
pixels wide on  
this monitor.

Above is a screenshot of an HTML email in Microsoft Outlook's preview pane. Outlook is maximized to full screen (1024x768 pixels), but as you can see, the preview pane is only about 440 pixels wide. In **AOL**

9, the “preview pane” is only about 194 pixels wide, then when you click “Full View,” it typically opens up a window that’s about **540 pixels** or so. This is why you shouldn’t design emails to be much larger than **600 pixels** in width.

### **Simple Layouts and Tables Work Best**

You wouldn’t believe all the different ways email applications garble up HTML email. You’ll need to keep your email designs nice and simple. Avoid complicated layouts, too many embedded tables, and tables with too many rows and columns. A simple two column table with a row across the top is about as complex as you can get. There are lots of email applications that don’t cooperate when you code tables with colspans (table cells that “span” across multiple columns) in them. One application that’s particularly finicky when it comes to tables (and just about everything *else*) is Lotus Notes. We can’t stress it enough. You’ve got to keep your layouts very, very simple, if you want it to work across all the major email apps.

If you’re working on an email layout and find the tables are getting complex, and you’re having to code too many “COLSPANS,” you might think about chopping it up into separate tables. For instance, we almost always use separate tables for the header, body, then footer of our emails. Better to go with more “simple” tables, than one big “complex” table.

### **Special Considerations for Browser-based Email Applications**

A lot of your recipients will be checking their email in their web browser, such as with Yahoo!Mail, Hotmail, or Gmail. Since they’re viewing email in browsers, there are certain things you need to know when you code your HTML:

- They’ll strip out your <HTML>, <HEAD>, and <BODY> tags, in order to keep your code from interfering with their web pages. That means...
- Any background colors you specify in your <BODY> tag will be lost. So wrap your entire email inside a larger, 100% wide table, and set a background color to that table.
- Any CSS that you place inside the <HEAD> tags will be lost. So don’t try to link to CSS files on your server. Use embedded CSS instead, and make sure it’s below the <BODY> tag. If you’re a web designer, that’ll definitely make you feel “kinda dirty” because that’s not exactly the “proper” place to put embedded CSS. But no worries---it’ll work. If you just can’t stomach that, use inline CSS.

### **CSS in HTML email**

CSS (Cascading Style Sheets) is a Godsend for websites, but don’t put too much faith in it with HTML email. For instance, you shouldn’t expect DIVs to work. And **definitely** stay away from CSS *positioning*. Won’t work. If you’re a CSS-standards freak, you are going to hate yourself after coding HTML email.

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 your CSS to "fail gracefully." That means if someone took away your CSS, your design and content would still display decently. Before you send your HTML email, delete the CSS and see what it looks like. Did you have tiny text that is now gigantic, and blowing out your tables?

CSS is *especially* a touchy situation if you're using a WYSIWYG to design your HTML email, because WYSIWYGs insert all kinds of crazy CSS and DIVs by default. It's one of the drawbacks of WYSIWYGs for HTML email. It may help you get things coded, but you still have to understand HTML enough to go back and *remove* some of the code, so things won't break in email applications.

If your email uses lots of CSS, be sure to check it in Google's Gmail. At the time of this writing, they don't fully support CSS, so you'll see your fonts defaulting to Arial, and black in color. Here's what our [MonkeyWrench Newsletter](#) looked like in Outlook 2003 (which supports CSS just fine) vs. Gmail...



Gmail stripped out virtually all of my CSS! Where did my font sizes, font colors, and line-height go? And since the fonts are so much larger in Gmail, it could have blown out my table cells (if I didn't know what I was doing). But notice the top line (next to the peeking chimp). It wrapped my text into 2 rows. Blech!

## **Flash, (and JavaScript, ActiveX, movies, and other stuff that won't work) in HTML email.**

You can *deliver* HTML emails with Flash in them, but most recipients won't be able to view them. This is because Microsoft Outlook and Apple Mail are the only major email applications that use the operating system's built-in browsers to render their email. Other email applications use their own, proprietary ways of rendering HTML. Plus, most people have **anti-virus applications** that block the code used to embed Flash movies.

### **Point emails to landing pages**

If you've got a great animation or movie to show to your recipients, just send a simple, intriguing GIF or JPEG graphic in your HTML email, then link it to a "landing page" on your website with the animation in it. Same goes for JavaScript, 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. The only email application on our test machines that will actually play fancy stuff (like Flash and movies) seems to be Outlook 2000 (which is kind of old, these days). 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 stuff from running, don't send Flash, JavaScript, ActiveX, or movies in HTML email. Just won't work reliably.

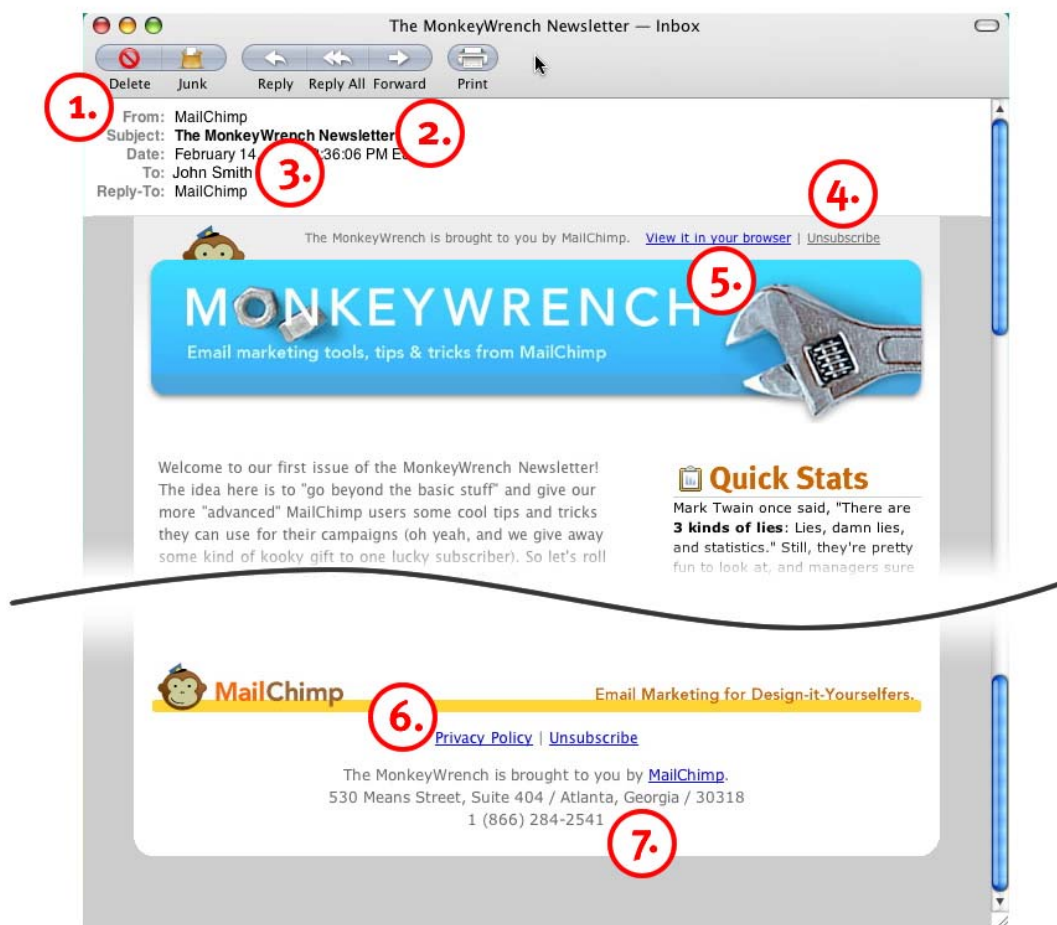
### **Background Colors**

Remember when we told you (above) about how browser-based email services (like Yahoo!Mail and Hotmail) strip out your <HTML>, <HEAD>, and <BODY> tags? That's what you need to remember when coding your background colors and images. You normally specify that stuff in your <BODY> tag, but you can't rely on that with HTML email. You'll need to create a big, 100% wide "table wrap" around your email. Set your background color in that table, and use some CELLPADDING as needed. If you're into CSS, you might be tempted to use a <DIV> instead. But we've seen DIVs break quite frequently in email applications, so we don't recommend them.

# Anatomy of a Good, Healthy HTML Email Newsletter

Here's what a well-built, happy, healthy HTML Email Newsletter looks like...

1. Your company name in the "From." Recipients should recognize who the email is from instantly. It can't be deceptive in any way (duh). If a recipient has to strain his brain to remember who you are, he'll click "this is spam" instead of opening.
2. A relevant subject line (don't be "spammy"). So they instantly know who the email is from, and what it's about (hence, "subject" line).
3. The "To:" field of your email should be personalized to the recipient's name, not their email address ([see how to do this in MailChimp](#)).
4. A one-click opt-out link that removes people from your list immediately. Consider placing it at the *top* of your email (as well as in the footer), so that people who want off your list can find it really easy (instead of clicking their "Junk" button). If you use MailChimp's Managed List functionality, you can use our **\*UNSUB\*** tag to generate the unsubscribe link.
5. In addition to your opt-out link, you might also include a link in your header for recipients to "view this email in your browser." Point it to an archived version of your email on your server. This helps if the email got forwarded to friends, and then got mangled along the way.
6. A link to your company's privacy policy should be in your email
7. Your valid, physical mailing address (P.O. Boxes aren't good enough), and as much contact information as possible. The more contact information you provide, the more reputable your email will look.



## Plain-text Email

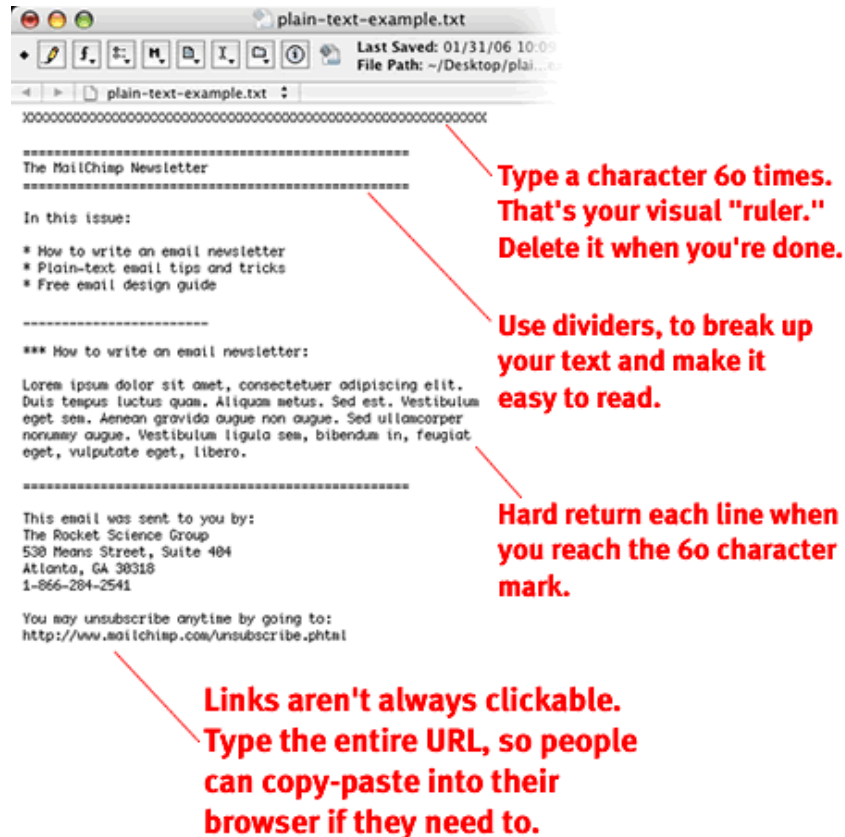
Don't forget your plain-text email! You need it for people who can't (or won't) view HTML. If you don't take the time to create the plain-text version of your HTML email, you'll just look like a lazy spammer (spam filters will penalize you for only sending HTML). When you send a campaign in MailChimp, you'll need to create both versions of your message (HTML & Plain-text), then paste them into our [Create Campaign screen](#). Then we "bundle" them together in "multipart format" and send them off. Your recipients' email application decides which version to display.

### The Art of Plain-text email

There's definitely an art to constructing your plain-text email. You can't just copy-paste, or batch-convert the text from your HTML email. It has to be formatted *juuust* right, so the best way to do it is to build it from scratch.

Here are some tips for your plain-text emails:

1. Use a plain text editor, like TextWrangler for the Mac, or Windows NotePad. Don't use something like Microsoft Word, because it won't give you the kind of "raw" text files you need. You need an application that gives no formatting whatsoever.
2. If you're using MailChimp, create a plain-text message in your text editor, then simply copy-paste it into the "plain text message" field in MailChimp. No uploading required. Or, you can just type straight into MailChimp's interface ([screenshot of Create Campaign page](#)).
3. By default, most email applications start to wrap your plain-text messages at about 60 characters or so. So you'll want to enter a hard return after each line gets about 60 characters long. One nice little trick is to type a letter 60 times across the top of your message, and use that as your visual "ruler."
4. People don't read email. They *scan* email. So make it really easy to find important information with bullet points and visual markers (see screenshot below)
5. You can't code a clickable link with plain-text email. You just have to type out the entire URL. Some email applications will make it clickable automatically. Some won't.



One thing we've noticed is that a lot of people spend so much of their time on the HTML version of their message, they're too "drained" to work on their plain-text email. So they either ignore it, or they slap some junk in, or they "cheat" a little, and type in something like, "Evidently, you can't view HTML email, so visit this URL in your browser to see our newsletter in all its glory." We think this is a big mistake (even though we've cheated a few times like that ourselves). So don't put **all** your energy into the HTML version of your email. Save some love for your plain-text message too.

### **MailChimp Trick:**

When you setup a managed list in MailChimp, you have the option to "let my users choose plain-text only" when they fill in your opt-in form. MailChimp will automatically deliver the plain-text formatted email to them when you send your campaigns.

[Read about it in our knowledge base.](#)



## **MonkeyBrains!**

**MailChimp's MonkeyBrains blog is chock full of HTML email design examples, tips, tricks, and hacks, plus email marketing news and best practices.**

<http://mailchimp.blogs.com>

# Common Mistakes to Avoid

Here are some of the most common mistakes people make when coding HTML emails:

1. Not coding “absolute paths” to their images. Remember, attaching graphics and using “relative paths” won’t work. You need to host the images on your server, then link to them in your code.
2. Using JavaScript, or ActiveX, or embedding movies. That stuff just doesn’t work in HTML email.
3. Getting over ambitious with designs. Designing HTML email isn’t the same as designing web pages. You can’t have all the CSS-positioning, DIVs, DHTML, and complex, embedded tables like you can in web pages.
4. Using free image hosting services. They don’t like it when they get lots of outside hits (such as from email campaigns). So they’ll sometimes start blocking your images.
5. Forgetting---or refusing---to include an opt-out link. It’s stupid and unprofessional not to allow recipients to unsubscribe from your list. Oh yeah, it’s illegal, too. Be sure to read and understand [The CAN-SPAM Act of 2003](#).
6. Linking to an external CSS file. You usually put CSS code in between the <HEAD></HEAD> tags of web pages, right? But browser based email applications (like Yahoo, Hotmail, and Gmail) strip <HEAD> and <BODY> tags from HTML email. Embed your CSS below the <BODY> tag.
7. Letting your permission “grow cold.” If you’ve been collecting opt-ins at your website like a good email marketer, but you haven’t sent an email in several months, your subscribers have forgotten all about you. So when they receive a full blown email newsletter from you “out of the blue” they’re going to click their “this is spam button” in their email application (related blog post: [Surviving the inbox whack-a-mole game](#)). Don’t let your permission grow cold. Send occasional emails. If you’ve been collecting emails for years, and you’re only *just now* sending your first email to everyone, you should send a little “re-invitation” to your list. Something like, “A really long time ago, you subscribed to my email list. Well, I’ve finally got some time to start sending my really cool newsletter, and I want to make sure you still want to receive it. Click here to sign up...” At the very least, send an email and place some kind of text at the top that reminds them of where and when they opted in. People usually forget about opting-in to something after about 6 months or so.
8. Sending to a list without permission. This is the worst offense. Lots of “innocent” marketers, who “mean well” commit this heinous crime. Here are some common ways legitimate marketers can inadvertently become seen as “evil spammers:”
  - Getting an email list from a tradeshow. “But I’m exhibiting there, and the tradeshow host said it would be okay, and when people purchased tickets, the fine print said that we could email them, and...” Nope. You’re spamming. Don’t do it. If they didn’t give you permission to email them, they didn’t opt-in to your list. If they didn’t opt-in, you’re



spamming. Even if you can legally send them email marketing, those recipients are more likely to report you for spamming them. Then, you'll get blacklisted. Plus, your company will look really slimy. Don't do it. If a tradeshow host is collecting email addresses, then they should be doing the emailing. It's all about permission, and setting expectations.

- Getting a list of “fellow members” from some trade organization. Just because they joined a club, and the club posted contact information so everyone could keep in touch with each other, it doesn't mean each member gives you permission to send them newsletters and offers. If you sent them a personal greeting from your own email account, they probably wouldn't mind. That's what the organization's “members list” is for. But add them to a list and send them a huge email newsletter, and you'll be reported for spamming. Don't do it.
  - They go to events, and swap business cards. The business cards just get thrown into a “prospects” pile. Years later, they get an intern to finally type all the contact info from those cards into a database. Then, one day out of the blue, they send a big, fat email newsletter to everyone. As if they actually want to hear from you! Trust us, they don't.
  - Fish bowl of business cards. Similar to the example above, but they hold an event, and collect business cards in a fishbowl to win a “door prize.” The people who dropped their business cards into the bowl wanted the coffee t-shirt you're giving away (plus all the fame & glory). They didn't opt-in to your list.
9. Using a WYSIWYG to “code” your HTML. WYSIWYGs are notorious for generating absolutely horrible HTML. They insert so much junk code, it's unbelievable. Even the ones that generate “clean” code don't know how to “rig” things to work in email applications (like sticking your embedded CSS *below* the <BODY> tag). To code HTML email properly, you need to learn a little HTML. It's really not that hard. Plus, you can download free HTML email templates at our website: <http://www.mailchimp.com/resources/templates/>
  10. Forgetting to test. Thoroughly. When you send HTML email, you've really got to test it in as many email applications you can. Then you have to test on different operating systems. Then different ISPs (we'll explain shortly). If you keep things simple, and build a rock solid, thoroughly tested **template** for each newsletter, you won't have to test so much. But you should *always* send at least a few campaigns to yourself before sending it out to your entire list.



## **The MonkeyWrench Newsletter**

**Subscribe to MailChimp's MonkeyWrench Email Newsletter. We'll keep you up-to-date on more "advanced" email coding tricks, obscure hacks, power tips from MailChimp programmers, and oh yeah---free prizes.**

**<http://www.mailchimp.com/monkeywrench/>**

# Designing around spam filters

A guide on email newsletter design wouldn't be complete without a section on spam filters. You have to design your emails a certain way, if you want your messages to make it past them. But you can't go overboard, either. Spam filters can smell fear. If you try too hard, they know it, and they attack. So just be cool, act normal, and don't make any obvious mistakes.

## How anti-spam systems work

Before getting into the nitty-gritty design tips, you need to know how all the various anti-spam mechanisms work out there. Once you get an overall understanding of how they work, designing around them will be much easier.

### Bayesian Filtering

This is one of the most important ones to learn about, since it's installed in so many email applications these days (like Outlook 2003, Mozilla Thunderbird, and Apple Mail). Bayesian filters work by watching users classify email as "junk" (such as when they click a "this is spam" button).

It reads the junk mail, compares it to other emails you called "junk," and looks for common traits in the subject line, the content, the hyperlinks, the sender, etc. Over time, Bayesian filters learn to scan for those "traits" in every email



message you receive. Every time they find something that looks "spammy," they assign a "score" to it. For instance, using "Click here!" might get you 0.7 points.

Using **bright red fonts** might get you 2 points. Including the word, "mortgage" might get you 1 point, but using the word, "Viagra" will get you 5 points. Once an

email exceeds some threshold (set by the user), the email is classified as spam, and thrown into the junk folder. It's amazing how many different things Bayesian filters look for. To read some, [go to the MailChimp Blog and search for "Funny Spam Assassin Criteria"](#)

### Blackhole Lists

A while back, some server admins got really, really, angry at all the junk mail they received. So they started to track the IP addresses of the servers that sent them spam, and put them on "blacklists." Anytime they received email from a server on their blacklist, it was deleted immediately. It worked pretty well. So they started to share their lists with other server admins. And other admins started to add on to those blacklists. Pretty soon, the blacklists got very, very big. Large ISPs started to sync up with them. If you send email that's very "spammy" you could end up on one of these blacklists (whether the recipients are opt-in or not). Once you make it on to a blacklist, good luck getting off. Geeks are notoriously difficult to negotiate with (we know, because we're geeks). This is why you need to make sure you only

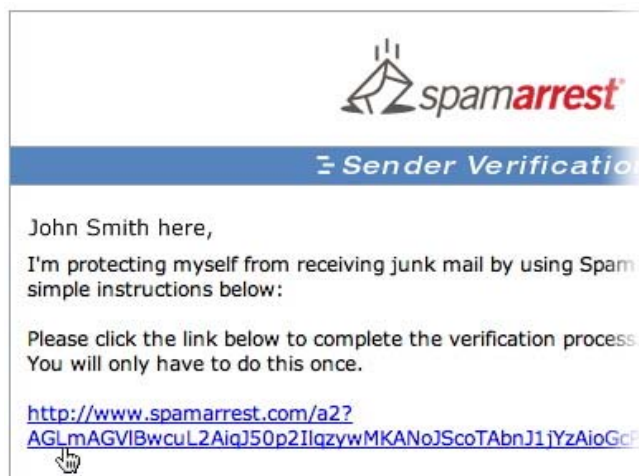
send to recipients who gave you explicit permission to email them, and you need to have *proof* that each one of them opted in to your list (such as through a double opt-in system). Want to know the easiest way to get blacklisted? Let your company's sales team blast out an email newsletter to a list of "prospects" that they collected from conferences, and off of websites. You'll get blacklisted, guaranteed.

## Email Firewalls

Managing email servers (and incoming spam, viruses, and phishing attacks) can be a lot of work. So large corporations usually install "email firewalls" or hire 3<sup>rd</sup> party vendors (Google the terms, "[Barracuda Firewall](#)" or "[Postini](#)" for some examples) to handle their incoming email. They often use a combination of Bayesian-style/adaptive filters, community reporting, blackhole lists, and a little bit of proprietary "magic pixie dust" to keep spam out of the company. Most of the time, when your email's not getting through to a larger company, it's their firewall. You can think of these firewalls as kind of Xenophobic and paranoid. They're all twitchy, and tend to ask questions like, "Okay, is this sender new to me? Why is he sending copies of the same, exact email, to a bunch of people in our company? Spammers do that kinda stuff. Hmm, how long has their server been around? Can I really trust this sender?" Spam firewalls are usually only a problem when you first start sending campaigns to a big client or something. You'll experience some deliverability issues in the beginning, because you're "new." They'll eventually "learn" to let you through. To expedite things, you may have to ask the IT people in charge of the firewall to "white list" your IP Addresses (or the IP address of your email service provider).

## Challenge/Response Filters.

These are more common among "at-home" recipients (because they're too intrusive to use at work). When you send email to someone with a challenge/response filter, here's what happens. If you are not already in that person's "buddy list" or "address book," then you're considered a stranger to him. And if you're a stranger, you *could* be a spammer. So their challenge/response filter sends you an automatic reply with a question that you have to answer, or some link you have to click (this is to prove you're human). The screenshot you see to the right is an example of a typical challenge/response reply, from Spamarrest, a very popular vendor.



The thing to remember here is that you have to be “white listed” if you want your emails to get through. So when people fill out your opt-in forms on your website, ask them to “please add our email address to your address book...” Use your opt-in process as a way to “[set expectations](#)” and get “whitelisted” up front. Whenever you send a newsletter, make sure the “reply-to” address that you use is valid, and that a human checks it after each campaign. You can expect to receive a couple auto-replies like this after every send.

## Tactics for Avoiding Spam Filters

So now you know how a lot of anti-spam systems work. You really have to “think like a spam filter” when you design your emails. Just like you “think like a search engine” when you design web pages. Spam filters read your emails, and look for similarities with known spam. You don’t want to do things that’ll get you accidentally thrown into the junk folder. And the best way to learn what **not** to do is:

1. Open up your junk email folder.
2. Look through all that spam you got. *Really* read it. Look at how they design things, and look at how they write their copy. Notice all the similarities.
3. Then, don’t do what they do.

### Okay, okay, to be more specific, here’s some stuff to avoid:

- Using **bright red fonts**, or going crazy with colors, styles, and formatting. Check out the article, “[Let’s dissect some spam!](#)” at our MonkeyBrains blog to see what I mean.
- Yelling with lots of exclamation points!!!!!!
- YELLING WITH ALL CAPS!!!!
- Using spammy words, like “mortgage, viagra, etc.”
- Saying, “free” or “click here!” or “click here now!” or “act now!” or “limited time!” (be especially careful of your unsubscribe link, where you might say something like, “click here to unsubscribe.”)
- If you use a WYSIWYG to code your HTML email, make sure the HTML code is “clean.” WYSIWYGs often throw in lots of useless code that’s invisible to the user, but spam filters look at it, and think you’re a sloppy spammer who doesn’t know HTML.
- Bad HTML in general can get you spam filtered. Missing table tags, content below the closing `</HTML>` tag, or empty `<TITLE></TITLE>` tags will get your message thrown into the junk folder. Don’t get sloppy with that code! Note: the empty `<TITLE>` information is very common if you use WYSIWYGs. They throw that in by default, and expect that you go back and fill it in. Don’t forget to go back and give the document a title!
- Don’t use “dummy text” in your message, even if you’re sending a test campaign. Spam filters get suspicious of all that “lorem ipsum” stuff. When sending test campaigns, use content that’s as real

as possible. Don't type in one paragraph, then copy-paste it ten times. Spam filters can see all the duplicate content, and they think it's spam.

- Don't use the word "test" in your subject line.
- Don't get too creative with your spelling, because the spammers do that all the time (how many m0rtg4g3 refl nanc3 spam messages are in your inbox right now?).
- Don't make an HTML email that's just a bunch of pretty graphics. You need some text in your message, too. Otherwise, the spam filters will have nothing to read, and will think your message is junk.
- Don't send the HTML email by itself. Always include that plain-text alternative message.

### **Learn More:**

Read our article, "Spam Filters and Salmon" to learn more about how spam filters think:

[http://www.mailchimp.com/resources/how\\_spam\\_filters\\_think.phtml](http://www.mailchimp.com/resources/how_spam_filters_think.phtml)

# Testing & Troubleshooting Your Email Designs

After you've finally figured out how to design and code your first HTML email template (and, of course, your plain-text alternative version as well), don't just plop in your content and start sending campaigns to your list. You need to test your template first. Make sure it'll work in all the different email applications out there. Once you've found all the little bugs, and your template is "rock solid," you can start sending.

## Web Designers: emails aren't the same as web pages!

If you're a web designer, you're probably used to testing web pages in a few different browsers, like Internet Explorer, Mozilla Firefox, and Mac's Safari. And you're probably familiar with a few annoying inconsistencies between all the browsers, and you have a couple hacks to make things look right.

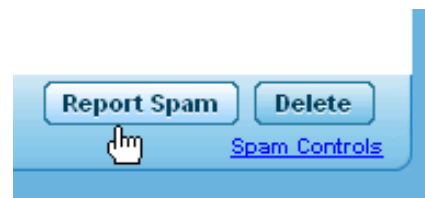
Multiply all that by ten, for email design. There are tons of email applications out there that you need to test on, and they all render HTML email in their own annoying ways. Do you need to test every single newsletter across every single application, every single time you send? Of course not. Just setup a good, simple, rock solid template and test it as much as possible. Then all you do is swap out content. Here are some tips for testing your email designs...

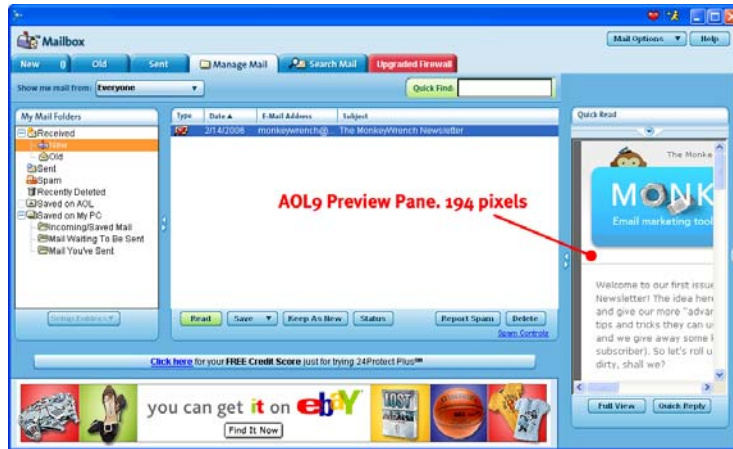
## Testing in Different Email Applications

It really helps to setup a test computer (or two) in your office. Here are some email applications you should install, and what to look for:

### AOL

Their free CDs are *everywhere*, and they offer some very cheap \$7 per month plans if you call and ask for them. They'll only give you 2 or 3 hours a month, but that's good enough for testing. AOL takes spam very seriously. So check to make sure your emails don't get filtered. Also, notice their "Report Spam" button. You've got to make sure your emails look relevant, or people will click that button (even if they opted in to your list!). At the time of this writing, AOL 9 has an extremely tiny "preview" pane. It's about 194 pixels wide, so make sure that your email shows up okay under those circumstances. Full view mode seems to open up at around 530 pixels, by default.





AOL's "preview pane" is *tiny*! Does your email peek out enough?

### Apple Mail

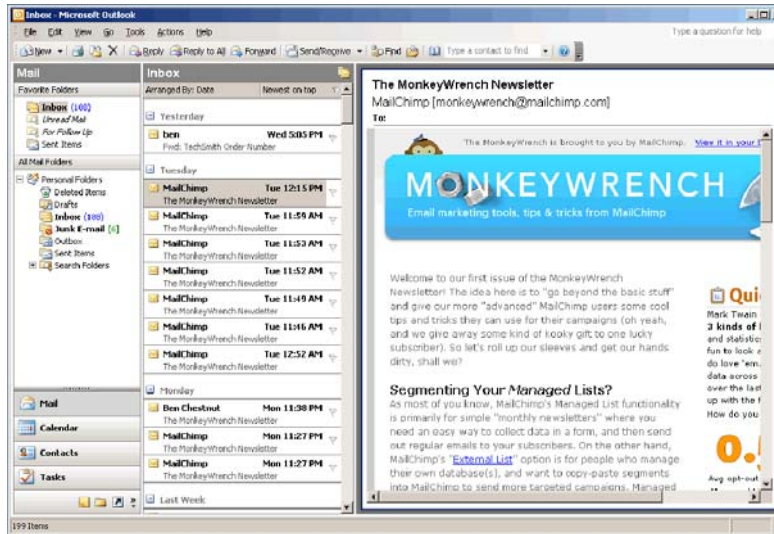
Apple's free email application. You get a nice big preview pane here, and they don't do much image blocking at all (yet). But it does come with its own spam filtering system that "learns" so you should test your emails for any deliverability issues.

### Apple's Entourage

Kind of like Microsoft Outlook, but for the Mac (it comes with Mac Office). Their HTML email support is fine, but we noticed in earlier versions of Entourage, if you send an email in HTML format, but it consists of nothing but text and hyperlinks (no graphics or tables), then the links actually display their URL next to them. Kind of odd. But if the email actually contains a graphic or table, then the links work as coded. The issue seems to be fixed in the latest version of Entourage 2004.

### Microsoft Outlook 2003

Outlook 2003 has better spam protection, so watch whether or not your emails get filtered. Also, this one's got 3 vertical panes, so your emails have less room in the preview area (see below).



Three vertical panes in Outlook 2003 leave little room: Make sure you fit!

### Microsoft Outlook 2000

There are lots of business users who haven't upgraded to Outlook 2003 yet. This version of Outlook predates all the vicious worms, spam, and viruses that we're used to now, so more of your emails with flash and videos and crazy stuff will actually work here. Don't let it build your confidence. That fancy stuff won't work in other programs.

### Microsoft Outlook Express

This is the free version of Microsoft Outlook, installed on all Windows machines. Used mostly by "at home" recipients. Not really many HTML email issues, besides the usual "block images by default" feature that's on just about every email application these days.

### Lotus Notes 6.5.3 and 6.5.4

If lots of your recipients work at very large companies, you'll need to check your emails in Lotus Notes. This is probably the most frustrating email application out there. It does the strangest things. You'll need to keep your layouts very simple. Even still, Lotus will find a way to mess things up. I'm still baffled about how they managed to make a cell in my table blink red. There are even big differences between Lotus Notes 6.5.3, vs. 6.5.4. Go to our [MonkeyBrains blog](#), and search for "Lotus Notes" for all the different things that can go wrong in Lotus.

### Eudora 6.2

Eudora has good, predictable HTML email support. We don't see too many issues to talk about here.

## **Mozilla Thunderbird**

A free email application that's gaining in popularity. Does some odd things with alt-text, and image blocking (search for "mozilla thunderbird" at the [MailChimp Blog](#)), but for the most part supports HTML email great. Thunderbird 1.5 has a feature that tries to detect "scams," where they look for hyperlinks that claim to point to a website, but the code actually links you somewhere else. If it detects a potential "scam," you get a warning across the top of the message, saying "Thunderbird thinks this message is a scam." You can click the "Not a scam" button if it was a false alarm. It's a nice idea, but it seems to give lots of false positives, especially when you use click tracking in your emails (all your links will point to a redirect script, in order to track clicks). In the Thunderbird support forums, they say that the feature is still kind of new, and they're working on refining it. We don't think it's much of a problem, as long as your emails look nice and professional.

## **Testing Browser-based Email Services**

Setup accounts with all the browser-based email services. They're usually free, so it's easy to setup lots of test accounts. You *especially* want to look at how browser based email services alter your HTML and CSS.

Also, since these services are used by bazillions of people, their anti-spam filters are set really high. We listed them below in order of most-to-least popularity (according to [a recent survey](#) across all MailChimp managed lists). When you setup your test accounts with these services, leave their "junk mail filter" settings to "default." It really pays to test on these accounts before sending your campaign...

### **Yahoo!Mail**

Pretty aggressive anti-spam filters (understandably). Other than that, they do the typical browser-based email stuff, like stripping your BODY and HEAD tags.

### **Yahoo!Mail Beta**

A much richer browser interface, with a preview pane. So far, support for HTML email looks great.

### **Hotmail**

Similar issues as Yahoo!Mail, where BODY and HEAD tags are stripped, so background colors and embedded CSS are lost. At one time, Hotmail was "expiring" hyperlinks in emails that were open for 5 minutes or longer.

### **Gmail**

Extremely aggressive spam filters. And very, very little CSS support. It strips embedded CSS, (even if you keep the CSS out of the <HEAD> tag). It also strips every "class" code that it finds (such as in <span class="header">). To get CSS to work in Gmail, you'll need to use inline styles. Otherwise, your fonts will

default to Arial, colored black, and about 13 pixels in size. If your fonts don't adhere to your CSS rules, check to see if they end up blowing out your tables cells, or wrapping in weird ways.

### **Check Different ISPs, too**

If possible, check your emails when they're sent through **different ISPs**. Different email servers will alter your messages *before* they even get to the recipient's email application. For instance, some ISPs use email servers that will strip any content below a line in your email that starts with a period (I know, weird, huh?). We've been surprised at how differently email looks when checked in Outlook 2003, but received through:

- Comcast
- Bellsouth
- Earthlink

### **Send tests to friends and co-workers**

If you can't setup a few test computers, keep your designs really simple, and send each campaign to a few friends or office-mates. Just ask them to let you know if anything looks "off" or broken to them.

#### **Free Download:**

We've posted an article in our blog about the most popular email services to test on, and included a handy little checklist you can download free at:

[http://mailchimp.blogs.com/blog/2005/11/aolcom\\_the\\_most.html](http://mailchimp.blogs.com/blog/2005/11/aolcom_the_most.html)

## 2. Recipients

- Send test message(s) to:
- Send to list on MailChimp: [Setup your MailChimp](#)

# Test With MailChimp!

To test your HTML emails, you can set up a MailChimp free trial account. It's quick and easy, and comes with 25 free email credits for testing.

If you like MailChimp, you simply purchase more email credits (only what you need). No contracts, no monthly fees, no spam, no hassles.

<http://www.mailchimp.com/tryit.phtml>

# Email Marketing Basics & Best Practices

If you're new to email marketing, we should go over some basic principles before you start sending campaigns. We'll go over best practices, legal issues, and how to measure your overall performance...

## E-marketing Etiquette (How Not To Be A Jerk)

There are 3 kinds of email marketers in this world. **The Good, The Bad, and The Clueless.** The "Bad" are the evil-spamming-scumbags. These jerks don't care what kind of damage they're doing. They send millions and millions of pieces of spam every day. They're evil, and they know it. But because they have to remain anonymous, and because they'd never spend even a fraction of a penny per email, and because they don't really care about best practices and features like tracking and unsubscribes, they never use email providers like us. We don't really have to worry about evil spammers.

The "Clueless" are much more common. These are the ones that keep us up at night. They are generally good people, who don't mean any harm. They're "just trying to get the word out." They're often very ethical business people, who have been in marketing and sales for years. But since they're new to email marketing, they make all kinds of mistakes, and they typically have absolutely no idea what kind of mess they're making.

Don't be a clueless email marketer. "Good" email marketers know their etiquette...

1. Email isn't like TV, or print advertising, or direct marketing, or radio. In those cases, the *advertiser* spends all the money getting the message in front of the consumer. But in email marketing, it's the *consumer* who pays the price. They pay through monthly ISP fees. They pay for bandwidth. And when ISPs spend the majority of their time managing incoming spam to their servers, you know that *someone* has to pay the price. Never forget **the number one rule of email marketing**: Since your recipients must pay for every email they receive (through monthly ISP bills, bandwidth costs, etc), you *must* obtain their *permission* before you email them. Permission is the foundation of "good" email marketing. Email isn't free. Say that to yourself 10 times.
2. Always provide a way to unsubscribe. And remove people **immediately** after they request it. Take them off your list as fast as you put them on.
3. Do not ever, ever use **anonymous email addresses** as the reply-to, or unsubscribe address in your campaigns. Use *your company domain*, and be sure emails to that address go to a live person. Things like, "unsubscribe\_acme\_newsletter@hotmail.com" are a big no-no. If you don't even have your own company domain, you're not ready to send email marketing. Sorry.
4. Don't send too often. You're going to make your customers sick of you (remember, they're paying for each email you send). Only send something when you've got something useful to say.

5. Don't send bulky email (Flash, Sounds, JavaScript, Videos, etc). Use HTML email wisely---not "just because you can".
6. Don't use rented email lists, and for the love of God, don't buy some CD full of "30,000,000 email addresses" that you bought from some shady website (refer to Rule #1). They are full of bad addresses, and people who didn't truly opt-in to receive anything from *you*. Good e-marketers collect their own email addresses, on their own sites. Building your own list may be harder and the list may take time to grow, but at least you're 100% positive everyone on the list actually *wants* to hear from you. Your response rate will be considerably higher, and since the list is smaller, the cost of delivery will be lower. Trust us. Just build your own list! If you're thinking of marketing to a partner's list, there's nothing wrong with that---but your partner must do the sending---from *their* system---on your behalf. It's not very ethical for them to just hand you an email list (it probably violates their privacy policy, if they have one). You give your content to the partner---they don't give you their list. They'd send an email to their list with, "Here's a message from one of our valued partners...please join their list at..."
7. Use double-opt-in (some call it "confirmed" opt-in) for your subscriber lists. When someone signs up for your list, send a confirmation email that requires the recipient to "click this link to verify your address". If they don't click, they're not added to your list. If they do click, you add them to your list, and you *also* save the date and time of confirmation, plus their IP address for your records. This keeps your list cleaner, and it also ensures that the email address really exists, has no typos, and wasn't maliciously or erroneously submitted. It's not a hassle to the end recipient at all---in fact, it makes your operation look much more ethical and secure. We've seen cases where "grandma" thinks someone else would absolutely *love* those daily email recipes she gets, so she signs them up. Or sweet, innocent little Billy subscribes Dad to some "fart joke of the day" list. Worse yet, some jerk from a competing company submits a bunch of people's email addresses into your signup form, trying to get you in trouble for spamming. Note: MailChimp's Managed List feature uses this "double opt-in" method by default.
8. Know the laws. Be sure to read and comply with the CAN-SPAM laws. Generally speaking, if your email is a marketing or sales type of message, you need to follow the rules, or you can be fined for each email sent.

## **The CAN-SPAM Act of 2003**

The United States federal CAN-SPAM Act became law on January 1, 2004. According to their website, the FTC says that if you violate the law, you could be fined \$11,000 for *each* offense (multiply \$11,000 times the number of people on your recipient list). ISPs around the country have already successfully sued spammers for millions and millions of dollars under this new law.

We can't give you much legal advice; MailChimp's an ape, not a lawyer (okay, insert lawyer joke here). But if you send commercial email, you should [read through the CAN-SPAM Act of 2003](#), and understand the rules. If you have a lawyer, consult with her. Here are a couple of points we'd like to highlight:

- If you're sending "**commercial**" email (where you're selling or promoting stuff), here are just a few rules you should know about:
  - Never use deceptive headers, from-names, reply-tos, or subject lines.
  - Always provide an unsubscribe link
  - Remove recipients from your list within 10 business days
  - That unsubscribe link must work for at least 30 days after sending your email
  - You must include your physical mailing address in the email (PO Boxes are not sufficient)
  - To learn more, go to: <http://www.ftc.gov/bcp/online/pubs/buspubs/canspam.htm>

## Double Opt-in

As you can probably tell by now, we *highly* recommend the double opt-in method when managing your email lists. In fact, it's the *only* way MailChimp's built-in list management system will work. Here's a quick overview of the process:

1. A customer signs up for your email newsletter through a form at your website.
2. He receives an email with a confirmation link.
3. If he doesn't click the link, he's not added to the list.
4. If they click the link, they're added to your list, and you store the IP address, date, and time of registration. Now you've got proof of opt-in, should you ever need it in the future (like if you receive a false or malicious abuse report).

Double opt-in is fast replacing the (single) opt-in method, where someone submits a form, and bam--- they're added to a list. There are too many chances for someone to get signed up to a list without his permission, either erroneously or maliciously. And there's no need to even discuss the old opt-out method anymore.

## Pros & Cons

There was once a big concern about double opt-in, where marketers complained, "but nobody's going to click the confirmation link! I'll lose all my prospects!" This isn't a huge issue anymore. People are used to the double opt-in method these days. Plus, it just looks more professional.

Sure, if you compare a double opt-in list to a *single* opt-in list side-by-side, the single opt-in list will be larger. It's just too easy to subscribe (anyone) to a single opt-in list. But it will also have more "accidental"

members, a higher bounce and unsubscribe rate, and lower response rate. Double opt-in ensures that every single recipient truly wants to hear from you.

## Segmenting & Experimenting

At MailChimp, we say, “There’s *email newsletters*, and then there’s *email marketing*.” On one hand, you can collect a list of subscribers and send a monthly newsletter to them. There’s nothing wrong with that. In fact, that’s why we built MailChimp’s double opt-in list management system. It works great for that kind of stuff. Real simple and convenient.

### Segment Your Lists

But if you want to get serious about email marketing, you should also learn to *segment* your customer database and send focused, relevant communications to them. And MailChimp’s “Externally Managed List” option is built specifically for this task. You can use your current customer database, CRM, or e-commerce system to manage your customers and segment them any way you want. Then, export your list in tab-delimited format, and copy-paste into MailChimp. Yes, it’s that easy. No database programming, no synchronization setup, no server access codes, and no begging your IT group for help. Just copy-paste. We’ll even handle delivery, reporting, bounce-backs, and unsubscribes.

You should try the “external list” feature to split out your customer segments, by:

- Buying habits, purchase history
- Location
- Account type
- ISP (at-home vs. at-work)
- Gender, age, etc.

For instance, one thing we’ve done at MailChimp is use our customer database to send a special email offer to “customers who are classified as non-profit, who have opted-in to receive emails from us, who are in companies from 1-5 employees, who have logged in within the past 6 months.” You can’t really get that kind of segmenting done through a simple little opt-in registration form (such as for monthly newsletters). You need to use your customer database, and that’s why the “external list” option is so powerful in MailChimp.

[Here’s a knowledge base article on copy-pasting externally managed lists into MailChimp.](#)

**2. Recipients**

Copy 'n Paste your recipient list below. Paste a tab-delimited file if you want to

bananas@mailchimp.com	John	Smith	598	
fred@mailchimp.com	Fred	Von Chimp IV	1,438	4,211
alpha-mail@mailchimp.com	Alpha	Mail	4,211	
chimpyi@mailchimp.com	Chimpy	Chang	987	
ben@mailchimp.com	Ben	Chestnut	342	
mark@mailchimp.com	Mark	Armstrong	7,654	

Personalize the "To:" Field?

**3. Plain Text Email**

Paste your plain-text message, which we'll automatically display for people who

Hello \*|MERGE1|\* \*|MERGE2|\*,

We noticed that your email address is: \*|MERGE0|\* and you have \*|MERGE3|\* bananas left in your account!

Microsoft Excel - my-customers.xls

	A	B	C	D
1	EMAIL	FNAME	LNAME	BANANAS
2	bananas@mailchimp.com	John	Smith	598
3	fred@mailchimp.com	Fred	Von Chimp IV	1,438
4	alpha-mail@mailchimp.com	Alpha	Mail	4,211
5	chimpyi@mailchimp.com	Chimpy	Chang	987
6	ben@mailchimp.com	Ben	Chestnut	342
7	mark@mailchimp.com	Mark	Armstrong	7,654
8	dan@mailchimp.com	Dan	Kurzus	543
9	bobo@mailchimp.com	BoBo	Jones	132
10	chester@mailchimp.com	Chester	Cheetah	0

Copy-paste your tab-delimited data into MailChimp, then personalize your email with \*|MERGE|\* Tags that correspond to each column from your database (starting at "zero" for "email address")

We think the easiest way to use MailChimp with your customer database is to segment your list however you need, then export it to "TAB DELIMITED FORMAT." Then, open that file in Microsoft Excel. You should see all your columns and rows, nice and neat. Always make sure that "email address" is the first column (move it all the way over to the 1<sup>st</sup> column if it's not already). Now, copy-paste from Excel into MailChimp.

Want to personalize each message? Use MailChimp's \*|MERGE\_TAGS|\* to merge data from each column in the Excel file into your message. It's really simple. [Read our knowledge base article for details.](#)

## Experiment Like Crazy

MailChimp's "Externally Managed List" option is an extremely powerful tool for experimenting with your email marketing. You don't have to be a rocket scientist or mathematician, and you don't have to program anything. You can simply use your own customer database to create multiple lists, and then copy-paste into MailChimp to send separate campaigns to them. We'll track clicks and opens, and generate reports per campaign, so you can measure the performance of each. Take your customer list, and send half in the morning, and the other half in the afternoon. Or split them into different days. Or try different subject lines. The sky's the limit, since you just copy-paste from your database.

## Personalizing Emails with MailChimp \*|MERGE\_TAGS|\*

You can also personalize all your externally managed lists, just like you can with MailChimp Managed lists. Simply copy-paste from a Microsoft Excel file, then use \*|MERGE\_TAGS|\* that correspond to each column in your spreadsheet (see below).

	*IMERGE0*	*IMERGE1*	*IMERGE2*	*IMERGE3*
my_recipients.xls				
	A	B	C	D
1	<b>Email Address</b>	<b>First Name</b>	<b>Last Name</b>	<b>Phone</b>
2	customerservice@mailchimp.com	Freddie	Chimp	678-999-0141
3	abuse@mailchimp.com	John	Monkey	555-123-4567
4	tech@mailchimp.com	Bubba	Ape	555-098-6543
5	accountservices@mailchimp.com	Oran	Gutar	678-999-0414
6	ben@mailchimp.com	Ben	Chestnut	678-999-0141

\*IMERGE1\*

\*IMERGE2\*

\*IMERGE3\*



Quarterly  
April 10, 2005

Products
Resources
Support
Newsroom
Home


Dear **\*IMERGE1\***,

We would like to thank you for your cooperation in our recent annual projects; your interest in the ACME Company is much appreciated.

**To ensure that you continue to receive correspondence, please take a moment to review your account information:**

Your last name is **\*IMERGE2\***, and your day phone is **\*IMERGE3\***

If this information is not correct, [update it here.](#)



We are proud to announce the J.D. Power and associates award for Highest Overall Satisfaction Among Industry Providers in the Southeast.

### Learn More:

See how MailChimp's "Merge Tags" work in our online knowledge base.

Go to: <http://www.mailchimp.com/help/> and search for "Merge Tags"

# Measuring Performance

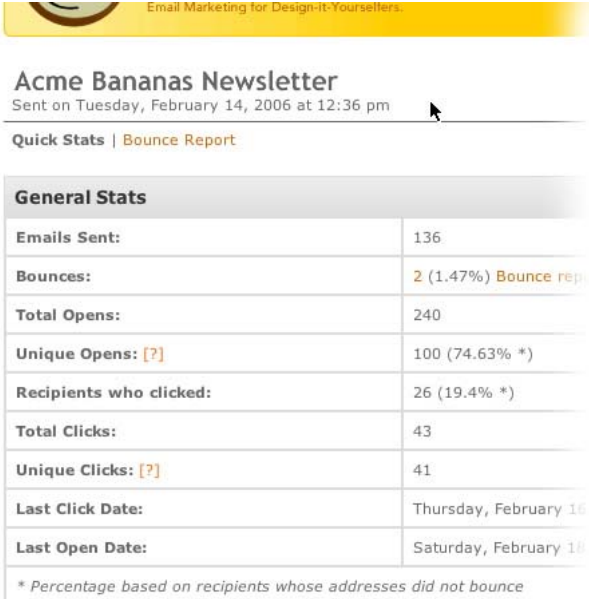
After you become a pro at HTML email design and coding, and you're sending campaigns on a regular basis, how do you measure performance? How do you know if your email marketing efforts are working or not? How can you experiment, and improve your performance? Here's a basic rundown of what you should be measuring, and how you can test your campaigns.

## Open Rates

How many people opened your email? An open rate of about 20% to 30% is considered "average." Watch your open rate over time. Your first email might get a great open rate (like 50% to 70% or more), but over time will level off some.

**Try experimenting with different things, like:**

- Your subject line: Make it longer (or shorter). Merge their first name into the subject. Use your company's name.
- Date of send. Do weekdays work best? Or is the email geared towards an "at-home" audience, who check emails on weekends? Or, do they typically only order while at work, on fast connections?
- Time of send. Mornings, just before the first wave of spam? Just before lunch? Just after lunch? Just before quitting time? If you were a restaurant owner, would you send a campaign for a Friday event on Monday, or Thursday?



The screenshot shows an email marketing dashboard for 'Acme Bananas Newsletter'. The dashboard includes a header with the newsletter name and send date, followed by a 'Quick Stats | Bounce Report' link. Below this is a 'General Stats' table with the following data:

General Stats	
Emails Sent:	136
Bounces:	2 (1.47%) <a href="#">Bounce report</a>
Total Opens:	240
Unique Opens: [?]	100 (74.63% *)
Recipients who clicked:	26 (19.4% *)
Total Clicks:	43
Unique Clicks: [?]	41
Last Click Date:	Thursday, February 16, 2006
Last Open Date:	Saturday, February 18, 2006

\* Percentage based on recipients whose addresses did not bounce

## Click Rates

How many people clicked links in your email? Which links did they click the most? Did they click on product links, or research links? Did you see a rise in purchases? How long after you sent the campaign do links keep getting clicked?

**Try experimenting with:**

- The number of links in your message. One big, giant link, or lots and lots of different links?

- Product photos for each link
- The wording of each link.
- Segmenting your list, and sending more focused emails (and links that are more relevant)

Example of a typical MailChimp report: You can see which links people are clicking...

Clicks by URL		
URL	Total Clicks	Uniques
<a href="http://www.acme-ind.com/index.html">http://www.acme-ind.com/index.html</a>	67	55
<a href="http://www.acme-ind.com/landing/preview.html">http://www.acme-ind.com/landing/preview.html</a>	28	25
<a href="http://www.acme-ind.com/updates/refer.phtml">http://www.acme-ind.com/updates/refer.phtml</a>	19	17
<a href="http://www.acme-ind.com/updates/unsubscribe.html">http://www.acme-ind.com/updates/unsubscribe.html</a>	15	13
<a href="http://www.acme-ind.com/privacy.html">http://www.acme-ind.com/privacy.html</a>	2	1

© Copyright 2005 The Rock at Science Group, LLC. All Rights Reserved. | [Privacy](#) | [Terms](#) | [Product Info](#) | [Contact Us](#)

## Unsubscribe Rate

What's your unsubscribe rate after each campaign? Less than one percent is average for lists that are contacted regularly, and well-maintained. If you send very infrequently, or if it's your very first send, your unsubscribe rate may be much higher. Check your unsub rate after each campaign. If you see it spike after a particular campaign, see if it had anything to do with your content. Maybe you're sending too frequently. Maybe not frequently enough. Keeping a good, fresh, quality list will result in a very low unsub rate.

### List Summary - Acme Bananas

List Stats:	Since Last Send:
Current Members: 147 ( <a href="#">view all</a> )	Signups: 0
Unsubscribes: 8 ( <a href="#">view all</a> )	Unsubscribes: 0
Cleaned: 0	Cleaned: 0

## Bouncebacks

Watch your bounceback rate after each campaign. A good list management system will break down your bouncebacks into "hard" vs. "soft" bounces, and clean your list for you. Soft bounces are emails that exist, but for some reason, they couldn't be delivered. For instance, their server might have been too busy at the time of your delivery. Hard bounces are undeliverable--perhaps the email account doesn't exist anymore, or there was a typo in the address. Your email marketing service should let you read the header information from each bounced message, so you can see exactly why they bounced.

Bounce Stats	
Total Bounces:	2 (1.47% of 136 sent)
Syntax Errors:	0
Soft Bounces:	2 (1.47% of 136 sent) <a href="#">Email Headers</a>
Hard Bounces:	0

### Try experimenting with:

- Re-sending your campaign (a little later) to the people who “soft-bounced.”
- Look through your hard bounces for any obvious typos. For instance, “hotmial.com” could probably be corrected to “hotmail.com”

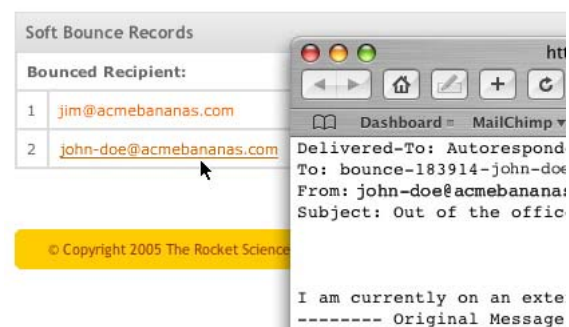
### MailChimp Helps Manage Your Bounces

Not all bounce-backs are bad. Soft bounces are mostly “temporary” and you can probably keep them on your list and try again. If MailChimp sends your campaign and an email address “soft bounces” back to our system, we’ll watch that email address over the next 3 campaigns. If it keeps soft-bouncing, we’ll automatically clean it from your list. But if an email address “hard bounces” (Like when it’s an undeliverable address), we *immediately* remove it from your list. You shouldn’t send campaigns to a bad email address over and over---ISPs block you for doing stuff like that.

### Viewing SMTP replies for bounced email

When a recipient’s email address bounces back to MailChimp, the server that bounced it usually sticks a little “message” in the email’s header, which is called the “SMTP Reply.” These SMTP replies tell us why the email was bounced (such as temporarily busy, overquota, on vacation, bad email address, account no longer exists, etc). This helps MailChimp classify the bounce as “soft” or “hard.” But sometimes, email servers don’t use standard codes and messages in their SMTP replies. It’s all up to how the server admin set things up. Worse yet, some anti-spam firewalls outright **lie**, by inserting “this email address no longer exists” into their SMTP replies. This is meant to trick list management systems into removing an email address from their list. Classifying bounce-backs as “soft” or “hard” isn’t a perfect system. That’s why MailChimp gives you the “View Full Headers” option in your bounce reports, so you can read the headers of each and every bounce-back, and determine exactly why each email bounced.

To learn the **exact** reason each email bounced, click to read their full

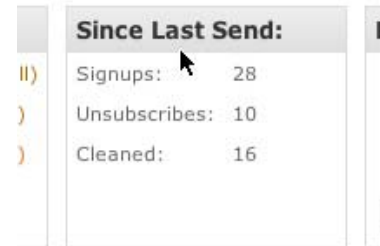


## Traffic to your website

Check your website traffic logs after each email campaign. Does traffic pick up? Do orders increase? Do the cash registers go, “cha-ching?” Check and see if you get a spike in traffic immediately, or if it’s more like a “gradual wave.” How long does the new traffic last (and how long should you keep the graphics and pages that your email points to hosted live)?

## Signups since last campaign

After each campaign, do you get lots of new subscribers? That could mean your wonderful recipients are forwarding your emails on to friends. How nice! Your content is actually useful! Don’t see any list growth at all? Make that content more interesting!



A screenshot of an email marketing dashboard. A mouse cursor is hovering over the 'Signups' row. The table shows the following data:

Since Last Send:	
Signups:	28
Unsubscribes:	10
Cleaned:	16

## Try experimenting with:

- Place some text near the top of your email, that says, “feel free to forward this to a friend”
- Place a “view this email in your browser” link at the top of your email. It should point to an archived version of the campaign on your website. Some people like to forward *that* link on to their friends.

## **MailChimp Campaign Reports**

**MailChimp can generate reports for each and every campaign that you send. We track open rates, URLs that were clicked, bouncebacks, unsubscribes, and more. You can also download every report into Microsoft Excel format, and generate your own graphs, or send them on to clients.**

<http://www.mailchimp.com/learnmore.phtml>

# Wrapping Up

So there you have it. Everything a beginner needs to know about designing, coding, sending, and managing email campaigns. We hope this guide helps get you on your way to sending many successful campaigns. Following is a “quick list” of the overall email newsletter creation process. Use it to send your very first email campaign (whether you use MailChimp or not). Finally, be sure to keep your skills honed by visiting all the useful email marketing websites and blogs that are out there now (we included a list of favorites in the Appendix).

## “My First Email Campaign” Checklist for Beginners

- ❑ Decide on your content and frequency. Will you be sending email newsletters every single month? Do you truly have the time to do that? What will the newsletter be about? Why would anybody want to subscribe to it? What will recipients gain from it?
- ❑ Get permission. Ask the customers in your database for permission to send them email newsletters, offers, and promotions. Don’t ever start “blasting” newsletters and email offers without permission.
- ❑ Make sure your privacy policy is in order. In particular, issues like “tracking personally identifiable information” or “marketing to children under 13.” If you’re tracking opens and clicks, and if it’s possible for children to subscribe to your emails, you need to cover those topics in your privacy policy.
- ❑ Know the laws. Become familiar with the CAN-SPAM rules. Consult with your attorney.
- ❑ Setup any email addresses you’re going to use for your campaigns. For our MailChimp company newsletter, we use newsletter@. For our MonkeyWrench newsletter, we use monkeywrench@. You might use sales@ or email@. Whatever. Just be sure it works, and be sure a human checks that account.
- ❑ Setup an abuse@ email account. No matter how clean you keep your list, you will always get complaints. It is *inevitable*. People forget they opted in. People get tired of hearing from you. People get mad from some lost shipment, or bad customer service. When that happens, they sometimes report your email as spam. If they subscribe to anti-spam services, or if they report it to their ISP, those representatives will want to contact you via email. And the 1<sup>st</sup> place they’ll try is [abuse@your-company-domain.com](mailto:abuse@your-company-domain.com). So make sure that abuse@ works. And while you’re at it, you should register that abuse@ address online (go to <http://www.abuse.net> ), because if/when they receive complaints about your emails, anti-spam organizations often use this to gauge how responsible your company is.
- ❑ Setup “feedback loops” with major ISPs. If you’re going to send lots of email marketing **from your own company server**, you should register your IP addresses with AOL, and MSN so that

they can send you a report if/when their users report your emails as spam. Google the term, “AOL feedback loop” for more information on how this works. MSN and other ISPs are said to be implementing similar reporting mechanisms. You should also go ahead and register for reports at <http://www.spamcop.net>. If users on their system report your emails as spam, they’ll send you automatic email alerts. **If you use MailChimp** to send your campaigns. Don’t worry---we’ve already got this covered. All our IP addresses for all our servers are registered, so we can handle any deliverability problems as they arise. If you use another service besides MailChimp, make sure they monitor this information too. Most reputable vendors do this.

- ❑ Setup test email accounts with Yahoo, Gmail, Hotmail, and AOL. Setup a test computer (or two) in the office, and download as many different email applications you can find, like Outlook, Mozilla Thunderbird, Lotus, Apple Mail, Entourage, and Eudora. You’ll want to tons of testing.
- ❑ Setup a double opt-in form on your website. Tip: create questions you might be able to use for future segmenting, like “industry” or “title” or “referral.” Be sure to customize your confirmation screens and confirmation emails appropriately. Use them to [set expectations](#) about the emails they’ll be receiving. Want a working example? [Subscribe to our MailChimp MonkeyWrench newsletter](#) just to see how we do it (then unsubscribe immediately, if you want). After you get it set up, you can start inviting your customers to opt-in. Post links all over your website, your invoices, and in your email signature.
- ❑ Design your HTML Email template. If you’re sending different kinds of communications (sales promotions, vs. monthly newsletters, vs. how-to’s and tips), you may want to setup multiple templates for each occasion. For some free examples, you can download MailChimp’s HTML Email templates at: <http://www.mailchimp.com/resources/templates/> If you’re not the “design-it-yourself” type, you might want to hire a professional to take care of this for you. Any web designer should be able to handle it. You can send them this guide, plus our templates, to get started.
- ❑ Insert real content. Don’t just use “lorem ipsum dummy text” when you test your campaigns. It’s not accurate, plus spam filters often throw away emails with garbled, nonsensical text in them.
- ❑ Build your plain-text alternative email. Plain-text email has its own peculiarities, so you’ll want to get a “template” for it refined from the beginning. Don’t let plain-text be an afterthought.
- ❑ Test your templates. And we mean **really test them** like you’ve never tested before. Send them to friends, family, and colleagues. Try the templates in as many different email applications you can. Open your own emails, and click them like crazy. Now go check your reports. Make sure all the tracking works like it should. Hit the “reply” button and see if you get replies to your account. You want to uncover any embarrassing mistakes before sending to your **real** customers.
- ❑ Setup a test list. If you’re using MailChimp’s Managed List function, import a “test list” first, and go nuts. When you’re ready to send to your real customers, just replace the list.

- ❑ On the subject of lists, make sure that **YOU** are subscribed to your own list. Even though systems like MailChimp allow you to send tests, it's best to be on the actual list, just like your customers.
- ❑ Prepare your website and landing pages for delivery. Before you send, you need to get everything in order. Did you create a copy/archive of the email for your website? Make sure that's all in place. Are all the landing or e-commerce pages that your email links to live, and working?
- ❑ Prepare the office. **What time** will you be sending your email? Here's a funny and embarrassing story. Once, we spent hours and hours prepping an email campaign for MailChimp. We sent it out at 9am EST, and got tons of calls and orders immediately after. It was great! So our entire office went to lunch together that day, to celebrate. But "lunchtime" on the East Coast is "just getting into work time" on the West Coast! We all came back to tons of angry voicemails and emails from our West Coast customers waiting for us. D'oh! Now, we've got these cool clocks across our wall, representing every major time zone around the globe. Learn from our mistake, and be sure you'll be near the phone/email after you send. Let your staff know about the campaign too, if they take orders via phone, email, or live chat.
- ❑ Deliver the campaign (finally!).
- ❑ After the campaign, it's always fun to log in to MailChimp, because we provide stats in real time. You can virtually watch the opens and clicks as they happen! It's always shocking to see how many people open and click in the first few minutes after you send. Then, it trails off. You'll still get opens and clicks weeks after! That's why MailChimp keeps your links and tracking alive for three months before archiving them. And in our reports, we provide "Date of most recent open/click."
- ❑ Review performance. After a few days, go back and analyze your reports. How many opens? How many clicks? Compare it to your web traffic logs, and most importantly, to sales. Did the email generate leads or sales? What was the ROI on that campaign?
- ❑ Plan the next campaign. Use your campaign reports to understand what people clicked, and what they didn't. That should serve as the foundation for your next campaign. Maybe you'll split up your list and segment next time around, or test different subject lines or delivery schedules. Get scientific!

# Appendix

# Useful Resources

Here are some links to handy resources on the Internet for you

## Email Marketing Sites

- <http://www.emailsherpa.com> - Great email marketing case studies and research. Be sure to sign up for their email newsletters.
- <http://www.clickz.com> - Lots and lots of online marketing and email marketing news.
- <http://www.marketingprofs.com> - Nice case studies and articles here. Premium membership is actually worth it, in my opinion. They also issue a Buyers Guide to email vendors.
- <http://www.btobonline.com> - Email and direct mail news. Check their “email marketer insight” section.
- <http://www.futurenowinc.com> - Look for their little alien mascot, and sign up for their Grokdotcom newsletter. These guys specialize in conversion, and they know their stuff.
- <http://www.marketingexperiments.com> - Great experiments and case studies here.

## Email & Marketing Related Blogs

- <http://sethgodin.typepad.com> - That familiar bald head, chock full of handy marketing ideas
- <http://www.tompeters.com> - As Tom would say, “WOW!”
- <http://www.email-marketing-reports.com> - Business reports, media articles and other relevant material for marketers. Great resource put together by Mark Brownlow.
- <http://emailmarketing.typepad.com> - Tamara Gielen’s blog on email marketing. She posts links to news all over the email marketing world.
- <http://mailchimp.blogs.com> - Gotta plug our own blog, “MonkeyBrains” has email design tips and tricks, hacks, troubleshooting tips, news, and more.

## Products & Tools

Text Editors for coding

- <http://www.bbedit.com> - BBEdit is a great text editor for the Mac. Use it to code HTML and plain-text emails. Or check out their TextWrangler product, a lite, free version of BBedit.
- <http://www.notetab.com> - NoteTab Pro is a great text editor for PC.
- <http://www.macromedia.com/dreamweaver/> - Used by web design pros to build websites. You can use it to build HTML emails, too. Use the coding interface for HTML, but be careful of the

WYSIWYG tool. It adds CSS and other things to your file (which you'll need to take out for HTML email)

FTP clients, for sending files to your server

- <http://fetchsoftworks.com> - FTP software for the Mac. (\$25)
- <http://www.cuteftp.com> - Great FTP software for PC (\$90, and worth every penny)

## Graphic design programs

- <http://www.macromedia.com/software/fireworks/> - Great for designing anything for the Web. I've got an old installation of Fireworks 4 that I still use religiously.
- <http://www.adobe.com/products/photoshop/> - A must-have for serious graphic designers. A bit overkill for HTML Email design, but you can't list "graphic design programs" and not include Photoshop. Also check out Illustrator (<http://www.adobe.com/products/illustrator/>)

## Miscellaneous

- <http://www.apple.com/downloads/dashboard/reference/picker.html> - A great Apple Dashboard widget for quickly picking websafe colors to use in your emails.
- <http://www.vmware.com> - We use vmware to setup testing computers, with multiple operating systems and email applications.
- <http://www.newsletterarchive.org> - Online repository of email newsletters. Great if you're looking for ideas and inspiration. Take a look at what some of the largest companies are doing with their email marketing
- <http://www.templatekit.com> - They sell tons and tons of templates for websites and emails
- <http://www.mailchimp.com/resources/templates/> - We posted some free HTML email templates, which are easily tweaked to match your brand, and pre-tested on all the major email applications.
- <http://www.mailchimp.com/help/> - Our Help Center, with searchable knowledge base.
- <http://www.istockphoto.com> - Really, really, really cheap stock photography. We get all our photos from here. It's a great site.

## Email Applications & Known Issues

Following are the major email applications that we test new templates in, plus the known issues that we've uncovered for each. Use it as a starting point, but be sure to test thoroughly on your own. Many years ago, when we first published our HTML email design guide, we listed all kinds of issues across different email apps. But these days, they've all gotten a lot better (as long as you keep your layouts in simple tables, and don't use CSS-positioning, DIVs, or anything that sets off antivirus apps, like Flash, JavaScript, and ActiveX). This table has become more a list of "peculiarities" than concerns.

Email Application	Known Issues
AOL 9	Nothing horrible here, but their "preview pane" is extremely tiny. Make sure to design your emails to peek out enough from the preview pane. Your HTML code needs an extra line break or two after the final </HTML> tag, or you'll see weird "equals signs" in your email
Outlook 2000	This one's getting replaced by Outlook 2003, so won't be around much longer. Default installations of Outlook 2000 will display just about anything you throw at it, including embedded movies (hence all their virus and spyware problems). Displays HTML email just fine, save for some obscure CSS properties. Like AOL, if you don't have an extra line break under the closing </HTML> tag, you may see mysterious equals signs appearing in your email.
Outlook 2003	Blocks images by default (as do most apps these days) to protect your privacy. Displays HTML email fine, but default installations of Outlook 2003 open up in 3 vertical panes, leaving little room for your preview pane.
Mozilla Thunderbird	Thunderbird blocks images like other apps, but doesn't use big, obvious placeholders. If you use "alt text" for a logo graphic for instance, the Alt text will display, instead of a big "broken/blocked image" graphic. The downside of this is that, depending on what alt text you use, people may

	<p>never know there's an image to display, and won't click the "show images" button. If you don't use alt text for your images, a placeholder seems to appear like other apps.</p> <p>Thunderbird 1.5 has a new "This looks like a Phishing scam" feature, where you get a big warning icon at the top of the email if it detects a link that appears to be deceptive. For instance, if it finds a link described as, "Paypal home page" but the URL actually points to some suspicious URL. Nice feature, but it gets triggered if your email uses click-tracking. Mozilla says they're working on improving it, but it's not a huge issue for you if your emails always look reputable.</p>
Lotus Notes	<p>Ah, Lotus Notes. Where do we begin? Actually, Lotus has gotten a lot better with HTML email, but they've still got peculiar issues that pop up now and then. You'll almost always get a random border box around something. A graphic, a table cell--- whatever. Don't ask me why. It just happens. Oh, 6.5.3 handles center alignment ok, but 6.5.4 doesn't. Go figure, And neither seems to handle animated GIFs with transparency. COLSPANs can also be touchy here. Keep the table structures very simple.</p>
Comcast, Bellsouth, and other "at-home" ISPs	<p>We've noticed some "at-home" ISPs use email servers that interpret lines beginning with "periods" to mean "<a href="#">end of email.</a>" So be careful not to start any lines with a period. Sounds weird to do such a thing anyway, but places where it often occurs include: plain-text emails, where someone uses periods as visual markers and divider lines. Embedded CSS code (.header, .footer, etc). Indent your CSS, or add a space in front of each line to get around this.</p>