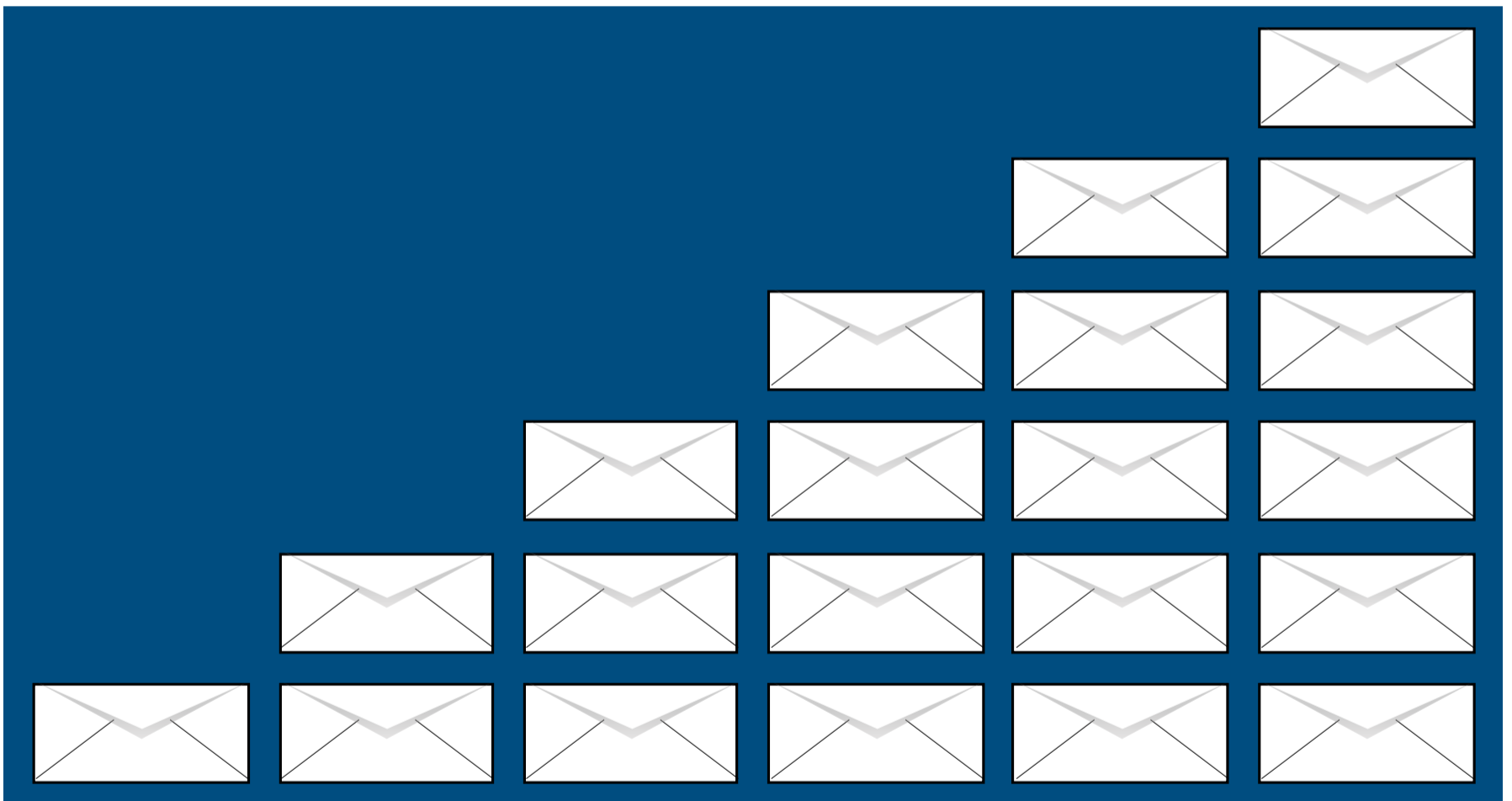


# Best Practices Enhanced

Volume 2: Design and Image Management  
*2017 Revised Edition*



This guide is published by:  
Goolara, LLC  
1030 Country Club Drive, Suite D  
Moraga, CA 94556  
Telephone: (510) 522-8000  
(888) 362-4575  
Fax: (510) 522-2457

Copyright © 2017 Goolara, LLC All rights reserved.

No part of the contents of this publication may be reproduced or transmitted in any form or by any means without the written permission of Goolara, LLC.

Goolara and the Goolara logo are registered trademarks in the United States, other countries or both. All Rights Reserved.

[www.goolara.com](http://www.goolara.com)

# TABLE OF CONTENTS

---

Introduction . . . . .	1
HTML Code . . . . .	3
Code pages. . . . .	3
HTML Compatibility . . . . .	4
Avoid Extraneous HTML . . . . .	5
Leave That Body Alone . . . . .	5
Use CSS in Email . . . . .	7
Inline vs. Style Tags . . . . .	7
Tables vs. DIV tags . . . . .	11
JavaScript. . . . .	11
Design For Subject Lines. . . . .	12
The long and short of it . . . . .	12
Start on the left . . . . .	13
Make the first text count. . . . .	13
Preheader divs . . . . .	14
Preheader best practices. . . . .	15
Special characters in Subject Lines . . . . .	16
Emojis . . . . .	16
Emoji or Emoticon? . . . . .	16
Where are they? . . . . .	17
I'm Not Mad I'm Happy!. . . . .	19
Politically Correct Emojis. . . . .	20
Emojis and Deliverability. . . . .	20
Test and Test Again. . . . .	21
Dynamic Subject Line . . . . .	21
Image Preparation . . . . .	22
There are no images. . . . .	22

Use ALT tags . . . . .	24
Text-to-image ratios . . . . .	24
Background images . . . . .	25
Layout Considerations . . . . .	26
Design for above-the-fold. . . . .	26
Don't look like a mass email. . . . .	29
Centering content . . . . .	29
Deliverability Considerations. . . . .	30
CAN-SPAM. . . . .	30
CASL . . . . .	31
A Clear Unsubscribe message . . . . .	31
Offer a view-in-browser option . . . . .	33
Create a text section . . . . .	34
Get Them to Reply . . . . .	35
Don't send attachments . . . . .	35
Testing . . . . .	36
Perform split tests . . . . .	36
Send tests . . . . .	37
Mobile Considerations . . . . .	38
Responsive Design. . . . .	38
Mobile First. . . . .	39
Keep the email 500-650 pixels wide. . . . .	40
Keep links separated for fat fingers . . . . .	41
Links on Mobile Phones . . . . .	41
Dynamic Content . . . . .	42
Summary. . . . .	44
Revision History . . . . .	47

# INTRODUCTION

---



The appearance of your email is almost as important as the content. If content is king, then design is the queen, and a powerful queen at that. Studies show that people are far more likely to respond to good looking email than plain text messages. Sending a good-looking email is ideal. Unfortunately, this is not always easy to do. There are many potential pitfalls that can turn your email into gibberish, and this isn't something you want to learn after a mailing has been sent. In this guide, we will address the most important issues, and address some important considerations that are too often overlooked by email designers. This guide is a companion volume to *Best Practices Enhanced, Volume 1: Content, List Management, and Testing*.

As we discussed in *Volume 1, Best Practices* simply means doing the things you should always do to make your email easy to read, regulation legal, engaging, and effective. Following Best Practices for design helps encourage more opens and higher clickthrough rates. It can also help eliminate mistakes and problems you might encounter when dealing with various mailbox providers.

Some of what is discussed here is intended for people who are familiar with the basics of web and email design. If you are already designing email, you've probably already encountered some of the potential pitfalls we list. If you've designed web pages, but are new to email design, then you should read this entire guide. As you'll learn, what's true for a web page, isn't always true for an email, and, in some cases, contradicts everything you've learned.

*Best Practices  
simply means  
doing the things  
you should  
always do to  
make your  
email easy to  
read, regulation  
legal, engaging,  
and effective.*

When we talk of design, we mean all the aspects that affect the appearance of an email. Some of these topics, such as subject line length, may not seem like design issues at first glance, but they are no less important than the images when it comes to the way an email will look in a recipient inbox.



# HTML CODE

---



Email can be sent as plain text, HTML, or both. These days, HTML is the preferred method of getting content to users. Over the years, HTML has added many new features that make website design easier and more attractive. While these new features usually work on web sites, the HTML that works consistently in emails is still a few years behind the times. New elements and attributes that aid in website development don't always work in email clients. Some HTML5 elements go unrecognized by email clients, and CSS properties are often only

seen if they are placed repeatedly in the starting tag for every element (inlined). JavaScript is ignored altogether.

## Code pages

When email was first created back in the 1970's, an email was a plain text message system, written in English with alphanumeric characters using a form of encoding called ASCII. This became the standard that we still use today; but what about HTML, special characters, or emails written in a foreign language? Over the years, a number of standards have been adopted to display other languages such as Arabic and Chinese, and to produce special characters such as emojis and em dashes. Fortunately, you don't need to understand the details of how this happens, but there are a few concepts that are useful to familiarize yourself with. One is that every email is sent in a particular code page (character encoding) that controls which characters are displayed. For most languages there are several choices, but the choice of characters that can be represented is limited to the characters of that language. For example, the English code page ASCII has all the characters needed to represent English, but would be unable to represent the character é, which appears in most European languages. Instead of this character, most email clients will substitute "?" or perhaps a blank box, like this: □.

*“There are many things used on web pages that would be nice to use in emails, but are not allowed.”*

But what if you want to send an email that includes characters from other languages? Even if you are writing in English, you may, from time to time, find you need to display a name or word that contains non-English characters. With a language-specific code page you'd be stuck. You can see this in early web design, where words that use umlauts, for instance, are either displayed as diphthongs (for instance, maedchen for mädchen), or the umlaut is simply dropped (uber for über). To solve this, an extended set was added to ASCII, but even here, only European characters were added. You couldn't write in Russian, or Chinese, or any other language that didn't use a Western European character set. So, Unicode was developed. Unicode includes the full compliment of characters needed to represent nearly every language in the world. Normally Unicode is sent in an encoding method called UTF-8. There is also a UTF-16 and a UTF-32, but these are still not that common, and not useful for email.

Although Unicode has been around for many decades, some older email clients are still unable to render contents in this character set, but they can display characters for the language in which they are intended. This used to be more of a problem than it is today. If your content is strictly in one specific language, with no special characters or emojis, then you are still better off using the code page for that language. Otherwise, use the UTF-8 version of Unicode.

### **HTML Compatibility**

Since most email is read in the HTML format, email clients are constantly working on ways to ensure that their customers don't experience viruses and other malicious attacks while they are reading the mail in their inboxes. One of the driving forces behind security for email is to make sure that an email recipient cannot get a computer virus simply by receiving or opening an email. The ILOVEYOU virus, a few years back, took advantage of the automation built into Microsoft Outlook, causing Microsoft to scale back Outlook's automation features and leading others to follow suit. Today, very few email clients will accept JavaScript. Including it your email code could be seen as an attempt at an attack and might send your mailings to the junk folder.

Likewise, CSS offers many features that would make it easier to control the design and appearance of email, but it too is



sometimes either removed, or reduced to a bare bones set of usable properties. Sometimes this is simply because an email client doesn't want your CSS to interfere with the CSS they use to layout their pages. CSS is still useful, though. The most compatible way to use it in email is to place it inline for each element within the body of the email. You can do this manually, or with a CSS inliner that takes the CSS in the head and moves it into each element automatically (more on this later). While many email clients nowadays will accept CSS that is placed between head tags, or is otherwise separated from the body of the email, not all of them will, so keep this in mind when you are designing your mailings.

Besides the inability to use JavaScript, the difference between email and website design is most apparent in the acceptance of HTML5 elements. HTML5 has resulted in the deprecation of many attributes, with sites such as w3schools.com recommending the use of CSS for certain actions such as centering or font choice.

### **Avoid Extraneous HTML**

It's not uncommon for designers—when working in HTML design software outside of their ESP's visual editor—to simply cut and paste the entire HTML document in their email marketing software. It is the path of least resistance. Consider, however, that if you are viewing your emails with an online email client (Gmail, for instance), that site already has its own attributes and properties that it uses to display the page. It has its own `<DOCTYPE>` and `<html>` tag specifications too, so having these in your code will, in the best case, accomplish nothing, and in the worst case, ruin the email. We have seen examples of email that wouldn't display properly simply because the `<DOCTYPE>` information was left in. Some email clients these days are sophisticated enough to strip out the ones that conflict with their own settings, or ignore the ones they aren't compatible with, but don't count on it. This is especially important when you are using pre-existing templates, and *especially* those intended for other purposes, such as web site page design.

### **Leave That Body Alone**

Sometimes designers try to put background colors, centering commands, or other values in the `<body>` tag, but we don't recommend this. Most ISPs ignore any style information in a



body tag. This can result in substantial changes to your email's appearance. Some mail readers can render these attributes, but most will not. The safest bet is to assume the body tag will be ignored. For maximum compatibility, we don't recommend trying to apply any style attributes to it. It is better to create a table with a height and width of 100%, and apply any other style properties you want for the entire page inside that <table> element instead.

To serve as an example, here is the HTML for an email with several unnecessary tags and other useless information. This is an example of how *not* to format an email. The errors are indicated in the text to the left of the HTML code.

DOCTYPE tag can interfere with proper email display.

HTML tag is unnecessary, and should not include styles.

Metatags will not be used in email. They probably won't hurt anything, but they won't help either.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" style="border-box: border-box; font-family: 'Helvetica Neue', 'Helvetica', Helvetica, Arial, sans-serif; font-size: 14px; line-height: 1.5">
  <head>
    <meta name="Description" content="Best products on the market." />
    <meta name="Keywords" content="shopping, products, buy, bargain, best price" />
    <title>Shoppers' Paradise</title>
    <meta property="og:title" content="Shoppers' Paradise">
    <meta property="og:description" content="Best products on the market!">
    <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
    <meta content="width=device-width" name="viewport">
    <style id="media-query" type="text/css">
      body {
        margin: 0;
        padding: 0
      }
      table, tr, td {
        vertical-align: top;
        border-collapse: collapse
      }

      @media only screen and (min-width:620px) {
        .block-grid {
          width: 600px!important
        }
        img.fullwidth {
```

Never use JavaScript in an email! →

It is safer to not add styles to body tags. Use 100% table instead.



```
max-width: 100%!important
}
<script src="js/script.js"></script>
```

```
</head>
<body style="background: #ffffff; font-family:
'Helvetica Neue', 'Helvetica', Helvetica, Arial,
sans-serif; bgcolor="#ffffff">
...
```

CSS includes many command and options. Probably more than most people will ever know or use, but can be helpful in the right circumstances. However, the commands that work consistently in email is a much more limited set.

The following style conventions work in most of the major email clients:

```
font
font-family
font-style
font-variant
font-size
font-weight
letter-spacing
line-height
text-align
text-decoration
text-indent
color
background-color
border
border-collapse
direction
table-layout
```

These are, for the most part, safe to use, although some, such as font-size may be affected by the viewing device (see above).

Technically, the html tag could also be in red. It's not really needed. But the bigger issue here is the application of style information to it, which is never a good idea. The biggest no-no in the example above is the inclusion of JavaScript information ("js/script.js"). While including metatags probably won't have any impact on how your mailing displays, the JavaScript instruction could actually break it.

## Using CSS in Email

CSS was developed to work around the fact that every computer has its own, occasionally unique, set-up. I may have the Helvetica font on one computer but not on another. Before CSS, web designers were at the mercy of the fates when it came to how a page would look. If a font was missing, the computer would replace it with whatever font it used as its default. Suddenly, those rows of Helvetica could turn into Times Roman, or worse: Courier. With CSS, the designer can specify exactly which fonts or font families should replace others, regardless of the computer defaults. If the computer can't find Helvetica, the CSS will tell it to use Arial, or some other sans serif font instead.

It wasn't long before the capabilities of CSS expanded to include nearly every aspect of design, ensuring that the finished product will look as much like the artist's original concept as possible.

## Inline vs. Style Tags

When building websites, it's standard operating procedure to use CSS files that are separate from the actual web page, but this won't work in email. For email, you need to make sure the CSS information is contained within the mailing. There are two ways to do this: as separate properties included within style tags, or styles that are set "inline" for each visible element in an

email (paragraphs, headings, etc.). As previously mentioned, CSS styles work more often when they are placed inline. Typing in all of this information repeatedly for each element is a time-consuming process, so most people prefer to place their CSS style information in style tags in the head section of an email, and then run a CSS inliner, which is a web page that automatically does the tedious work for you.

First, let's look at an example of some email text with the CSS information appearing between style tags in the head area of an HTML document. Note that I've removed most of the content and CSS to make the HTML easier to follow. You'll also notice three different color models used in the CSS (RGB, color name, and hex). This is simply to show that you can use any of these three color value system, along with RGBA, HSL, and HSLA, although, for maximum compatibility, we recommend either RGB, or **six** digit hex values.

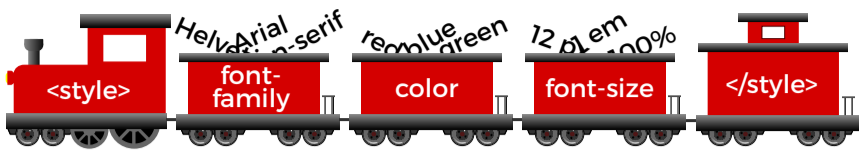
```
<html>
  <head>
    <style>
      .paral {
        color: rgb(100, 48, 48);
        font-family: Arial, Helvetica, sans-serif;
        margin-bottom: 10px;
        font-weight: normal;
        font-size: 14px;
        line-height: 1.6;
        margin-right: 20px;
        margin-left: 20px;
      }
      h4 {
        font-weight: 500;
        font-size: 23px;
      }
      .btn {
        text-decoration: none;
        color: white;
        font-family: Arial, Helvetica, sans-serif;
        background-color: #666666;
        padding: 10px 16px;
        font-weight: bold;
        margin-right: 10px;
        text-align: center;
        cursor: pointer;
        display: inline-block;
      }
      @media only screen and (max-width: 600px) {
        a[class="btn"] { display: block !important;
          margin-bottom: 10px !important; background-image: none !important; margin-
```

*There are two ways to place CSS information in an email: as separate properties, or as styles that are set “inline” for each visible element in an email.*

```
        right:0!important;}
    }
</style>
</head>
<body>
  <table>
    <tr>
      <td style="vertical-align: top; padding-right:10px;">
        
      </td>
      <td>
        <h4>Title Ipsum</h4>
        <p class="para1">Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. </p>
        <a class="btn">Go to Link &rarr;</a>
      </td>
    </tr>
  </table>
</body>
</html>
```

This version will work in many places, but if you open it in a client that doesn't accept separate CSS information, you'll end up with a generic looking email. Now here's that same email after running it through a CSS inliner:

```
<html>
<head>
  <style>
@media only screen and (max-width: 600px) {
  a[class="btn"] {
    display: block!important;
    margin-bottom: 10px!important;
    background-image: none!important;
    margin-right: 0!important;
  }
}
</style>
</head>
<body>
  <table>
    <tr>
      <td style="vertical-align: top; padding-right:10px;">
        
      </td>
      <td>
        <h4 style="font-weight: 500; font-size: 23px;">Title Ipsum</h4>
        <p class="para1" style="color: rgb(100, 48, 48); font-family: Arial, Helvetica, sans-serif;
```



```

margin-bottom: 10px; font-weight:
normal; font-size: 14px; line-height:
1.6; margin-right: 20px; margin-left:
20px;"/>Lorem ipsum dolor sit amet,
consectetur adipisicing elit, sed do
eiusmod tempor incididunt ut labore et
dolore magna aliqua. </p>
<a class="btn" style="text-decoration:
none; color: white; font-family: Arial, Helvetica,
sans-serif; background-color: #666666; padding: 10px
16px; font-weight: bold; margin-right: 10px; text-
align: center; cursor: pointer; display: inline-
block;"/>Go to Link &rarr;</a>
</td>
</tr>
</table>
</body>
</html>

```

You can see that all of the style formats except the @media query have been removed from the head tag area and placed inline, so that the header, paragraph, and button tags each contain all the information they need as to font families, sizes, colors, etc. Since the @media query can't be placed inline, it remains in the style section for use on the pages that can recognize it.

Keep in mind, though, that even if you do this, some email clients will *still* not display the style information correctly and may even replace specified color and header information with choices of their own. If you use standard header tags (<h1>, <h2>, etc.), Outlook.com, for instance, will adjust the color of the headers in your email to match the color scheme chosen by the recipient for their Outlook interface. You can avoid this by not using the header tags and simply define the font, color, size, etc. inline for each section. This is more time consuming, but it will help eliminate unwanted surprises.

You should also be aware that not all CSS inliners are created equal. While all of them will inline the styles for the various elements in an email, some also remove the media queries, wrecking any responsive design parameters you've created. Check the HTML code after you've run it through a CSS inliner, and if the media queries are missing, you'll have to put them back in. We've also seen at least one CSS inliner that added the style information to the head and style tags. In all likelihood, this will have no effect on the mailing's appearance, but it does add unnecessary bulk to your email, and could cause problems for certain email readers. If you see the Inliner

you are using do this, you should probably use a different one. Better safe than sorry.

## Tables vs. DIV tags

Over the past few years, there's been a strong push for web designers to move away from tables in favor of DIV tags. While they may be preferable for websites, tables are still preferred for controlling element placement in email. As is often the case, the email clients lag behind the rest of the Internet when it comes to implementing the most recent developments in HTML. You can use DIV tags in email, and more and more ISPs have no trouble with them, but tables are still the safest way to construct an email that will render properly across all platforms.

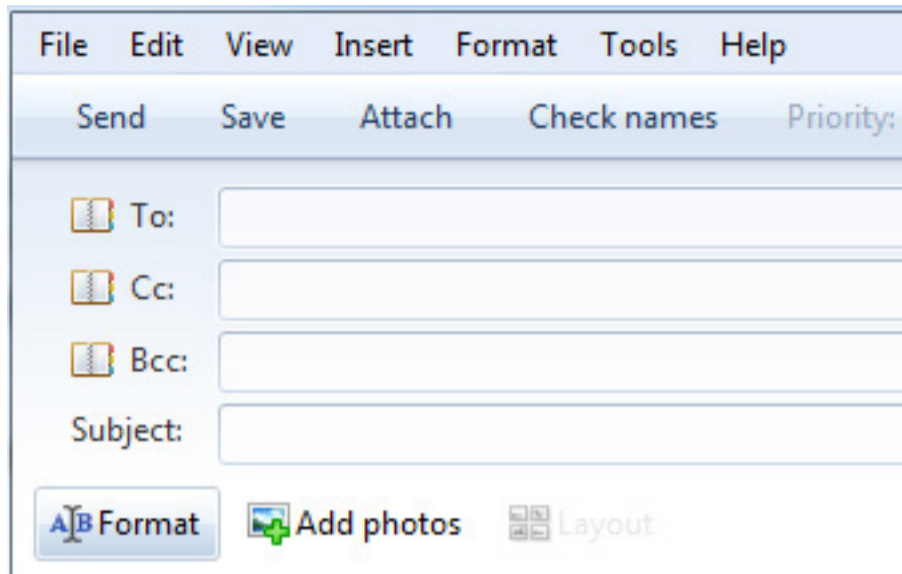
## JavaScript



As mentioned previously, you should *never* use JavaScript in your email. Made to make your online experience more interactive, JavaScript is too easily exploited by hackers and phishers and is virtually always stripped out of email by ISPs. It can even get you red flagged no matter how innocuous your email is. Even if your mailing does not end up in the trash, it is likely that the email client will strip the scripting out of the message, which could leave it incomprehensible.

# DESIGN FOR SUBJECT LINES

---



Pivotal to the success of an email is its subject line. The success of an email can rise or fall on the strength of its subject line. If the subject line inspires the reader to open the email, then it's done its job. But there are some hidden factors in determining what makes a good subject line: Length, characters, and what device the recipient is using can all play important factors in the making of a successful subject line.

## The long and short of it

There are several articles in magazines and online that attempt to tackle the question, "How long should a subject line be?" There is no consensus. One school of thought says that short subject lines leave people intrigued and will get them to open the email to learn more. Another says that long email subject lines are better because they give people enough information to make an informed decision, so that when they do open the email, they are more likely to click through to purchase.

But there is another factor involved in the length of the subject line, and that has to do with how the various email providers display information. Gmail, for instance, displays the subject line followed by as much text as it can read from the email copy.

□ ☆ Jim Morton

How long is your subject line? - Knowing how email clients work can help you take advantage of the available real estate

4:38 pm

Smart phones do something similar, with the Sender's name appearing first with as much of the email message that can fit below that, and the first lines of email body copy below that.

Some online email clients, such as Yahoo and Outlook.com display only the Sender and the subject line. Knowing this, there are a few basic Best Practices to follow:





## Start on the left

If your subject line is long, there's a good chance that some of it will be cut off, replaced with an ellipsis. Make sure that the most important information is at the beginning of the subject line. Starting an email with some standard phrase can leave you with this situation:

*Welcome to the September 2017 company newsletter. In this issue, we...*

How much will be cut off depends on the email client and the device upon which the email is viewed. Yahoo mail, for instance, cuts the message depending on the length of the browser window and your screen resolution.

A very long subject line might appear okay in Yahoo on a large screen with a high resolution and a large enough browser window. Gmail, on the other hand, always provides a certain amount of space for the email text copy, so anything over around 85 characters runs the risk of being cut off. Email clients that allow alternative lay-outs for message and reading panes, such as Outlook, Live Mail, and Outlook.com, are more variable. A side-by-side layout lets you change the size of the message pane, allowing for more text to appear, but at the expense of the reading pane. The default is always in favor of the reading pane. Subject lines can be cut off after as few as thirty characters. Phones face even greater limitations of space. On the iPhone, for instance, a message is cut off after 35-40 characters, but the first text in the email is given around 90 characters.

## Make the first text count

Some email viewers show you the subject line followed by the first text. It's a shame to waste this valuable real estate on things like "Click here to view as Web page," "Unsubscribe," and random alt tags when you could be using it to further promote your message. It's possible to construct your email so that the text logically follows the subject line. This only works in certain situations and whether this technique has any value for you depends entirely on your recipients and how they view their email. Even if you don't plan to apply this technique, you don't want to waste that valuable real estate with something like

*“Whether you choose to use a first text preheader, or the div version is up to you, but you should be doing one or the other.”*

“Please click “Not Spam” if delivered to Spam Folder.” In fact, starting with this particular message could actually increase your chances of ending up in the trash. While “View in Browser,” and “Unsubscribe,” are important links to keep near the top, make sure you’ve got something to say before them for the sake of the email message pane displays.

### **Preheader divs**

An alternative to the first element approach is to use a non-displaying div preheader that contains the information you want to appear in the snippet. This method affords greater flexibility because the text in a div preheader does not have to appear anywhere else in the message. Using a preheader div lets you expand on the meaning of the subject line, give people the most important piece of information from the email, or paraphrase the text on the page as needed.

For example, you may be offering a contest, but you’ve placed the details below the main message of the email. While the subject line and opening text talk about the primary subject, you still want your readers to be aware of the contest. By using a preheader div, you can keep the top of the message and subject line on topic, and allow the snippet to act as an enticement to encourage registration.

But preheader divs come with one drawback. As we discussed in the previous chapter, not all email clients handle divs well—especially not invisible ones. Some ISPs will place the text on the page in spite of everything. For this reason, preheader div tags are usually configured something like this:

```
<div style="font-size: 1px; color: #ffffff; display: none !important; mso-hide:all;">Preheader snippet text goes here.</div>
```

The color attribute shown in the example is based on an email with a white background. Some marketers prefer to omit this color information since text on a background of the same color is seen by some spam filters as a negative quality and can affect the email’s deliverability. Some marketers set the font size to zero, but that comes with the same caveat as color on color.



Most email marketers use the first text approach. It is simpler to implement and is in less danger of being flagged as spam. It also eliminates the problem of compatibility. A preheader that appears as text on a page will work anywhere, whereas non-displaying divs might not.

On the other hand, if your email design is not conducive to the first text approach, and there is some message further down in the content that makes a better teaser line than the first text on the page, then the div approach is the better choice.

### **Preheader Best Practices**

Whichever method you use, it is a good idea to make sure that the snippet is not misleading. If the snippet reads “Learn how to receive your personalized coffee mug,” there better be a way to do so in the email or you are only going to alienate your audience and come across as dishonest.

You also need to be careful about the wording in your preheader. A preheader sentence such as “All clothing is now 50% off for all platinum club members,” could shorten to “All clothing is now 50% off...,” which might make non-platinum club members angry when they finally see the full sentence. Rewording the sentence (e.g., “Platinum club members now get 50% off on all clothing”) so that the modifying information comes first will help avoid this problem. Also be careful with statements where the second half of the sentence contradicts the first half (e.g., “The Ford Fiesta is a great car, if you like visiting the repair shop”). Unless you are intentionally intending for the message to be cut off for a humorous effect, it’s a good idea to reword the line so that this won’t happen.

Another thing to watch out for is the Title tag, which the iPhone will include as the first text in the snippet, but other email clients, such as Gmail and Yahoo, will not. There may be times when you can take advantage of this idiosyncrasy, but in most cases you’ll want to remove the title tags from your email’s HTML to ensure that your preheader appears the same across all platforms.



On the iPhone, several Unicode symbols are converted to three dimensional versions. Here is a chart of those symbols. Below each line of standard symbols are the iPhone versions of the same symbols.

Whether you choose to use a first text preheader, or the div version is up to you, but you should be doing one or the other. The various email clients give you a certain amount of text to get your message across, and if you're not taking advantage of it, you are missing out on a simple and easy method to add to the potential selling power of your email.

### Special characters in Subject Lines

There was a time when using anything other than the basic ASCII character set was a bad idea. More and more, however, Unicode is becoming the standard for character sets across the Internet and we're starting to see some clever uses of special characters appearing in subject lines. Hearts (♥) and stars (★) are the most popular, but some businesses are taking advantage of the extensive character set to use symbols that are relevant to their industries. Some examples are restaurants (☺), travel industry (✈), and, of course, email services (✉). This doesn't mean that everyone will be able to see these symbols in their subject line, so some caution is still advised. Make sure that if the symbol doesn't appear, the message is still understandable. At this time, the use of symbols appears to have little or no impact on the deliverability of email. Of course, that may change if spammers decide to use them extensively.

### Emojis

More prevalent of late is the use of emojis in subject lines—those emails with little pictographs in the subject lines. At Halloween, they are jack-o-lanterns and ghosts (🎃, 👻), further into winter they might be snowmen or Christmas trees (🧊, 🎄). Sometimes they relate to the sender's industry. Guitar Center, for instance, regularly uses the guitar pictograph (🎸), while Webdesigner News starts every subject line with the image of a pencil (✎). These are emojis, and have become popular tools for spicing up subject lines to make them more appealing.

### Emoji or Emoticon?

First let's get the out of the way the inevitable question, "What is the difference between an emoji and an emoticon?" An emoticon is a facial expression created using the limited assortment of punctuation that is available in basic English text. The most well-known example is the colon and right parenthesis indicating a smiling face: :) . The alternative to basic text is Unicode, a character coding system designed to



include every character in every language. In Unicode, there is an emoji for a smiling face (☺), along with a large assortment of other tiny pictographs. Unlike a smiling face created with a colon and a right parenthesis, the emoji is one character, not two. There are several emojis you can choose from to indicate various levels of mirth (😄😃😁😆😅), along with characters for nearly every other human emotion (😱😨😩😪😫).

Emojis got their start on Japanese mobile phones, where they were used to replace emoticons. Although the names sound similar, the word emoji has nothing to do with emotions. It is a combination of the Japanese words for “picture” (e – 絵) and “character” (mo-ji – 文字). Their worldwide acceptance began when Apple decided to include emojis as a feature on its iPhones in 2009. Then in 2010, hundreds of emojis were encoded and introduced in the Unicode Standard, and more are added every day. As of this writing, there are 722 emojis available with Unicode character coding. Emojis have popped up everywhere from Android phones to Gmail.

As befitting their Japanese roots, some emojis are specifically aimed at Japanese culture and leave westerners scratching their heads. Emojis for foods such as Dango (🍡) and Oden (🍢), and festivals such as Tanabata (🎋) and Tsukimi (🌕) presumably don’t see much use in America and Europe, while other symbols, such as the white flower (🌸) might be used, but in an entirely different context from how it’s used in Japan (in Japan it is used to mean “well done”).

### Where are they?

Unless you are using an iPhone to write your mailings, which is highly doubtful, finding the emojis on a keyboard can be tricky. You can type in the Unicode directly, but that is a pain in the neck, and you first have to know these codes to type them. For instance, to add an airplane (✈️) to your subject line, you’d need to type in U+2708 (hold down the Ctrl+Alt+Shift keys, type U, type 2708, then hit enter). It’s a lot of work for one character, and it doesn’t always work anyway. Some desktop systems have shortcuts for inserting emojis, or special pull-down menus, but these are still slow. The easiest way to add emojis that we’ve found is iEmoji.com, which lets you compose

the subject line on their web page, then copy and paste it to your email marketing software. But care should be taken when doing this, which leads us to the next point: Why do some emojis work in subject lines and others don't?

Have you tried using emojis in your subject lines, only to have them replaced by small squares or question marks? There are two primary causes for this. The first is that you are using a newer, unusual emoji that is not included across all systems. The country flags, for instance, do not show up in most email reader subject lines, and often not in the content either. In most email readers, the newer ability to choose the skin tones of certain emojis isn't available, and will add blank squares or question marks to a subject line (more on this below). When using emojis in the subject line, it is safest to stick to the default emojis, which usually appear in yellow.

With a few email readers, such as Live Mail, how it displays can even depend on where it is in the software. Take this example:



All three emojis appear in the list window on the left, but not in the title window on the right. The first emoji (the umbrella) appears correctly in both areas, while the others (the cat and dog) appear as empty boxes on the right. The reason for this is because the umbrella is one of the original emojis that were introduced in 1995. As a rule, these will appear in your subject lines more often than the newer emojis will. Some of these characters, such as the smiley face (☺), musical notes (♪ 🎵), and card suits (♠ ♣ ♥ ♦) were added early on, and are available as symbol characters in basic English character sets. Here is a list of the original 1995 emojis:



*As a rule,  
it is never a  
good idea to  
use emojis to  
replace words  
in a subject  
line.*

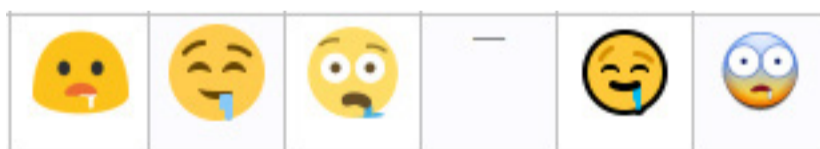
A second, and more likely cause of question marks in the subject line, is that your email is set to something other than Unicode. If the character you want to use is not available in the character set you are using, it will not appear in the subject line. Go to the settings while in your email marketing software and check the character encoding choice. If it doesn't say "UTF-8" it's probably not going to work in the subject line, even if it works in the content.

As a rule, it is never a good idea to use emojis to replace words in a subject line. If the emoji is replaced with a question mark, you might end up with a subject line that still makes sense, but says something you don't want it to. For instance, if you replaced the word "love" with a heart in the subject line "You'll ♥ our deals," you could end up with this: "You'll ? our ideas," which isn't exactly a confidence builder, and could be read as "You'll question our deals." In fact, a scan of various emails—and even web pages—shows that using the heart symbol to replace the word love might just be the number one gaffe. I even found the following line in an online article about emojis: "There's a lot of ? for emoji these days..."

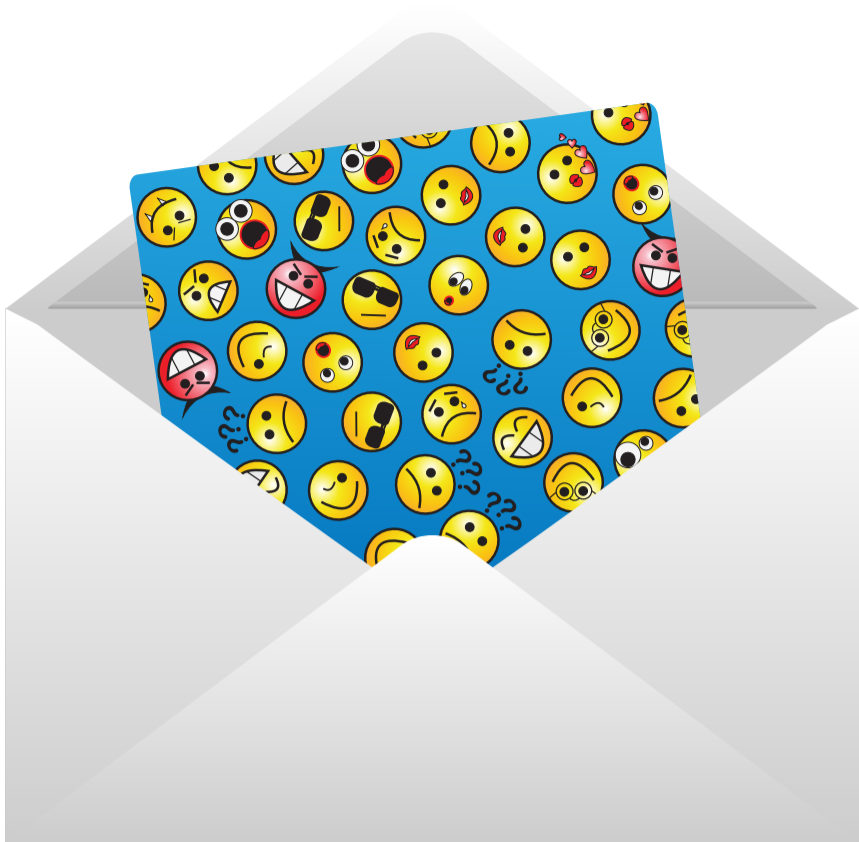
It is safer to put the emojis at the beginning and the ends of the subject lines, or as breaks between words. Even so, you should ask yourself: If a question mark appears instead, will it affect the subject line's meaning?

### **I'm Not Mad. I'm Happy!**

In some cases, the emojis from one operating system are different enough from the emojis in another to cause confusion. Here, for instance is the emoji labeled "drooling face":



Two appear happy, two appear unhappy, and the last one looks downright scared. One doesn't even appear at all. While it is unusual for emojis to vary this much across platforms, it doesn't hurt to check the emoji you plan to use to make sure it doesn't change too drastically when viewed on different devices and operating systems. The easiest place to do this is at the Unicode Consortium's Full Emoji Data page. There, you'll find all the emojis—including a few that appear animated, such as the Gmail emojis, which sometimes cry, bounce up



and down, or wink. The Unicode Consortium's Data page also lists the date when each emoji was introduced, which can help you determine how safe it is to use that emoji. An emoji introduced in the past couple years is probably not going to show up in a subject line, and might not even show up in the content.

### **Politically Correct Emojis**

While animation is more site specific, and doesn't affect the individual emojis, there is another recent addition to the emojis that will affect how and emoji behaves in a subject line. After people complained that the emojis of hands and faces were not ethnically inclusive enough, a feature was added

whereby you can specify the gender of an emoji and its skin color. Care must be taken when using skin tones and genders as these add additional code to each emoji. For instance, the code for the left pointing finger emoji is U+1F448, while the code for the same emoji with pale skin is U+1F448+U+1F3FB. In subject lines, even if the original default emoji appears, the gender and color information will in most cases appear as empty square blocks or question marks. For this reason, it is best to stick to the basic emojis and avoid skin tones and gender additions until more mail readers are compatible with these features.

### **Emojis and Deliverability**

As always, the most important question is: Can emojis affect the deliverability of an email? Our tests suggest that, under some circumstances, emojis do appear to have a negative effect on an email's deliverability, but a minor one. Mailings with large quantities of emojis in the subject line and contents were more likely to end up in the spam folder, while those that used them more judiciously appeared to have no problems getting through. Obviously a subject line that is nothing but emojis is probably not a good idea. Some spam filters can identify if a subject line is nonsense, and a string of emojis looks just like gibberish. We recommend restricting the use of emojis in subject lines to no more than three, and to make sure there is actual text in the subject line as well. Keep in mind also that there may be aspects of your content that



*If you plan to use emojis in your subject lines, our advice is, as always with any format experiments: test and test again.*

are pushing your mailings close to a negative rating, and the emojis won't do anything to improve the situation. For information on what to look out for, check out our white paper, Deliverability Enhanced.

As to which emojis provide the best open rates, a quick scan of the articles that discuss this shows that there is no consensus here. In all likelihood, this data changes from month to month anyway. The only meaningful answer is to see how they do in your own tests, and proceed accordingly. Like those articles that tell you which day of the week is best for sending, any article that claims to know which emoji performs best is working from a limited data set and should be taken with a grain of salt.

### **Test and Test Again**

If you do plan to use emojis in your subject lines, our advice is, as always with any first time format experimentation, test and test again. We would also recommend paying closer attention than usual to the deliverability results in your tests. Some A/B split testing against subject lines without emojis or with different emojis isn't a bad idea either. Emojis can be a fun way to enliven your subject lines and increase open rates, but it will still require testing with your own recipients to see if they'll work for you.

### **Dynamic Subject Lines**

One feature in Goolara Symphonie you won't see in many other email marketing systems is the ability to create subject lines dynamically. This is different from mail merge (i.e., the ability to add a field, such as the first name, to a subject line), which is very common. With dynamic content, you have the ability to tailor a subject line based on logical queries (e.g., "Is the recipient over forty?") to further personalize your message. Best practices dictate that whatever logic arguments you put in place, you also include a fall-back subject line in case the software encounters a field that is either blank or contains faulty data. Needless to say, the same hold true for any other dynamic content in your mailing. See the *Dynamic Content* chapter for more information on this subject.

# IMAGE PREPARATION

---



Images are nice. They're pretty, they get us to pay attention to the email and they can even get us to click on things that text alone cannot. But the truth is, most email providers automatically block images, allowing the user to decide whether he or she wants to view them. You should always keep this in mind while you are designing your email. Assume that none of your images will appear then ask yourself, does my message still have meaning? Is it still compelling? Sometimes marketers create email comprised completely

of images. This is never a good idea. An email comprised of nothing but images is going to look like a stack of empty boxes (see page 22).

## There are no images

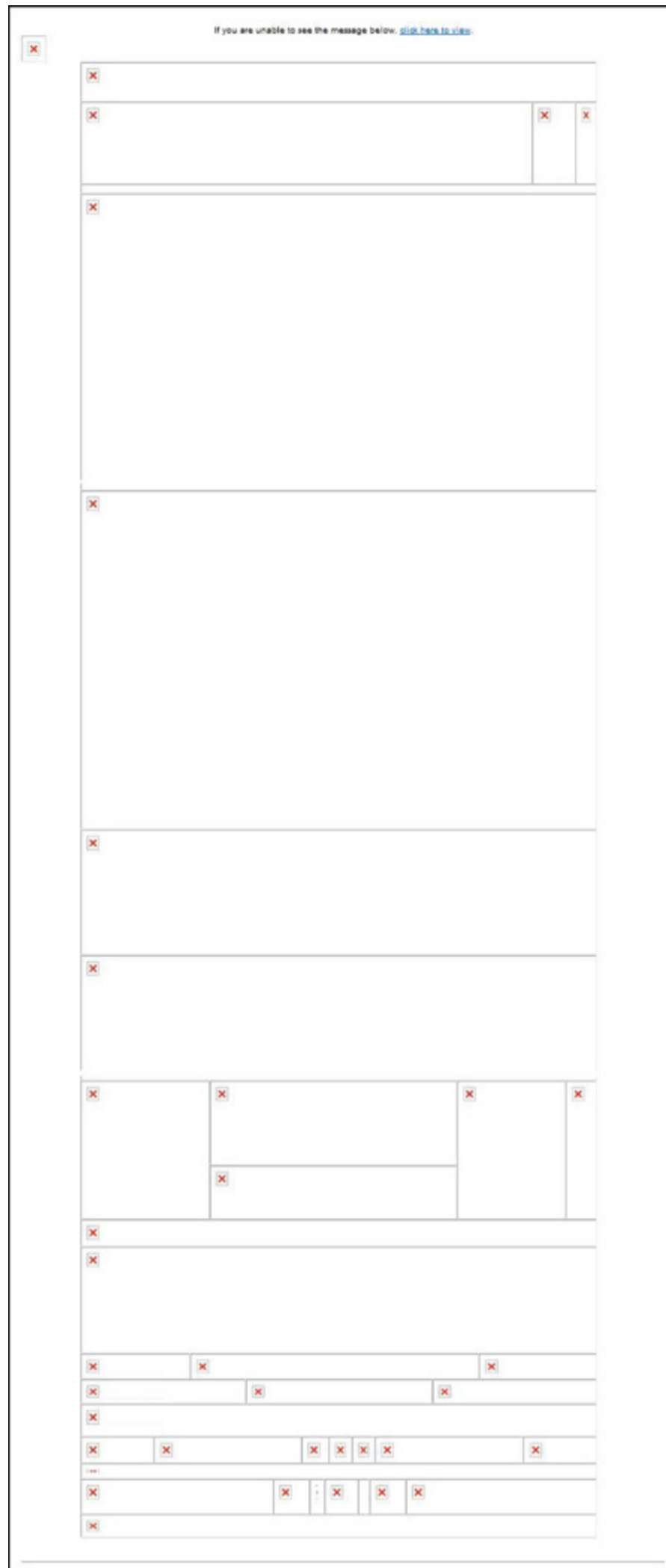
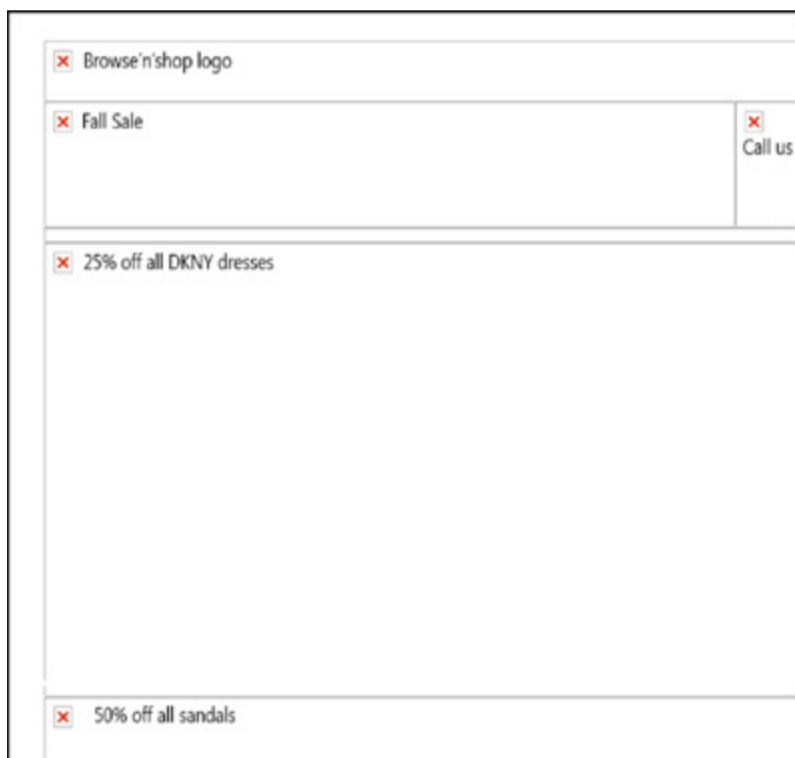
If your content is entirely, or even mostly, comprised of images, the recipient will see almost nothing informative if they view the content without images enabled. Until recently, most email clients required the user to either add your email to their address book, or click the option to enable images. Either way, the user had to make an effort to view the images and few recipients ever bothered with it. While this is less of an issue than it used to be, thanks, in part, to Google's decision to cache and automatically display images, there are still plenty of mailbox providers out there that keep the images turned off.

Making an engaging email that is not based largely on images does require more effort. It means taking more time to create HTML that has content that will render well without images, so your designer will need to take more time to do this properly. However, remember that not everyone will see your content if it is primarily image based, so it won't matter how beautiful your

*(Continued on page 20.)*



**Worst case scenario.** Here is an actual email sent by a company that specializes in teaching digital marketing techniques. Aside from the subject line, there is no textual information in this email, and thus no compelling reason to display the missing images. Aside from the “view as web page” statement at the top of the page, the only other text in this email was the boilerplate information at the bottom. Even simple Alt tags describing each image would have been an improvement. Below is shown the top part of the email with Alt tags added to the images.



*(Continued from page 18.)*

images are, since so few people will see them initially. Email content which is not as attractive, but has HTML which can present your message without the use of images, will get seen by many more people and will spread your message more effectively than one that is primarily image based. Since it will cost more for the designer to make content which is not primarily image based, you must decide if the cost is worth the increase in the number of recipients that will view the email.

### **Use ALT tags**

Every image you place in an email should have an ALT tag. The ALT tag is a piece of text that will display when the images are disabled. Ideally an ALT tag should make the recipient either want to display the images, or, at least, clickthrough based on what the ALT tag says the picture represents. For example, which of these seems more intriguing:



Clearly, the image on the right stands a better chance of being displayed. How in the world would a Chihuahua attack a bear anyway? The other one tells us nothing. We know it's an image, but, beyond that it could be anything. Is it an advertisement, or something not quite safe for work? We just don't know.

But the ALT tag does more than just give the missing image a context. It gives ISPs more text to consider for filtering purposes. Some designers like to split images into slices to make the HTML line up, but this doesn't help deliverability. It makes the HTML design more complex, and makes things like the ALT tags more difficult to name.

### **Text-to-image ratios**

Most ISPs use the ratio of text to images as a factor in the Reputation score. It is usually not a major factor, but every little bit counts. An email that is more than half images may have issues with deliverability and inbox placement. The issue arose when some spammers discovered that they could bypass

keyword spam filters by using images that contained the offending words or message. As a result, image-laden emails are considered more risky by ISPs, even if you have a good Reputation score. It is a best practice to avoid the use of many images for the reason described above for recipient's deciding whether to enable images and see your content. But another reason to avoid too many images is the calculation of text-to-image ratios for deliverability purposes (and inbox placement).

## Background images



One of the biggest frustrations for designers is that background images do not work in email. Even if the recipient clicks "Show Images," the background images will not appear in most email clients. If you design an email in which the background image plays an important part, you can end up with a mess. While more and more email clients add the ability to display background images, it is, by no means, universal yet, so it is best to avoid them.

How do you make that beautiful layout then, where the image is shown and text is superimposed over it? There are two possibilities, one good, and one difficult. The easy but wrong answer is to just turn everything into an image, with the text you want integrated directly into the image. This would only work for unpersonalized text, but gets back to the issue of using too many images and the host of problems that introduces. Unfortunately, the correct answer is to redesign your content to separate the area that holds the image from the area with the text. Instead of making a larger image with text within the image, reduce the size of the image and put the text in HTML next to the image. Background colors render well across all email clients, so your designer can match the colors of the image with the background color, so it is less obvious that the image has ended and the text has begun.

This is more difficult than creating a big image, we know, but to get the best results out of email, you should always have a balance between text and images. (For more guides to using images in email, please visit the Resources section at <https://goolara.com/Resources/WhitePapers.aspx>.)

# LAYOUT CONSIDERATIONS

---

What does it mean to have content that has a 50% or greater text-to-image ratio? Most ISPs use the ratio of text to images as a factor in the Reputation score\*, but they don't define exactly how that is calculated. We take it to mean that you look at the actual page size and compare how much of the area is taken by images versus other textual data. So if you had an email which was 4" square, as an example, and it had a 2" square image, you would have a ratio of 25% images.

## Design for above-the-fold



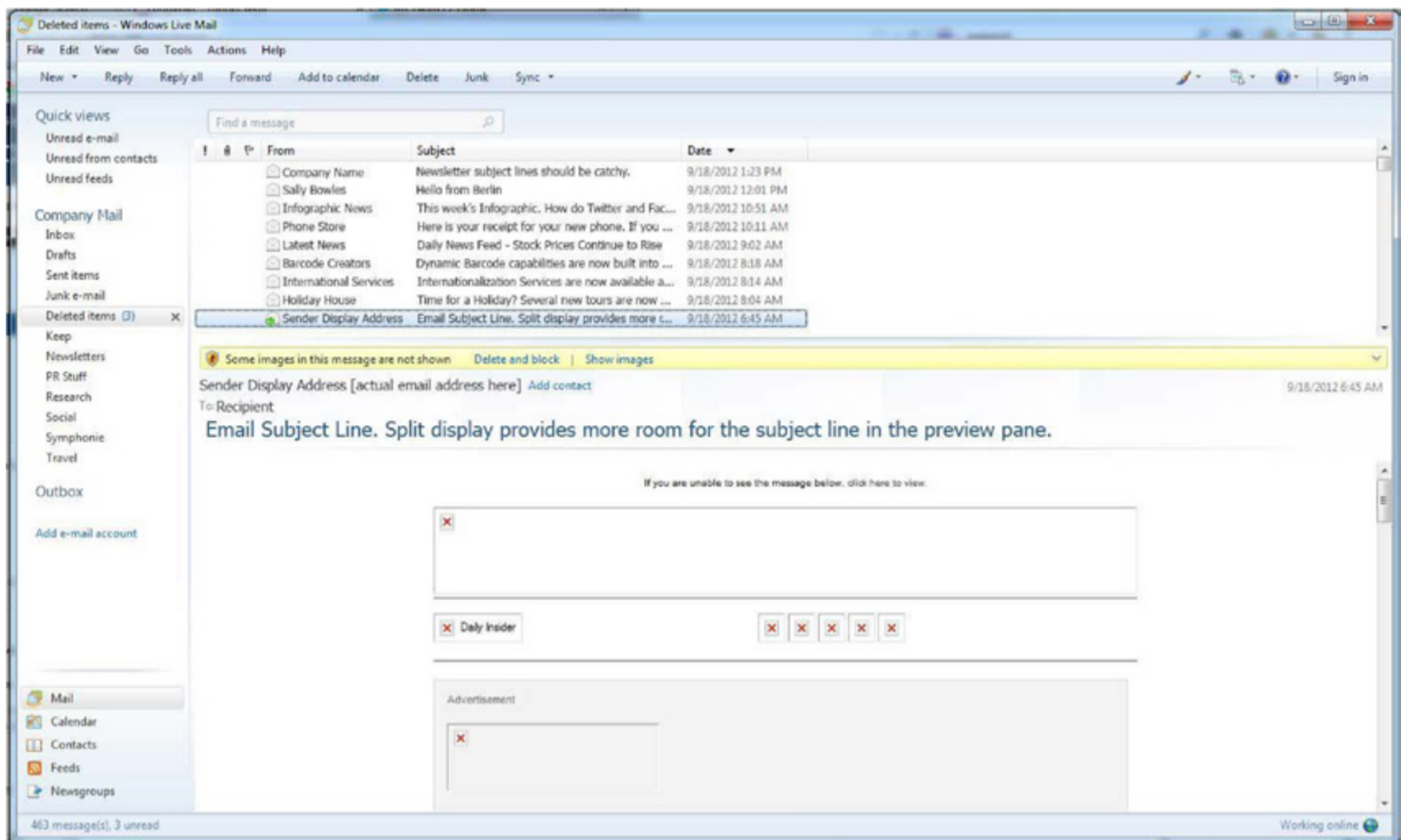
"Above-the-fold" is an old concept. It dates back to the days when people got their news from actual newspapers. At newsstands, newspapers are normally folded in half, with the top half showing. Whether or not someone buys a paper is often based on what they see on the top half of the newspaper. That is the reason for the large, enticing headlines, and photos. Marketers discovered that a similar phenomenon occurs when people view web pages. It has become a driving principle in Search Engine Optimization (SEO) with everyone jockeying for the top spot on page one.

The same holds true for email. The default set up common to many email viewers is to show a preview of the selected email in a separate window either below or in-line with the email. Unless it's a very short email, some of the message will not appear in this window. What does appear is the portion that would be considered "above the fold." Here is where your most compelling message(s) should appear. Many people will only

\* **Reputation score:** An accrued score of negative and positive factors that determine whether or not a mailing should go to the Inbox or the Bulk folder. To learn more about Reputation scores, please see our white paper *Deliverability Enhanced* in the Resources section of the Goolara website.

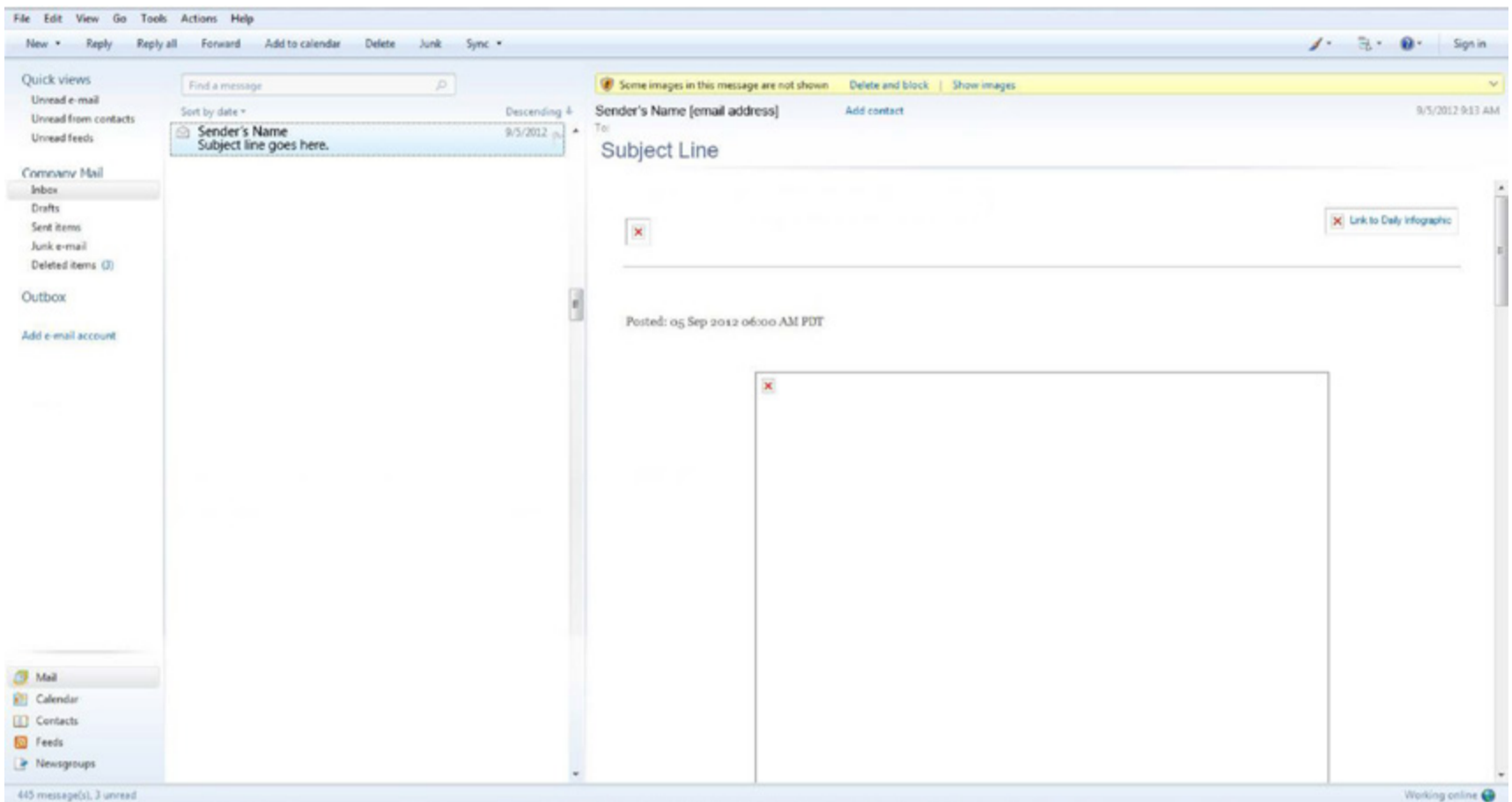
view what they can see in this window before making their decision whether to read further or delete the email, so make it count.

Here is an example of one of the problems posed by the combination of email client limitations and poor placement of the images. This email contains quite a bit of text, but because the recipient is viewing it in a horizontally split email frame with the images turned off, the results are unsatisfactory. The addition of a third-party advertisement near the top of the page doesn't help either.

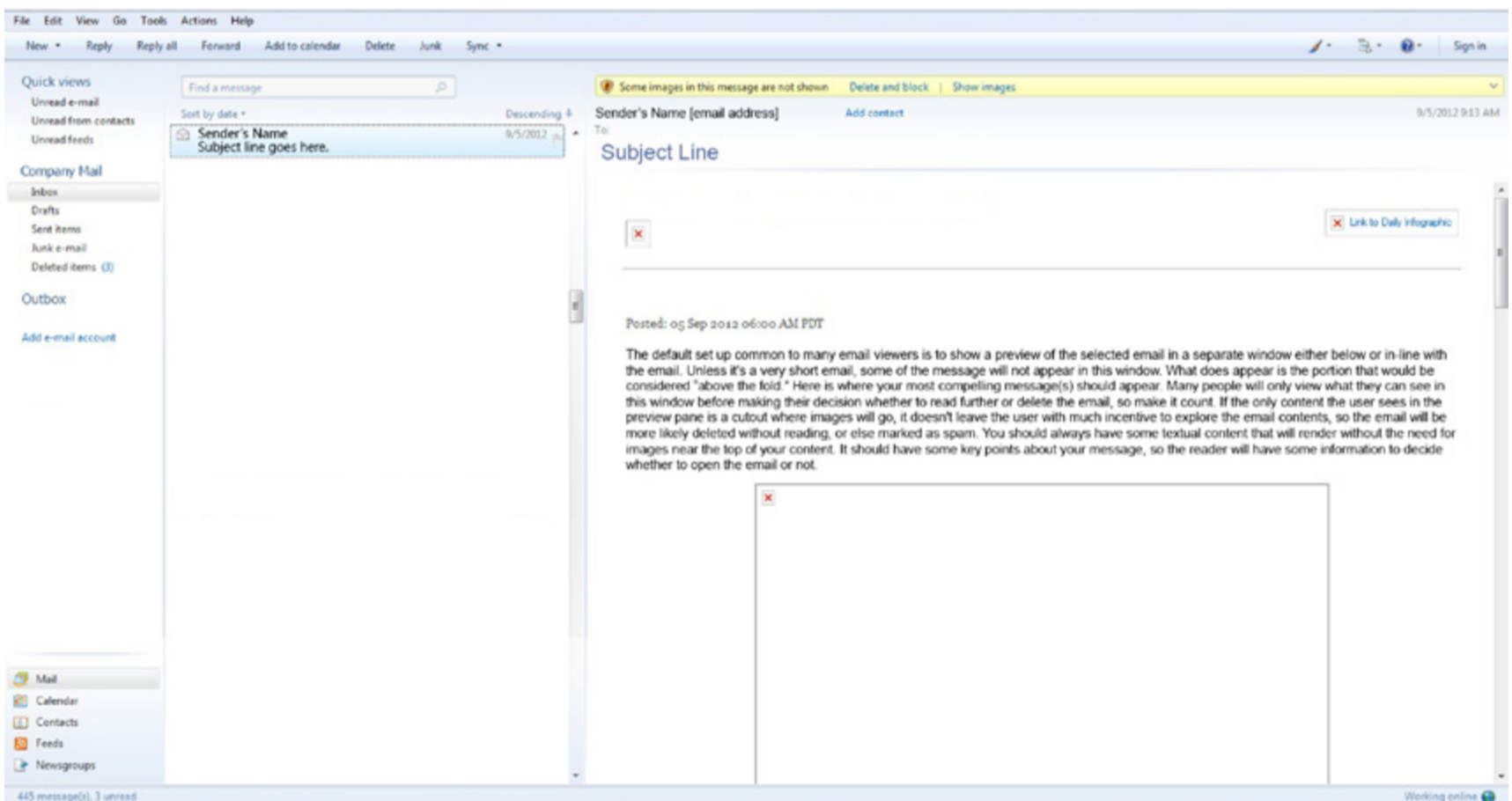


If the only content the user sees in the preview pane is a cutout where images will go, it doesn't leave the user with much incentive to explore the email contents, so the email will be more likely deleted without reading, or else marked as spam. You should always have some textual content that will render without the need for images near the top of your content. It should have some key points about your message, so the reader will have enough information to decide whether to open the email or not. As an example, take a look at the email shown on the next page.

This email only contains one image, but it is a big one. If you scroll down below the image box, there is text on this page, but the recipient doesn't see anything at first.

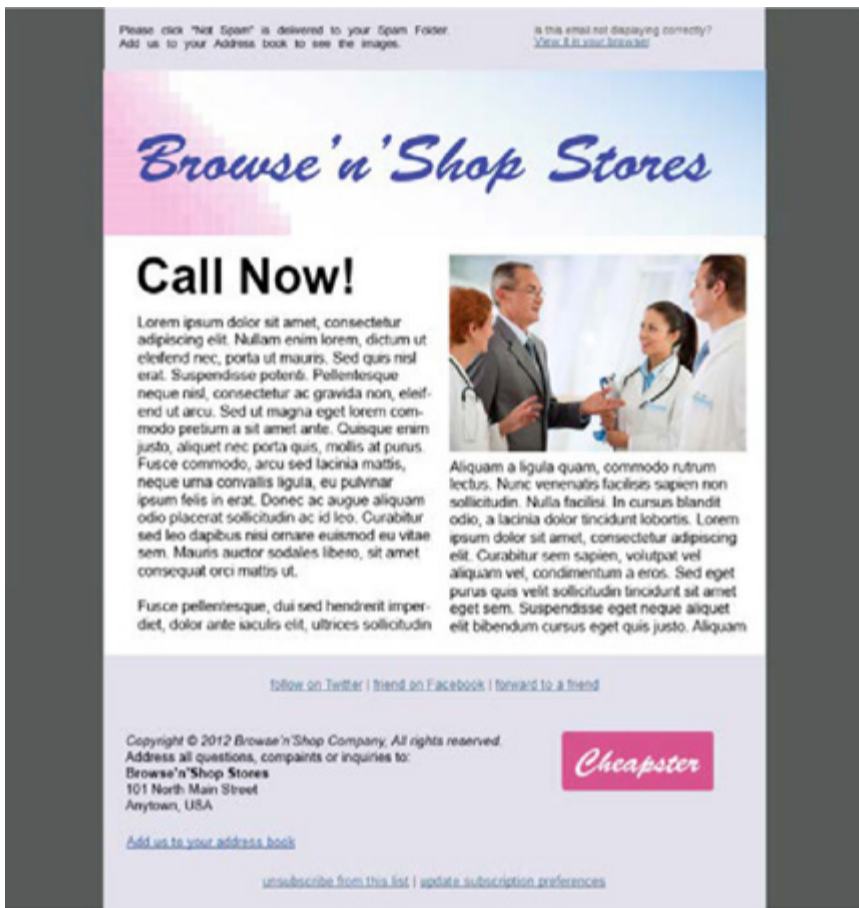


Now here's the same email with the text *before* the image:



The text tells the recipient what to expect from the image, which increases the chances of the recipient choosing "Display image," and the potential for a clickthrough, even without the image.





## Don't look like a mass email

Some cut-rate email service providers automatically insert their own preformatted headers and footers into any content you create. The end result is bad. No matter how carefully you write and design your email, the generic ISP information makes the end result look mass-produced. They do this to make sure the things required by CAN-SPAM are included in the email, such as the physical address information and the unsubscribe link. This is especially true with email marketing software that offers a free version with their logo on the page. While this may be acceptable for community center newsletters, it is not acceptable in a business environment. It's a bit like wearing clothing with the price tag still on it.

With better email marketing software you have complete control over the email contents. The wording used for the unsubscribe link should be up to you, and there is no reason that you should be limited to a preselected list of possible colors and styles.

The surprising thing we find is when customers move from these kinds of templates to better email marketing software, but want to reproduce the look of a mass email. They create header and footer borders that surround their content, making the email have that "place content here" feel. The more you make your email look like a mass mailing, the less you are telling the recipient that they are unique and deserve better attention.

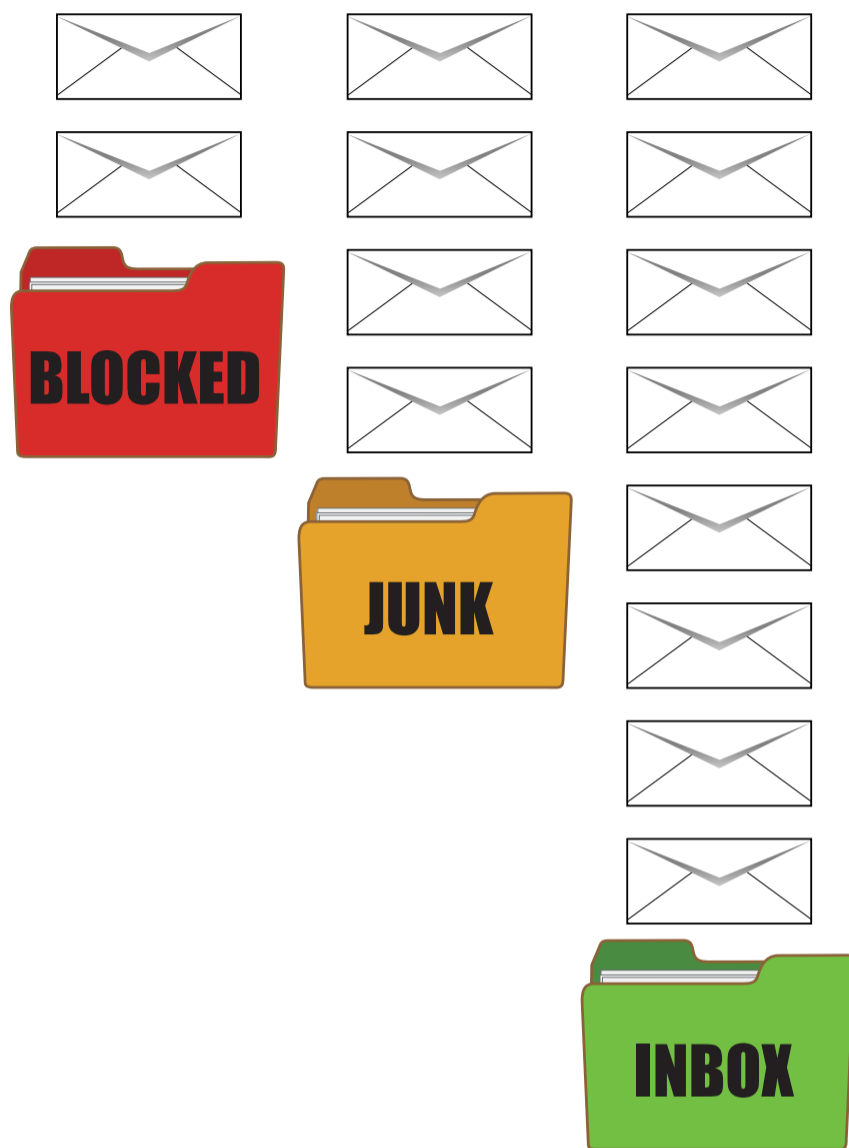
If you want engaged users, don't send them content that makes them think they are just one of many mass recipients.

## Centering content

Many content creators want the HTML centered on the page. This is possible, but needs to be done with HTML commands that will work across ISPs and email clients. A common way to do this is with a command in the Body tag, but remember that this tag will be stripped out by most ISPs, removing your formatting. It is recommended to create an HTML table, and then use centering commands in the cells of that table.

# DELIVERABILITY CONSIDERATIONS

---



There are many factors involved in email deliverability. Understanding these factors and what you can do to control them will go a long ways toward ensuring that your emails are delivered to the inbox and read by the recipients. We've covered most of these topics in our Deliverability Enhanced white paper. There are a few things, however, that might be considered design issues, and are important enough to warrant inclusion in this Best Practices guide.

## CAN-SPAM

The law requires, among other things, a way to unsubscribe (which must be complied with within ten days), the senders physical address (as text, not as a graphic) and a display name (sometimes called a "Friendly From") that is not misleading. It should be noted that these regulations only apply to email with the primary purpose of advertisement or promotion of a commercial product or service. They do not apply to transactional

emails since advertising and promotion are not their primary purpose.

You must comply with these rules for email delivered into the United States, and different countries around the world have different laws, but few are so easy to comply with as is the US CAN-SPAM act.

Ideally, as we have mentioned before, your unsubscribe information and your physical mailing address should be formatted by you, using your look and feel, and not inserted automatically by some software that knows nothing of your design.

Finally, just to be clear, the unsubscribe and address information cannot be included only in an image. They must

be within the text itself, so every recipient can see it without requiring images to be enabled. And they must be easily readable by an average person, so they should be in the same font, at the same point size, as the other text you display around them.

## CASL



Enacted July 2014, the Canadian Anti-Spam Law (CASL) puts restrictions on any email intended for commercial profit and has many similarities to the U.S. CAN-SPAM act. As with CAN-SPAM, there must be an easy way to unsubscribe, but the physical address requirements are less stringent. CASL is a bit more stringent about what is considered a commercial electronic message. It doesn't

distinguish between a mass mailing and an individually-sent email, and, according to at least one Canadian law blog, even a request for college funds sent to your cousins could constitute a violation. Unlike CAN-SPAM, it looks as if Canada is going to actually try and enforce the law, and in 2017, the law is scheduled to expand so that anyone can bring a CASL violation before the courts. For now, if you're following the regulations put forth by the CAN-SPAM act, and your recipients have all opted in, you should be in the clear. One of the more problematic features of the law is the requirement for email senders to keep an actual record of all opt ins, which has prompted some to send re-opt-in requests to their subscribers.

### A Clear Unsubscribe message

Somewhere on the page, usually at the bottom, it is a Best Practice to include information that explains why the recipient is receiving that email, along with the Unsubscribe link. This helps with deliverability and makes you seem more legitimate. If a deliverability specialist contacts an ISP to work on deliverability issues, the ISP



***“If you offer multiple topics for unsubscribing, it is a best practice to also offer users a single click method to unsubscribe from all lists.”***

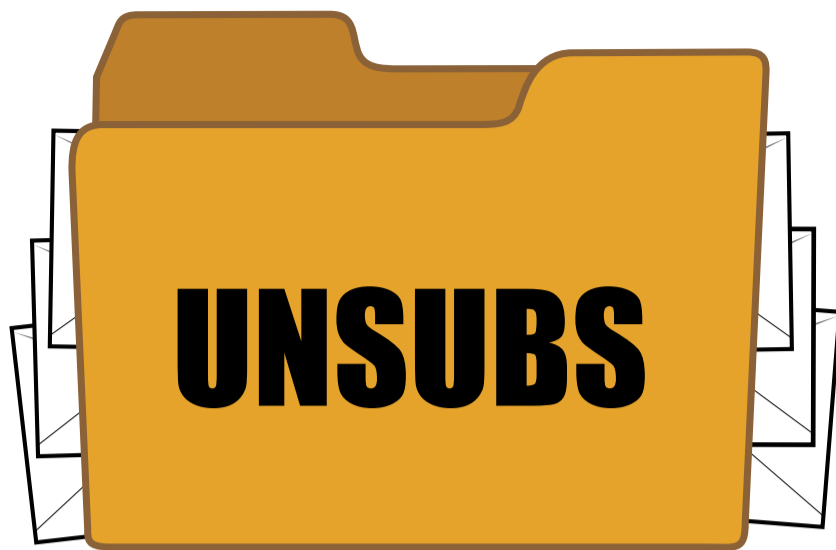
will often ask when the subscriber provided their information, from what website, and using what IP address. If the email content itself has the date the person subscribed, and the website that was used, it will simplify some of the work for the ISP, and gives users more of a sense of commitment to the email. Users are a little slower to hit the spam button if they see the details of when and where they requested the email. It helps establish you as a reputable marketer if you provide these details within the email itself.

No marketer is eager to have users unsubscribe, but it is important to recognize that an unsubscribe request from a user is far better than a spam complaint, or worse, a blacklist filing. Attempting to keep users from unsubscribing by making the unsubscribe link small, not clearly labeled, or hidden far below the text just increases the chances that the user will mark the email as spam. When working on deliverability issues with an ISP, they will look at the contents of the email, and if they don't think the unsubscribe link is clearly labeled and made accessible, they will be very unlikely to help with any deliverability issues.

We even heard of a case where an ESP told a marketer that he would get better deliverability if he converted his address and unsubscribe information into an image—a clear violation of CAN-SPAM. We can't imagine much worse advice than this. If your ESP ever tells you to convert your CAN-SPAM compliance information into an image, it's time to find a new ESP.

Every spam complaint hurts your Reputation Score, but unsubscribes do not. Making it easy for uninterested recipients to get off your list is actually a good thing for you, as ISPs look at the number of unengaged recipients and the number that mark your email as spam as part of their Reputation Score calculations. If the user doesn't want your email, let them go easily and without any resistance. If they change their mind they can always sign up again, but once they are marked as a spam complaint they shouldn't be allowed back into your email distribution.

If you offer several topical areas for users to unsubscribe from it is very important that you make it clear in the unsubscribe area what topic they are unsubscribing from, and what they will continue to receive. One of the frustrations of users is that they select to unsubscribe, but continue to receive email. This



often happens when the user is subscribed to multiple lists, and may not know that the unsubscribe only removed their email address from one of the distributions. Some users get fed-up with this and always hit the spam button instead, which removes them from all your distributions, whether the user wanted this or not, and hurts your Reputation score with a spam hit.

If you offer multiple topics for unsubscribing, it is a best practice to also offer users a single click method to unsubscribe from all lists. This means that you would have two links, one for the current topic, and one to remove from all topics.

We recommend that you include a sentence in your unsubscribe line that tells users exactly what they are unsubscribing from. Something along these lines:

“This distribution is being sent to those who have subscribed to our East European News list. If you want to unsubscribe from this distribution, click [here](#). Your unsubscribe request will not affect your subscriptions to other lists. If you want to unsubscribe from ALL lists we send, you may click [here](#).”

### **Offer a view-in-browser option**

Somewhere near the top of the page (although, perhaps not the first thing, for the reasons already mentioned), you should include a “View in Browser” link. Many email clients are full of ads and other distractions that a user may not want to deal with, so being able to jump to a web page that simply shows the content of the email can be an advantage. Additionally, web pages will not block images, so it is a way that users can enable images without following instructions on their email client.

Keep in mind that the view-in-browser functionality causes some issues for dynamic content with some email marketing software. Some systems won't render the dynamic content properly when the email is viewed in a browser. Others will revert to an unpersonalized version that the marketer must also create. Good email marketing software, such as Symphonie, will render the content exactly as the recipient received it, in its personalized form.

**TIP: Use a preference center**

When users go to unsubscribe, or to change which topics they receive, it is ideal if you can take them to a screen that offers them options to select from. You can show which topics you currently have as options for the user, and which ones they are currently subscribed to. You can offer options for message frequency, if this is something you can manage in your marketing program. It may be that a user decides to remain subscribed when they are reminded of the description of the topic area, they may choose to only unsubscribe from one topic, or, better yet, they may get interested in a different topic and subscribe to that one.

A good preference center may take some effort to build, but it can help keep more of your recipients engaged with your company. For legal as well as Best Practices reasons, it must be kept simple to use. If the user is there to unsubscribe, that option should be the default one, requiring no other selections and a single button to execute the unsubscribe. If users get confused, or are uncertain how to unsubscribe, not only will you risk the user returning to their email to mark it as spam, you are risking a lawsuit for not complying with CAN-SPAM.

**Create a text section**

Years ago it was critically important to send email that includes a text-only section for those email clients that couldn't read HTML. Remember dial-up AOL and Web-TV? Luckily those days are nearly gone. Most every email client created in the last ten years can display HTML. New technology, such as mobile devices, many times don't even offer an option to view the plain text, defaulting to always show the HTML.

Although the plain text version of your email will not be seen by many users, it can still be useful if your content has too many images. We discussed previously that there are many reasons why you should limit the number and size of images used in your content, but if your designers insist on the use of many images, a plain text section can help with deliverability. The ISPs want some text to read to know what your email is about, and since they cannot easily "read" your images, giving a plain text section provides them with text to analyze.

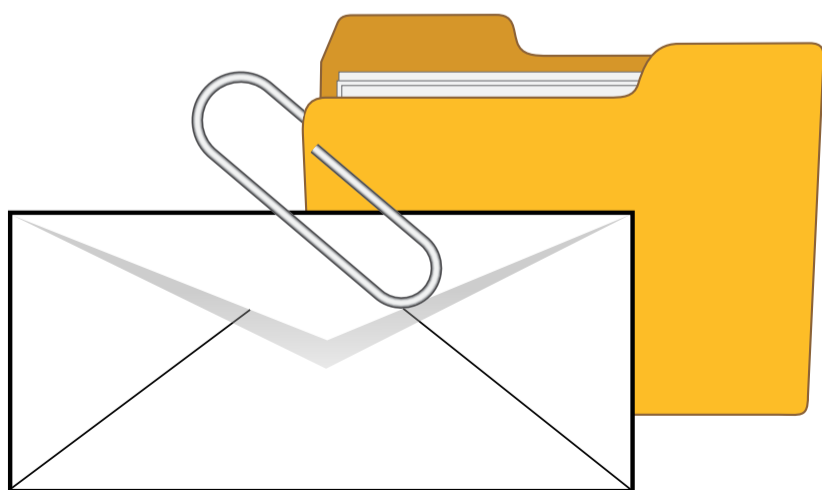
It is recommended that you create your plain text section from the same content used in the HTML section. The ISPs look to see if the same words appear in the HTML and plain text sections, and if they aren't the same or similar, the ISP may conclude you are trying to fool their filters with your plain text section.

Some marketers are very resistant to creating a plain text section, for understandable reasons. It is very difficult to make plain text look good, and convey the marketing message as efficiently as can be done with HTML. However, if you recognize that the number of recipients that see the plain text will be small, and that the main purpose is to help with deliverability, it may lessen the concern about the appearance of the plain text.

### **Get Them to Reply**

According to the top ISPs, one of the strongest boosts to deliverability is when the recipient replies to your email. As discussed in our white paper, *Deliverability Enhanced*, many factors are used to build a Reputation Score. One of the best factors for ensuring deliverability is when the recipient replies to an email. It isn't practical, in a sales environment, to create every mailing with the express purpose of having recipients respond to it. In fact, many promotional mailings are intentionally set up to use a "donotreply" email address to avoid unwanted responses. But it is worth considering as an occasional tactic to ensure your deliverability numbers stay high. Just be sure that you have a good, organic reason for the recipients to respond. Otherwise you may find the technique backfiring on you in a big way if your recipients decide you are simply being sneaky.

### **Don't send attachments**



Goolara Symphonie will let you send email with attachments, but we don't recommend it. Like JavaScript, attachments often fall prey to hackers and spammers, and any value this feature once had has been destroyed by its misuse. The rule of thumb these days is: Don't open an attachment unless it is something you are specifically expecting. A better approach is to provide a link where the recipient can either view or download the file.

# TESTING

---



A good marketer recognizes that there is only so much she can know about how an audience will react. There are hundreds of examples of marketers discovering after testing that the email they thought would be a hit fell flat, while the one they created as an afterthought produced higher open rates and clickthroughs. During the 2012 presidential election, the staff on Obama's marketing team placed bets on which subject line would yield the best results, only to find they were wrong more often than not. The importance of testing cannot be overstated, and is, therefore, a best practice

for the conscientious marketing professional.

## **Perform split tests**

Sometimes the best marketing person in the world cannot accurately predict how an audience will react to a message. How many huge marketing campaigns from well known advertising agencies, costing millions of dollars, do you know of that were basically flops? It certainly happens, so you should take advantage of the real-time nature of email to test your subject line ideas. Create two or more options for subject lines and send them out as a A/B split test. Give recipients a day to receive and open the email, and then send the version with the most effective subject line to the remaining audience. There is no other form of marketing that can give such quick, real-time results, so you really should use this feature to maximize your marketing efforts.

Above all, the subject line should, in some way, reflect the content of the email. CAN-SPAM actually requires this, but using a subject line that is not relevant to the content is also a good way to alienate your audience and increase your risk for getting tagged as a spam sender.





Whether you use a long subject line or a short one, special characters or English ASCII safe ones, you should still test an email on a smaller sub-set of recipients before you send it to everyone on your list.

### **Send tests**

Most email marketing software offers the ability to see what your email will look like in various email viewers. These are very good for getting an idea of how your email will look when it is finally sent, but the fact is, the only way to be absolutely

sure that what you see is what you mailed, is to actually send the email and see what arrives. Email providers, such as Google and Microsoft, are constantly tweaking their interface and the smallest thing can trip you up. There is really no better way to catch some problems ahead of time than by an actual mailing. We therefore recommend setting up accounts with the major email services, such as Gmail, Yahoo, Hotmail/Outlook/LiveMail, AOL, and any other that has many of your recipients using it. In some cases, these might be local services.

Testing becomes especially important if there are aspects of your email that, for whatever reason, are not appearing properly in your email marketing software. When this happens, it is always safer to make sure that everything is correct in your code. Some services, such as those from Microsoft and Google, are smart enough to throw away any information that they think is unnecessary, but not all email providers will do this, and an email that looks okay in one email provider's site may turn into gibberish in another.

Sometimes, you will get some clues that there is something wrong in your email design. If, for instance, the hierarchical formatting of the HTML suddenly disappears and the code is bunched up into one big ball of code, you can bet there is a problem in the code somewhere. In Goolara Symphonie, the first thing to do at this point is to run the Content Checker. If there are problems with the HTML, the Content Checker will find them.

# MOBILE CONSIDERATIONS

---



Since the introduction of the first version of this guide back in 2012, no segment has grown more than the mobile market. Every year, more and more people are using mobile phones and tablets to check and send email. Nowadays, people read their email everywhere. They read it on their phones while heading to work, on their laptops at the local coffee shop, and on their tablets while waiting for lunch. Some statistics go so far to say that by the end of 2014, over half of all email was opened on a mobile device. While statistics like this can be tricky to verify, there

is no doubt that more email is opened on mobile devices now than ever before, and you are going to want to make sure that your email is easy to read on these devices. So what's the best approach?

Some vendors try to determine whether a user is opening emails on a mobile device, and if so, allow the email marketer to target them differently. Our experience is that people who have mobile devices rarely use them exclusively. They may triage their email on the mobile device during the day, but look at other emails in more detail at home on their desktop machine. And even if someone has shown a pattern of opening on desktops or mobile devices, that pattern may suddenly change, and then the content would look worse on that other device.

## Responsive Design

It's for this reason that we've seen responsive design become more popular in email design over the past few years. With Responsive Design, you set up a media query that determines the size of your viewing window. If the size falls below a certain pixel width, the page reorganizes itself and its elements into a format that is easier to read in a small viewing area. This is particularly handy for reading email on smart phones, where a design runs the risk of becoming unreadable when it is viewed on a such a screen. Currently, not all platforms support

*“When Responsive Design won’t work, you need to make sure that your mailing is still readable. This is the thinking behind Mobile First.”*

responsive design, although more and more are beginning to accept it. Unlike other style attributes, there is no way to inline media queries, so an email reader that ignores style tags won’t render email responsively.

Responsive Design can be very satisfying when it performs the way it should, but the learning curve is steep, and thorough testing across as many email clients as possible is mandatory (see our Responsive Design guide available in the Resources section of the Goolara website for information on this subject). When Responsive Design won’t work, you need to make sure that your mailing is still readable. This is the thinking behind “Mobile First.”

### **Mobile First**

For those of you familiar with the basics of programming, it is the same principle as progressive enhancement—you make sure that your mailing is readable on the smallest device under the worst of circumstances (which is to say, no responsive design available) and then enhance the email from there. Relying entirely on responsive design to make your mailing readable on a smaller device is a recipe for disaster.

You should apply the Mobile First principle to your mailings whether you plan to use responsive design or not. You don’t absolutely have to use responsive design to make your mailings readable on small mobile devices. Large retail sellers, such as Bed Bath & Beyond, Kohl’s, New Egg, Macy’s, and others skip the media queries in favor of designs that are still readable when scaled down. Our general advice is to reserve the labor intensive responsive design to those mailings that use a standard template—newsletters and such—and use a mobile-readable design for the ones that don’t use a common template.

There is also the question of what constitutes a mobile device, as different sized devices are being released. Is an iPad a mobile device? Its screen is as big as many desktops. Are tablets, ultrabooks, and other models mobile devices? If you had to pick one screen size to represent any device you label as mobile, what size would that be? The best answer to this is to pay attention to the feedback from your clients, but here are a few guidelines that we’ve found to be consistently accurate.



## Keep the email 500-650 pixels wide

How wide should your email be? That is a question which has always been difficult, as screen resolutions have expanded over the years, but now are shrinking some due to the use of mobile devices. Your content may be opened on a mobile device that has only 300 pixels across, and it may be opened on a desktop computer with a screen that measures 1920 pixels or more across. With HTML you can control how wide an email will appear. You have very little control over the width of text messages, since, by their design, there are not any formatting commands available.

With a web site you can control the formatting of the page on-the-fly, allowing the user with a big screen to see more content than is shown to someone with a smaller screen. However, much of the logic to make this happen exists in JavaScript, which does not work in email, so your choices are limited. When you send a text message, such as a note to a coworker, you would normally not bother with HTML commands to limit the width of the text. If you write a 200 word paragraph that takes ten visible lines on your display, it may be opened by your coworker in a maximized window that shows the text spanning the entire page and only needs two visible lines. In standard personal email communications this doesn't seem wrong. However, if you don't limit the size for marketing messages, and the text stretches to the right side of the screen, it will likely appear incorrect, especially if any images are included, since images don't change in size.

Most marketers have settled on a content size of 500-650 pixels across. This looks reasonable on small screens, and on a very large screen, is not so narrow that it looks funny. By restricting the width of your emails to 650 pixels or less, you keep things in a readable format that works for mobile phones as well as desktop computers. If your email contains masthead or similar logo running across the top of the page, the 500 minimum ensures that the images will not be reduced in size to fit.



## **Keep links separated for fat fingers**

Clicking on links on a desktop is rarely a problem when you have a highly precise tool (a mouse) to make your selection. But a nearly invisible clump of tightly packed links on a phone might go unclicked, or the wrong link might be selected, muddying your analytic results. Users who get frustrated trying to click a link with their fingers may even give up and delete the email entirely. To avoid this, make sure your links are separated with

enough space so that a user attempting to click the link will not accidentally hit the others around it. According to Apple, the space needed for a finger tap is exactly 44 pixels square, so make this the starting point for creating your buttons. To accommodate this, you may need to reform portions of your email, especially the bottom part, where many marketers place a dense series of links.

## **Links on Mobile Phones**

It's worth mentioning at this time statistics still show that while many more people are reading their emails on mobile phones now, they are clicking on links less often with these devices, preferring to do any web browsing with larger formats. The number drops even further when you look at the statistics for people completing purchases on their phones. While it is certainly important to make sure your mailings are easy-to-read on mobile devices, keep in mind that the opening of an email on a phone might be only half the story. Of course, this may change in the future; no other aspect of marketing is changing as rapidly as that of mobile marketing.

# DYNAMIC CONTENT

---

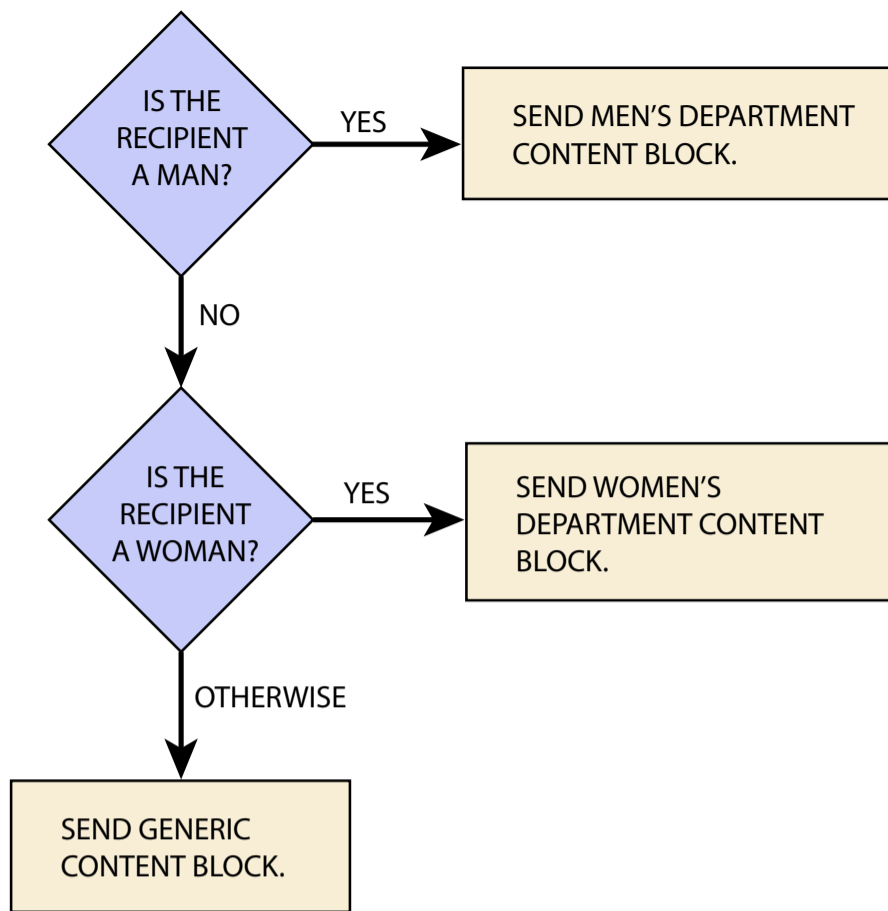


**I**f your email contains dynamic content, there are other issues that will affect design. Dynamic content can change the amount of text in an email dramatically. If one person is slated to receive a large block of content, while another is slated to receive no content in that section at all, the design of your email layout has to take this into account.

The simplest option is to include content blocks of roughly the same size for every person, so the email layout doesn't need to change. Those that have an appointment see the details of the appointment, and those that do not are shown text saying they don't have any currently scheduled appointments. This method works well when you have something to say for every possible audience, and makes the email layout easier. The dynamic content can be put into an HTML table cell, and you know that each substitution is the same size, so the layout of the email will remain consistent.

However, it is not always easy or ideal to show some content for every person. Maybe the people with appointments need enough text to provide the details that would leave the non-appointment emails with too much filler space. In these cases, you must work with your designer to create an HTML layout that can adjust to variable content. One solution is to put content in a table or other layout that is oriented vertically. If content is substituted, the content below the substitution is pushed down to lower in the page, and if no content is substituted, the text flows smoothly.

Laying out the content vertically can work well if you have a design that can stretch vertically without breaking the layout. It means you cannot have images or other areas that enter into the content area of the vertical column, and your layout has to be in columns of content. If you feature several columns, you must remember that content in fixed column one may not line up with the dynamic content in column two, and



unless you are careful, column two could be considerably longer or shorter than column one.

With good email marketing software you should be able to provide the full HTML of content to be substituted dynamically. This content can include HTML elements like tables or divs to give you the layout choices you need to format the content correctly.

The final layout option is to create HTML that can be substituted at specific points in the layout, and the browser will move the content around, as needed, to format for your content. For example, you might have a three-column design for a portion of the email, and then conditionally

substitute a new HTML table at the end of the columns that presents a new section of a one column layout, perhaps using a wide image as a separator. The image, textual content, and all of the HTML to make this possible would be substituted as one block, and those recipients who did not meet the criteria would not be shown anything there. But that would not lead to a blank space, as the HTML that followed the substitution location would still be rendered.

# SUMMARY

---

**B**est Practices means creating email that engages the recipient, making them want to open the email and click through on the links. Many factors go into what constitutes Best Practices but there are certain broad categories:

**HTML Code** – When email was first created it was restricted to basic ASCII characters. Many email service providers now allow for UTF-8 characters which broadens the capabilities of email text, although if you don't need the extra capabilities, it is still safer to use the basic character set for your chosen language.

Eventually, HTML capabilities were added to email, enabling the use of images and tables. Since most mailbox providers will override some of the pre-header information, it is safest to use only HTML that falls between the body tags. If you are planning to create responsive designs, you will need to add a section with style attributes at the top, but keep in mind that not all ISPs recognize this information. The more extraneous HTML you add to your mailing, the greater chance for conflicts between your code and the ISP's. Additional style information (such as media queries) should be placed in the <head> area or above the main copy. Avoid using standard header tags (h1, h2, etc.) as these will be interpreted differently by each email provider according to their settings. Tables are preferred over div tags and JavaScript should *never* be sent.

**Design for Subject Lines** – The success of an email can rise or fall on the strength of its subject line. When writing subject lines, the text at the beginning of the subject line is the most important, as is the text at the beginning of the email. These are the things that show up first. Start on the left when designing a subject line, to avoid having your main message cropped.

One way to add text the subject line display in an email client is through the use of a pre-header. As with the subject, you need to make sure that if the message is cropped, it doesn't lead to a misinterpretation.



You can often use special characters in subject lines, and some devices, such as iPhones, will actually translate these characters into small pictograms. More recently, emojis have become popular, although not all email clients recognize these characters. While some emojis are recognized by most email clients, you should still be careful not to build the comprehension of your subject line on an emoji. Also keep in mind that some emojis appear very different in different email clients, occasionally appearing to mean the opposite of what is intended.

A few ESPs, such as Goolara Symphonie, will let you use dynamic content in your subject lines, which can boost email open rates.

**Image Preparation** – It is a good idea to assume that the images won't appear since this is the default setting of many ESPs. All images should have alt tags that are descriptive of the content of the image. Well written alt tags can entice readers to allow the images to be displayed. Always start your email with some text. Do not use background images.

**Layout Considerations** – Always design for the top half of the email ("above the fold"). If your email has large images, make sure you've put some descriptive text above the image. Avoid, if possible, using ready-made templates.

**Deliverability Considerations** – Some design features, such as a non-image version of the company address on the email, and unsubscribe information, are required by the CAN-SPAM Act. Make sure that the unsubscribe information is clear and easy to use. A "view in browser" feature is a good way to engage your recipients. Although they are rarely used anymore, you should always create a text version of the email to be safe. Don't send attachments.

**Testing** – It is important to test your email before sending to the general public. With A/B splits, you can find out which subject lines or content works best. Always send test emails to accounts you've set up for this purpose. This is the surest way to catch potential problems.

**Mobile Considerations** – As more and more people use smart phones to read their email, it is important to design your email so that it can be easily read on these devices. Responsive

Design is one way to do this, but it is more labor intensive. “Mobile First” is an approach to design that says you should always make sure your email is readable on a mobile phone whether it is built to be responsive or not. Keep your images below 650 pixels wide. Responsive design allows email to change its display characteristics when it is viewed on a mobile phone, but not all email providers work with this feature. It is a good idea to set up your email for responsive design, but make sure that the message is still readable on a phone in case the responsive design features don’t work. A minimum of 44 pixels should be allowed for clicking on buttons with fingers.

**Dynamic Content** – When using dynamic content, design considerations become even more important than with non-personalized email. Allowance must be made for all options, so that if some information is missing for some people, the page layout isn’t affected adversely. Tables can help isolate these elements.

# REVISION HISTORY

---

## **OCTOBER 2017**

Revised technique for placing HTML in an ESP's visual editor.

Add emoji information to subject line section



## About Goolara

Goolara has been in the email marketing business since 2005. Symphonie, Goolara's premiere email marketing solution is available in on-premise and cloud-based, SaaS deployments. The powerful software features many advanced capabilities, such as full-featured dynamic content, transactional and triggered email, and customizable report generation features. It is easy to use and runs from a browser-based interface using Chrome, Firefox, Internet Explorer, or Safari. Goolara is headquartered in Moraga, California and can be found online at [www.goolara.com](http://www.goolara.com).

Goolara, LLC  
1030 Country Club Suite D  
Moraga, CA 94556  
Telephone: (510) 522-8000  
(888) 362-4575  
Fax: (510) 522-2457

Copyright © 2017 Goolara, LLC All rights reserved.

No part of the contents of this publication may be reproduced or transmitted in any form or by any means without the written permission of Goolara, LLC.

Goolara and the Goolara logo are registered trademarks in the United States, other countries or both. All Rights Reserved. All other company and product names and logos may be trademarks of the respective companies with which they are associated.

[www.goolara.com](http://www.goolara.com)