

Last modified January 23, 2009 | [www.epsilon.com](http://www.epsilon.com)

# **DREAMmail**

## **Advanced Messaging Guide**

### **Version 4.9**

## **Copyrights**

This documentation and related technology are governed by a user agreement and shall remain the sole and exclusive property of Epsilon. No part of this documentation or related technology may be used, reproduced, translated, displayed, distributed, disclosed, stored in a retrieval system or transmitted in any form or by any means without the written permission of Epsilon, unless otherwise stated in the user agreement. The information contained in this documentation is confidential and proprietary to Epsilon.

## **Disclaimer**

Epsilon does not warrant, guarantee or make any representations or otherwise concerning the contents of this documentation or the applicability thereof. Epsilon reserves the right to change the contents of this document at any time without prior notification of such updates.

## **Trademarks**

DREAMmail is a registered service mark of Epsilon.

# TABLE OF CONTENTS

<b>Preface to the Advanced Messaging Guide .....</b>	<b>v</b>
<b>Overview of Advanced Messaging .....</b>	<b>1</b>
What is Advanced Messaging? .....	2
Using Content Segments to Personalize Messages.....	5
<b>Working With Script Content Segments .....</b>	<b>7</b>
Process of Creating Script Content Segments .....	8
Creating Script Content Segments in <i>DREAMmail</i> .....	9
Creating Script Content Segments in <i>DMConnect</i> .....	11
Referencing a Script Content Segment in <i>DREAMmail</i> .....	12
Referencing a Script Content Segment in <i>DMConnect</i> .....	13
Editing and Deleting Content Segments .....	14
<b>Syntax of the DREAMmail Scripting Language.....</b>	<b>15</b>
Elements Used to Create Script Content Segments.....	16
Numeric, Relational, and Date/Time Operators.....	18
Formatting Text and Messages .....	23
IF Statement .....	25
SWITCH Statement .....	27
Multiple Conditions in Statements .....	29
VARIABLE Statement .....	33
<b>Rich Media in Dynamic Personalization .....</b>	<b>37</b>
Flash Compatibility with Email Applications .....	38
Flash Compatibility with Web-Based Email Services.....	39
Inserting Flash into an HTML Message .....	40
Personalization of Flash Content .....	42
Tracking Links in Flash Movies.....	44
Inserting Flash with Personalization and URL Tracking...46	
<b>Example of Dynamically Personalized Messages.....</b>	<b>48</b>

Information for Personalizing the Example Messages.....49

Example of Entertainment Content .....52

Content Segments Used in the Example Messages .....56

Results of the Dynamic Personalization Scenarios .....70

Appendix - Scripting Language Limitations .....72

Index.....73

# PREFACE TO THE ADVANCED MESSAGING GUIDE

## About this guide

This document is intended for content developers who are responsible for creating personalized messages. It is part of the set of documents that describe *DREAMmail* and explains the two most commonly used advanced messaging techniques: scripting and implementation of Macromedia Flash.

Advanced messaging can be created in either the *DREAMmail* application or using *DMConnect*, the *DREAMmail* API. This document assumes that you are familiar with the basic concepts and terminology in *DREAMmail*, and that you have experience using *DMConnect*. This document also assumes that you have basic programming or scripting skills.

For more information about *DREAMmail*, consult the online help and for information about *DMConnect*, see the *Programming Guide*.

This guide is divided into the following sections:

- [Overview of Advanced Messaging](#) introduces the concept of dynamic personalization, what it can be used for, and its limitations.
- [Working With Script Content Segments](#) explains what content segments are, how to create them, and how to put them in your messages.
- [Syntax of the DREAMmail Scripting Language](#) explains the scripting language is, how it works, what functions are available, and how to write scripts.
- [Rich Media in Dynamic Personalization](#) explains how rich media, in particular, Flash, can be included in messages and can contain dynamic personalization
- [Example of Dynamically Personalized Messages](#) shows examples of the *DMConnect* requests that can be used with scripts to produce personalized messages based on user profile fields.
- [Appendix - Scripting Language Limitations](#) explains some of the limitations that you must take into account when creating script content segments.

# OVERVIEW OF ADVANCED MESSAGING

Messages targeted to a subscriber's specific needs and interests generate higher response rates than generic messages. From simply addressing a subscriber by name to customized offers or editorial content, advanced messaging enables you to peak interest and increase sales.

By using scripting languages and rich media in messages, you can leverage data that you have collected, including your subscribers' demographic information and preferences, to create more dynamic, personalized, and focused messages. When a message is sent, DREAMmail matches the specific conditions contained in the dynamic personalization script with each subscriber in your mailing list, according to his or her profile data. The relevant content is inserted into the message, and the subscriber receives a personalized email.

This section discusses the following topics:

- [What is Advanced Messaging?](#)
- [Using Content Segments to Personalize Messages.](#)

# What is Advanced Messaging?

Advanced messaging is the process of creating personalized messages based on subscriber profiles and preferences. This personalization can range from something as simple as putting in a subscriber's first name in the greeting, offers based on demographics or subscriber preference, or interactive ads customized for a subscriber.

Another way of personalizing messages is to deliver rich media content that enhances your marketing promotions. This typically involves sending a message that includes streaming audio or video. In [DREAMmail](#), you can place Flash content directly in your email messages and then pass variables that are personalized for each subscriber, to the movie.

This section discusses the following topics:

- [Uses of advanced messaging](#)
- [Creating a more interactive experience](#)
- [What is dynamic personalization and how does it work?](#)

## Uses of advanced messaging

Advanced messaging can be used to target specific subscribers by:

- personalizing offers based on geographic location
- personalizing offers based on demographic information
- combining multiple messages in one email
- adding personalized or relevant URLs to messages.

**Personalizing offers based on geographic location:** Using city, state, or country information you can:

- market product lines that vary by region
- direct subscribers to a specific retail outlet
- test offers on a regional basis.

**Personalizing offers based on demographic information:** Using demographic information you can:

- market product lines that vary by gender or age
- send out specialized offers based on special occasions as birthdays or anniversaries
- focus offers based on family status.

**Combining multiple messages in the same email:** Dynamic personalization can be used to assemble multiple blocks of content within the same email, based on subscriber preferences. This enables you to:

- build a newsletter or message unique to each subscriber's preferences
- combine several relevant offers in a single email
- help reduce list fatigue and lower deployment costs.

**Adding personalized or relevant URLs to your messages:** In addition to content, you can include URLs to specific pages on your website in a personalized email. This enables you to direct subscribers:

- straight to the purchase point on your website
- to a personalized web experience.

## Creating a more interactive experience

Rich media, whether it be in the ads you insert or the content of messages, tends to be more engaging than traditional GIF images and has been shown to generate higher click rates than static content. Rich media in messages enhances your subscribers' experience with games, audio, and so on. Rich media can also help to present your company as a leader in new technologies. Both advertisers and web users have come to expect rich media as part of the web experience.

## What is dynamic personalization and how does it work?

Dynamic personalization is a method of customizing email messages based on subscriber information that is stored in the *DREAMmail* database. You can use dynamic personalization to target messages to a subscriber's personal interests, therefore making a message more relevant to a given subscriber. You can vary the complexity of dynamic personalization, from addressing a subscriber by name to including personalized sales offers or editorial content.

Dynamic personalization includes profile fields and content segments, which you can include in the body and subject line of a message by inserting references to them. When the message is sent, *DREAMmail* replaces the reference with the content of the profile field or content segment that is stored in the *DREAMmail* database.

**Profile fields:** Profile fields are fields in the *DREAMmail* database in which you can save subscriber information, such as first name, last name, email address, mailing address, gender, and so on.

**Content segments:** A content segment is a unit of information that you can include in a message. Content segments can contain any type of information that you normally include in a message such as URLs, profile fields, references to other content segments, and in some cases, programmatic logic.



A content segment can be either static or dynamic. ***Static content segments*** contain a single block of editorial content that is included in **all** messages that reference the content segment. ***Dynamic content segments*** can contain multiple blocks of editorial information along with a series of condition statements or scripting that determine which block is to be inserted into messages that reference the content segment.

You create and submit content segments using either [DREAMmail](#) or *DMConnect*. [DREAMmail](#) automates the entire process, matching specific content with each subscriber in your list, based on their profiles. When the message is sent, [DREAMmail](#) inserts the appropriate content into the message to create personalized email messages.

# Using Content Segments to Personalize Messages

There are three types of content segments, all of which allow for some degree of personalization.

This section discusses the following topics:

- [Static content segments](#)
- [Conditional content segments](#)
- [Script content segments](#).

## Static content segments

Static content segments enable you to specify a single piece of information that is displayed in all messages that reference the segment. For example, if you want every subscriber to receive the same 10% discount, you can create a static content segment. (Static content segments can contain profile fields that are dynamically personalized.) If, however, you want different subscribers to receive different discounts based on their demographic information or preferences, create a conditional content segment or a script content segment. You can create static content segments in either [DREAMmail](#) or *DMConnect*.

## Conditional content segments

Conditional content segments enable you to insert information into messages according to subscriber demographic information or preferences stored in your database. For example, if your database stores subscriber addresses, you can create a conditional segment that sends one type of information to subscribers in Miami and a different type of information to subscribers in New York. You can also send default information to subscribers who do not live in either location.

A conditional content segment allows you to test multiple conditions (for example, `IF City equals Montreal`). However, if you plan to create a conditional content segment that tests more than 15 conditions, Epsilon Interactive recommends that you create a script content segment instead. You use the *Content Manager* in [DREAMmail](#) to create conditional content segments.

## Script content segments

Script content segments are more complex than conditional content segments because they enable you to use a range of programming logic when you create a content segment. In addition to standard programming statements such as `IF...ELSE` and `SWITCH`, you can use numeric, relational, and date and time operators to control the dynamic information that is included in your messages. You can create script content segments in either [DREAMmail](#) or *DMConnect*.

**Related topics**

[Creating Script Content Segments in DREAMmail](#)

[Creating Script Content Segments in DMConnect](#)

[Content Segments Used in the Example Messages.](#)

# WORKING WITH SCRIPT CONTENT SEGMENTS

Script content segments consist of scripts you create using the [DREAMmail](#) scripting language. Each script contains blocks of editorial content and programming statements that determine what information is inserted into messages that reference the content segment. Editorial content can consist of text, URLs, profile fields, and HTML code.

For example, a dynamic content segment might contain multiple sales offers and condition statements that query a subscriber's geographical location to determine which offer to insert into a message sent to the subscriber. For examples of script content segments, see [Content Segments Used in the Example Messages](#).

Content segments are created and maintained using the *Content Manager* in [DREAMmail](#) or in [DMConnect](#).

This section discusses the following topics:

- [Process of Creating Script Content Segments](#)
- [Creating Script Content Segments in DREAMmail](#)
- [Creating Script Content Segments in DMConnect](#)
- [Referencing a Script Content Segment in DREAMmail](#)
- [Referencing a Script Content Segment in DMConnect](#)
- [Editing and Deleting Content Segments](#).

# Process of Creating Script Content Segments

Creating a script content segment for dynamic personalization involves six stages.

Stage	Description
1	<b>Choose the profile fields that you want to use in creating a script content segment.</b> Decide which profile fields in your database you need to use to customize your message; for example, gender or location.
2	<b>Create an outline of the script in a series of logical statements.</b>
3	<b>Rewrite the outline in the DREAMmail scripting language.</b> For information, see <a href="#">Syntax of the DREAMmail Scripting Language</a> .
4	<b>Create the script content segment using DREAMmail or DMConnect.</b> For information, see either <a href="#">Creating Script Content Segments in DREAMmail</a> or <a href="#">Creating Script Content Segments in DMConnect</a>
5	<b>Create the message.</b> Create the message using either <i>Message Manager</i> in DREAMmail or <i>CreateMailing</i> request in DMConnect. <b>Important:</b> If you are using scripts across several similar messages, you may want to set up a message template instead of creating individual messages. For more information about templates, see the DREAMmail help system.
6	<b>Include the content segment in the message.</b> Using reference tags, define locations in the message where you want the dynamic personalization scripts to be inserted. When the message is sent, DREAMmail queries the subscriber's profile data and inserts the correct content into the message. For more information, see <a href="#">Referencing a Script Content Segment in DREAMmail</a> or <a href="#">Referencing a Script Content Segment in DMConnect</a> .

## Related topics

[Creating Script Content Segments in DREAMmail](#)

[Creating Script Content Segments in DMConnect](#)

[Referencing a Script Content Segment in DREAMmail](#)

[Referencing a Script Content Segment in DMConnect](#).

## Creating Script Content Segments in *DREAMmail*

You create a script outside of *DREAMmail* and insert it into a script content segment in *DREAMmail*.

Step	Action
1	<b>Plan how you want to dynamically personalize your messages.</b>
2	<b>From the <i>DREAMmail</i> menu, click <i>Messages &gt; Content Manager</i>.</b>
3	<b>From the <i>Create New</i> list, select <i>Script Content Segment</i> and then click <i>Add</i>.</b>
4	<b>In the <i>Segment Name</i> field, type a name for the content segment.</b> A content segment's name must be unique within the site. When you create names, be sure to use: <ul style="list-style-type: none"> <li>• short meaningful names with 32 characters or less</li> <li>• underscores (_) instead of spaces in the content segment name.</li> </ul>
5	<b>Select a character set from the <i>Language</i> drop-down list.</b> The character set helps <i>DREAMmail</i> to interpret message content. The default value is English (US-ASCII). If your content contains characters outside of the US-ASCII character set (extended ASCII or double-byte), you must specify the character set that you use. For more information, see the <i>Programming Guide</i> .
6	<b>Enter the script into the <i>Content</i> field.</b> The script must be in a CDATA section. Make sure that you: <ul style="list-style-type: none"> <li>• include all tags and the URLs of any websites that you cite; URLs must begin with http:// or https://</li> <li>• include all necessary tags and position them correctly.</li> </ul>
7	<b>If you include URLs in the content segment and you want to track the click-through rates for these URLs, click <i>Select URLs</i> in the <i>URL Functions</i> section.</b> If you do not want to track URLs, go to <a href="#">Step 11</a> .
8	<b>In the <i>URL Tracking</i> window, enable the individual URLs that you want to track, or enable the <i>Select</i> checkbox to select all of the URLs.</b>
9	<b>To assign a label to the URL, click <i>Generate</i>.</b> <i>DREAMmail</i> automatically generates labels for selected URLs in the <i>Label</i> field. You can edit these URL labels.
10	<b>Click <i>Save</i>.</b> The URLs that you are tracking are displayed in the <i>URL Functions</i> window.
11	<b>To save the segment, click <i>Save</i>.</b>

**Related topics**

[Creating Script Content Segments in DMConnect](#)

[Referencing a Script Content Segment in DREAMmail](#)

[Referencing a Script Content Segment in DMConnect.](#)

# Creating Script Content Segments in *DMConnect*

You can create script content segments using *DMConnect*. *DMConnect* is an XML-based programming interface that enables you integrate external content management systems with *DREAMmail*. For more information about using *DMConnect*, see the *Programming Guide*.

Step	Action
1	<b>Create your script content segment.</b>
2	<b>Write the <i>DMConnect</i> <i>CreateContent</i> request.</b> Create an XML file and include all mandatory elements and any optional elements. The <code>&lt;ContentType&gt;</code> element must be <i>ScriptContent</i> .
3	<b>In the <code>&lt;ContentName&gt;</code> element, type in the name of the segment.</b> A content segment's name must be unique within the site. When you create names, be sure to use: <ul style="list-style-type: none"><li>• short meaningful names with 32 characters or less</li><li>• underscores ( <code>_</code> ) instead of spaces in the content segment name.</li></ul>
4	<b>Enter the script content segment into the <code>&lt;ContentText&gt;</code> element.</b> The script must be in a CDATA section. Make sure that you: <ul style="list-style-type: none"><li>• include all tags and the URLs of any websites that you cite. URLs must begin with <code>http://</code> or <code>https://</code></li><li>• include all necessary tags and position them correctly.</li></ul>
5	<b>To track click-through rates for URLs included in the segment, place each selected URL in a <code>&lt;URL&gt;</code> element.</b>
6	<b>Save the XML file and send using an HTTP post.</b> <i>Note: The file size cannot exceed 2 MB.</i>

## Related topics

- [Creating Script Content Segments in DREAMmail](#)
- [Referencing a Script Content Segment in DREAMmail](#)
- [Referencing a Script Content Segment in DMConnect.](#)



## Referencing a Script Content Segment in *DREAMmail*

Content segments can be created in either *DREAMmail* or *DMConnect*. Once you create and save content segments, you can use either *DREAMmail* or *DMConnect* to add them to a message or the subject line of a message. This procedure explains how to reference script content segments in *DREAMmail*.

Step	Action
1	From the <i>DREAMmail</i> menu, click <i>Messages &gt; Message Manager</i> .
2	From the <i>Messages for campaign list</i> , select the campaign that contains the message in which you want to insert a segment.
3	From the list of messages, select the message into which you want to insert a segment.
4	Click the yellow action button next to the message name and select <i>Edit</i> . The <i>Message Composer</i> window is displayed.
5	In the <i>Compose</i> form, click <i>Insert &gt; Content Segment</i> .
6	In the <i>Content Segment</i> dialog box, select the content segment that you want to insert into the message.
7	Click <i>Add to Body</i> or <i>Add to Subject</i> , depending on where you want to insert the content segment. <i>DREAMmail</i> inserts a reference to the content segment; for example, <i>#CONTENT:offer_fall05#</i> .
8	Verify that the location of the reference to the content segment in the message body or subject line is correct.
9	Click <i>Save</i> .

### Related topics

[Creating Script Content Segments in \*DREAMmail\*](#)

[Creating Script Content Segments in \*DMConnect\*](#)

[Referencing a Script Content Segment in \*DMConnect\*](#).

## Referencing a Script Content Segment in *DMConnect*

Content segments can be created in either *DREAMmail* or *DMConnect*. Once you have created and saved the content segments, you can use either *DREAMmail* or *DMConnect* to add them to a message or the subject line of a message.

This procedure explains how to reference script content segments in *DMConnect*. For information about using *DMConnect*, see the *Programming Guide*.

Step	Action
1	<b>Create your message in either <i>DREAMmail</i> or <i>DMConnect</i>.</b>
2	<b>Write a <i>DMConnect</i> <i>CreateMailing</i> request.</b> Create an XML file and include all mandatory and any optional elements.
3	<b>Type the name of the message in the <code>&lt;MailingName&gt;</code> element.</b> A message's name must be unique within the site. When you create names, be sure to use: <ul style="list-style-type: none"><li>• short meaningful names with 32 characters or less</li><li>• underscores (_) instead of spaces in the content segment name.</li></ul>
4	<b>Enter the message into the appropriate <code>&lt;BodyText&gt;</code> element.</b> The message must be entered into a CDATA section. Make sure that you: <ul style="list-style-type: none"><li>• include all tags and the URLs of any websites that you cite; URLs must begin with <code>http://</code> or <code>https://</code></li><li>• include all necessary tags and position them correctly.</li></ul>
5	<b>Place the cursor where you want to insert the reference to the content segment in the message, and type in the reference.</b> The reference to the segment must be entered in the format: <code>#CONTENT:name_of_segment#</code> ; for example, <code>#CONTENT:offer_fall05#</code> .
6	<b>To track click-through rates for URLs included in the message, place each selected URL in a <code>&lt;URL&gt;</code> element.</b>
7	<b>Save the XML file and send using an HTTP post.</b> <i>Note: The file size cannot exceed 2 MB.</i>

### Related topics

[Creating Script Content Segments in \*DREAMmail\*](#)

[Creating Script Content Segments in \*DMConnect\*.](#)

# Editing and Deleting Content Segments

Content segments can be updated if the information in them is out of date. Segments can also be deleted once they have been used or are no longer needed.

## Procedure to edit or delete a content segment

You can edit a content segment if it needs to be changed or delete it if it has become obsolete.

Step	Action
1	Log in to DREAMmail.
2	From the DREAMmail menu, click <i>Messages &gt; Content Manager</i> . A list of content segments is displayed
3	Click the yellow action button next to the content segment and choose either <i>Edit</i> or <i>Delete</i> .
4	If you are editing the content segment, make the necessary changes to it.
5	Click <i>Save</i> or <i>save changes</i> or <i>OK</i> to confirm that you want to delete it.

# SYNTAX OF THE DREAMMAIL SCRIPTING LANGUAGE

You can add basic personalization to messages using the *Content Manager* in DREAMmail or you can use more sophisticated personalization in messages using the DREAMmail scripting language. After you write scripts, you need to include them in messages using DREAMmail or *DMConnect*. This section explains the syntax of the DREAMmail scripting language.

This section discusses the following topics:

- [Elements Used to Create Script Content Segments](#)
- [Numeric, Relational, and Date/Time Operators](#)
- [Formatting Text and Messages](#)
- [IF Statement](#)
- [SWITCH Statement](#)
- [Multiple Conditions in Statements](#)
- [VARIABLE Statement](#).

## Elements Used to Create Script Content Segments

The DREAMmail scripting language uses tags to create logic statements. You insert these statements into messages by reference. DREAMmail uses these statements to determine the content that is dynamically inserted when messages are sent.

This section discusses the following topics:

- [Tags and statements](#)
- [Expressions](#)
- [Operators](#)
- [Data types](#).

### Tags and statements

The scripting language tags are the building blocks for dynamic personalization scripting and determine which piece of content is inserted into a message in place of another. There are four statements that you can use to build a script.

- IF statements determine the content to place in a message. These statements are used to compare database fields, or conditions, where each field is independent. IF statements often use the AND and OR operators.
- SWITCH statements determine the content to place in a message. These statements compare one database field against multiple values. These statements execute faster than IF statements, because IF statements must examine multiple fields.
- VARIABLE statements are used to insert content based on a variable rather than subscriber profile data.

For more information, see [IF Statement](#), [SWITCH Statement](#), [Multiple Conditions in Statements](#), and [VARIABLE Statement](#).

### Expressions

An expression, or *expn*, is comprised of one or more values separated by an operator. The value to which a profile field is compared must be placed within either single or double quotation marks. For example, the expression "#city='New York'" applies to all subscribers who live in New York.

There are two types of expressions that enable you to perform formatting in scripts:

- text formatting expressions format the text of personalized data
- message formatting expressions specify what content is to be used for each message format being deployed.

For more information, see [Formatting Text and Messages](#).

Operators

Expressions use operators to perform mathematical operations, manipulate numbers and text, test the truth or falsity of conditions, or compare values. For more information, see [Numeric, Relational, and Date/Time Operators](#).

Data types

The *type* attribute specifies the data type expected in an expression. For example, you can have the following statement: `<Switch type = 'stringi' value="#City#">`. When DREAMmail examines the City profile field, it ignores the case of the letters in the field. This means, for example, that "New York", "New york", "NEw York" and "new york" all match the search string "New York". This is useful if you are not sure about the quality of your data. The following data types are supported in the scripting language.

Type	Description
int	integer
float	a number that includes decimals
string	a case-sensitive alphanumeric character string
stringi	a non-case-sensitive alphanumeric character string
date	a date

Related topics

- [Numeric, Relational, and Date/Time Operators](#)
- [Formatting Text and Messages](#)
- [IF Statement](#)
- [SWITCH Statement](#)
- [Multiple Conditions in Statements](#)
- [VARIABLE Statement](#).

## Numeric, Relational, and Date/Time Operators

Expressions use operators to perform mathematical operations, manipulate numbers and text, test the truth or falsity of conditions, or compare values.

This section discusses the following topics:

- [Relational operators](#)
- [Mathematical operators](#)
- [Date and time operators](#).

### Relational operators

Relational operations, sometimes referred to as *comparison operators*, compare two values. Expressions that contain relational operators are called *relational expressions*. In the scripting language, *DREAMmail* supports the operators shown in the table below.

Operator	Definition	Example
AND	Boolean AND	"#Gender='Female' AND City='Seattle'#" returns all the female subscribers in Seattle.
NOT	Boolean NOT	"#NOT State='California'#" returns all the subscribers who are not from California.
OR	Boolean OR	"#Gender='Female' OR City='Seattle'#" returns all female subscribers and all subscribers from Seattle.
<	less than	"#JulySales< AugSales#" returns all subscribers whose July sales are less than the August sales.
>	greater than	"#JulySales>AugSales#" returns all subscribers whose July sales are greater than the August sales.
=	equal to	"#JulySales=AugSales#" returns all subscribers whose July sales are equal to the August sales.

Operator	Definition	Example
!=	not equal to	"#JulySales!=AugSales#" returns all subscribers whose July sales are not equal to the August sales.
<=	less than or equal to	"#JulySales<= AugSales#" returns all subscribers whose July sales are less than or equal to the August sales.
>=	greater than or equal to	"#JulySales>=AugSales#" returns all subscribers whose July sales are greater than or equal to the August sales.
Contains(first term, second term)	The statement contains the text string. This function is case sensitive.	"#Contains(City,'New')#" returns all cities with New in the name.  <b>Note:</b> The first and second terms can be any combination of profile fields, strings and local variables.
EndsWith(first term, second term)	The statement ends with the text string. This function is case sensitive.	"#EndsWith(EmailAddr,'ru') #" returns all email addresses whose domain ends in ".ru"  <b>Note:</b> The first and second terms can be any combination of profile fields, strings and local variables.
BeginsWith(first term, second term)	The statement begins with the text string. This function is case sensitive.	"#Begins with(EmailAddr,'John')#" returns all email addresses beginning with "John".  <b>Note:</b> The first and second terms can be any combination of profile fields, strings and local variables.



Operator	Definition	Example
IsNull()	<p>The statement returns true if the field being compared does not contain any data.</p> <p><b>Note:</b> If the profile field has been updated, it is considered empty and not null and the field will not return true. In order to capture all the subscribers that do not have any information in the field, you should use the following:</p> <p>"#IsNull(FieldName) OR FieldName=""#"</p>	"#IsNull(FirstName)#" returns true if there is no First Name entered in the profile.

## Mathematical operators

Mathematical operations can be applied in simple personalization or as part of the scripting language in a complex script. Mathematical functions can be applied to user defined variables, profile field, constants, and literals. They can be nested and can contain bracketed functions for more complicated mathematical operations. Standard order of operations applies to mathematical operators.

**TABLE: Mathematical Operator format**

Operator	Operation and Usage	Example
+ (addition)	Op1 + Op2 Returns Op1 plus Op2. If both Op1 and Op2 are integers, then the answer is an integer; otherwise, it is a floating data type.	Your orders for the summer are <Print expn = "#OrdersJune + OrdersJuly#">
- (subtraction)	Op1 - Op2 Returns Op1 minus Op2. If both Op1 and Op2 are integers, then answer is an integer; otherwise, it is a floating data type.	You ordered <Print expn = "#OrdersJune - OrdersJuly#"> items more in June than in July.
* (multiplication)	Op1 * Op2 Returns Op1 multiplied by Op2. If both Op1 and Op2 are integers, then the answer is an integer; otherwise, it is a floating data type.	Your total amount including tax is <Print expn = "#OrdersAmt * 1.15#">.

**TABLE: Mathematical Operator format**

Operator	Operation and Usage	Example
/ (division)	Op1 / Op2 returns Op1 divided by Op2. Answer is a floating data type.	Your average purchases per day for June is <Print expn = "#OrdersAmtJune / 30#">.
Round	<Round(Op1, precision)> returns a floating representation of Op1 with precision decimal places. For example, if precision is 0 or not specified, then <i>Round</i> rounds to the nearest integer. If precision is specified then this rounds to the number of decimal places specified.	Your savings are \$<print expn="#Round(PurchaseAmt, 2)#"> This rounds the purchase amount to the next cent.

For an explanation of the syntax used to output these expressions, see [Print statement](#).

## Date and time operators

Scripting expressions that use date and time operators can insert content based on current date and/or time, as well as make logical comparisons based on profile dates and system dates.

**TABLE: Date operators**

Function	Explanation	Example
#System:Date#	Returns the system date of the server that is currently sending the message.	#System:Date# returns the date in the GMT time zone
Month(date)	Month(date) returns the number of the month of the year; where January = 1.	The month of the year is <Print expn = "#Month(System:Date)#">
DayOfWeek(date)	DayOfWeek(date) returns the number of the day of the week; where Sunday = 1.	The day of the week is <Print expn = "#DayOfWeek(System:Date)#">

TABLE: Date operators

Function	Explanation	Example
Day(date)	Day(date) returns the day from the date; for example 21 is Day(05212001).	<If Condition="#Day(BirthDay) = Day(System:Date) and Month(BirthDay) = Month(System:Date)#"> <i>Happy Birthday!</i> <Else> <EndIf>
DateDiff(date1,date2)	DateDiff(date1,date2) returns the number of days that date2 is away from date1. When date2 is a later date than date1, then the returned value is positive; otherwise, the value is negative. The value is always an integer.	You have been a customer for <Print expn = "#DateDiff (JoinedDate,System:Date)#"> days.

Related topics

- [Data types](#)
- [Print statement.](#)

## Formatting Text and Messages

DREAMmail provides statements and expressions that enable you to format the text of personalized data and specify content specific to a message format.

This section discusses the following topics:

- [Print statement](#)
- [Text formatting expressions](#)
- [Message format expressions](#).

### Print statement

The *Print* statement can be used to output and format text. It contains one tag and can use any of the three text formatting expressions.

**Syntax of the Print statement:** The syntax of the Print statement is:

```
<Print expn="#expn#">
```

where "expn" is either a text or message formatting expression; see [Text formatting expressions](#) and [Message format expressions](#).

### Text formatting expressions

When the data in profile fields is not stored in the way that you want it to be displayed in a message, you can use text formatting expressions to modify it.

These expressions enable you to change text to:

- all uppercase
- all lowercase
- initial capital letter.

**Changing text to uppercase:** The *UCase(string)* expression takes a string as input and converts it all to uppercase. The string can be a literal or a profile field variable, or another content segment. The syntax of this expression is:

```
<Print expn="#UCase('hello world')#">
```

Returns: HELLO WORLD

**Changing text to lowercase:** The *LCase(string)* expression takes a string as input and converts it all to lowercase. The string can be a literal or a profile field variable, or another content segment. The syntax of this expression is:

```
<Print expn="#LCase('HelLo WoRlD')#">
```

Returns: hello world

**Changing a phrase to initial capital letters:** The *Capitalize(string)* expression takes a string as input and capitalizes the first letter of all words (a word is a text block separated by spaces) then converts the remaining letters to lowercase. The string can be a literal or a profile field variable, or another content segment. The syntax of this expression is:

```
<Print expn="#Capitalize('HelLo WoRld') #">
```

Returns: Hello World

This function could be used to personalize a message such as the following:

```
<If condition="#IsNull(FirstName) or FirstName=""#">
    Dear Customer
<Else>
    Dear <Print expn="#Capitalize(FirstName) #">
<EndIf>
```

## Message format expressions

You can specify which content segments are used for each message format (text, HTML, or AOL) being deployed. That is, you do **not** have to create separate content segments depending on the format of a message. You use the message format expressions to identify the message format being used. These expressions enable *DREAMmail* to insert the appropriate text:

- #Message:IsHTML#

This expression returns true if *DREAMmail* is currently composing an HTML message (or the HTML portion of an autosense message). Otherwise, it returns false.

- #Message:IsText#

This expression returns true if *DREAMmail* is currently composing a text message (or the text portion of an autosense message). Otherwise, it returns false.

- #Message:IsAOL#

This expression returns true if *DREAMmail* is currently composing an AOL message. Otherwise, it returns false.

**Example of message formatting using expressions:** This example shows how you can use the message format expressions to test which type of format a subscriber can receive. When *DREAMmail* determines the format, the appropriate content is inserted into the message.

```
<If condition="#Message:IsText#">
    Hello World
<ElseIf condition="#Message:IsHTML#">
    <strong>Hello World</strong>
<EndIf>
```

# IF Statement

IF statements compare different database fields, or conditions in a script, where each field is independent. Often they are used to create OR or AND conditions.

This section discusses the following topics:

- [Tags in IF statements](#)
- [Syntax of tags used in IF statements](#)
- [Example of an IF statement.](#)

## Tags in IF statements

IF statements use the following tags:

- <If>
- <Else>
- <ElseIf>
- <EndIf>.

## Syntax of tags used in IF statements

This table describes the syntax of the tags used in IF statements.

Tag	Syntax	Explanation
If	There are two possible formats: <If condition="#expn#"> <If op1="expn" op="operator" op2="expn" type="type">	Used to compare different database fields for each condition.
Else	<Else>	If none of the conditions of the <If> statement are met, the text that follows the <Else> statement is used.
Elseif	<ElseIf condition="#expn#">	Is equivalent to <Endif><Else><If condition="#expn#">.
EndIf	<EndIf>	Finishes the <If> statement.

## Example of an IF statement

You want to send a specialized greeting to everyone living in New York and a standard greeting to everyone else. To do this, use an IF statement.

```
<If condition="#City='NewYork'#">
  Hey New Yorkers
<Else>
  Hello
<EndIf>
```

## Related topics

[Expressions](#)

[SWITCH Statement](#)

[Multiple Conditions in Statements](#)

[VARIABLE Statement.](#)

# SWITCH Statement

SWITCH statements compare one field in the database against several values. SWITCH statements execute faster than IF statements, because IF statements must examine multiple fields.

This section discusses the following topics:

- [Tags in SWITCH statements](#)
- [Syntax of tags used in SWITCH statements](#)
- [Example of a SWITCH statement.](#)

## Tags in SWITCH statements

SWITCH statements used the following tags:

- <Switch>
- </Switch> or <EndSwitch>
- <Case>
- </Case> or <EndCase>
- <Default>
- </Default>.

## Syntax of tags used in SWITCH statements

This table describes the syntax of the tags that are used in SWITCH statements.

Tag	Syntax	Explanation
Switch	<Switch type="type" value="#expn#">	Compares one field in the database, or a variable, against multiple values. This executes faster than an <If> statement, which must examine multiple fields.  <i>Value</i> and <i>type</i> must correspond to each other because the switch statement looks for an exact match. For example if you use <i>int</i> and give the value as <i>#JanSales#</i> , you will not get a match for 1.98, you would have to use <i>float</i> instead.
/Switch or EndSwitch	</Switch> or <EndSwitch>	Finishes the <Switch> statement.



Tag	Syntax	Explanation
Case	<Case match= "value">	Compares the value in this tag to the value of the expression in the <Switch> tag.
/Case or EndCase	</Case> or <EndCase>	Finishes the <Case> statement.
Default	<Default>	If none of the <Case> elements are a match to the value in the <Switch> statement, the text following the <Default> tag is used.
/Default	</Default>	Finishes the <Default> statement.

Example of a SWITCH statement

You want to send a personalized greeting to subscribers living in New York and Chicago, and a standard greeting to all the other subscribers. Since DREAMmail has to search in only one field for the variable, it executes faster using the SWITCH statement than an IF statement.

```
<Switch type="stringi" value="#City#">
<Case match="New York">
  Hey New Yorkers!
</Case>
<Case match="Chicago">
  Hey Windy City!
</Case>
<Default>
  Hello
</Default>
</Switch>
```

Related topics

- [Expressions](#)
- [IF Statement](#)
- [Multiple Conditions in Statements](#)
- [VARIABLE Statement.](#)

## Multiple Conditions in Statements

Statements can be combined so that multiple conditions can be handled. Very rarely do you use a single IF or SWITCH statement. Most likely, you combine conditions to create more complex personalization. The order of the operations is extremely important, especially when using multiple conditions. When checking your statements, make sure that both your primary and secondary audiences are correct.

The most frequently used multiple conditions are:

- [IF...OR condition](#)
- [IF...AND condition](#)
- [Combined AND and OR conditions](#)
- [Nested conditions](#).

### IF...OR condition

An IF...OR condition inserts content if only one of the conditions is met. IF...OR conditions can be created in one of three ways, using:

- <If> tags and an OR operator
- <ElseIf> tags
- nested conditions.

**Example of an <If> tag and an OR operator:** You send a personalized greeting to all of your subscribers in the city of Seattle and the state of California, while sending all other subscribers a standard greeting.

```
<If condition="#City='Seattle' OR State='California'#">
  Hello coffee drinkers
<Else>
  Hello
<EndIf>
```

**Example of an <ElseIf> tag:** Using the <ElseIf> tag, you send a personalized greetings to all of your subscribers in the city of New York and in the state of California, while all other subscribers get a standard hello as the greeting.

```
<If condition="#City='New York'#">
  Hey New Yorkers!
<ElseIf condition="#State='California'#">
  How's the sun?
<Else>
  Hello
<EndIf>
```

## IF...AND condition

An IF...AND condition inserts content only if **all** of the conditions are met. There are two methods of creating an AND condition:

- using an <If> tag and an AND operator (see [Example of an <If> tag and an AND operator](#))
- using nested conditions (see [Nested conditions](#)).

**Example of an <If> tag and an AND operator:** You can send a personalized message to all female subscribers in the city of New York to advertise the latest romantic comedy, while sending all other subscribers an advertisement of your website.

```
<If condition="#City='New York' AND Gender='female'#">
  See the latest romantic comedy in our big screen theaters.
<Else>
  Check our website for the latest movies showing in our big screen
  theaters.
<EndIf>
```

## Combined AND and OR conditions

You can use a combination of AND and OR conditions to create complex scripting. For example, you can send a personalized message to all females subscribers in the city of New York and the state of California advertising the latest romantic comedy, send other subscribers in New York city and California a message advertising the latest action movies, and send all other subscribers an advertisement of your website. This means that all people in New York City or California whose gender profile field is male, left blank, has other words entered, or has female misspelled receives the action movie advertisement. All people outside of New York City and California receive the generic website advertisement.

```
<If condition="#(City='New York' OR State='California') AND
Gender='female'#">
  See the latest Romantic Comedy in our big screen theaters.
<ElseIf condition="#(City='New York' OR State='California') AND
Gender!='female'#">
  Action Movies look better in our big screen theaters.
<Else>
  Check our website for the latest movies showing in our big screen
  theaters.
<EndIf>
```

## Nested conditions

By nesting conditions, you can use multiple conditions to create complex personalization. The conditions can be of any type, and can be comprised of a combination of <If> and <Switch> tags. When you nest conditions, do not exceed 10 conditions.

The following examples show how <If> and <Switch> tags can be nested.

**Example of a SWITCH statement nested in an IF statement:** You can insert <If> tags within <Switch> tags in scripts. To send a movie promotion to female subscribers in certain cities, a different promotion to all other female subscribers, and a third promotion to all other subscribers, you sort the female subscribers using an <If> tag and, since you use the same field to determine the location of the subscriber, use <Switch> to filter the desired cities.

```
<If condition="#Gender='female'#">
  <Switch type="stringi" value="#City#">
    <Case match="New York">
      See the sneak preview of the latest romantic comedy in our 42nd
      Street big screen theater.
    </Case>
    <Case match="Los Angeles">
      See the sneak preview of the latest romantic comedy in our Rodeo
      Drive big screen theater.
    </Case>
    <Case match="Miami">
      See the sneak preview of the latest romantic comedy in our South
      Beach big screen theater.
    </Case>
    <Case match="Chicago">
      See the sneak preview of the latest romantic comedy in our South
      Side big screen theater.
    </Case>
    <Default>
      See the latest romantic comedy in our big screen theaters.
    </Default>
  </Switch>
<Else>
  Action movies are more exciting in our big screen theaters.
</EndIf>
```

**Example of an IF statement nested in a SWITCH statement:** To send promotions to both females and males in each of four cities and a general promotion to everyone else, you must modify the script shown in [Example of a SWITCH statement nested in an IF statement](#). In this case, you first filter the four cities using a <Switch> tag, then determine the gender using an <If> tag.

```
<Switch type="stringi" value="#City#">
  <Case match="New York">
    <If condition="#Gender='female'#">
      See the sneak preview of the latest romantic comedy in our 42nd
      Street big screen theater.
    <Else>
      Action movies are more exciting in our big Broadway theater.
```

```
<EndIf>
</Case>
<Case match="Los Angeles">
  <If condition="#Gender='female'#">
    See the sneak preview of the latest romantic comedy in our new
    Rodeo Drive big screen theater.
  <Else>
    Action movies are more exciting in our new Rodeo Drive theater.
  <EndIf>
</Case>
<Case match="Miami">
  <If condition="#Gender='female'#">
    See the sneak preview of the latest romantic comedy in our South
    Beach big screen theater.
  <Else>
    Action movies are more exciting in our South Beach theater.
  <EndIf>
</Case>
<Case match="Chicago">
  <If condition="#Gender='female'#">
    See the sneak preview of the latest romantic comedy in our South
    Side big screen theater.
  <Else>
    Action movies are more exciting in our South Side theater.
  <EndIf>
</Case>
<Default>
  Check our website for the latest movies showing in our big screen
  theaters.
</Default>
</Switch>
```

## Related topics

[IF Statement](#)

[SWITCH Statement](#)

[VARIABLE Statement.](#)

# VARIABLE Statement

Sometimes the subscriber profile that you target involves the relationship of several variables within your users’ profiles. Rather than creating a deeply nested script, you can use the VARIABLE statement, which enables you to create variables to simplify your scripting. VARIABLE statements also allow you to dynamically insert content based on a random condition that has been met in the process of sending out the message.

This section discusses the following topics:

- [Syntax of VARIABLE statement tag](#)
- [Description of the <SetVar> tag](#)
- [Example of the <SetVar> tag for users who match multiple conditions](#)
- [Modification of the <SetVar> example where gender is ignored](#)
- [Description of the RandInt expression](#)
- [Example of the RandInt expression.](#)

## Syntax of VARIABLE statement tag

This table describes the syntax of the VARIABLE statement tag.

Tag	Syntax	Explanation
SetVar	<SetVar name="variable-name" value="expn">	Enables you to create variables in scripts.  Note: The <SetVar> tag does not require an end tag.

## Description of the <SetVar> tag

The <SetVar> tag sets a variable in a message. You can place multiple <SetVar> tags anywhere in the message. You reference these variables in the format, `var:content1`.

The <SetVar> tag has two properties:

- `name`, which defines the name of the variable  
The name must be prefixed by `var:` which indicates that the variable is not a profile field. For example, `var:movie_promo`. This name must not conflict with a profile field name.
- `value` is an expression that sets the variable. This can be a random variable; see [Description of the RandInt expression](#). The format of `value` can be either an explicit string (for example, `value="2"`) or an expression (for example, `value="#var:which_movie#"`).

## Example of the <SetVar> tag for users who match multiple conditions

Users often fall into more than one category but you want to send them only one promotion. For example, to send a romantic comedy promotion to all females, an action movie promotion to all males in New York, Chicago, Miami, and Los Angeles, and general movie promotions to all other subscribers, you can create the following message.

```
<SetVar Name="var:which_movie" Value="3">
  <Switch type="stringi" value="#City#">
    <Case match ="New York">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
    <Case match="Los Angeles">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
    <Case match="Miami">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
    <Case match="Chicago">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
  </Switch>
<If condition="#Gender='female'#">
  <SetVar Name="var:Which_movie" Value="1">
<EndIf>
  <Switch type="int" value="#var:Which_movie#">
    <Case match="1">
      See the latest Romantic Comedy in our big screen theaters.
    </Case>
    <Case match="2">
      Action movies are more exciting in our big screen theaters.
    </Case>
    <Case match="3">
      Check our website for the latest movies showing in our big screen
      theaters.
    </Case>
  </Switch>
```

In this scenario, the "#gender ='female'#" condition overrides the location condition, since all subsequent conditions always override previous conditions.

## Modification of the <SetVar> example where gender is ignored

If you want to send the action movie promotion to all subscribers in Chicago, Los Angeles, Miami, and New York, regardless of gender, you can rewrite the previous <SetVar> script as follows.

```
<SetVar Name="var:which_movie" Value="3">
<If condition="#Gender='Female'#">
  <SetVar Name="var:Which_movie" Value="1">
```

```

<EndIf>
  <Switch type="stringi" value="#City#'">
    <Case match="New York">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
    <Case match="Los Angeles">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
    <Case match="Miami">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
    <Case match="Chicago">
      <SetVar Name="var:which_movie" Value="2">
    </Case>
  </Switch>
  <Switch type="int" value="var:Which_movie">
    <Case match="1">
      See the latest Romantic Comedy in our big screen theaters.
    </Case>
    <Case match="2">
      Action movies are more exciting in our big screen theaters.
    </Case>
    <Case match="3">
      Check our website for the latest movies showing in our big screen
      theaters.
    </Case>
  </Switch>

```

## Description of the *RandInt* expression

The *RandInt* expression is a special variation of the `<SetVar>` tag. It assigns a random value that inserts content based on a condition that is met during the process of sending the message.

The *RandInt* expression sets the value of the *variable\_name* to a range of integer values in the following `<SetVar>` format:

```
<SetVar name="variable_name" value="#RandInt (expn1,expn2) #">
```

where:

expn1	defines the lowest value of the integer that can be assigned to the counter
expn2	defines the highest value of the integer that can be assigned to the counter



Example of the  
*RandInt*  
expression

You want to randomly send one of the four offers to all females in your mailing list. Using the <If> statement, you verify whether a subscriber is female. Using the <SetVar> tag, *DREAMmail* randomly assigns a number to the #RandNum# field for all female subscribers. The <Switch> tag compares the #RandNum# field against several cases as shown in the following table.

Statement	Result
If #Gender# = female AND #AdNum# = 1	Enter promotion ABC
If #Gender# = female AND #AdNum# = 2	Enter promotion DEF
If #Gender# = female AND #AdNum# = 3	Enter promotion UVW
If #Gender# = female AND #AdNum# = 4	Enter promotion XYZ
If #Gender# != female	Enter promotion PDQ

```
<SetVar name="var:leftValue" value="1">
<SetVar name="var:RandNum" value="#RandInt (var:leftValue,4) #">
<If condition="#Gender='female' #">
<Switch type="int" value="#var:RandNum#">
  <Case match="1">
    Special offer on ABC!
  </Case>
  <Case match="2">
    Special offer on DEF!
  </Case>
  <Case match="3">
    Special offer on UVW!
  </Case>
  <Case match="4">
    Special offer on XYZ!
  </Case>
</Switch>
<Else>
  Special offer on PDQ!
<EndIf>
```

Related topics

- [IF Statement](#)
- [SWITCH Statement](#)
- [Multiple Conditions in Statements](#)
- [VARIABLE Statement.](#)

# RICH MEDIA IN DYNAMIC PERSONALIZATION

Rich media in email messages typically involves sending a message that includes streaming audio or video. There are several technologies involved to create these messages but most require support for scripting languages (typically JavaScript) as well as animation and video (for example, Flash, Shockwave, and so on.) While it is possible to deliver rich media messages through email, there are many challenges involved (such as support for scripting languages by email applications).

Macromedia Flash is a popular way to deliver rich media content to enhance your marketing promotions. Traditionally, web publishers have placed Flash content on their websites and then sent a message inviting subscribers to the website. Using [DREAMmail](#), you can place Flash content directly in your email messages since many HTML-capable email applications can play Flash content.

[DREAMmail](#) can only personalize the HTML message and not the actual Flash content. You can, however, design a Flash movie that takes advantage of the personalization within the HTML message.

This chapter discusses the following topics:

- [Flash Compatibility with Email Applications](#)
- [Flash Compatibility with Web-Based Email Services](#)
- [Inserting Flash into an HTML Message](#)
- [Personalization of Flash Content](#)
- [Tracking Links in Flash Movies](#)
- [Inserting Flash with Personalization and URL Tracking](#)

## Flash Compatibility with Email Applications

Most email applications use browser technology to display HTML-formatted messages. Depending on the type of email application that your subscribers use, you must add specific tags to run the Flash content.

This section discusses the following topics:

- [MS Outlook and Netscape Messenger](#)
- [Eudora](#)
- [Lotus Notes](#).

### MS Outlook and Netscape Messenger

Microsoft Outlook, Microsoft Outlook Express and Netscape Messenger use standard browser technology to display HTML-formatted messages. These email applications can display Flash content. They control this content by using JavaScript as long as the security settings are set to enable both scripting and object and/or plug-in support. To display Flash content:

- in Microsoft Internet Explorer, Outlook or Outlook Express, use an ActiveX control, referenced with the `<OBJECT>` tag
- in Netscape Navigator, use a plug-in, referenced with the `<EMBED>` tag.

Note that some of the latest security patches for these applications disable these settings.

### Eudora

The HTML rendering engine in Qualcomm Eudora does not support objects and plug-ins or JavaScript. The engine also does not support `<SPAN>` tags. If Internet Explorer is installed, however, Eudora uses it as the default display engine for HTML, and can support Flash. Like Outlook, Eudora provides an option to disable scripting and plug-in support.

### Lotus Notes

Although Lotus Notes is not generally a primary target for email messages, the latest versions use Internet Explorer as a rendering engine and can display Flash.

## Flash Compatibility with Web-Based Email Services

Many subscribers use web-based email services, such as Hotmail or Yahoo!. To ensure that the scripting content of messages does not interfere with the features of the web-based email application, many of these systems modify messages before they are displayed.

Neither Yahoo! nor Hotmail can display Flash content. Many of the smaller web-based email systems also cannot display Flash content. For example, both Hotmail and Yahoo! use various techniques to disable the `<OBJECT>`, `<EMBED>` and `<SCRIPT>` tags. Therefore, it is vital that an effective static HTML version of your message be supported.

This section discusses the following topics:

- [Why Yahoo! email cannot display Flash](#)
- [Why Hotmail email cannot display Flash](#)

### Why Yahoo! email cannot display Flash

Yahoo! removes the opening `<SCRIPT>` tag, causing the script itself to be displayed unless it is contained within HTML comment tags. Yahoo! also changes the `<OBJECT>` and `<EMBED>` tags to `<xOBJECT>` and `<xEMBED>` tags respectively, preventing browsers from processing them. This means that the browser also ignores alternate content inserted within the `<OBJECT>` tags.

### Why Hotmail email cannot display Flash

Hotmail changes the opening `<SCRIPT>` tag to `<xSCRIPT>`, preventing browsers from processing the script. To prevent scripting errors, Hotmail also changes `onClick`, `onMouseOver`, and so on, to `xonClick`, `xonMouseOver`, and other tags. Hotmail also removes both the `<OBJECT>` and `<EMBED>` tags.

## Inserting Flash into an HTML Message

Although over 95% of Internet users have the Flash player installed on their computer, many email applications either do not support the use of this player, or, based on security settings, disable its use. Therefore, your Flash-enabled messages must account for the fact that not all users can experience the Flash content, and must display instead a meaningful static HTML version.

This section discusses the following topics:

- [Flash in an HTML message](#)
- [Example of an HTML Flash insertion script](#).

### Flash in an HTML message

To insert Flash in messages, you create an HTML message that detects whether a subscriber's email application can display Flash, and if so, the Flash part of the message is inserted. If the email application cannot display Flash, only the static image is displayed. When you design your HTML message, make sure that the message is displayed correctly, whether the Flash is displayed or not.

After you create the Flash movie, you can insert the Flash script directly in the message where you want Flash to be displayed. You can also save the script as a script content segment and reference the content segment where you want the Flash to be displayed in a message.

### Example of an HTML Flash insertion script

Create a table with a static GIF background that is displayed if the subscriber's computer cannot display Flash. Then, using the `<EMBED>` tag, insert the Flash script into the cell. Do not include the `codebase` parameter in the `<OBJECT>` tag because some computers try to install the ActiveX control when the subscriber opens the message. For the same reason, do not include the `pluginspage` parameter in the `<EMBED>` tag because a dialog box pops up asking the subscriber to perform an installation and this process could scare your subscribers.

If a subscriber has the Flash player installed, the Flash movie covers the background of the table cell. If the player is not installed, or the subscriber's email application does not display Flash, the subscriber sees the table's background GIF. Since the background image is not clickable, you add a link that takes the subscriber to the website containing the Flash movie.

```
<HTML>
<SCRIPT language=JavaScript>
<!--
    //Detect Flash Plug-in
    var FlashDetected= false;
```

```

        var plugin = (navigator.mimeTypes &&
        navigator.mimeTypes["application/x-shockwave-flash"] ?
        navigator.mimeTypes["application/x-shockwave-
        flash"].enabledPlugin : 0);
        if (plugin &&
        parseInt(plugin.description.substring(plugin.description.indexOf (
        ".")-1)) >= 4) {
            //Flash 4 or greater is available
            FlashDetected = true;
        }
    -->
</SCRIPT>
<SCRIPT language=VBScript>
<!--
        'Detect Flash ActiveX
        on error resume next
        FlashDetected =
        IsObject(CreateObject("ShockwaveFlash.ShockwaveFlash"))
    -->
</SCRIPT>
<BODY>
    <TABLE border=0 cellpadding=0 cellspacing=0>
        <TR>
            <TD background="http://www.Epsilon Interactive.com/
            alternate_image.gif" width=550 height=400>
<SCRIPT language=JavaScript>
<!--
if (FlashDetected)
document.write("<OBJECT classid=clsid:D27CDB6E-AE6D-11cf-96B8-
444553540000 width=550 height=400 border=0><PARAM NAME='Src'
VALUE='http://www.Epsilon Interactive.com/flashmovie.swf'><EMBED
src='http://www.Epsilon Interactive.com/flashmovie.swf' width=550
HEIGHT=400 TYPE='application/x-shockwave-flash'></EMBED></OBJECT>");
-->
</SCRIPT>
        </TD>
    </TR>
</TABLE>
    If you cannot see the movie, <a href="http://www.Epsilon
    Interactive.com/flashmovie.swf">click here.</a>
</BODY>

```

# Personalization of Flash Content

DREAMmail can personalize only the HTML in a message and not the actual Flash content. You can, however, design a Flash movie that takes advantage of the personalization within the HTML message.

This section discusses the following topics:

- [Example of a personalized Flash movie](#)
- [Passing data to Flash movies.](#)

## Example of a personalized Flash movie

Flash supports passing variables into the movie, such as parameters for the movie name. For example, the movie name in the `<EMBED>` tag can be:

```
Movie.swf
```

You can pass variables to the movie, using a standard Internet query strings. For example, if you wanted to create a Flash variable called *myName* and populate it with the string "John", name your movie as:

```
Movie.swf?myName=John
```

**Profile fields:** In DREAMmail you can insert any profile field from mailing lists into the body of the message using `#FieldName#` tags. These tags can also be used within the HTML tags. As a result, you can create a movie name that passes a variable that is personalized for each subscriber:

```
Movie.swf?myName=#FirstName#
```

**Variables:** You can pass multiple variables by separating them with the ampersand (&) character:

```
Movie.swf?myName=#FirstName#&myColor=#FavoriteColor#
```

## Passing data to Flash movies

Once you have created an HTML file that passes data to the Flash movie, you can use that data for something useful within the Flash content. The *ActionScript* guide that comes with Flash explains how to use variables in more detail. Using the profile fields example (see [Profile fields](#)), create a text object within the movie, setting its properties to *Dynamic Text* and assign its name to be the variable name that you passed into the movie (in this case *myName*).

When DREAMmail deploys the HTML message, the movie name is personalized with the subscriber's name. When the subscriber opens the message, his or her name is passed into the Flash movie, and is displayed in the text object that contains the variable *myName*.

***Important:*** You can also use variables to change the colors within the Flash movie, to change which frame in the movie you start with, or to change which symbols you use on stage. You are only limited by your imagination and Flash scripting abilities.



## Tracking Links in Flash Movies

Flash movies can provide links that can be tracked in *DREAMmail*. To track a link in the Flash content, you must define the URL within the HTML and not in the Flash movie itself.

This section discusses the following topics:

- [Example of link tracking](#)
- [Example of the URL tracking script.](#)

### Example of link tracking

You first create a variable as part of the query string for the movie name:

```
movie.swf?myurl=http://www.Epsilon Interactive.com
```

**Adding URLs:** Next, in the Flash movie include the button or link that launches the web page to use this variable, rather than defining the URL within the Flash movie. To get a URL-based variable, you must flag the link as an expression and enter an expression that provides the URL's location. For example, the following ActionScript statement indicates that the URL is the value of the variable myurl:

```
getURL(myurl);
```

**Tracking URLs:** You now change the query string to pass a tracked version of the URL. Since the URL is now part of the HTML, and not the Flash movie, *DREAMmail* can reference it. Since the tracked version of the URL is part of another URL (the movie path), *DREAMmail* does not know where this URL actually ends. This is also true if you have multiple URLs that you want to track. You can submit the content using *DMConnect* (which enables you to explicitly state which URLs are to be tracked), or you can force tracking. You force URL tracking by inserting the following `<URL>` tag set in your query string:

```
<URL URL=... Variant=... Label=...>
```

where

URL	is the URL to track
Variant	is any personalized part of the URL query string that you do not want track. (This is optional in the query string. It is not used in the example below.)
Label	is the label to use to represent the URL in reports.

## Example of the URL tracking script

If you add personalization and URL tracking to the query string for the <OBJECT> and <EMBED> tags in the [Example of an HTML Flash insertion script](#), the result is:

```
<OBJECT classid=clsid:D27CDB6E-AE6D-11cf-96B8-444553540000 width=550
height=400 border=0><PARAM NAME='Src' VALUE='http://www.Epsilon
Interactive.com/flashmovie.swf?myfirstname=#FirstName#&myurl=http://
www.Epsilon Interactive.com Label=Home'><EMBED src='http://
www.doublclick.com/flashmovie.swf?myfirstname=#FirstName#&myurl=http://
www.Epsilon Interactive.com Label=Home' WIDTH=550 HEIGHT=400
TYPE='application/x-shockwave-flash'></EMBED></OBJECT>
```

When the message is sent, DREAMmail replaces the URL with a redirect URL, which is passed into the movie. The movie then uses the redirect URL when the user clicks the button or link. The redirect URL records the click, then redirects the user to the actual URL, in this example, [www.epsilon.com](http://www.epsilon.com). In DREAMmail reports, you see the click-through activity for these URLs together with the click-through activity for URLs contained directly in the static part of the HTML.

## Inserting Flash with Personalization and URL Tracking

If you add personalization and URL tracking scripting to the Flash insertion code, the result is:

```
<HTML>
<SCRIPT language=JavaScript>
<!--
    //Detect Flash Plug-in
    var FlashDetected= false;
    var plugin = (navigator.mimeTypes &&
    navigator.mimeTypes["application/x-shockwave-flash"] ?
    navigator.mimeTypes["application/x-shockwave-
    flash"].enabledPlugin : 0);
    if (plugin &&
    parseInt(plugin.description.substring(plugin.description.indexOf(
    ".")-1)) >= 4) {
        //Flash 4 or greater is available
        FlashDetected = true;
    }
-->
</SCRIPT>
<SCRIPT language=VBScript>
<!--
    'Detect Flash ActiveX
    on error resume next
    FlashDetected =
    IsObject(CreateObject("ShockwaveFlash.ShockwaveFlash"))
-->
</SCRIPT>
<BODY>
    <TABLE border=0 cellpadding=0 cellspacing=0>
        <TR>
            <TD background="http://www.Epsilon Interactive.com/
            alternate_image.gif" width=550 height=400>
        <SCRIPT language=JavaScript>
        <!--
        if (FlashDetected)
        document.write(<OBJECT classid=clsid:D27CDB6E-AE6D-11cf-96B8-
        444553540000 width=550 height=400 border=0><PARAM NAME='Src'
        VALUE='http://www.Epsilon Interactive.com/
        flashmovie.swf?myfirstname=#FirstName#&myurl=<URL URL=http://
        www.Epsilon Interactive.com Label=Home>'><EMBED src='http://www.Epsilon
        Interactive.com/flashmovie.swf?myfirstname=#FirstName#&myurl=<URL
        URL=http://www.Epsilon Interactive.com Label=Home>' WIDTH=550
        HEIGHT=400 TYPE='application/x-shockwave-flash'></EMBED></OBJECT>
        -->
        </SCRIPT>
```

```
</TD>
</TR>
</TABLE>
If you cannot see the movie, <a href="http://www.Epsilon
Interactive.com/flashmovie.swf">click here.</a>
</BODY>
```

# EXAMPLE OF DYNAMICALLY PERSONALIZED MESSAGES

Script content segments help to target specific groups of subscribers, based on information in their profile fields that can often be used in various combinations. You can create script content segments based on information for which you have profile data.

This chapter shows how simple messages can be personalized using different types of script content segments.

This chapter discusses the following topics:

- [Information for Personalizing the Example Messages](#)
- [Example of Entertainment Content](#)
- [Content Segments Used in the Example Messages](#)
- [Results of the Dynamic Personalization Scenarios.](#)

# Information for Personalizing the Example Messages

This section discusses the following topics:

- [Message scenario](#)
- [Subscriber profile fields](#)
- [Requests used in generating the messages.](#)

## Message scenario

Tantex, a media holding company, has expanded into the entertainment field. Every week it sends out a newsletter detailing what is hot, what is not, and what is on. The following sample message and dynamic personalization scripting represent a simplified version of the newsletter. This newsletter uses several of the methods for personalizing offers including:

- geographic location
- demographic information
- combinations of multiple messages
- URLs specific to the message.

## Subscriber profile fields

Before you can create dynamic personalization scripts, you must choose the profile fields that you want to use to create your scripts.

For the example, we are using only part of the subscriber profile. Fields that are not required for the example, such as email address, have been left out. A *1* indicates that the subscriber expressed interest in the area when he or she subscribed to the list. If no interest was shown, the field is blank.

TABLE: Profile fields used in examples

First Name	Tanya	Ann	David	Rick
City	Seattle	Chicago	Las Vegas	New York
Gender	Female	Female	Male	Male
Movies	1	1	1	1
Music	1	1		1
Pop	1	1		1
Classical	1			1

TABLE: Profile fields used in examples

First Name	Tanya	Ann	David	Rick
Jazz		1		1
Sports		1	1	
Hockey		1	1	
Basketball			1	

Requests used in generating the messages

The requests used in these examples have been written as *DMConnect* requests. These request are referenced in the table below.

TABLE: CreateContent requests used to personalize the message

Content	Description
<a href="#">MyFirstName</a>	This request uses an IF Statement to include either the subscriber's first name or a generic greeting.
<a href="#">MovieTips_March_19_2005</a>	This request places a script content segment for subscribers interested in movies, in text-format messages.
<a href="#">MovieText_March_19_2005</a>	This request places a script content segment for movie tips.
<a href="#">MovieTips_March_19_2005H</a>	This request places a script content segment for subscribers interested in movies, in HTML- and AOL-format messages.
<a href="#">MovieText_March_19_2005HH</a>	This request places a script content segment for movie tips.
<a href="#">MusicTips_March_19_2005</a>	This request places a script content segment for subscribers interested in music, in text-format messages.
<a href="#">MusicTips_March_19_2005H</a>	This request places a script content segment for subscribers interested in music, in HTML- and AOL-format messages.
<a href="#">SportsTips_March_19_2005</a>	This request places a script content segment for subscribers interested in sports, in text-format messages.

TABLE: CreateContent requests used to personalize the message

Content	Description
<a href="#">SportsTips_March_19_2005H</a>	This request places a script content segment for subscribers interested in sports, in HTML- and AOL-format messages.



## Example of Entertainment Content

This *DMConnect* request creates the message, called *WeekendTips\_March\_19\_2005*, which contains script content segment tags.

```
<DMConnect>
  <CreateMailing>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <MailingType>StandardMailing</MailingType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <CampaignName>EntertainmentTips</CampaignName>
    <MailingName>WeekendTips_March_19_2005</MailingName>
    <ToList>
      <ListName>MovietainList</ListName>
      <ListName>SportstainList</ListName>
      <ListName>MusictainList</ListName>
    </ToList>
    <Subject>Weekend Tips for March 19 2005</Subject>
    <MessageContent>
      <TextBody>
        <BodyText>
```

Dear #CONTENT:MyFirstName#,

Don't just sit around the house on the weekend of March 19.

```
<If condition="#Movies='1'#">
#CONTENT:MovieTips_March_19_2005#
<EndIf>
```

```
<If condition="#Music='1'#">
#CONTENT:MusicTips_March_19_2005#
<EndIf>
```

```
<If condition="#Sports='1'#">
#CONTENT:SportsTips_March_19_2005#
<EndIf>
```

Enjoy life to the fullest. Keep up to date with the latest in the world of entertainment. Go to our website and find out what to do this weekend.

<http://www.weekends.com>

Become one of our VIPs and receive all the latest information regularly. It's easy; simply go to our website to register.

Enjoy your weekend!

Tantex ]]>

```

        </BodyText>
        <TrackedUrls>
            <URL>
                <URLName>http://www.weekends.com</URLName>
                <URLLabel>weekends</URLLabel>
            </URL>
        </TrackedUrls>
    </TextBody>
    <HtmlBody>
        <BodyText>
<![CDATA[ <HTML> <body>

```

Dear #CONTENT:MyFirstName#, <br>

<br>

Don't just sit around the house on the weekend of March 19.<br>

<br>

<If condition="#Movies='1'#">

```

    <font face="arial">#CONTENT:MovieTips_March_19_2005H# </font>
<EndIf>

```

<If condition="#Music='1'#">

```

<font face="arial">#CONTENT:MusicTips_March_19_2005H# </font>
<EndIf>

```

<If condition="#Sports='1'#">

```

    <font face="arial">#CONTENT:SportsTips_March_19_2005H# </font>
<EndIf>

```

Enjoy life to the fullest. Keep up to date with the latest in the world <br>of entertainment. Go to our website and find out what to do this weekend. <br>

<br>

<font face="arial"><a href="http://www.weekends.com">Click here for the latest info. </a></font><br>

<br>

Become one of our VIPs and receive all the latest information <br>regularly. It's easy; simply go to our website to register.<br>

<br>

Enjoy your weekend! <br>

```

<br>
Tantex </body></HTML> ]]>

    </BodyText>
    <TrackedUrls>
    <URL>
        <URLName>http://www.weekends.com</URLName>
        <URLLabel>weekends</URLLabel>
    </URL>
    </TrackedUrls>
</HtmlBody>
<AolBody>
    <BodyText>
<![CDATA[<HTML> <body>

Dear #CONTENT:MyFirstName#, <br>

<br>

Don't just sit around the house on the weekend of March 19. <br>
<br>

<If condition="#Movies='1'#">
    <font face="arial">#CONTENT:MovieTips_March_19_2005H# </font>
<EndIf>

<If condition="#Music='1'#">
    <font face="arial">#CONTENT:MusicTips_March_19_2005H# </font>
<EndIf>

<If condition="#Sports='1'#">
    <font face="arial">#CONTENT:sportsTips_March_19_2005H# </font>
<EndIf>

Enjoy life to the fullest. Keep up to date with the latest in the world <br>
of entertainment. Go to our website and find out what to do this weekend. <br>
<br>
<font face="arial"><a href="http://www.weekends.com">Click here for the
latest info. </a></font><br>
<br>
Become one of our VIPs and receive all the latest information <br>
regularly. It's easy; simply go to our website to register.<br>
<br>
Enjoy your weekend! <br>
<br>
Tantex </body></HTML> ]]>

```

```
</BodyText>
<TrackedUrls>
  <URL>
    <URLName>http://www.weekends.com</URLName>
    <URLLabel>weekends</URLLabel>
  </URL>
</TrackedUrls>
</AolBody>
</MessageContent>
<HtmlOpenRateFlag>True</HtmlOpenRateFlag>
<AutosenseFlag>True</AutosenseFlag>
<RetryAutosenseFlag>True</RetryAutosenseFlag>
<IncludeCampaignFooterFlag>True</IncludeCampaignFooterFlag>
</CreateMailing>
</DMConnect>
```

## Content Segments Used in the Example Messages

This section contains the following topics:

- [MyFirstName](#)
- [MovieTips\\_March\\_19\\_2005](#)
- [MovieText\\_March\\_19\\_2005](#)
- [MovieTips\\_March\\_19\\_2005H](#)
- [MovieText\\_March\\_19\\_2005H](#)
- [MusicTips\\_March\\_19\\_2005](#)
- [MusicTips\\_March\\_19\\_2005H](#)
- [SportsTips\\_March\\_19\\_2005](#)
- [SportsTips\\_March\\_19\\_2005H](#)
- [Message to Tanya](#)
- [Message to Ann](#)

### MyFirstName

Tantex wants to send out the message addressed to their subscribers' first name, but are concerned that not all of the profiles include the first name. This dynamic personalization script inserts the text *friend* if the *FirstName* field is empty. If the field is not empty, it inserts the contents of the *FirstName* field from the subscriber's profile.

This *DMConnect* request creates the content segment called *MyFirstName*, referenced in the *CreateMailing* request.

```
<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MyFirstName</ContentName>
    <ContentText><![CDATA[
<If condition="#IsNull(FirstName) OR FirstName=''">
  Friend
<Else>
  #FirstName#
```

```

<EndIf>
]]>
    </ContentText>
  </CreateContent>
</DMConnect>

```

## MovieTips\_March\_19\_2005

This *DMConnect* request creates the content segment called *MovieTips\_March\_19\_2005* in the *CreateMailing* request.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MovieTips_March_19_2005</ContentName>
    <ContentText><![CDATA[
      <Switch type="stringi" value="#City#">
        <Case match="New York">
          #CONTENT:MovieText_March_19_2005#
        </Case>

        <Case match="Los Angeles">
          #CONTENT:MovieText_March_19_2005#
        </Case>

        <Case match="Miami">
          #CONTENT:MovieText_March_19_2005#
        </Case>

        <Case match="Chicago">
          #CONTENT:MovieText_March_19_2005#
        </Case>

        <Default>
          Check out http://www.movietan.com for the latest movie reviews.
        </Default>
      </Switch>
    </ContentText>
    <TrackedUrls>
      <URL>
        <URLName>http://www.movietan.com</URLName>
        <URLLabel>movies</URLLabel>
      </URL>
    </TrackedUrls>
  </CreateContent>
</DMConnect>

```

```

        </URL>
    </TrackedUrls>
</CreateContent>
</DMConnect>

```

## MovieText\_March\_19\_2005

This *DMConnect* request creates the content segment called *MovieText\_March\_19\_2005*, referenced in the *MovieText\_March\_19\_2005* request.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MovieText_March_19_2005</ContentName>
    <ContentText><![CDATA[
      <If condition="#Gender='Female'#">
        See the latest Romantic Comedy in a big screen theater! Check out
        http://www.movietan.com for the latest movie reviews.
      <Else>
        Action Movies are more exciting on a big screen! Check out http://
        www.movietan.com for the latest movie reviews.
      <EndIf>
    ]]>
    </ContentText>
    <TrackedUrls>
      <URL>
        <URLName>http://www.movietan.com</URLName>
        <URLLabel>movies</URLLabel>
      </URL>
    </TrackedUrls>
  </CreateContent>
</DMConnect>

```

## MovieTips\_March\_19\_2005H

This *DMConnect* request creates the content segment called *MovieTips\_March\_19\_2005H* in the *CreateMailing* request. It is the HTML version of *MovieTips\_March\_19\_2005* and adds the HTML coding necessary for use in the message.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>

```

```

        <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MovieTips_March_19_2005H</ContentName>
    <ContentText><![CDATA[
    <Switch type="stringi" value="#City#">

        <Case match="New York">
            #CONTENT:MovieText_March_19_2005H#
        </Case>

        <Case match="Los Angeles">
            #CONTENT:MovieText_March_19_2005H#
        </Case>

        <Case match="Miami">
            #CONTENT:MovieText_March_19_2005H#
        </Case>

        <Case match="Chicago">
            #CONTENT:MovieText_March_19_2005H#
        </Case>

        <Default>
        <a href="http://www.movietan.com">Click here for the latest movie
        reviews.</a><br><br>
        </Default>
    </Switch> ]]>
    </ContentText>
    <TrackedUrls>
        <URL>
            <URLName>http://www.movietan.com</URLName>
            <URLLabel>movies</URLLabel>
        </URL>
    </TrackedUrls>
    </CreateContent>
</DMConnect>

```

## MovieText\_March\_19\_2005H

This *DMConnect* request creates the content segment called *MovieText\_March\_19\_2005H*, referenced in the *MovieText\_March\_19\_2005H* request.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>

```



```

        <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
        <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MovieText_March_19_2005H</ContentName>
    <ContentText><![CDATA[
    <If condition="#Gender='Female'#">
    See the latest Romantic Comedy in a big screen theater! <br>
    <a href="http://www.movietan.com">Click here for the latest movie
    reviews.</a><br><br>
    <Else>
    Action Movies are more exciting on a big screen! <br>
    <a href="http://www.movietan.com">Click here for the latest movie
    reviews.</a><br> <br>
    <EndIf>
    ]]>
    </ContentText>
    <TrackedUrls>
        <URL>
            <URLName>http://www.movietan.com</URLName>
            <URLLabel>movies</URLLabel>
        </URL>
    </TrackedUrls>
    </CreateContent>
</DMConnect>

```

## MusicTips\_March\_19\_2005

This *DMConnect* request creates the content segment called *MusicTips\_March\_19\_2005*, referenced in the *CreateMailing* request.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MusicTips_March_19_2005</ContentName>
    <ContentText><![CDATA[
    <If condition="#Jazz='1'#">
      <Switch type="stringi" value="#City#">

        <Case match="New York">
          Come and hear the funky jazz trio at New York's primo jazz club.

```

```
Check out http://www.musicstan1.com for the latest on the world of
blues and jazz.
</Case>

<Case match="Chicago">
This Saturday night come down and hear the blues as they are meant
to be played. Check out http://www.musicstan1.com for the latest on
the world of blues and jazz.
</Case>

<Case match="New Orleans">
Traditional New Orleans jazz and zydeco music festival. Check out
http://www.musicstan1.com for the latest on the world of blues and
jazz.
</Case>

<Default>
Check out http://www.musicstan1.com for the latest on the world of
blues and jazz.
</Default>
</Switch>
<EndIf>

<If condition="#Pop='1'#">
  <Switch type="string" value="#City#">

    <Case match="New York">
First stop of their new World Tour! Check out http://
www.musicstan2.com for the latest on the world of pop music.
</Case>

    <Case match="Seattle">
Putting the grunge back into rock! Check out http://
www.musicstan2.com for the latest on the world of pop music.
</Case>

    <Case match="San Francisco">
"The Summer of Love" - a 60's revival concert! Check out http://
www.musicstan2.com for the latest on the world of pop music.
</Case>

    <Default>
Check out http://www.musicstan2.com for the latest on the world of
pop music.
</Default>
  </Switch>
</If>
```

```

<If condition="#Classical='1'#">

  <Switch type="stringi" value="#City#">

    <Case match="New York">
      Hear opera at its finest! Check out http://www.musictan3.com for
      the latest on the world of classical music.
    </Case>

    <Case match="Boston">
      Let's pop back to Bach! Check out http://www.musictan3.com for the
      latest on the world of classical music.
    </Case>

    <Case match="Los Angeles">
      A starry night and classical music, what could be better! Check
      out http://www.musictan3.com for the latest on the world of
      classical music.
    </Case>

    <Default>
      Check out http://www.musictan3.com for the latest on the world of
      classical music.
    </Default>
  </Switch>
</If>
]]>
</ContentText>
<TrackedUrls>
  <URL>
    <URLName>http://www.musictan1.com</URLName>
    <URLLabel>jazz</URLLabel>
  </URL>
  <URL>
    <URLName>http://www.musictan2.com</URLName>
    <URLLabel>pop</URLLabel>
  </URL>
  <URL>
    <URLName>http://www.musictan3.com</URLName>
    <URLLabel>classical</URLLabel>
  </URL>
</TrackedUrls>
</CreateContent>
</DMConnect>

```

## MusicTips\_March\_19\_2005H

This *DMConnect* request creates the content segment called *MusicTips\_March\_19\_2005H*, referenced in the *CreateMailing* request. It is the

HTML version of the previous script and adds the HTML coding necessary for use in the message.

```
<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>MusicTips_March_19_2005H</ContentName>
    <ContentText><![CDATA[
<If condition="#Jazz='1'#">

  <Switch type="stringi" value="#City#">

    <Case match="New York">
      Come and hear the funky jazz trio at New York's primo jazz
      club.<br>
      <a href="http://www.musictan1.com"> Click here for the latest on
      the world of blues and jazz.</a><br><br>
    </Case>

    <Case match="Chicago">
      This Saturday night come down and hear the blues<br>as they are
      meant to be played.<br>
      <a href="http://www.musictan1.com"> Click here for the latest on
      the world of blues and jazz.</a><br><br>
    </Case>

    <Case match="New Orleans">
      Traditional New Orleans jazz and zydeco music festival. <br>
      <a href="http://www.musictan1.com"> Click here for the latest on
      the world of blues and jazz.</a><br><br>
    </Case>

    <Default>
      <a href="http://www.musictan1.com"> Click here for the latest on
      the world of blues and jazz.</a><br><br>
    </Default>
  </Switch>
</EndIf>

<If condition="#Pop='1'#">

  <Switch type="stringi" value="#City#">
```

```
<Case match="New York">
First stop of their new World Tour! <br>
<a href="http://www.musictan2.com"> Click here for the latest on
the world of pop music.</a><br><br>
</Case>

<Case match="Seattle">
Putting the grunge back into rock! <br>
<a href="http://www.musictan2.com"> Click here for the latest on
the world of pop music.</a><br><br>
</Case>

<Case match="San Francisco">
"The Summer of Love" - a 60's revival concert! <br>
<a href="http://www.musictan2.com"> Click here for the latest on
the world of pop music. </a><br> <br>
</Case>

<Default>
<a href="http://www.musictan2.com">Click here for the latest on
the world of pop music.</a><br><br>
</Default>
</Switch>
<EndIf>

<If condition="#Classical='1'#">

<Switch type="stringi" value="#City#">

<Case match="New York">
Hear opera at its finest! <br>
<a href="http://www.musictan3.com">Click here for the latest on
the world of classical music.</a><br><br>
</Case>

<Case match="Boston">
Let's pop back to Bach! <br>
<a href="http://www.musictan3.com">Click here for the latest on
the world of classical music.</a><br><br>
</Case>

<Case match="Los Angeles">
A starry night and classical music, what could be better? <br> <a
href="http://www.musictan3.com"> Click here for the latest on the
world of classical music.</a> <br><br>
</Case>
```

```

        <Default>
        <a href="http://www.musictan3.com">Click here for the latest on
        the world of classical music.</a><br><br>
        </Default>
    </Switch>
</EndIf>
]]>
</ContentText>
<TrackedUrls>
    <URL>
        <URLName>http://www.musictan1.com</URLName>
        <URLLabel>jazz</URLLabel>
    </URL>
    <URL>
        <URLName>http://www.musictan2.com</URLName>
        <URLLabel>pop</URLLabel>
    </URL>
    <URL>
        <URLName>http://www.musictan3.com</URLName>
        <URLLabel>classical</URLLabel>
    </URL>
</TrackedUrls>
</CreateContent>
</DMConnect>

```

## SportsTips\_March\_19\_2005

This *DMConnect* request creates the content segment called *SportsTips\_March\_19\_2005*, referenced in the *CreateMailing* request.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>SportsTips_March_19_2005</ContentName>
    <ContentText><![CDATA[
<If condition="#Basketball='1'#">

    <Switch type="stringi" value="#City#">

      <Case match="New York">
        Come and see basketball at its finest! Check out http://
        www.sportstan1.com for the latest on the world of basketball.
      </Case>

```

```

    <Case match="Miami">
    Here come the play offs and we're on fire! Check out http://
    www.sportstan1.com for the latest on the world of basketball.
    </Case>

    <Case match="Las Vegas">
    Wild horses can't stop us and we're leading our division! Check
    out http://www.sportstan1.com for the latest on the world of
    basketball.
    </Case>

    <Default>
    Check out http://www.sportstan1.com for the latest on the world of
    basketball.
    </Default>
  </Switch>
</Switch>
<EndIf>

<If condition="#Hockey='1'#">

  <Switch type="stringi" value="#City#">

    <Case match="New York">
    Battle for New York! See who gets the biggest bite out of the Big
    Apple! Check out http://www.sportstan2.com for the latest on the
    world of hockey.
    </Case>

    <Case match="Phoenix">
    Join the pack, come out and support your team! Check out http://
    www.sportstan2.com for the latest on the world of hockey.
    </Case>

    <Case match="Detroit">
    The Motor City's team is in the Driver's seat! Check out http://
    www.sportstan2.com for the latest on the world of hockey.
    </Case>

    <Default>
    Check out http://www.sportstan2.com for the latest on the world of
    hockey.
    </Default>
  </Switch>
</Switch>
]]>
  </ContentText>
</TrackedUrls>

```

```

        <URL>
        <URLName>http://www.sportstan1.com</URLName>
        <URLLabel>basketball</URLLabel>
    </URL>
    <URL>
        <URLName>http://www.sportstan2.com</URLName>
        <URLLabel>hockey</URLLabel>
    </URL>
</TrackedUrls>
</CreateContent>
</DMConnect>

```

## SportsTips\_March \_19\_2005H

This *DMConnect* request creates the content segment called *SportsTips\_March\_19\_2005H*, referenced in the *CreateMailing* request. It is the HTML version of the previous script (see [SportsTips\\_March\\_19\\_2005](#)) and adds the HTML coding necessary for use in the message.

```

<DMConnect>
  <CreateContent>
    <AcknowledgementsTo>
      <EmailAddress>anyone@epsiloninteractive.com</EmailAddress>
      <Option>9</Option>
    </AcknowledgementsTo>
    <ContentType>ScriptContent</ContentType>
    <ClientName>Tantex</ClientName>
    <SiteName>Weekends</SiteName>
    <ContentName>SportsTips_March_19_2005H</ContentName>
    <ContentText><![CDATA[
<If condition="#Basketball='1'#">

    <Switch type="stringi" value="#City#">

      <Case match="New York">
        Come and see basketball at its finest! <br><a href="http://
        www.sportstan1.com"> Click here for the latest on the world of
        basketball. </a><br><br>
      </Case>

      <Case match="Miami">
        Here come the play offs and we're on fire!<br><a href= "http://
        www.sportstan1.com"> Click here for the latest on the world of
        basketball.</a><br><br>
      </Case>

      <Case match="Dallas">
        Wild horses can't stop us and we're leading our division! <br><a
        href= "http://www.sportstan1.com"> Click here for the latest on

```



```

        the world of basketball.</a><br><br>
    </Case>

    <Default>
    <a href="http://www.sportstan1.com">Click here for the latest on
    the world of basketball.</a><br><br>
    </Default>
</Switch>
<EndIf>

<If condition="#Hockey='1'#">

    <Switch type="stringi" value="#City#">

        <Case match="New York">
        Battle for New York! See who gets the biggest bite out of the Big
        Apple! <br>
        <a href="http://www.sportstan2.com"> Click here for the latest on
        the world of hockey.</a><br><br>
        </Case>

        <Case match="Phoenix">
        Join the pack, come out and support your team! <br>
        <a href="http://www.sportstan2.com"> Click here for the latest on
        the world of hockey.</a><br><br>
        </Case>

        <Case match="Detroit">
        The Motor City's team is in the Driver's seat! <br>
        <a href="http://www.sportstan2.com"> Click here for the latest on
        the world of hockey.</a><br><br>
        </Case>

        <Default>
        <a href="http://www.sportstan2.com">Click here for the latest on
        the world of hockey.</a><br><br>
        </Default>
    </Switch>
<EndIf>
]]>

</ContentText>
<TrackedUrls>
    <URL>
        <URLName>http://www.sportstan1.com</URLName>
        <URLLabel>basketball</URLLabel>
    </URL>
    <URL>
        <URLName>http://www.sportstan2.com</URLName>

```

```
        <URLLabel>hockey</URLLabel>
    </URL>
</TrackedUrls>
</CreateContent>
</DMConnect>
```

## Results of the Dynamic Personalization Scenarios

The messages created with the requests shown in this chapter are generated and when sent, are displayed in the subscriber's email application as shown below.

### Message to Tanya

Dear Tanya,

Don't just sit around the house on the weekend of March 19.

[Click here for the latest movie reviews.](#)

Put the grunge back into rock!

[Click here for the latest on the world of pop music.](#)

[Click here for the latest on the world of classical music.](#)

Enjoy life to the fullest. Keep up to date with the latest in the world of entertainment. Go to our website and find out what to do this weekend.

[Click here for the latest info.](#)

Become one of our VIPs and receive all the latest information regularly. It's easy; simply go to our website to register.

Enjoy your weekend!

Tantex

### Message to Ann

Dear Ann,

Don't just sit around the house on the weekend of March 19.

See the latest Romantic Comedy in a big screen theater!

[Click here for the latest movie reviews.](#)

This Saturday night come down and hear the blues as they were meant to be played.

[Click here for the latest on the world of blues and jazz.](#)

[Click here for the latest on the world of pop music.](#)

[Click here for the latest on the world of hockey.](#)

Enjoy life to the fullest. Keep up to date with the latest in the world of entertainment. Go to our website and find out what to do this weekend.

[Click here for the latest info.](#)

Become one of our VIPs and receive all the latest information regularly. It's easy; simply go to our website to register.

Enjoy your weekend!

Tantex

# APPENDIX - SCRIPTING LANGUAGE LIMITATIONS

Script content segments and the scripting language in general have several limitations that you must take into account.

This section discusses the following topics:

- [Nesting limitations](#)
- [Recursions](#)
- [Number of conditions](#)
- [Error messages](#)
- [Related topics](#).

## Nesting limitations

Although there are no direct limits on the number of dynamic personalization conditions or on the number of levels of nesting, the logic can become confusing when you exceed 10 nested elements. For more information about nesting conditions, see [Nested conditions](#).

## Recursions

Scripted content segments can reference each other, but cannot reference themselves, as this would cause an endless loop.

## Number of conditions

Dynamic personalization scripts are processed in memory the first time that a dynamic personalization condition is encountered for a message. As a result, the more content in each condition, the less conditions that you can have.

## Error messages

Messages that contain or reference a syntax error in dynamic personalization scripting cannot be sent. The message appears in *Message Manager* with a status of *error*. This error message does not define either the nature or the location of the error.

## Related topics

[Creating Script Content Segments in DREAMmail](#)  
[Creating Script Content Segments in DMConnect](#).

## A

- ActiveX controls 38
- advanced messaging, description of 2
- AND operator 18
- AND...OR conditions 30
- AOL format
  - expression for detecting 24

## B

- Boolean operators 18

## C

- capitalize, changing text to initial capitals 24
- <Case> tag 28
- comparison operators 18
- conditional content segments 5
- conditions in scripting language
  - limits to number of 72
  - order and priority of 34
- content segments
  - conditional 5
  - editing 14
  - names of 11
  - referencing 12, 13
  - script 5
  - static 5
  - types of 5

## D

- data types, scripting language 17
- date operators 21
- <Default> tag 28
- DMConnect*
  - referencing script content segments in 13
- DREAMmail*
  - creating script content segments in 9
- dynamic personalization
  - definition of 3
  - error messages 72
  - examples of 48–71
  - script content segments 8
  - using Flash 42

## E

- <Else> tag 25
- <ElseIf> tag 25, 29
- email applications
  - Flash, support of 38
  - HTML, support of 38
  - web-based 39
- <EndIf> tag 25
- error messages in dynamic personalization 72
- Eudora
  - default HTML rendering engine 38
  - Flash in 38
- examples, scripting language
  - <ElseIf> tag 29
  - AND ...OR conditions 30
  - IF statement nested in a SWITCH statement 31
  - IF statements 26
  - IF...AND conditions 30
  - IF...OR conditions 29
  - RandInt expression 36
  - SWITCH statement nested in an IF statement 31
  - SWITCH statements 28
  - VARIABLE statements 34
- expressions in scripting language
  - about 16
  - date operators 21
  - mathematical operators 20
  - message format detection 24
  - operators 18
  - random variable 35
  - relational operators 18
  - specifying message format 16
  - text formatting 23, 24

## F

- Flash
  - compatibility with email applications 38
  - compatibility with web-based email services 39
  - detecting support of 40
  - dynamic personalization 42
  - Eudora support of 38
  - Hotmail support of 39
  - inserting URLs in 44
  - Internet Explorer support of 38
  - Microsoft Outlook Express support of 38
  - Microsoft Outlook support of 38

- Netscape support of 38
- passing variables to 42
- personalizing 42
- rich media, dynamic personalization using 37
- scripts for detecting support of 40
- tracking links 44
- Yahoo! support of 39

formatting in scripting language

- changing text to initial capitals 24
- changing text to lowercase 23
- changing text to uppercase 23
- text in messages 23

## G

- geotargeting, personalizing messages 2
- GIFs, substituting for Flash 40

## H

- Hotmail, Flash in 39
- HTML format
  - expression for detecting 24
  - Flash in messages 40

## I

- <If> tag 25
- IF statements 16, 25, 31
- IF...AND conditions 30
- IF...OR conditions 29
- Internet Explorer
  - ActiveX control 38
  - Flash in 38

## L

- LCase expression 23
- limitations, number of conditions in scripting language 72
- links, see *URLs*
- Lotus Notes, Flash in 38
- lowercase, changing text to 23

## M

- Macromedia Flash, see *Flash*
- mathematical operators in scripting language 20
- message formats
  - AOL 24
  - expressions that detect 24
  - HTML 24
  - text 24
- messages
  - combining multiple messages 3

- demographic information in 2
- expressions that detect format 24
- Flash in 42
- geographic targeting of 2
- personalizing 1
- referencing content segments in *DREAMmail* 12
- rich media in 3
- URLs in 3, 9, 11, 44
- Microsoft Outlook email, Flash in 38
- Microsoft Outlook Express email, Flash in 38
- multiple conditions in statements
  - AND...OR 30
  - IF...AND 30
  - IF...OR 29

## N

- names of content segments 11
- nested conditions 30
  - IF statement in a SWITCH statement 31
  - SWITCH statement in an IF statement 31
- nesting 72
- Netscape email, Flash in 38
- NOT operator 18

## O

- operators
  - date 21
  - expression 18
  - mathematical 20
  - relational 18
  - scripting language 17, 18
- OR operator 18
- Outlook Express, see *Microsoft Outlook Express*
- Outlook, see *Microsoft Outlook email*

## P

- personalizing messages
  - demographics 2
  - Flash movies 42
  - geography 2
- plug-ins, support of 38
- print statement 23

## R

- RandInt expression 35, 36
- random values, see *RandInt expressions*
- redirect URLs 45
- referencing script content segments
  - in *DMConnect* 13
  - in *DREAMmail* 12

- relational expressions 18
- relational operators 18
- rich media 3
  - about 37
  - Flash 42
  - personalizing email 42
  - types of 37

## S

- script content segments 5
  - process of creating 8
  - referencing in *DMConnect* 13
  - referencing in *DREAMmail* 12
- scripting language 16
  - date operators 21
  - dynamic personalization of messages with 5
  - elements of 16
  - examples of statements in 26–36
  - expressions 16
  - formatting text in messages 23
  - limitations 72
  - message format detection 24
  - operators 17, 18
  - print statements 23
  - relational operators 18
  - specifying formats in messages 16
  - statements 16
  - tags 16
  - text formatting expressions 24
- segments, see *content segments*
- <SetVar> tag 33, 35
- sniffer code, see *Flash, detecting support of*
- statements, scripting language
  - AND...OR conditions 30
  - IF 16, 25
  - IF...AND conditions 30
  - IF...OR conditions 29
  - multiple conditions in 29
  - nested conditions 30
  - overview of 16
  - print 23
  - SWITCH 16, 27
  - VARIABLE 16, 33
- static content segments 5
- SWITCH statements 16, 27, 31
- syntax, Flash
  - passing variables to 42
  - tracking links in 44
- syntax, statements in scripting language
  - <ElseIf> tags 29
  - AND...OR conditions 30

- IF statement tags 25
- IF...AND conditions 30
- IF...OR conditions 29
- print statement 23
- RandInt expression 35
- SWITCH statement tag 27
- VARIABLE statement tag 33

## T

- tags
  - IF statements 25
  - scripting language 16
  - SWITCH statements 27
  - VARIABLE statement 33
- text format, expression for detecting 24
- text formatting expressions in scripting language
  - initial capital letters 24
  - lowercase 23
  - uppercase 23
- tracking URLs 9, 11, 44
- type attribute in dynamic personalization 17

## U

- UCase expression 23
- uppercase, changing text to 23
- URLs
  - adding to messages 3, 44
  - inserting in Flash 44
  - redirect 45
  - tracking 9, 11, 44

## V

- VARIABLE statements 16, 33, 35, 36

## W

- web-based email services, Flash in 39

## Y

- Yahoo! email, Flash in 39