

# HTML Mail with embedded pictures

By Thomas Maul, General Manager, 4D Germany.

TN 05-37

## Summary

---

この **Tech Note** では、画像の組み込まれた **HTML** メールを送信する方法を取り上げます。実際の応用例としては、日次あるいは月次の売上日報を管理グループに配信する場合などが考えられます。**HTML** テキストは、**HTML** テンプレートを使用して作成します。また、**4D** のクイックレポートを活用すれば、**HTML** の知識がなくても、簡単に **HTML** テキストを生成することができます。

## About HTML emails

---

今回の内容では **HTML** 形式の電子メールについて詳しく取り上げることはせずに、**HTML** メールの自動生成だけに焦点を絞っています。**HTML** エディタをユーザに提供し、自由に **HTML** メールを作成できるようにすることを思案中であれば、**MarketBlast** プラグインを利用したメールクライアントである「**4D Email 3.0**」(ベータ版)について検討することをお勧めします。

**HTML** メールの画像は、外部ファイルまたは添付ファイル参照、どちらの形態を取ることができます。大抵の広告メール(**SPAM** および定期購読を含む)は、通常の **WEB** サーバの画像を参照しています。この利点は、メールのサイズを大幅に節約できることです。さらに、サーバへのアクセスが発生するので、相手がメールを読んだことを検知することもできます。そのような理由で、最近では、ユーザのプライバシーを保護するために画像をダウンロードしないように設定されているメールクライアントが増えています。

そのようなメールクライアントでも確実に画像を表示し、**HTML** メールを完全に表示させるための方法として、画像をメールに組み込むことが考えられます。

## About the example

---

サンプルデータベースが自動配信するメールには、販売実績の統計、**ZIP** コード別の売上を示した円グラフ、製品コードで分類した統計とグラフが含まれています。グラフは **4D Chart** で作成したものを変換した **GIF** 画像です。

サンプルでは、**HTML** を生成する **2** 種類の方法を網羅しています。ひとつ目は、**4D** のクイックレポートを使用する方法で、**HTML** の知識がない人でも簡単に利用することができます。ふたつ目は、デベロッパが提供する **HTML** テンプレートとバージョン **2004** の **4D** コマンドである **PROCESS HTML TAGS** を使用する高度な方法です。

いずれの場合も、最後の段階で **PROCESS HTML TAGS** によって画像と表組を合成し、**4D Internet Commands** で配信しています。

結果として、次のようなレポートメールが送信されます。

### 4D Sales Report - 05.11.16

This is an example sales report created by 4th Dimension.

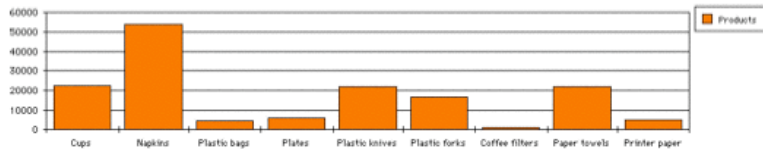
#### Customer Sales Statistic

Company	Zip	Price
Grace Medical Associates	95128	<<<<<<<
Grace Medical Associates	95128	<<<<<<<
Zip Subtotal	95128	<<<<<<<
The Cup and Saucer Cafe	98104	<<<<<<<
Zip Subtotal	98104	<<<<<<<
Edwards, Berry, and Tuft LLP	95118	<<<<<<<
Zip Subtotal	95118	<<<<<<<
Seaside Lodge	95128	<<<<<<<
Zip Subtotal	95128	<<<<<<<
Total		<<<<<<<



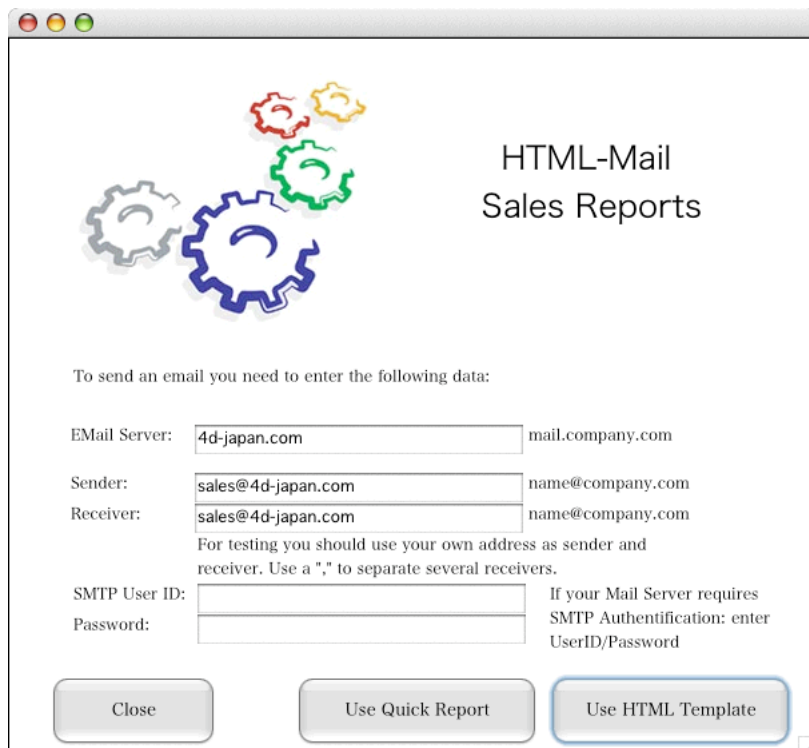
#### Product Sales Statistic

Subtotal		80	4800
Plastic forks	500	8	4800
Plastic forks	500	20	12000
Subtotal		28	16800
Plastic knives	800	150	22050
Subtotal		150	22050
Plates	700	20	6240
Subtotal		20	6240
Printer paper	800	50	5000
Subtotal		50	5000
Total		779	154685



## Using the example

サンプルを起動すると、説明文のダイアログに続いて入力画面が表示されます。



HTML-Mail  
Sales Reports

To send an email you need to enter the following data:

EMail Server:  mail.company.com

Sender:  name@company.com

Receiver:  name@company.com

For testing you should use your own address as sender and receiver. Use a "," to separate several receivers.

SMTP User ID:

Password:

If your Mail Server requires SMTP Authentication: enter UserID/Password

プログラムを実行するには、電子メールクライアントに必要な情報、つまりメールサーバの名前または **IP** アドレスと、**SMTP** 認証が必要であれば、アカウント **ID** とパスワードが必要です。

テスト目的では、送信者と宛先の両方を自分のアドレスにすることが勧められています。

「**Use Quick Report**」「**Use HTML Template**」どちらのボタンをクリックしても画像の組み込まれた **HTML** メールが生成されますが、若干、スタイルとカラーが異なります。

## Customizing the report

以降の部分では、実際のアプリケーションで応用するためにレポート生成のコードをカスタマイズする方法を解説してゆきます。最初にサンプルデータベースのレポートを変更する方法、次に別のアプリケーションにコードを移植する場合の手順を考慮します。最後にユーザがレポートをカスタマイズできるようにする方法を取り上げます。

## Customizing the Quick Report statistic

デモウィンドウを閉じてユーザモードに移行します。

- 1 テーブルリストで「**Invoices**」テーブルを選択します。
  - 2 クイックレポートを起動します。
  - 3 マニュアル操作、あるいはウィザードを使用して、新しいレポートを作成します。
  - 4 プリントプレビューを実行して結果を確認します。
  - 5 どちらの方法でレポートを作成したにしても、ウィザードを開きます。
  - 6 「次のステップ」ボタンを「出力タイプ」になるまでクリックします。
  - 7 「**HTML ファイル**」を選択します。
  - 8 「次のステップ」ボタンをクリックし、「表示形式」で任意の変更を施します。
  - 9 「次のステップ」ボタンをクリックし、「仕上げ」ステップに進みます。
  - 10 「**4D コードをビルド**」をクリックします。バージョン **2004** の新機能であるこのボタンは、インタプリタ版のユーザモードでのみ利用することができます。
- 「コードオプション」はすべてのチェックボックスを有効にし、「コードをビルド」ボタンをクリックします。



このオプションを実行することにより、編集したクイックレポートと同じレポートを作成するための **4D** コードが自動生成されます。

- 11 最後に「クリップボードにペースト」ボタンをクリックしてダイアログおよびクイックレポートを終了します。

- 12 デザインモードに移行し、「**Mail\_Report\_test1**」メソッドを作成してクリップボードの内容

をペーストします。

コードはほとんど完成しており、次の箇所だけの変更を必要としています。

変更前

**QR SET DESTINATION(\$ID;qr HTML file ;")**

変更後

**QR SET DESTINATION(\$ID;qr HTML file ;\$2)** `最後の引数を**\$2**に変更

**13**編集メニューで「置換」を選択し、検索語句を「**=¥**」、置換語句を「**=3D¥**」にして「一括変換」を実行します。

**4D** クイックレポートは、**Web** ブラウザで閲覧されることを想定した **HTML** テンプレートを生成します。**HTML** テンプレートを電子メールで使用するためには、上記の置換が必要です。この変更点に関する情報は「マルチパート **MIME** メール」の項で補足しています。

作成したレポートを実際に使用してみましょう。**Mail\_CreateReport** メソッドを開いて **15** 行目の **Mail\_Report1** を **Mail\_Report\_test1** に変更します。

カスタムモードでファイルメニューの「**HTML Mail**」を選択してデモを起動します。「**Use Quick Report**」ボタンをクリックすると、カスタマイズしたレポートが送信されます。

## **Sending HTML reports from your application**

---

サンプルのコードは、簡単な手順で他のアプリケーションにコピーすることができます。デザインモードで **HTMLMail\_send** メソッドを開き、メソッドメニューで「メソッド書き出し」を実行してメソッドをデスクトップなどに書き出します。

それまで **4D Internet Commands** を使用していなかったのであれば、プラグインをインストールします。

アプリケーションを起動してデザインモードで「**HTMLMail\_send**」を作成します。メソッド読み込みを実行して、書き出してあったメソッドを読み込みます。

サンプルのデータベースフォルダの中にある **HTML\_Mail\_Template** フォルダを **Explorer** または **Finder** で開きます。 **1report.txt** ファイルをストラクチャと同じ階層にコピーし、 **htmlmail\_template.txt** に名称を変更します。このテンプレートには、 **1** または **2** 個の表組と **0-2** 個のピクチャを組み込まれた **HTML** メールの基本形となります。

前項で **Mail\_Report\_test1** を作成したように、クイックレポートを作成してコードをメソッドにペーストします。

ひとまず結果をテストするためにハードコーディングでコードを実行します。実際には、ユーザインタフェースを作成したり、サーバプロセスで自動実行されるようにプログラムを変更したりすることができます。

```
HTMLMail_Title:="This is the title of our mail"
HTMLMail_Text1:="We created this mail to show how the tech note works."

$qreport:=QR New offscreen area
QUERY([Invoice];[ Invoice]Day>(Current date-30)) ` replace this with your table/field names
$path:="HTMLMail_Report1.html"
Mail_Report_test1 ($qreport;$path)
QR EXECUTE COMMAND($qreport;qr cmd new )
QR DELETE OFFSCREEN AREA($qreport)
$err:=HTMLMail_send ("ThomasMaul@de.4D.com";"ThomasMaul@de.4D.com";HTMLMail_Title;
"mail.de.4D.com";"userID";"password")
```

上の例では、件名と本文がハードコードされています。次にクエリを実行して実際のサイズ  
のセクションを作成しています。レコード数が少なければ、**ALL RECORDS** を実行しても構  
いません。最後に、送信者、宛先、サーバをすべてハードコードしてメールを送信します。結  
果が成功であれば、変数で置き換えて柔軟なコードに書き換えることができます。

## Include pictures

---

グラフは **4D Chart** を使用することによって、ダイナミックに生成することができます。 **4D Chart** でグラフを作成する方法には色々あり、今回のサンプルのコードにもグラフを自動生成するコードが含まれています。

始めにスタティックな画像で練習し、次にダイナミックに生成されたグラフの場合について考  
えます。すでに **4D Chart** で作成したグラフがあるのであれば、 **CT Area to picture** を使用して  
変換した画像を使用することができます。

テストには、ピクチャライブラリの画像を使用することができます。ピクチャライブラリには、大抵、数個のボタンピクチャが含まれています。はじめにピクチャの番号を書き留めておきます。

テストコードの最後の行(**HTMLMail\_Send**)の前に次のコードを追加します。

```
C_PICTURE($mypict)
GET PICTURE FROM LIBRARY(851;$mypict)
PICTURE TO GIF($mypict;HTMLMail_Pict1Base64) `convert 4D picture to plain GIF, result is a blob
ENCODE(HTMLMail_Pict1Base64) `convert blob in Base64 encoding
```

2 行目でピクチャを読み込んでいますが、後でこの部分はダイナミックに生成したグラフに置き換えることになります。通常、ビジネスで使用するグラフは **256** 色以下で作成されるので、**GIF** 形式に変換します。そして最後に **BASE64** フォーマットにエンコードします。

最後にピクチャの参照を含む **HTML** テンプレートを用意します。サンプルのデータベースフォルダの中にある **HTML\_Mail\_Template** フォルダを **Explorer** または **Finder** で開き、**1reports\_1picture.txt** ファイルをストラクチャと同じ階層にコピーして名称を **htmlmail\_template.txt** に変更します。(前のテストで使用したものを上書きします。)

テストコードを実行すると、次のような画像を含むメールが送信されます。

## More statistics – more pictures – customize the email

複数の表組または複数のグラフを含む **HTML** メールを作成する場合も同様です。ここではサンプルに含まれている **2reports\_2pictures.txt** テンプレートを基にしながら、オリジナルのテンプレートを作成する方法を説明します。ファイルをテキストエディタで開くと、**HTML** のソースコードをみることができます。

```
<!--4DInclude HTMLMail_Report1.html-->
<P><BR></P>
<IMG align=3Dbaseline border=3D0 hspace=3D0=20
src=3D"cid:001@<!--4DVar HTMLMail_MailID-->">
```

中程にある上記の箇所では、**4D** クイックレポートで作成された **HTML** レポートを組み込んでい

ます。その次の部分は空行を意味しています。最後の行は画像を組み込んでいる部分です。

表組を増やしたければ、**1** 行目をコピーしてドキュメントの名前を変更します。

**4D** のクイックレポートでレポートを作成し、**HTML** ファイルに書き出して、生成したコードから表組を生成します。コードの **\$path** はテンプレートのドキュメント名に修正しておきます。

**HTML** の知識があれば、**HTML** ドキュメントを編集してスタイルを変更したり、テキストを追加したりすることもできます。

画像を増やしたければ、**<IMG>** で始まる行をコピーして、任意の場所に配置します。**cid:001** となっている箇所は、順に **002**、**003** で置き換えます。この番号はユニークな **ID** として使用されます。

ドキュメントを最後までスクロールし、次の箇所をまるごと複製します。

```
-----=_NextPart_001_<!--4DVar HTMLMail_MailID-->
Content-Type: image/gif; name="Pict1.gif"
Content-Transfer-Encoding: base64
Content-ID: <001@<!--4DVar HTMLMail_MailID-->>
Content-Description: Pict1.gif
Content-Disposition: inline; filename="Pict1.gif"
Content-Location: Pict1.gif

<!--4DHTMLVar HTMLMail_Pict1Base64-->
```

**Content-ID** は順に **002**、**003** で置き換えます。ピクチャの名前(**Pict1.gif**)は任意のものを使用できますが、必ずユニークな名前であればなりません。最後に **4D** の **BLOB** 変数名 **HTMLMail\_Pict1Base64** を **HTMLMail\_Pict2Base64** などに変更します。

あとは **4D** メソッドで **HTMLMail\_Pict2Base64** という **BLOB** 変数を作成すれば完了です。

## Allow the end user to modify the report

エンドユーザにクイックレポートエディタを提供して、自由にレポートを編集させるという方法もあります。クイックレポートエディタは、フォームに組み込んでも構いません。作成されたレポートは外部ファイルまたは **BLOB** に保存されます。サンプルでは、**QR BLOB TO REPORT** でオフスクリーンエリアにレポートを作成しました。**QR SET DESTINATION** から



**QR SET HTML TEMPLATE** の間にあるコードをコピーし、**QR RUN** コマンドでコードを実行すれば、自由にレポートが編集できるようになります。**DESTINATION** と **TEMPLATE** が正しくセットされたことをプログラムでチェックするようにすれば、あとはサンプルと同じように動作します。

## Fully customizable HTML reports

---

**4D** のクイックレポートを利用する代わりに、**HTML** テンプレートを自分で作成するという方法もあります。これには **HTML** の基本知識に加えて **4D HTML** タグの理解も必要です。**4D HTML** タグについてはランゲージリファレンスの **Web** サーバの章、**4D HTML** タグの項を参照して下さい。

サンプルにはこの方法の例として **Mail\_ManualReport1** メソッドが含まれているので、自分で作成した **HTML** テンプレートを利用する場合の参考にすることができます。メソッドでは、レポートに出力するデータで4つの配列を作成し、**PROCESS HTML TAGS** で配列を **HTML** レポートに展開しています。**HTML** タグを理解していれば、このようにして複雑なレポートを作成することができます。

## Inside multipart MIME Mails

---

この部分では、画像を組み込まれたメールの仕組みについて解説しています。今回のコードを使用する上で必要ないものですが、背景に興味のある人のために補足しました。

サンプルでは、デバッグのために送信したメールを外部ファイルに書き出すようになっています。メールを送信すると、**htmltest.txt** というテキストファイルが作成されるので、テキストエディタで開いてみて下さい。

ファイルは、**-----=\_NextPart\_002\_05.11.16\_8561095** というヘッダで区切られた幾つかの部分で構成されています。

ファイルには含まれていませんが、冒頭には次のような **SMTP** ヘッダが挿入されています。

**Content-Type: multipart/related; boundary="-----=\_NextPart...."**

この部分は、メールが複数の部分で構成されていることをメールクライアントに知らせ、その区切りとなるセパレータを定義しています。次の部分は **HTML** メールをサポートしないメール

クライアントで表示されるテキストのみのメッセージです。**4D Internet Commands** の **MSG\_GetBody** コマンドを使用すると、この部分が返されます。

**This is a multi-part message in MIME format.**

実際の場面では、この部分にテキストのみで作成されたレポートを含めると良いでしょう。

```
-----_NextPart_001_05.11.16_8561095
Content-Type: multipart/alternative;
boundary="-----_NextPart_002_05.11.16_8561095"
```

**NextPart** の後には **ID** 数値である **001**、および日付とティック数で表現された時刻から成るユニークな番号が続いています。この部分はメールごとに生成され、残りは同一の内容となります。

```
-----_NextPart_002_05.11.16_8561095
Content-Type: text/html;
charset="iso-8859-1"
Content-Transfer-Encoding: quoted-printable
```

次のセクションには、実際の **HTML** 部分が含まれています。ヘッダは、使用キャラクターセットとエンコーディングに関する情報です。

```
-----_NextPart_001_05.11.16_8561095
Content-Type: image/gif;
name="Pict1.gif"
Content-Transfer-Encoding: base64
Content-ID: <001@05.11.16_8561095>
Content-Description: Pict1.gif
Content-Disposition: inline;
filename="Pict1.gif"
Content-Location: Pict1.gif
```

続くセクションは **BASE64** で暗号化された画像を含んでいます。この部分のヘッダには、画像が **GIF** 形式であり、**BASE64** でエンコードされ、**Pict1.gif** という名前であるということを知らせています。**content-ID** は、画像が **HTML** のどこに挿入されるものであるかを知るために必要です。

もう一度 **HTML** 部分に戻り、通常の **HTML** とは異なる箇所がある点に注目して下さい。

```
<div align=3D"right">
```

これはテキストが **quoted-printable** フォーマットであるため、すべての**"=**"を**"=3D"**で置き換える必要があるためです。**3D**は**"=**"に相当する**16**進数コードです。

```
<IMG align=3Dbaseline border=3D0 hspace=3D0=20  
src=3D"cid:002@05.11.16_8561095">
```

画像の参照も幾らか異なっています。ここでも "=" が "=3D" で代替されていることに加え、参照が **URL** ではなく内部の **content-ID** になっています。

このように、**HTML** メールの構造は比較的明快です。 "=" の処理にさえ気をつければ、ほとんどすべての **HTML** をメールにすることができます。画像を組み込む場合はメールの結末にエンコードされたものを参照し、カウンターと日付によるユニークな **content-ID** を使用します。