



# Technical Note 04-49

## *PROCESS HTML TAGS*

By Julien Feasson, Software Engineer, 4D Inc.  
Technical Note 04-49

(原題: Process HTML Tags and more!)

### 概要

バージョン 2004 の新コマンド中、最大の目玉をひとつだけあげるとしたら、それは恐らく PROCESS HTML TAGS でしょう。このコマンドはメール編集、Word との連携から定期メンテナンスにいたるまで、極めて幅広く魅力的なポテンシャルを有しています。

### 説明

PROCESS HTML TAGS は、Web コンテキストからは独立してテキストに存在する 4D タグを処理するコマンドです。4D Web サーバ機能の一部として発展した 4D タグは、4D 変数や 4D 表記への参照や処理命令を HTML に組み込んで、サーバサイドスクリプトによるセミダイナミックな Web ページを生成する能力を提供するものです。これには 4D メソッドの実行や、パラメータの受け渡しなどの機能も含まれます。

PROCESS HTML TAGS の登場によって、これらのタグをいつでも使用できるようになり、データベースのカレントセレクションに応じたテキストを生成したり、メソッドをスクリプトから実行したりすることが可能になりました。

### 4D タグ

4D タグは、HTML のコメントとして記述されます。(<!--#4D...-->) タグの種類は次のとおりです。

4DVAR	4D 変数や 4D 表記を挿入
4DHTMLVAR	上記を HTML コードとして挿入
4DSCRIPT	メソッドを実行
4DINCLUDE	ページを挿入
4DIF,4DELSE,4DENDIF	条件分岐
4DLOOP,4DENDLOOP	ループの実行

これらのタグはどのようなテキストにも組み込むことができ、たとえば電子メールのテンプレートを作成するためにも利用できます。

## 4DSCRIPT

<!--#4DSCRIPT/MyMethod/MyParam-->の形式で記述されたタグは、引数\$1にMyParamを渡してメソッドMyMethodを実行します。はじめにOn Web Authenticationが実行され(存在する場合)、Trueが返されたときにメソッドが実行されます。対象メソッドのプロパティは「4DACTION、4DMETHODおよび4DSCRIPTで利用可能」オプションが有効にされている必要があります。メソッドはテキストを\$0に返し、これがASCIIコード1番ではじまればHTMLとみなされます(変数の場合も同じ)。スクリプトの手前で値を定義した変数を後のメソッドコールで使用することもできます。

PROCESS HTML TAGSには、パラメータとして入出力データをBLOBあるいはテキストで渡します。コマンドの使用目的はSEND HTML BLOBによる送信あるいは.shtml拡張子のついたリクエストに対する返信に限定されず、様々な用途に利用することができます。Webサーバを起動する必要もありません。

Webプロセスから独立して使用された場合、4DINCLUDEにとってのデフォルトフォルダはストラクチャの置かれた階層になります。

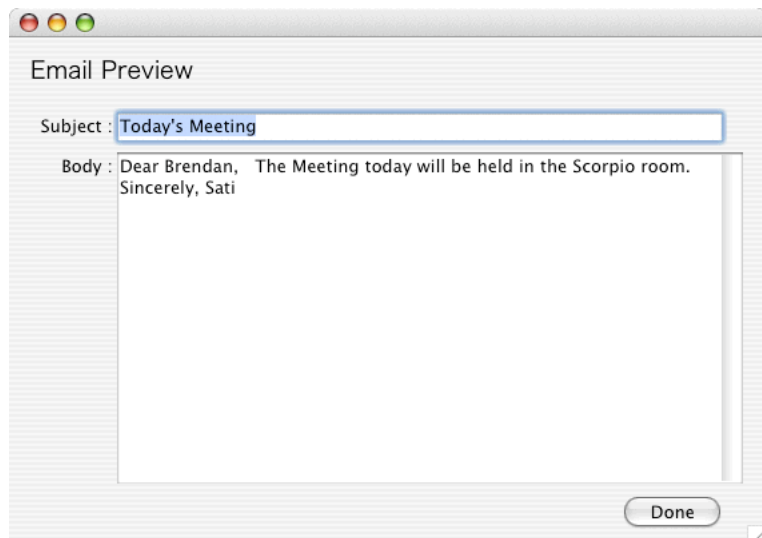
このコマンドの最大の魅力は、どんなテキストに対しても使用できるということであり、とりわけテンプレートとして高い利用価値があります。テンプレートを外部ファイルとして管理すれば、後に4Dタグを編集することによってロジックや処理対象を簡単に編集することができます。

### 例題 1：電子メールのテンプレート

この例では、電子メールのテンプレートに4Dタグを使用しています。

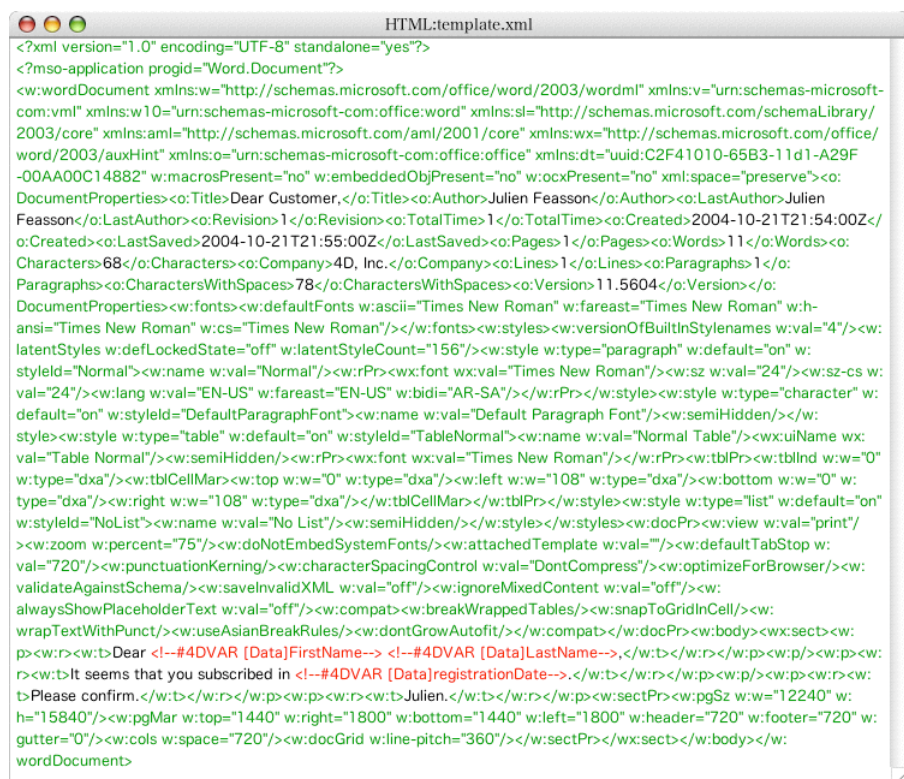
The screenshot shows a window titled '更新: EMailTemplates' (Update: EMailTemplates). The window has a toolbar with navigation buttons (First, Previous, Next, Last) and action buttons (Cancel, Validate). Below the toolbar, the 'EMail Templates' list shows '1 of 2' templates. The selected template is 'Basic Letter' with ID 3. The 'Name' field is 'Basic Letter', 'Email Tags' is empty, 'From' is 'admin@4d.com', and 'Subject' is 'Today's Meeting'. The 'Body' field contains the text: 'Dear <!--4DVAR [People]Name-->, The Meeting today will be held in the Scorpio room. Sincerely, Sati'.

送信時にはカレントセクションに基づいて 4D タグが展開され、本文が完成します。



## 例題 2: Word との連携

Microsoft Word 2003 はファイル形式として XML をサポートするようになりました。XML はテキストファイルであるので、PROCESS HTML TAGS の恩恵を受けることができます。サンプルの Word テンプレートは XML 形式で一部に 4D タグを含んでいます。4D タグをコマンドで処理すれば、データベースを参照した Word ファイルが完成します。



### 例題 3：外部スクリプトで 4D メソッドを実行

4D タグを使用した単純なテキストを 4D の外部スクリプトとして使用することも可能です。スクリプトにはデータベースとのパラメータ受け渡しや条件分岐も含めることができます。PROCESS HTML TAGS の出力結果は破棄するか、4D 内部のエラー処理に活用することができるでしょう。この手法の最大のポイントは、スクリプトが外部ファイルとして管理されていることであり、コンパイルされたデータベースであっても簡単に変更を施すことができるという点です。これによりサーバの停止や再コンパイルなどの手間が省かれます。

たとえばバックアップ、データバリデート、Web サーバ再起動などの定期メンテナンスパッケージをスクリプトとして用意する場合があります。データベースは特定のトリガコールによってこのスクリプトを PROCESS HTML TAGS で展開し、所定のアクションを実行します。

#### セキュリティ対策

4DSCRIPT でコールできるのは、プロパティで許可されたメソッドだけです。さらに、使用者はメソッド名を知らなくてはなりません。加えて実行前には On Web Authentication の認証を通過する必要があります。しかしながら、4D のコマンドはいつでもコールできるという点に留意しなくてはなりません。たとえば、以下の 4D タグはアプリケーションを終了します。

```
<!--#4DIF (QUIT 4D=0)--><!--#4DENDIF-->
```

したがって、外部スクリプトの改ざんを検出するためにチェックサムなどの対策を施すことが望ましいといえますが、それではデベロッパ自身もスクリプトを変更できないので、さらによいのは ENCRYPT BLOB で暗号化したスクリプトを使用するということです。実行前にデベロッパだけが知っているプライベートキーでスクリプトを解読するようにすれば、他人による改ざんを阻止できます。スクリプトを変更する場合は、同じプライベートキーで暗号化することができます。