



# Technical Note 05-31

## HTTP ダウンロード用メソッド

By Thomas Maul, General Manager, 4D Germany.  
Technical Note 05-31

(原題: HTTP Download with a single method)

### 紹介文

4D で HTTP ダウンロードを行なう方法は、Tech Note 02-05、05-18 などで紹介されていますが、より手軽な方法を求める声が依然として多いのが実情です。この Tech Note では HTML ページのダウンロード、データの POST、ピクチャの受け取り、ファイルのダウンロードまで、すべて単一のコマンドで処理することのできるメソッドを紹介しています。4D Insider を使用しなくても、メソッドのコピー&ペーストだけで済むので、インストールは非常に簡単です。

### インストール

サンプルデータベースフォルダに含まれている HTTP\_Download.txt をテキストエディタなどで開いてコピーし、新規メソッド HTTP\_Download を作成してペーストします。もちろん、サンプルデータベースから直接コピーしても構いません。メソッドは 4D Internet Commands プラグインを使用しています。またローカル変数へのポインタを使用しているので、大幅な書き換えをしない限り、バージョン 2004.1 以降でなければ動作しません。

### 例題

Web からピクチャをダウンロードする。

```
$result:=HTTP_Download("picture";->[Logos]Bildfeld;"http://www.test.com/mypict.jpg")
```

Web ページのテキスト(32000 文字)を受け取る。

```
$result:=HTTP_Download("text";->$mytext;"127.0.0.1:8080/4DAction/test")
```

PDF をダウンロードする。

```
$result:=HTTP_Download("blob";->$myblob;"http://www.test.com/demo.pdf")  
BLOB TO DOCUMENT("demo.pdf";$myblob)
```

第一引数には、動作の種類を渡します。text は最初の 32000 バイトを返し、picture は QuickTime によって変換された 4D ピクチャを返し、blob はデータをそのまま返します。第二引数は、返される値に対応するタイプの 4D オブジェクトへのポインタ、第三引数は URL です。

## 詳細

通常は、例題のシンタックスを応用すれば用が足りるはずですが、メソッドにはオプションの引数や機能があり、より高度な処理、たとえばデータの chunk 処理、リダイレクトやクッキーの処理などが可能です。

```
HTTP_Download(Job;Resultpointer;URL;{{Referer;{{Cookie;{{Postdata;{{Timer;{{Redirect}}}}}}}) ->Header
```

引数	タイプ	説明
Job	文字列	動作の種類、たとえば"picture"
Resultpointer	ポインタ	データを受け取る変数へのポインタ
URL	文字列	URL、http://または https://
Referrer	文字列	リファラーURL
Cookie	テキスト	カンマ区切りのクッキー
Postdata	テキスト	POST リクエストのデータ
Timer	倍長整数	タイムアウト、秒単位
Redirect	ブール	リダイレクトの有無
Function result	テキスト	HTTP ヘッダ

HTTP\_Download メソッドは、HTTP または HTTPS の Web サーバから HTML ページ、画像、文書をダウンロードすることができます。

引数 Job は、ダウンロードの種類を指定するために使用します。

text は最初の 32000 バイトをテキスト形式で返します。

picture は QuickTime によって変換された 4D ピクチャを返します。

blob は未処理のデータをそのまま返します。

引数 Resultpointer には、結果を受け取るフィールドまたは変数(ローカル/プロセス)へのポインタを渡します。タイプは Job と一致している必要があります。

引数 URL には、リクエスト対象の URL を渡します。

残る 5 つの引数は任意です。

引数 Referrer には、Web サーバに渡すリファラーを渡します。通常、それは直前に閲覧したページの URL です。

引数 Cookie には、クッキーを渡します。たとえば、4D-font、ID というクッキーを使用する場合、4D-font=sizemediumtext, ID=5 を渡します。Cookie: 4Dfont=sizemediumtext, ID=5 という形式でも構いません。

引数 Postdata は、HTTP フォームの POST アクションを擬似的に実行するためのものです。使用すると、通常の GET ではなく POST メソッドで HTTP リクエストが送信されます。データは x-www-form-urlencoded で送信されるので、スペースが含まれる場合は+に置換する必要があります。8 ビット文字はエンコードする必要がありますが、メソッドにはそのためのツールが用意されています(後述)。特殊な利用に対応できるように、メソッドによる自動変換は行なわれません。以下はスペースおよびウムラウトを含むデータの変換例です。

```
$name:= "Herbert Grönemeyer"  
$err:=HTTP_Download("URL_Encoder";->$name;Mac to Win($name))  
$name:=Replace String($name;" %20";"+")
```

HTTP\_Download メソッドは、何もデータが返されないとデフォルトで 30 秒後にタイムアウトを迎えます。この設定は引数 Timeout で変更することができます。

メソッドは、標準の HTTP リダイレクト(HTTP 301、302、307)に従いますが、引数 Redirect に False を渡すことにより、これを無効にすることができます。リダイレクトの際、メソッドはクッキーを自動的に処理するようになっています。www.amazon.com など一部の Web サーバは、接続後、直ちにリダイレクトを送信し、クッキーの設定を試みています。ブラウザは設定されたクッキーで応答するので、これにより Web サーバはブラウザがクッキーに対応していることを判別できるからです。リダイレクトを無効にすることにより、こうした自動反応を直接的に制御できるようになります。

実行結果として返されるテキストは HTTP のヘッダです。問題はなければ HTTP/1.1 200 OK, から始まります。URL が無効な場合は HTTP/1.1 404 Not found が返されますが、Amazon のように 200 OK を返し、内容に<title>404 - Document Not Found</title>を含むサーバもあります。いずれにしても @200 OK@ でヘッダを解析し、結果を知ることができます。HTTP\_Download メソッド自体が問題に遭遇した場合、以下のエラーが返されます。

- HTTP\_Download - Error: URL required  
パラメータが足りない。
- HTTP\_Download - Error: Wrong result type, blob expected  
ポインタが BLOB でない。
- HTTP\_Download - Error: Wrong result type, picture expected  
ポインタがピクチャでない。
- HTTP\_Download - Error: Wrong result type, text expected  
ポインタがテキストでない。
- HTTP\_Download - Error: HTTP Request not sent, please check the URL  
TCP 接続が確立できないがホスト名は存在する。
- HTTP\_Download - Error: Connection failed  
ホストが見つからない、DNS 失敗、接続失敗。

3 つのダウンロード処理に加え、第一引数に渡すことのできるオプションのツール機能が用意されています。

#### URL\_Encoder

URL や POST で渡すデータに使用できる文字は A-Z、0-9、-/ \* などに限定されており、その他の文字はエンコードする必要があります。たとえば www.test.com/my method という URL は www.test.com/my%20method にエンコードする必要があります。次のコードはその変換を行なうものです。

```
$url:= "www.test.com/my method"  
$err := HTTP_Download("URL_Encoder";->$encodedurl;$url)
```

## Get\_Cookie

返された HTTP ヘッダからクッキーを抽出することができます。クッキーの区切り記号はデフォルトでカンマですが、任意の第 4 引数を使用することにより、自由に指定することができます。

```
HTTPHeader:=HTTP_Download ("text";->TextResult;"http://www.amazon.com/")
$result:=HTTP_Download ("Get_Cookie";->HTTPCookies;HTTPHeader;char(13))
```

## File\_name

通常。URL の末尾にはファイル名が納まりますが、Web サーバによっては URL に番号を使用してファイル名をヘッダに含めるものもあります。このコードは最初にヘッダを調べ、ファイル名が見つけれなければ URL の末尾を返します。

```
$url:=http://www.test.com/song.mp3
$HTTPHeader:=HTTP_Download("blob";->$myblob;$url)
$Filename:=HTTP_Download ("File name";->$HTTPHeader;$url)
```

## 追記

今回はインストールの利便性を考慮し、あえてすべてをひとつのメソッドにまとめました。メソッドは 500 行を超え、筆者が 18 年間の 4D 開発経験で書いた最も長いメソッドになります。長いメソッドはデバッグや管理が大変であり、本来、望ましいものではありません。それでもコンポーネントではなくメソッドにこだわったのは、その用途からして既存のアプリケーション、それも大掛かりなプロジェクトではなく、ちょっとしたプロジェクトに簡単できるようなものを作りたかったからです。4D を一旦、終了し、4D Insider を起動してストラクチャを開くよりも、メソッドのコピー&ペーストだけでインストールできたほうが、その目的にかなっていると考え、今回、このような形での提供となりました。