Apache モジュール mod_mime

説明:	リクエストされたファイルの拡張子とファイルの振る舞い（ハンドラとフィルタ）、内容（MIME タイプ、言語、文字セット、エンコーディング）とを関連付ける
ステータス:	ベース
モジュール識別子:	mime_module
ソースファイル:	mod_mime.c

概要

このモジュールは拡張子を使っていろいろな「メタ情報」をファイルに 関連付けるために使用されます。この情報はドキュメントのファイル名と MIME タイプ、言語、文字セット、エンコーディングとを関連付けます。 この情報はブラウザに送られますし、複数のファイルの中からユーザの好みの ものが選ばれるように、コンテントネゴシエーションでも使われます。 コンテントネゴシエーションに関する詳しい情報は mod_negotiation を見てください。

AddCharset ディレクティブ、 AddEncoding ディレクティブ、 AddHandler ディレクティブ、AddLanguage ディレクティブ、 AddType ディレクティブはすべて、 ファイルの拡張子をメタ情報にマップするために使用されます。 それぞれ、ドキュメントの文字セット（訳注: charset）、content-encoding, content-language, MIME タイプ（content-type）を設定します。 TypesConfig ディレクティブは拡張子を MIME タイプにマップするファイルを指定するために使用されます。

さらに、mod_mime はコンテンツを作成、処理する ハンドラ [1] と フィルタ [2] を設定することができます。AddHandler ディレクティブ、AddOutputFilter ディレクティブ、AddInputFilter ディレクティブは ドキュメントを扱うモジュールやスクリプトを制御します。 MultiviewsMatch ディレクティブは これらのディレクティブが指定したファイルの拡張子を mod_negotiation が Multiviews のマッチをとるときに 考慮するようにできます。

mod_mime はメタ情報をファイル名と関連付けますが、 core サーバにはあるコンテナ（たとえば, <Location>, <Directory>, <Files>）の中のすべてのファイルを特定の メタ情報と関連付けるディレクティブがあります。これらのディレクティブには ForceType, SetHandler, SetInputFilter, SetOutputFilter があります。 コアのディレクティブは mod_mime により定義された ファイル名の拡張子のマッピングすべてを上書きします。

ファイルのメタ情報を変えても Last-Modified ヘッダの値は変わらないことに注意してください。ですから、 それらを変更した場合は、クライアントやプロキシで以前にキャッシュされた コピーがそのときのヘッダとともに使われる可能性があります。 メタ情報（言語、コンテントタイプ、文字セット、エンコーディング）を 変更したときは、すべての訪問者が正しいコンテントヘッダを 受け取るように、影響を受けるファイルに 'touch' コマンドを実行する（最終更新日を更新する）必要があるかもしれません。

トピック

ディレクティブ

参照

- MimeMagicFile
- AddDefaultCharset
- ForceType
- DefaultType
- SetHandler
- SetInputFilter
- SetOutputFilter

複数の拡張子のあるファイル

ファイルは複数の拡張子を持つことができ、拡張子の順番は通常は関係ありません。例えば、ファイル welcome.html.fr がコンテントタイプは text/html に、言語はフランス語にマップされる場合、welcome.fr.html もまったく同じ情報にマップされます。 同じメタ情報にマップされる拡張子が複数あるときには、 右側にあるものが使用されます。たとえば、".gif" が MIME タイプ image/gif にマップされ、".html" が MIME タイプ text/html にマップされる場合は、ファイル welcome.gif.html は MIME タイプ "text/html" に関連付けられます。

複数の拡張子のあるファイルが MIME タイプとハンドラの両方に関連付けられているときは注意する必要があります。 その場合、普通はリクエストがハンドラに関連付けられた モジュールによって扱われることになります。たとえば、拡張子 .imap が (mod_imap の) "imap-file" にマップされていて、 .html が MIME タイプ "text/html" にマップされているときは、ファイル world.imap.html は "imap-file" ハンドラと "text/html" MIME タイプに関連付けられます。ファイルが処理されるときは "imap-file" ハンドラが使用されますので、そのファイルは mod_imap のイメージマップファイルとして扱われることになります。

コンテントエンコーディング

特定の MIME タイプのファイルはインターネットでの転送を簡単にするために、 さらに符号化することができます。これは通常は gzip の ような圧縮のことを指しますが、pgp のような暗号化や、 バイナリファイルを ASCII (テキスト) 形式で送るために考案された UUencoding のことを指すこともあります。

MIME RFC は次のように言っています:

> Content-Encoding エンティティヘッダフィールドはメディアタイプの 修飾子として使われます。それが存在していれば、値はリソースに どの追加の符号化が適用されたか

を示し、Content-Type ヘッダフィールドに 書かれているメディアタイプを得るためにどの復号機構を適用すべきか、も 示していることになります。Content-Encoding は主に、元のメディアタイプの 同一性を失うことなくドキュメントを圧縮することを可能にするために 使用されます。

複数のファイル拡張子（複数の拡張子については 上の節 を参照）使うことで、 ファイルのタイプやエンコーディングを指定することが できます。

たとえば、Microsoft Word のドキュメントがあり、サイズを小さくするために pkzip されているとします。.doc 拡張子が Microsoft Word の ファイルタイプと関連付けられていて、.zip 拡張子が pkzip ファイルエンコーディングと関連付けられていると、ファイル Resume.doc.zip は pkzip された Word ドキュメントである ということがわかります。

クライアントのブラウザにエンコーディング方法を知らせるために、 Apache はリソースと共に Content-Encoding ヘッダを 送ります。

```
Content-encoding: pkzip
```

文字セットと言語

ファイルタイプとファイルエンコーディングの他に重要な情報は ドキュメントの書かれている言語と、どの文字セットでファイルが表示 されるべきか、というものです。たとえば、ドキュメントはベトナムの アルファベットやキリル文字で書かれていて、そのように表示される 必要があるかもしれません。この情報もまた、HTTP ヘッダで 送信されます。

文字セット、言語、エンコーディング、mime タイプはすべて コンテントネゴシエーション (mod_negotiation 参照) の最中に、複数の文字セット、言語、エンコーディング、MIME タイプからなる 代替物があるときにどのドキュメントをクライアントに送るのかを 決定するときに使われます。AddCharset, AddEncoding, AddLanguage, AddType の各ディレクティブで作成された 拡張子の関連付け（と MimeMagicFile でリストされている 拡張子）がこの選択に参加します。AddHandler, AddInputFilter, AddOutputFilter の 各ディレクティブでのみ関連付けられている拡張子は MultiviewsMatch ディレクティブを 使うことでマッチの 処理に含めることも外すこともできます。

Charset

さらに情報を伝えるために、Apache は文書の言語を Content-Language ヘッダで送ることもあります。 また、情報を正しく表示するために使用すべき文字セットを示すために Conten-Type ヘッダに情報を追加することもあります。

```
Content-Language: en, fr
Content-Type: text/plain; charset=ISO-8859-2
```

言語の指定は二文字の短縮形で行なわれます。charset が 使用すべき文字セットの名前です。

AddCharset ディレクティブ

説明:	ファイル名の拡張子を指定された文字セットにマップする
構文:	AddCharset charset extension [extension] ...
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime
互換性:	AddCharset は Apache 1.3.10 以降でのみ使用可能

AddCharset ディレクティブは、与えられた拡張子を指定された charset にマップします。charset は、拡張子 extension を含んでいるファイル名の MIME charset パラメータです。新しいマッピングは既にある他のマッピングに追加され、同じ拡張子 extension のためのマッピングを上書きします。

例:

```
AddLanguage ja .ja
AddCharset EUC-JP .euc
AddCharset ISO-2022-JP .jis
AddCharset SHIFT_JIS .sjis
```

この場合、ドキュメント xxxx.ja.jis は charset が ISO-2022-JP の日本語のドキュメントとして扱われます (xxxx.jis.ja も同様)。AddCharset ディレクティブは、ドキュメントが適切に解釈され表示されるように、 ドキュメントの charset の情報をクライアントに教えるために役に立ちます。 また、サーバがクライアントの charset の優先度に基づいて複数のドキュメントの中からドキュメントを選ぶコンテントネゴシエーション[3]のためにも役に立ちます。

引数 extensionは大文字小文字を区別せず、 最初のドットはあってもなくても構いません。

参照

- mod_negotiation
- AddDefaultCharset

AddEncoding ディレクティブ

説明:	ファイル名の拡張子を指定されたエンコーディング にマップする
構文:	AddEncoding MIME-enc extension [extension] ...
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime

AddEncoding ディレクティブは、与えられた拡張子を指定された エンコーディングにマップします。MIME-enc は、拡張子 extension を含んだドキュメントに使用する MIME エンコーディングです。この新しいマッピングは既にある他のマッピングに追加され、 同じ拡張

子 extension のためのマッピングを上書きします。例:

```
AddEncoding x-gzip .gz
AddEncoding x-compress .Z
```

これは、拡張子 .gz を含むファイル名が x-gzip エンコーディングを使ってエンコードされていることと、拡張子 .Z を含むファイル名が x-compress でエンコードされていることを指定します。

古いクライアントは x-zip と x-compress が返ってくることを期待しますが、標準規格ではそれぞれ gzip と compress と等価であることになっています。Apache は、コンテントエンコーディングの比較をするときには、先頭にある x- を無視します。Apache がエンコーディング付きで応答を返すときは、クライアントが要求した形式 (すなわち、x-foo や foo) を使用します。要するに、この二つのエンコーディングの場合は常に x-gzip と x-compress を使うべきである、ということです。deflate のようなより新しいエンコーディングでは、x- なしで指定してください。

引数 extension は大文字小文字を区別せず、 最初のドットはあってもなくても構いません。

AddHandler ディレクティブ

説明:	ファイル名の拡張子を指定されたハンドラにマップする
構文:	AddHandler handler-name extension [extension] ...
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime

拡張子 extension が名前にあるファイルは指定された handler-name [1] に扱われます。 この新しいマッピングは既にある他のマッピングに追加され、 同じ拡張子 extension のためのマッピングを上書きします。たとえば、拡張子 ".cgi" で終わるファイルを CGI スクリプトとして扱いたいときは、以下の設定をします。

```
AddHandler cgi-script .cgi
```

これを srm.conf か httpd.conf ファイルに記述することで、拡張子 ".cgi" を含むファイルは CGI プログラムとして扱われます。

引数 extension は大文字小文字を区別せず、 最初のドットはあってもなくても構いません。

参照

- SetHandler

AddInputFilter ディレクティブ

説明:	ファイルの拡張子をクライアントのリクエストを処理する フィルタにマップする
構文:	AddInputFilter filter[;filter...] extension [extension ...]
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
ステータス:	ベース
モジュール:	mod_mime
互換性:	AddInputFilter は Apache 2.0.26 以降のみで使用可能。

AddInputFilter はファイルの拡張子 extension を クライアントのリクエストや POST がサーバに来たときに 処理をするフィルタ[2]にマップします。 これは、SetInputFilter[4] ディレクティブも 含め、他の場所で定義されているフィルタに加えられます。 このマッピングはすでにあるものより優先されてマージされ、 同じ extension に対する既存のマッピングを上書きします。

複数のフィルタを指定するときは、データを処理する順番にセミコロンで 繋いで書く必要があります。フィルタと extension との 両方の引数は大文字小文字を区別せず、拡張子の最初のドットは あってもなくても構いません。

AddLanguage ディレクティブ

説明:	ファイル名を指定された言語にマップ
構文:	AddLanguage MIME-lang extension [extension] ...
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime

AddLanguage ディレクティブは、与えられた拡張子を指定された content language にマップします。MIME-lang は、拡張子 extension を含んでいるファイル名の MIME における言語です。 この新しいマッピングは既にあるマッピングに追加され、同じ拡張子 extension のためのマッピングを上書きします。

例:

```
AddEncoding x-compress .Z
AddLanguage en .en
AddLanguage fr .fr
```

この場合、xxxx.en.Z ドキュメントは compress された英語のドキュメントとして扱われます (xxxx.Z.en も同様)。content language はクライアントに通知されますが、 ブラウザがこの情報を使うことはおそらくありません。AddLanguage ディレクティブは、サーバがクライアントの言語の優先度に基づいて複数の ドキュメントの中からドキュメントを選ぶコンテントネゴシエーション[3]のためにより役に立ちます。

複数の言語が同じ拡張子に割り当てられているときは、 最後のものが使用されます。すなわち、次のような場合、

```
AddLanguage en .en
AddLanguage en-uk .en
AddLanguage en-us .en
```

拡張子 ".en" のあるドキュメントは "en-us" として扱われます。

引数 extension は大文字小文字を区別せず、 最初のドットはあってもなくても構いません。

参照

- mod_negotiation

AddOutputFilter ディレクティブ

説明:	ファイル名の拡張子をサーバからの応答を処理するフィルタに マップする
構文:	AddOutputFilter filter[;filter...] extension [extension ...]
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	
ステータス:	ベース
モジュール:	mod_mime
互換性:	AddOutputFilter は Apache 2.0.26 以降でのみ使用可能。

AddOutputFilter ディレクティブは 拡張子 extension をサーバの応答がクライアントに送られる 前に処理するフィルタ[2]を定義します。 これは SetOutputFilter ディレクティブ を含め、他の場所で定義されているフィルタに加えられます。 この新しいマッピングは既にあるマッピングに追加され、同じ拡張子 extension のためのマッピングを上書きします。

例えば、以下の設定はすべての .shtml ファイルを SSI で処理し、 その出力を mod_deflate を使って圧縮します。

```
AddOutputFilter INCLUDES;DEFLATE shtml
```

複数のフィルタを指定するときは、データを処理する順番にセミコロンで 繋いで書く必要があります。フィルタと extension との 両方の引数は大文字小文字を区別せず、拡張子の最初のドットは あってもなくても構いません。

AddType ディレクティブ

説明:	ファイル名の拡張子を指定されたコンテントタイプにマップ
構文:	AddType MIME-type extension [extension] ...
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime

AddType ディレクティブは、 与えられた拡張子を指定されたコンテントタイプにマップします。 MIME-type は拡張子 extension を含んだドキュメントに使用する MIME タイプです。 この新しいマッピングは既にあるマッピングに追加され、同じ拡張子 extension のためのマッピングを上書きします。 このディレクティブは MIME タイプファイル（TypesConfig ディレクティブを参照）に無いマッピングを追加するために使用することができます。

例:

```
AddType image/gif .gif
```

新しい MIME タイプは、TypesConfig ファイルを変更するのではなく、AddType ディレクティブを使って追加することが推奨されています。

引数 extension は大文字小文字を区別せず、 最初のドットはあってもなくても構いません。

参照

- DefaultType
- ForceType

DefaultLanguage ディレクティブ

説明:	あるスコープのすべてのファイルを指定された言語に 設定する
構文:	DefaultLanguage MIME-lang
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime
互換性:	DefaultLanguage は Apache 1.3.4 のみで使用可能

DefaultLanguage ディレクティブは、Apache がディレクティブのスコープ（例えば、その時点の <Directory> の範囲）にある、明示的な言語拡張子（AddLanguage で設定される .fr や .de）のない全てのファイルを、指定された MIME-lang 言語であるとみなすようにします。 これにより、すべてのファイル名を変えることなく、 ディレクトリがオランダ語のコンテントを含んでいる、 というようなことを指定することができます。 拡張子を使用して言語を指定する方法と違い、 DefaultLanguage は一つの言語しか指定できないことに注意してください。

DefaultLanguage ディレクティブが有効でなく、ファイルに AddLanguage で設定された言語の拡張子がないときは、 ファイルには言語属性がないとみなされます。

例

```
DefaultLanguage en
```

参照

- mod_negotiation

ModMimeUsePathInfo ディレクティブ

説明:	path_info コンポーネントをファイル名の一部として扱うように mod_mime に通知する
構文:	ModMimeUsePathInfo On|Off
コンテキスト:	ディレクトリ
ステータス:	ベース
モジュール:	mod_mime
互換性:	Apache 2.0.41 以降

ModMimeUsePathInfo ディレクティブは、 mod_mime の持つディレクティブを リクエストに適用させるために、ファイル名と path_info URL コンポーネントを結合させるために使用します。 デフォルトでは「 off 」で、path_info コンポーネントは無視されます。

このディレクティブは、バーチャルファイルシステムを使用している際に 推奨されるディレクティブです。

例

```
ModMimeUsePathInfo On
```

/bar が存在して（foo.shtml は存在しない）ModMimeUsePathInfo が有効であるとして、/bar/foo.shtml に対するリクエストを発行した場合、 mod_mime は入ってきたリクエストを /bar/foo.shtml として扱い、 AddOutputFileter INCLUDES .shtml のようなディレクティブは INCLUDES フィルタをリクエストに付加させます。 ModMimeUsePathInfo が設定されなければ、 icludes フィルタは付加されません。

MultiviewsMatch ディレクティブ

説明:	MultiViews でのマッチングの検索に含ませる ファイルのタイプを指定する
構文:	MultiviewsMatch [NegotiatedOnly] [Handlers] [Filters] [Any]
コンテキスト:	サーバ設定ファイル, バーチャルホスト, ディレクトリ, .htaccess
上書き:	FileInfo
ステータス:	ベース
モジュール:	mod_mime
互換性:	Apache 2.0.26 以降

MultiviewsMatch を使用することで、 mod_negotiation [5] の Multiviews に 3 種類の異なる挙動をさせることができます。 Multiviews を使用すると、ファイル（例 index.html）に対するリクエストに対して、 ネゴシエーションする拡張子がベースに付いたもの (index.html.en, index.html.fr や index.html.gz) をマッチさせることができます。

NegotiatedOnly オプションでは、ベース名に続く拡張子全てが コンテントネゴシエーションで mod_mime が認識する拡張子（例 文字セット、コンテントタイプ、言語やエンコーディ

ング）に関連付けられていなければなりません。これは副作用の最も少ない 最も厳密な実装で、デフォルトになっています。

ハンドラとフィルタの両方もしくは片方と関連付けられた拡張子を含めるには、MultiviewsMatch ディレクティブに Handler, Filters またはその両方のオプションをセットします。 もし他の条件が同じであれば、最も小さいファイルが送信されます。 例えば、500 文字の index.html.cgi と 1000 バイトの index.html.pl であれば、.cgi のファイルが優先されます。.asis ファイルを利用しているユーザは、 .asis ファイルが asis ハンドラに関連付けられているときには、 Handler オプションの使用を好むでしょう。

最後に、mod_mime が認識しない拡張子であろうとも、 どんな拡張子でもマッチさせるようにすることができます。この挙動は Apache 1.3 の ときと同じもので、予期しない動作、例えば .old や .bak ファイルといったウェブマスタが送信を意図していない ファイルを送信する、といった動作を行なう可能性があります。

例えば次の設定では、ハンドラやフィルタが Multiviews に参加することが できますし、未知のファイルは除外することができます。

```
MultiviewsMatch Handlers Filters
```

参照

- Options

RemoveCharset ディレクティブ

説明:	ファイルの拡張子に関連付けられたすべての文字セット を解除する
構文:	RemoveCharset extension [extension] ...
コンテキスト:	ディレクトリ, .htaccess
ステータス:	ベース
モジュール:	mod_mime
互換性:	RemoveCharset は Apache 2.0.24 以降で使用可能

RemoveCharset ディレクティブ は与えられた拡張子に関連付けられた文字セットを取り消します。 これにより、サブディレクトリにある .htaccess ファイルが親ディレクトリやサーバの設定ファイル から継承した関連付けを取り消すことができます。例えば:

extension は大文字小文字を区別しません。 また、最初のドットはあってもなくても構いません。

例

```
RemoveCharset .html .shtml
```

RemoveEncoding ディレクティブ

説明:	ファイルの拡張子に関連付けられたすべてのコンテントエンコーディングを解除する

構文:	RemoveEncoding extension [extension] ...
コンテキスト:	バーチャルホスト, ディレクトリ, .htaccess
ステータス:	ベース
モジュール:	mod_mime
互換性:	RemoveEncoding は Apache 1.3.13 のみで使用可能

RemoveEncoding ディレクティブは、 与えられた拡張子に関連付けられたエンコーディングを取り消します。 これにより、サブディレクトリにある .htaccess ファイルが親ディレクトリやサーバの設定ファイルから継承した関連付けを 取り消すことができます。例えば:

```
/foo/.htaccess:


AddEncoding x-gzip .gz
AddType text/plain .asc
<Files *.gz.asc>
  RemoveEncoding .gz
</Files>
```

これは、foo.gz は gzip でエンコードされていることを指定しますが、foo.gz.asc はエンコードされていないプレーンテキストの ファイルであるということを指定します。

注意: RemoveEncoding は AddEncoding ディレクティブの後で処理されますので、 同じディレクトリの設定中に両方が現れると、 後者の効果が打ち消される可能性があります。

extension は大文字小文字を区別しません。 また、最初のドットはあってもなくても構いません。

RemoveHandler ディレクティブ

説明:	ファイルの拡張子に関連付けられたすべてのハンドラを 解除する
構文:	RemoveHandler extension [extension] ...
コンテキスト:	バーチャルホスト, ディレクトリ, .htaccess
ステータス:	ベース
モジュール:	mod_mime
互換性:	RemoveHandler は Apache 1.3.4 以降でのみ使用可能

RemoveHandler ディレクティブ は与えられた拡張子に関連付けられたハンドラを取り消します。 これにより、サブディレクトリにある .htaccess ファイルが親ディレクトリやサーバの設定ファイル から継承した関連付けを取り消すことができます。たとえば:

```
/foo/.htaccess:
AddHandler server-parsed .html
```

```
/foo/bar/.htaccess:
RemoveHandler .html
```

これは、/foo/bar ディレクトリの .html ファイルは SSI ではなく (mod_include [6] モジュール参照)、 普通のファイルとして扱われるようにする効果があります。

extension は大文字小文字を区別しません。 また、最初のドットはあってもなくても構いません。

RemoveInputFilter ディレクティブ

説明:	ファイル拡張子に関連付けられた入力フィルタを解除する
構文:	RemoveInputFilter extension [extension] ...
コンテキスト:	バーチャルホスト, ディレクトリ, .htaccess
ステータス:	ベース
モジュール:	mod_mime
互換性:	Apache 2.0.26 以降

RemoveInputFilter ディレクティブは 指定されたファイル拡張子に関連付けられた入力フィルタを解除します。 これを利用することで、親ディレクトリやサーバ設定ファイルから 継承した関連付けを サブディレクトリ内において .htaccess ファイルで取り消すことができます。

extension 引数は大文字小文字を区別しません。また、 最初のドットはあってもなくても構いません。

RemoveLanguage ディレクティブ

説明:	ファイル拡張子に関連付けられた言語を解除する
構文:	RemoveLanguage extension [extension] ...
コンテキスト:	バーチャルホスト, ディレクトリ, .htaccess
ステータス:	ベース
モジュール:	mod_mime
互換性:	Apache 2.0.24 以降

RemoveLanguage ディレクティブは 指定されたファイル拡張子に関連付けられた言語を解除します。 これを利用することで、親ディレクトリやサーバ設定ファイルから 継承した関連付けを サブディレクトリ内において .htaccess ファイルで取り消すことができます。

extension 引数は大文字小文字を区別しません。また、 最初のドットはついてもつかなくても構いません。

RemoveOutputFilter ディレクティブ

説明:	ファイル拡張子に関連付けられた出力フィルタを解除する

構文:	RemoveOutputFilter extension [extension] ...
コンテキスト:	バーチャルホスト, ディレクトリ, .htaccess
上書き:	
ステータス:	ベース
モジュール:	mod_mime
互換性:	RemoveOutputFilter は Apache 2.0.26 以降でのみ使用可能

RemoveOutputFilter ディレクティブは 指定されたファイル拡張子に関連付けられた出力フィルタを解除します。 これを利用することで、親ディレクトリやサーバ設定ファイルから 継承した関連付けを サブディレクトリ内において .htaccess ファイルで取り消すことができます。

extension は大文字小文字を区別しません。 また、最初のドットはあってもなくても構いません。

例

```
RemoveOutputFilter shtml
```

参照

- AddOutputFilter

RemoveType ディレクティブ

説明:	ファイルの拡張子と関連付けられたコンテントタイプを 解除する
構文:	RemoveType extension [extension] ...
コンテキスト:	バーチャルホスト, ディレクトリ, .htaccess
上書き:	
ステータス:	ベース
モジュール:	mod_mime
互換性:	RemoveType は Apache 1.3.13 以降でのみ使用可能。

RemoveType ディレクティブは与えられた拡張子の MIME タイプの関連付けを取り消します。 これにより、 サブディレクトリにある .htaccess ファイルが親ディレクトリやサーバの設定ファイルから継承した 関連付けを取り消すことができます。たとえば:

/foo/.htaccess:

```
RemoveType .cgi
```

これは /foo/ ディレクトリ以下の .cgi ファイルの特別な扱いを取り消します。ファイルはデフォルトタイプ [7]として扱われます。

注意: RemoveType ディレクティブは AddType ディレクティブの後に処理されますので、 両方が同じディレクトリの設定中に現れた場合、 後者の効果が打ち消される可能性があります。

extension は大文字小文字を区別しません。 また、最初のドットはあってもなくても構いません。

TypesConfig ディレクティブ

説明:	mime.types ファイルの位置
構文:	TypesConfig file-path
デフォルト:	TypesConfig conf/mime.types
コンテキスト:	サーバ設定ファイル
ステータス:	ベース
モジュール:	mod_mime

TypesConfig ディレクティブは、MIME タイプ設定ファイルの位置を設定します。filename は ServerRoot [8] からの相対パスです。 このファイルはファイルの拡張子からコンテントタイプへの デフォルトのマッピングを設定します。 ほとんどの管理者は、よく使われるファイル名の拡張子を IANA に登録されたコンテントタイプに関連付けている、 Apache の mime.types ファイルを使います。 現在の一覧は http://www.isi.edu/in-notes/iana/assignments/media-types/media-types で管理されています。これは、主要なメディアタイプの定義を提供して、 必要ところを AddType で 上書きする、という方法で httpd.conf を簡略にします。 mime.types はサーバをアップグレードしたときに 置き換えられるかもしれないので、そのファイルを直接 編集しないでください。

ファイルは、AddType ディレクティブの引数と同じ形式の行で構成されます。

```
MIME-type extension extension ...
```

拡張子の大文字小文字は区別されません。空行やハッシュ（`#'）で始まる行は無視されます。

(1) IANA に既に登録されている、あるいは (2) 広く受け入れられていてプラットホーム間でファイル拡張子に衝突がない、 という場合でなければ、配布中の mime.types ファイルに新たなものを登録するように Apache HTTP Server Project にリクエストしないでください。 category/x-subtype のリクエストは自動的に却下されますし、 言語や文字セットの名前空間で既に使用されていて、衝突の可能性のある 2 文字の拡張子も却下されます。

参照

- mod_mime_magic

URI References

[1] http://httpd.apache.org/docs-2.1/handler.html
[2] http://httpd.apache.org/docs-2.1/filter.html
[3] http://httpd.apache.org/docs-2.1/content-negotiation.html
[4] http://httpd.apache.org/docs-2.1/mod/core.html#setinputfilter

[5] http://httpd.apache.org/docs-2.1/mod/mod_negotiation.html
[6] http://httpd.apache.org/docs-2.1/mod/mod_include.html
[7] http://httpd.apache.org/docs-2.1/mod/core.html#defaulttype
[8] http://httpd.apache.org/docs-2.1/mod/core.html#serverroot