

## Apache モジュール mod\_alias

説明:	ホストファイルシステム上のいろいろな違う場所を ドキュメントツリーにマップする機能と、 URL のリダイレクトを行なう機能を提供する
ステータス:	Base
モジュール識別子	alias_module
:	
ソースファイル:	mod_alias.c

### 概要

このモジュールのディレクティブはサーバにリクエストが到着したときに URL の操作や制御をすることを可能にします。Alias ディレクティブと ScriptAlias ディレクティブは URL とファイルシステムのパスをマップするために使用されます。これは DocumentRoot の下にないドキュメントをウェブのドキュメントツリーの一部として 送られるようにします。ScriptAlias ディレクティブにはマップ先のディレクトリが CGI スクリプトのみであることを示すという追加の効果があります。

Redirect ディレクティブは クライアントに違った URL に新しいリクエストを送るように指示します。これは、 リソースが新しい場所に移動したときによく使用されます。

### トピック

URI References..... 5

### ディレクティブ

Alias.....	1	RedirectPermanent.....	4
AliasMatch.....	2	RedirectTemp.....	4
Redirect.....	2	ScriptAlias.....	4
RedirectMatch.....	3	ScriptAliasMatch.....	5

### 参照

- [mod\\_rewrite](#)
- [URL からファイルシステム上の位置へのマッピング](#)<sup>1</sup>

## Alias ディレクティブ

説明:	URL をファイルシステムの位置にマップする
構文:	Alias URL-path file-path directory-path
コンテキスト:	サーバ設定ファイル, バーチャルホスト
ステータス:	Base
モジュール:	mod_alias

Alias ディレクティブはドキュメントを ローカルファイルシステムの DocumentRoot 以外の場所に保管することを可能にします。URL の (% が復号された) パスが url-path で始まるものは directory-filename で始まるローカルファイルにマップされます。

### 例

## Apache モジュール mod\_alias

```
Alias /image /ftp/pub/image
```

http://myserver/image/foo.gif へのリクエストに対して、サーバはファイル /ftp/pub/image/foo.gif を返します。

もし url-path の最後に / を書いたなら、サーバはエイリアスを展開するために最後の / を要求するという事に注意してください。すなわち、Alias /icons/ /usr/local/apache/icons/ というものを使用していると、/icons という url はエイリアスされません。

エイリアスの行き先を含んでいる <Directory> セクションを追加する必要があるかもしれないことに注意してください。エイリアスの展開は <Directory> セクションを調べる前に行なわれますので、エイリアスの行き先の <Directory> セクションのみ 効果があります。(しかし、<Location> セクションはエイリアスが処理される前に実行されますので、こちらは適用されます。)

## AliasMatch ディレクティブ

説明:	正規表現を使って URL をファイルシステムの位置にマップする
構文:	AliasMatch regex file-path directory-path
コンテキスト:	サーバ設定ファイル, バーチャルホスト
ステータス:	Base
モジュール:	mod_alias

このディレクティブは `Alias` とほとんど同じですが、簡単な先頭からのマッチを行なうのではなく、標準正規表現を利用します。ここで指定された正規表現と URL のパス が合うかどうかを調べ、合う場合は括弧で括られたマッチを 与えられた文字列で置き換え、それをファイル名として使用します。たとえば、/icons ディレクトリを使う ためには以下のようものが使用できます:

```
AliasMatch ^/icons(.*) /usr/local/apache/icons$1
```

## Redirect ディレクティブ

説明:	クライアントが違う URL を取得するように外部へのリダイレクトを 送る
構文:	Redirect [status] URL-path URL
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:	Base
モジュール:	mod_alias

Redirect ディレクティブは古い URL を新しいものへマップします。新しい URL がクライアントに返されます。そして、クライアントは新しいアドレスをもう一回取得しようとして、URL-path (% が復号された) パスで始まるドキュメントへの すべてのリクエストは URL で始まる新しい (% が符号化された) URL へのリダイレクトエラーが返されます。

**例**

```
Redirect /service http://foo2.bar.com/service
```

クライアントは `http://myserver/service/foo.txt` へのリクエストを行なうと、代わりに `http://foo2.bar.com/service/foo.txt` をアクセスするように告げられます。

**注意**

設定ファイル中の順番に関わらず、Redirect 系のディレクティブは Alias ディレクティブと ScriptAlias ディレクティブよりも優先されます。また、.htaccess ファイルや <Directory> セクションの中で使われていたとしても、URL-path は相対パスではなく、絶対パスでなければなりません。

もし status 引数が与えられていなければ、リダイレクトは "temporary" (HTTP ステータス 302) になります。これはクライアントに リソースが一時的に移動したということを示します。Status 引数は 他の HTTP のステータスコードを返すために使用することができます:

**permanent**

永久にリダイレクトをするステータス (301) を返します。これはリソースが永久に移動したということを意味します。

**temp**

一時的なリダイレクトステータス (302) を返します。これがデフォルトです。

**seeother**

"See Other" ステータス (303) を返します。これはリソースが他のもので置き換えられたことを意味します。

**gone**

"Gone" ステータス (410) を返します。これはリソースが永久に 削除されたことを意味します。このステータスが使用された場合、url 引数は省略されなければなりません。

Status の値にステータスコードを数値で与えることで 他のステータスコードも返すことができます。ステータスが 300 と 399 の間にある場合、url 引数は存在していなければいけません。その他の場合は省略されなければなりません。ただし、ステータスは Apache のコードが知っているものである必要があります (http\_protocol.c の関数 send\_error\_response を見てください)。

例:

**例**

```
Redirect permanent /one http://example.com/two
Redirect 303 /three http://example.com/other
```

---

## RedirectMatch ディレクティブ

説明:	現在の URL への正規表現のマッチにより 外部へのリダイレクトを送る
-----	-------------------------------------

## Apache モジュール mod\_alias

構文:	RedirectMatch [status] regex URL
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:	Base
モジュール:	mod_alias

このディレクティブは `Redirect` とほとんど同じですが、簡単な先頭からのマッチを行なうのではなく、標準正規表現を利用します。ここで指定された正規表現と URL-path が合うかどうかを調べ、合う場合は括弧で括られたマッチを 与えられた文字列で置き換え、それをファイル名として使用します。たとえば、すべての GIF ファイルを別サーバの同様な名前の JPEG ファイルにリダイレクトするには、以下のようなものを使います:

```
RedirectMatch (.*)%.gif$ http://www.anotherserver.com$1.jpg
```

## RedirectPermanent ディレクティブ

説明:	クライアントが違う URL を取得するように外部への永久的な リダイレクトを送る
構文:	RedirectPermanent URL-path URL
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:	Base
モジュール:	mod_alias

このディレクティブはクライアントに `Redirect` が永久的なもの (ステータス 301) であることを知らせます。 `Redirect permanent` とまったく同じです。

## RedirectTemp ディレクティブ

説明:	クライアントが違う URL を取得するように外部への一時的な リダイレクトを送る
構文:	RedirectTemp URL-path URL
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:	Base
モジュール:	mod_alias

このディレクティブはクライアントに `Redirect` が一時的なものである (ステータス 302) ことを知らせます。 `Redirect temp` とまったく同じです。

## ScriptAlias ディレクティブ

説明:	URL をファイルシステムの位置へマップし、マップ先を CGI スクリプトに指定
-----	--

Apache モジュール mod\_alias

---

構文:	ScriptAlias URL-path file-path directory-path
コンテキスト:	サーバ設定ファイル, バーチャルホスト
ステータス:	Base
モジュール:	mod_alias

`ScriptAlias` ディレクティブは、対象ディレクトリに `mod_cgi` の `cgi-script` ハンドラで処理される CGI スクリプトがあることを示す以外は `Alias` ディレクティブと同じ振る舞いをします。URL の (% が復号された) パスが URL-path で始まるものは ローカルのファイルシステムの フルパスである二番目の引数にマップされます。

## 例

```
ScriptAlias /cgi-bin/ /web/cgi-bin/
```

`http://myserver/cgi-bin/foo` へのリクエストに対してサーバはスクリプト `/web/cgi-bin/foo` を実行します。

## ScriptAliasMatch ディレクティブ

---

説明:	URL を正規表現を使ってファイルシステムの位置へマップし、マップ先を CGI スクリプトに指定
構文:	ScriptAliasMatch regex file-path directory-path
コンテキスト:	サーバ設定ファイル, バーチャルホスト
ステータス:	Base
モジュール:	mod_alias

このディレクティブは `ScriptAlias` とほとんど同じですが、簡単な先頭からのマッチを行なうのではなく、標準正規表現を利用します。ここで指定された正規表現と URL-path が合うかどうかを調べ、合う場合は括弧で括られたマッチを 与えられた文字列で置き換え、それをファイル名として使用します。たとえば、標準の `/cgi-bin` を使用するようにするためには、以下のようなものを使います:

```
ScriptAliasMatch ^/cgi-bin(.*) /usr/local/apache/cgi-bin$1
```

## URI References

---

[1] <http://httpd.apache.org/docs-2.1/urlmapping.html>