

Apache のハンドラの使用

Apache のハンドラの使用に関して記述しています。

トピック

ハンドラとは.....	1
例.....	1
プログラマ向けのメモ.....	2
URI References.....	2

ハンドラとは

関連モジュール	関連ディレクティブ
<code>mod_actions</code>	<code>Action</code>
<code>mod_asis</code>	<code>AddHandler</code>
<code>mod_cgi</code>	<code>RemoveHandler</code>
<code>mod_imap</code>	<code>SetHandler</code>
<code>mod_info</code>	
<code>mod_mime</code>	
<code>mod_negotiation</code>	
<code>mod_status</code>	

「ハンドラ」とは、ファイルが呼ばれたときに実行される動作の Apache における内部表現です。通常、ファイルはファイル型に基づいた暗黙のハンドラがあります。普通はすべてのファイルは単にサーバに扱われますが、ファイルタイプの中には別に「ハンドル」（訳注：扱う）されるものもあります。

Apache 1.1 では、ハンドラを明示的に使用する機能が追加されました。ファイルの拡張子や置いている場所に基づいて、ファイル型と関係なくハンドラを指定することができます。これはより優雅な解決法という点と、ファイルにタイプとハンドラの両方を関連付けることができるという点で優れています。（複数の拡張子のあるファイル¹も参照してください）。

ハンドラはサーバに組み込んだり、モジュールとして含めたり、`Action` ディレクティブとして追加したりすることができます。以下は標準配布に組み込まれているハンドラです。

- `default-handler:default_handler()` を使ってファイルを送ります。静的なコンテンツを扱うときにデフォルトで使用されるハンドラです。（`core`）
- `send-as-is`: HTTP ヘッダのあるファイルをそのまま送ります。（`mod_asis`）
- `cgi-script`: ファイルを CGI スクリプトとして扱います。（`mod_cgi`）
- `imap-file`: イメージマップのルールファイルとして解析します。（`mod_imap`）
- `server-info`: サーバの設定情報を取得します。（`mod_info`）
- `server-status`: サーバの状態報告を取得します。（`mod_status`）
- `type-map`: コンテントネゴシエーションのためのタイプマップとして解析します。（`mod_negotiation`）

例

Apache のハンドラの使用

CGI スクリプトを用いて静的なコンテンツを変更する

以下のディレクティブによって、拡張子が `html` であるファイルは `footer.pl` CGI スクリプトを起動するようになります。

```
Action add-footer /cgi-bin/footer.pl
AddHandler add-footer .html
```

CGI スクリプトは希望の修正や追加を行なって、元々要求された文書（環境変数 `PATH_TRANSLATED` で指されています）を送る責任があります。

HTTP ヘッダのあるファイル

以下のディレクティブは `send-as-is` ハンドラを使用するように指示します。このハンドラは自分自身の HTTP ヘッダを持っているファイルに使用されます。ここでは、拡張子に関わらず、`/web/htdocs/asis` ディレクトリにある全てのファイルは `send-as-is` ハンドラによって扱われます。

```
<Directory /web/htdocs/asis>
SetHandler send-as-is
</Directory>
```

プログラマ向けのメモ

ハンドラの機能を実装するために、利用すると便利かもしれないものが Apache API² に追加されました。詳しく言うと、`request_rec` 構造体に新しいレコードが追加されたということです。

```
char *handler
```

もしモジュールがハンドラに関わりたい場合、やらなければならないことは、リクエストが `invoke_handler` ステージに達する以前に `r->handler` を設定することだけです。ハンドラはコンテンツタイプの代わりにハンドラ名を使うようになっていること以外は、以前と同じように実装されています。必ず要求されているわけではありませんが、メディアタイプの名前空間を侵さないように、ハンドラの名前にはスラッシュを含まない、ダッシュ（訳注：“-”）で分離された名前を付ける習慣になっています。

URI References

- [1] http://httpd.apache.org/docs-2.1/mod/mod_mime.html#multipleext
- [2] <http://httpd.apache.org/docs-2.1/developer/API.html>