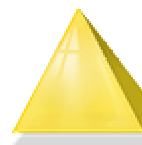


日本語対応版ユーザーガイド:

Zend Optimizer™



By Zend Technologies Inc.

注意書き

本書に記述されている内容は、予告なしに変更されることがあります。また、本書に記述されている内容を、Zend Technologies Ltd 社がすべて保証するものではありません。本書は、Zend Technologies Ltd 社の許可なしに改変あるいは、いかなる方法による第三者へ送付、写真などを含む機械的あるいは、電子的な複写や記録、記憶媒体や検索システムへの保存および購入者の個人使用を除く他の目的で使用することはできません。

本書に記述されているすべての権利は、以下に記述したものに帰属します。

© 1999-2006 Zend Technologies Ltd. All rights reserved.

Zend Optimizer™ User Guide issued April 2006.

Product Version: Zend Optimizer 3.0

DN: OUG-T-290306-3.0-004

目次

Zend Optimizer	1
基本情報	1
概要	1
システム要件	2
Zend Optimizer のインストール	3
アンインストール	4
インストール後の確認方法	5
トラブルシューティング	6
Zend Optimizer の設定一覧	8
Zend Optimizer PHP API 関数	10
付加情報	13
Zend Optimizer に関する良くある質問 (技術編)	14

Zend Optimizer

基本情報

Zend Optimizer は、Zend 社標準のランタイムコンパイラによって生成される中間バイナリコードを最適化し、より高速に実行できるようにすることで PHP のパフォーマンスを向上します。加えて、Zend Optimizer を使用すると、Zend Guard(TM)でエンコードされた中間コードファイルを読み込めるようになります。

Zend Optimizer 日本語対応版は、ゼンド・ジャパンの Web サイト (http://www.zend.co.jp/myzend/myzend_index.php) から無償で入手できます。ダウンロードするには、MyZend ページでログインする必要があります。もし、ID / パスワードをお持ちでない場合は、新規に作成することも可能です。

概要

Zend Optimizer は、PHP の拡張モジュールとしてロードされます。この拡張モジュールは、PHP スクリプトを自動的に最適化します。加えて、透過的にエンコードファイルを検知し、実行します。

Zend Optimizer ユーザーガイドは、以下のセクションで Zend Optimizer のインストール方法や操作方法に関する情報を提供する目的で作成されています。

- システム要件
- インストール
- トラブルシューティング
- 付加情報

システム要件

正常に動作させるためには、以下の条件を満たす必要があります。

1. サポートされている OS 上に Zend Optimizer がインストールされていること。
2. サポートされている PHP 4 系あるいは、PHP 5 系の PHP バージョンが設定され、それらの環境に Zend Optimizer がインストールされていること。
3. PHP を稼働させるマシン上に Web サーバがインストールされていること。

システム要件情報は、PHP バージョンの更新や変更、新バージョンのリリースなど絶えず変化します。そのため、最新の Zend Optimizer に関するシステム要件は、下記のサイトをご覧ください。

<http://www.zend.co.jp/products/compatibility.php>

Zend Optimizer のインストール

本章では、サポートされているそれぞれの OS 上で、どのように Zend Optimizer をインストールするか、その方法について説明します。

メモ：

Zend Optimizer のインストール後は、正常にインストールが完了していることをご確認ください。確認方法は、「インストール後の確認方法」をご覧ください。

Zend Optimizer のダウンロードとインストール：

1. MyZend ログインページにアクセスします。
http://www.zend.co.jp/myzend/myzend_index.php
2. MyZend ID / パスワードを入力し、「ログイン」ボタンをクリックします。
正常にユーザー認証が行われると「MyZend ユーザーメニュー」ページが表示されます。もし、MyZend ID / パスワードをお持ちで無い方は、「新規作成」をクリックして、それらを取得後、MyZend ログインページでログインを行ってください。
3. 「MyZend ユーザーメニュー」から「Zend Optimizer 日本語版のダウンロード」右横の「次へ」をクリックします。
4. Zend Optimizer を稼働させる OS 用のモジュールを選択し、「保存」をクリックしてダウンロードします。

インストール方法は、Zend Optimizer を稼働させる OS によって異なります。それらについて次に説明します。

Windows 環境上でのインストール方法：

1. ダウンロードした、ZendOptimizer-x.x.x-windows-i386.exe をダブルクリックします。
Zend Optimizer のインストーラが開始します。
2. インストーラの手順にしたがってインストール処理を続行してください。Zend Optimizer をインストールするには、Web サーバや PHP がすでにインストールされていることが前提となります。特にインストール中に php.ini ファイルのパスなどの入力が必要されます。

インストール処理完了後は、必ず 4 ページの「インストール後の確認方法」に記述されている方法で Zend Optimizer が正しくインストールされたか、ご確認ください。

Linux/Unix/Mac OS X 環境でのインストール方法：

1. 各 OS のターミナルウィンドウを開き、ダウンロードした Zend Optimizer パッケージファイルを任意のディレクトリに格納します。

次の例のようにパッケージファイルを解凍します。

```
gunzip -c <パッケージファイル名> | tar xf -
```

2. パッケージファイルを解凍したディレクトリに移動し、次のコマンドを実行します。
./install、ここでインストーラが起動します。
3. インストーラの手順にしたがってインストール処理を続行してください。Zend Optimizer をインストールするには、Web サーバや PHP がすでにインストールされていることが前提となります。特にインストール中に php.ini ファイルのパスなどの入力が必要されます。

インストール処理完了後は、必ず 4 ページの「インストール後の確認方法」に記述されている方法で Zend Optimizer が正しくインストールされたか、ご確認ください。

重要なメモ：

何らかの理由により、./install 起動によるグラフィカルイメージのインストーラが起動できないあるいは、そのような環境が使用できない場合がございます。その場合は、テキストベースのインストーラを用意しております。次のコマンドを実行してください。

```
./install-tty
```

また、SSH/Telnet により、リモートサーバからアクセスし、インストールを行う場合もこのコマンドを使用します。

重要なメモ：

Zend Optimizer をインストール/アンインストールする際は、そのシステムに管理者権限を持つユーザーあるいは、root でシステムにログインすることが必要です。それ以外のユーザーにより、インストール処理を行うと Zend Optimizer は、正しくインストールすることができません。これは、すべてのプラットフォームで共通事項です。

アンインストール

Windows 環境でのアンインストール

Zend Optimizer のアンインストール方法：

1. 「コントロールパネル」->「プログラムの追加と削除」で Zend Optimizer を選択し、「変更と削除」をクリックします。これにより、アンインストール処理が開始されます。

Linux/Unix/Mac OS X 環境でのアンインストール方法：

Zend Optimizer のアンインストール方法：

1. インストール時に以前の php.ini ファイルのバックアップファイル (php.ini-zend_optimizer.bak) を現在ご使用中の php.ini ファイルと置換します。
2. Zend Optimizer をインストールしたディレクトリをすべて削除します。インストール時にこのディレクトリ情報を保存しておくことをお勧めします。
3. Web サーバを再起動します。

重要なメモ：

Zend Optimizer をインストールしたディレクトリに他の Zend 製品をインストールしている場合は、該当ディレクトリをすべて削除することを行わないでください。このような場合は、安全性を考慮して、php.ini ファイルを以前のものと置換するだけにしてください。

インストール後の確認方法

インストールが成功したか否かを調べるには、phpinfo() 関数を使用した以下のスク립トを Web サーバのドキュメントルートに配置し、Web ブラウザからアクセスします。

```
<?php
    phpinfo();
?>
```

Web ブラウザからアクセスすると phpinfo() 関数が実行され、次のような PHP 関連の情報を表示します。「with the Zend Optimizer v3.0.0」のようにインストールしたバージョンが表示されると正常に Zend Optimizer がインストールされたと判断することができます。

```
Zend Engine vX.Y.Z, copyright (c) 1998-2006, Zend Technologies
Ltd.,
with the Zend Optimizer vX.Y.Z, copyright (c) 1998-2006 Zend
Technologies Ltd.
```

トラブルシューティング

本章ではモジュールの非互換により、発生しうる問題について説明します。また、Zend Optimizer を正常に動作させるために、非互換に関する問題を未然に防ぐ方法や解決方法などについても説明します。

PHP バージョンとの非互換

Zend Optimizer は、インストールされている PHP バージョンと完全に互換性がなければなりません。互換性がないと、Zend Optimizer はロードされずエラーメッセージが出力されます。（Web サーバのエラーログをご覧ください）。

ダウンロードした Zend Optimizer がインストールされている PHP バージョンに対応しているか、ご確認ください。対応していない場合は、Zend Optimizer をアンインストールし、ご使用中の PHP バージョンに対応した Zend Optimizer モジュールを再度、ダウンロードおよびインストールしてください。

メモ：

Zend Optimizer 3.0 は、最新の PHP 4 と PHP 5 をサポートしています。したがって、リリースされていないバージョンや PHP 4.0.x のような古いバージョンには対応しておりません。

Web Server との非互換

Zend Optimizer は、インストールされている Web サーバと完全に互換性がなければなりません。互換性がないと、Zend Optimizer はロードされずエラーメッセージがログに出力されます。（Web サーバのエラーログをご覧ください）。インストール処理中に選択した Web サーバがダウンロードした Zend Optimizer に対応しているか、ご確認ください。対応していない場合は、Zend Optimizer をアンインストールし、ご使用中の Web サーバに対応した Zend Optimizer モジュールを再度、ダウンロードおよびインストールしてください。

メモ：

Zend Optimizer は、サポートされている Web サーバでのみ動作可能です。

構成オプション

以下の PHP 設定が有効な場合、Zend Optimizer は、動作いたしません。

- **Debug Mode** - ご使用中の PHP が non-debug モードでコンパイルされていることをご確認ください。 - `phpinfo()` 関数によって出力された上部の “Debug Build”（古い PHP バージョンでは、`ZEND_DEBUG`）フィールドが “no” と表示されていることを確認してください。
- **Thread Safety** - ご使用中の PHP が Windows の場合、ZTS モード、UNIX/Linux/Mac OS X の場合は、non-ZTS モードでコンパイルされていることをご確認ください。 - `phpinfo()` 関数によって出力された上部の “Thread Safety” フィールドが、Windows の場合、 “enabled”、UNIX/Linux/Mac OS X の場合、 “disabled” と表示されていることをご

確認ください。

- **Enable Versioning** - ご使用中の PHP が Enable Versioning が無効でコンパイルされていることをご確認ください。 - phpinfo() 関数によって出力された上部の “Configure Command section “ 内に “--enable-versioning” の存在の有無より、確認することができます。このパラメータが存在している場合は、このコンパイルオプションを無効にして、PHP 再コンパイルする必要があります。

他の PHP 拡張モジュールとの非互換

Zend Optimizer が正しく動作しない他の原因として、PHP 拡張モジュールとの非互換があります。php.ini ファイルに定義されている、php 拡張モジュールが外部のソースから参照されている場合、そのモジュールと Zend Optimizer が使用するリソースの衝突や Zend Optimizer がクラッシュするような問題が発生します。一般的に、デバックや最適化に関連する拡張モジュールは、サードパーティによって php.ini ファイルに追加されており、そのため、Zend Optimizer の動作に影響を与える可能性があります。したがって、それらの拡張モジュールを php.ini ファイルから無効あるいは、削除することをお勧めします。

Zend Guard の非互換

Zend Guard でエンコードされたファイルは、そのファイルの実行するために対応した Zend Optimizer が必要です。対応した Zend Optimizer のバージョンが分からない場合は、通常、最新の Zend Optimizer をご使用ください。Zend Guard と Zend Optimizer に互換性がないと、Zend Optimizer は、エンコードファイルの実行時にエラーとなり、エラーメッセージ（「Zend Optimizer がインストールされていません。」や一般的な Zend Optimizer エラー）が出力されます。

メモ：

Zend Guard 4.0 以上で PHP スクリプトをエンコードした場合は、それらを動作させるためには、最新の Zend Optimizer が必要です。

Zend Optimizer の設定一覧

以前に記述したように、Zend Optimizer は、標準的な Zend ランタイム・コンパイラによって php スクリプトの中間コードを生成し、実行速度を向上するために、それを最適化します。Zend Optimizer のパフォーマンスを向上させるに、以下のようなディレクティブや拡張 API が用意されています。

Zend Optimizer の設定一覧

ディレクティブ	説明
<code>zend_optimizer.enable_loader</code>	エンコードされたファイルをロードするために、Zend Optimizer を使用しない場合は、 <code>zend_optimizer.enable_loader = 0</code> に設定することにより、Zend Optimizer のパフォーマンスをわずかではありますが向上させることができます。これは、Zend Optimizer に組み込まれている透過的な自動ローディング機能を無効にすることです。
<code>zend_optimizer.disable_licensing</code>	ライセンス機能を使用しない場合は、Zend Optimizer のライセンス要求を無効にすることができます。このオプションを設定すると、Zend Optimizer のメモリ使用量を減少させ、わずかですがパフォーマンスを向上させます。
<code>zend_optimizer.licence_path</code>	<p>ライセンスファイルは、ライセンスオプション付きでエンコードされた PHP スクリプトをロードする時に要求されます。</p> <p>このオプションを無効にすると、ライセンスを要求するサーバ上のエンコードされたスクリプトは、ロードされないかもしれません。</p> <p>このディレクティブは、Zend Optimizer がエンコードされたファイルに対するライセンスファイルの検索パスを指定するものです。パスのセパレータは、Windows は、<code>;</code>； Unix/Linux/Mac OS X は、<code>:</code> です。</p> <p>ディレクトリがこのパスに追加されると、ファイル拡張子が、<code>.zi</code> のものをライセンスファイルとしてロードします。ファイル名がこのパスに追加された場合、そのファイル名のものがライセンスファイルとしてロードされます。</p> <p>ライセンスは、サーバ起動時ロードされます。但し、PHP API によってロードされていない場合です。ライセンスファイルを追加したり、このパスを変更した場合、サーバの再起動が必要です。</p>

zend_optimizer.obfuscation_level_support Zend Optimizer でサポートされる難読化レベルです。

これらのディレクティブは、zend.ini ファイルに定義します。また、Zend Platform や Zend Core を通して、これらのディレクティブを簡単に設定することができます。

メモ：

ファイルをエンコードするために、Zend Guard とともに Zend Optimizer を使用している場合は、「Zend Guard 日本語対応版ユーザーガイド」をご覧ください。より詳細な情報が得られます。

Zend Optimizer PHP API 関数

以下の一覧は、Zend Optimizer で使用可能な PHP API 関数です。

関数名	形式	説明
zend_loader_enabled	bool zend_loader_enabled()	Zend Optimizer ロードが有効であるか、チェックします。 Zend_optimizer.enable_loader が on の場合、true となります。
zend_loader_file_encoded	bool zend_loader_file_encoded()	現在のファイルが、Zend Guard でエンコードされている場合、true となります。
zend_loader_file_licensed	array zend_loader_file_licensed()	現在のファイルがエンコードされており、かつ有効なライセンスファイルを保持している場合、ライセンス情報を配列で返します。その他の場合は、false を返します。
zend_loader_current_file	string zend_loader_current_file()	現在実行しているフルパス名を返します。
zend_loader_install_license	bool zend_loader_install_license(string license_file[, bool override])	Zend Guard でエンコードされたアプリケーションのために、動的にライセンスファイルをロードします。 この API により、同一プロダクトの古いライセンスファイル情報を上書きすることもできます。
obfuscate_function_name	string obfuscate_function_name(string function_name)	指定された関数名を内部的に装備されている難読化関数によって難読化し、難読化した文字列を返します。
current_obfuscation_level	int current_obfuscation_level()	現在の実行環境における設定されている難読化レベルを返します。
メモ :		
Zend Guard の難読化レベルが「基本」で難読化されたスクリプトファイルと古いバージョンのエンコードされたスクリプトの互換性を保証するために、そのような場合、戻り値は、0 となります。		
zend_runtime_obfuscate	bool zend_runtime_obfuscate()	難読化されたコードと非難読化コードが混在して動作することを許可する場合にこの API を使用します。

zend_get_id	array zend_get_id([bool all_ids = false])	HOST ID の配列を返します。 all_ids = true の場合、すべての HOST ID を返し、その他の場合は、"primary" の HOST ID を返します。
-------------	---	---

付加情報

Zend Optimizer に関する情報は、下記のサイトをご覧ください。

Zend Optimizer に関する情報

<http://www.zend.co.jp/products/optimizer/>

Zend 製品互換表

<http://www.zend.co.jp/products/compatibility.php>

技術情報

<http://www.zend.co.jp/tech/>

尚、Zend Technologies 社の Web サイトでは、Zend 製品全般に関するバグやサポート情報など蓄積されたページがございます。（英語のみ）。

バグ、サポートに関する蓄積情報

<http://www.zend.com/support/knowledgebase.php>

Zend サポートセンター

<http://www.zend.com/support/>

Zend Optimizer に関するお問い合わせは、下記、Web サイトのサポートページからお問い合わせください。

<http://www.zend.co.jp/callus/>

Zend Optimizer に関する良くある質問 (技術編)

Zend Optimizer とは? どのような機能があるのですか?

Zend Optimizer は、Zend 社が提供する標準ランタイムコンパイラによって生成される中間バイナリコードを最適化し、より高速に実行できるようにします。加えて、Zend Optimizer は、Zend Guard (TM) によってエンコードされたファイルを PHP にロードし、実行できるようにします。

Zend Optimizer は、どのように入手するのですか?

Zend Optimizer 日本語対応版は、ゼンド・ジャパンの My Zend サイト (http://www.zend.co.jp/myzend/myzend_index.php) にログインすることによって無償で入手することができます。また、他の Zend 製品の日本語対応版 (Zend Platform など) に同梱されています。

Zend Optimizer をインストールしましたが、起動はどのように行うのですか?

Zend Optimizer は、PHP システム内で動作する受動的なモジュールです。受動的なモジュールとは、PHP エンジンが動作したとき、ともに動作するライブラリと考えてください。Zend Optimizer は、自動的に PHP スクリプトを最適化し、透過的にエンコードされたファイルを検出してロードします。したがって、Zend Optimizer を明示的に起動させる必要はありません。

Zend Optimizer はどの程度の最適化を行ないますか?

Zend Optimizer はおよそ 20 種類の異なる最適化手法を実行することができます。そして、その幾つかは非常に複雑な最適化処理です。Zend Optimizer が実施する比較的単純な最適化の 1 つは、後置加算 (ポスト・インクリメント) 演算を前置加算 (プリ・インクリメント) 演算に変換することです。(但し、適用可能な場合に限りです) その理由は、後置加算演算と前置加算演算では、前置加算演算の方がより高速に演算を行うからです。

Zend Optimizer のパフォーマンスを更に向上させる方法がありますか?

エンコードされたファイルのロード用に Zend Optimizer を使用する予定がないのであれば、`zend_optimizer.enable_loader = 0` ディレクティブを `php.ini` に加えることによって若干ですが Zend Optimizer のパフォーマンスを向上させることができます。この変更は、Zend Optimizer に組み込まれている透過的な自動ローディング機能を無効にします。

エンコード済みの PHP ファイルとエンコードされていない PHP ファイル (ソースファイル) を一緒に使用できますか?

エンコード済みの PHP ファイルとエンコードされていない PHP ファイルは、透過的に一緒に使用することができます。

エンコードされたファイルを（ソースファイルの代わりに）使用すると、実行時の処理速度は改善されますか？ また、ファイルサイズは変化しますか？

スクリプトの実行ごとのコンパイル工程が節約されるので、速度の向上が期待できます。しかし、速度の向上はスクリプトの内容によって異なるため一概には言えません。また、エンコードされたファイルのサイズは、ソースファイルより若干小さくなるかもしれませんが、若干大きくなるかもしれません。小さくなるか大きくなるかはスクリプトの内容によって異なります。ファイルサイズを減少させる要因と実行時の速度を向上させる要因は、かならずしも同じ要因であるとは限りません。

サーバ起動時、Zend Optimizer がエラーメッセージを表示し、実行されません。どのような原因が考えられますか？

PHP のバージョンがインストールした Zend Optimizer と互換性がない可能性があります。ダウンロードした Zend Optimizer のバージョンが、PHP のバージョンと互換性があることを確認してください。

エンコード済みのファイルを実行しようとするエラーメッセージが表示されます。どのような原因が考えられますか？

最も一般的な原因は PHP のバージョン、あるいは Zend 中間バイナリコード（エンコードしたファイルの Zend Guard バージョン）のいずれかに互換性がないことに起因するものです。

エンコードファイルが動作しない他の原因として、システム間のファイル転送により、エンコードファイルが破損していることも考えられます。これは、Windows から Unix システムへ、FTP コマンドなどによりファイル転送を行った時、転送モードが “text mode” になっているような場合に発生します。

このようにシステム間で Zend Guard により、エンコードファイルを転送する場合は、それらを zip や tar.gz で圧縮し、転送モードは、常に “binary mode” で行うことにより、このような問題を未然に防ぐことができます。