

# PDCA グループウェア

## 「インタラクト Inter@ct」環境設定ガイド

2002 年 11 月発行

### CONTENTS

- 1、 WEB サーバ：Apache 1. 3. 2 6 の設定
- 2、 データベース： P o s t g e r S Q L 7. 1. 2 の設定
- 3、 スクリプトエンジン： P H P 4. 1. 2 の設定
- 4、 その他： 各種設定
- 5、 その他： Zend Optimizer の設定

ブレンウェア株式会社

<http://www.brain-web.co.jp/>

## 1、WEB サーバ： Apache 1.3.26 の設定

(注) root アカウントで設定します。

### ① apache\_1.3.26.tar.gz の解凍

```
[root@xxxx]#tar xvzf apache_1.3.26.tar.gz
```

### ② Apache のインストール

```
[root@xxxx]# OPTIM=" -02" ./configure --enable-module=so
```

(注) “-02” は、「マイナス オー ツー」です。

```
[root@xxxx]#make
```

```
[root@xxxx]#make install
```

### ③ インストール後 so モジュールがあることを確認

```
[root@xxxx]#/usr/local/apache/bin/httpd -l
```

一覧の中に

mod\_so.c

が表示されていればモジュールの組み込みは成功しています。

## 2、データベース： PostgreSQL 7.1.2 の設定

### ① postgres のユーザーアカウントを作成

```
[root@xxxx]#useradd postgres
[root@xxxx]#passwd postgres
```

パスワードは postgres

### ② pgsql ディレクトリの追加

/usr/local の下に pgsql ディレクトリを作成します。

```
[root@xxxx]#mkdir /usr/local/pgsql
```

所有者を postgres に変更

```
[root@xxxx]#chown -R postgres.postgres /usr/local/pgsql
```

### ③ postgresql-7.1.2tar.gz の解凍

(注) postgres アカウントで作業を進めます。

```
[root@xxxx]#tar xvzf postgresql-7.1.2tar.gz
```

### ④ PostgreSQL のインストール

```
[postgres@xxxx]$ ./configure --enable-multibyte=EUC_JP
[postgres@xxxx]$make
[postgres@xxxx]$make install
```

以上でインストールは終了ですが、postgres ユーザがデータベースを使えるようにするための環境設定を行います。

⑤ postgres の環境設定

postgres ユーザの.bashrc(隠しファイルとなっているので”ls -al” で確認)に以下の記述を追加します。

```
[postgres@xxxx]$vi .bashrc
export PATH="$PATH" :/usr/local/pgsql/bin
export POSTGRES_HOME=/usr/local/pgsql
export PGLIB=$POSTGRES_HOME/lib
export PGDATA=$POSTGRES_HOME/data
export MANPATH="$MANPATH" :$POSTGRES_HOME/man
export LD_LIBRARY_PATH="$LD_LIBRARY_PATH" :"$PGLIB"
保存し終了します。その後
[postgres@xxxx]$source .bashrc
エラーが出るようならば見直してください。
```

⑥ データベースの初期化

```
[postgres@xxxx]$initdb
```

⑦ postmaster の起動

```
[postgres@xxxx]$postmaster -i -S
```

⑧ レグレッションテスト

システムが正常に動作しているかを確認します。

```
[postgres@xxxx]$cd /usr/local/src/postgresql-7.1.2/src/test/regress
[postgres@xxxx]$make runtest
```

項目によってはエラーが発生します。

Turbolinux7Server の場合には、**abstime** と時々**random**にエラーが出ますが、問題なく稼動しているので、あまり深刻ではないようです。

⑨ pg\_hba.conf の編集

```
[root@xxxx]#vi /usr/local/pgsql/data/pg_hba.conf
```

181 行目の下に行を追加しサーバが所属しているネットワークアドレスを記述してください。

```
host    all    IP アドレス    サブネットマスク    trust
```

保存し終了します。

### 3、スクリプトエンジン： PHP 4.1.2 の設定

(注) root アカウントで設定します。

#### ① PHP-4.1.2.tar.gz の解凍

```
[root@xxxx]#tar xvzf PHP-4.1.2.tar.gz
```

#### ② PHP のインストール

(注) “ ” はスペースを表しています。

```
[root@xxxx]#. /configure--with-pgsql--enable-mbstring--enable-mbstr-enc
-trans--enable-track-vars--with-apxs=/usr/local/apache/bin/apxs--enabl
e-versioning--without-gd--enable-trans-sid
```

```
[root@xxxx]#make
```

```
[root@xxxx]#make install
```

#### ③ /tmp にディレクトリ phpwork を作成

```
[root@xxxx]#mkdir /tmp/phpwork
```

#### ④ php.ini のコピー

/usr/local/src/php-4.1.2/php.ini\_dist を /usr/local/lib/php.ini にコピーしま  
す。

```
[root@xxxx]#cp/usr/local/src/php-4.1.2/php.ini_dist /usr/local/lib/php.ini
```

## ⑤ php.ini の編集

```
[root@xxxx]#vi /usr/local/lib/php.ini
```

251 行目

修正前 `track_errors = Off`

修正後 `track_errors = On`

375 行目

修正前 `upload_tmp_dir =`

修正後 `upload_tmp_dir = /tmp/phpwork`

378 行目

修正前 `upload_max_filesize = 2M`

修正後 `upload_max_filesize = 8M`

700 行目

修正前 `session.gc_maxlifetime = 1440`

数値を **3600** 以上に設定します。

修正後 `session.gc_maxlifetime = 3600`

719 行目

修正前 `session.cache_expire = 180`

数値を **360** 以上に設定します。

修正後 `session.cache_expire = 360`

保存し終了します。

#### 4、その他： 各種設定

##### ① httpd.conf の設定

```
[root@xxxx]#vi /usr/local/apache/conf/httpd.conf
```

204 行目 `LoadModule foo_module libexec/mod_foo.so` の下に以下を追記します。  
`LoadFile /usr/local/pgsql/lib/libpq.so`

775 行目 `AddType application/x-tar .tgz` の下に以下を追記します。  
`AddType application/x-httpd-php .php .php3`  
`AddType application/x-httpd-php-source .phps`  
保存し終了します。

##### ② Apache と PostgreSQL の起動

Apache 起動(root)

```
[root@xxxx]#/usr/local/apache/bin/apachectl start
```

PostgreSQL 起動(postgres)

```
[postgres@xxxx]$postmaster -i -S
```

##### ③ PHP の動作確認

phpinfo.php の作成

```
[root@xxxx]#vi phpinfo.php
```

```
<? phpinfo() ?>
```

と入力し保存し終了します。

パーミッションの変更をします。

```
[root@xxxx]#chmod 755 phpinfo.php
```

ドキュメントルート以下に置きブラウザで確認

`http://「サーバー名又はIPアドレス」/phpinfo.php`

## 5、Zend Optimizer の導入

ZEND 社のホームページより linux 用 Zend Optimizer 2. 0. 1 以降をダウンロードしてください。(http://www.zend.co.jp/products/optimizer/)

現在の最新バージョンは Optimizer 2. 0. 2 です。

インストールの詳細は、ZEND 社 (http://www.zend.co.jp/) までお願いします。

### ① ZendOptimizer-2.0.2-Linux\_glibc21-i386.tar.gz の解凍

```
[root@xxxx]#tar xvzf ZendOptimizer-2.0.2-Linux_glibc21-i386.tar.gz
```

### ② ZendOptimizer.so のコピー

```
[root@xxxx]#cd ZendOptimizer-2.0.2-Linux_glibc21-i386/data/4_1_x_comp
[root@xxxx]#mkdir /usr/local/Zend
[root@xxxx]#mkdir /usr/local/Zend/lib
[root@xxxx]#cp ZendOptimizer.so /usr/local/Zend/lib
```

### ③ php.ini の編集

```
[root@xxxx]#vi /usr/local/lib/php.ini
以下の行を追加
zend_optimizer.optimization_level=15
zend_extension="/usr/local/Zend/lib/ZendOptimizer.so"
```

### ④ Apache の再起動

```
[root@xxxx]#/usr/local/apache/bin/apachectl stop
[root@xxxx]#/usr/local/apache/bin/apachectl start
```



## ⑤ 確認

phpinfo.php から Zend Optimizer が適用されているか確認します。  
適用されていると ZEND のロゴの左側が以下のような表示になります。

**This program makes use of the Zend Scripting Language Engine:  
Zend Engine v1.1.1, Copyright (c) 1998-2001 Zend Technologies  
with Zend Optimizer v2.0.2, Copyright (c) 1998-2002, by Zend Technologies**

以上で、インタラクト構築をする前の準備は終了です。