

PalmデバイスにおけるPostgreSQL アプリケーションの開発

日本PostgreSQLユーザー会
永安 悟史
<snaga@snaga.org>

PalmデバイスにおけるPostgreSQLアプリケーションの開発

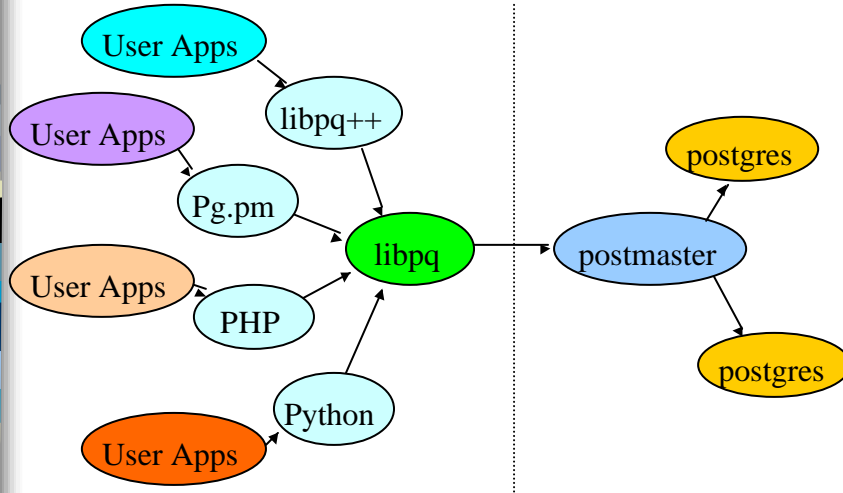
PostgreSQLとlibpq

- PostgreSQL
 - オープンソースRDBMS(www.postgresql.org)
- libpq
 - PostgreSQLへのC言語インターフェース
 - UNIXドメインソケット、INETソケットで通信
 - Windows, UNIX, Linux, ...
 - 各言語バインディングの基本(PHP, Perl, ...)
 - ソースアーカイブに標準添付

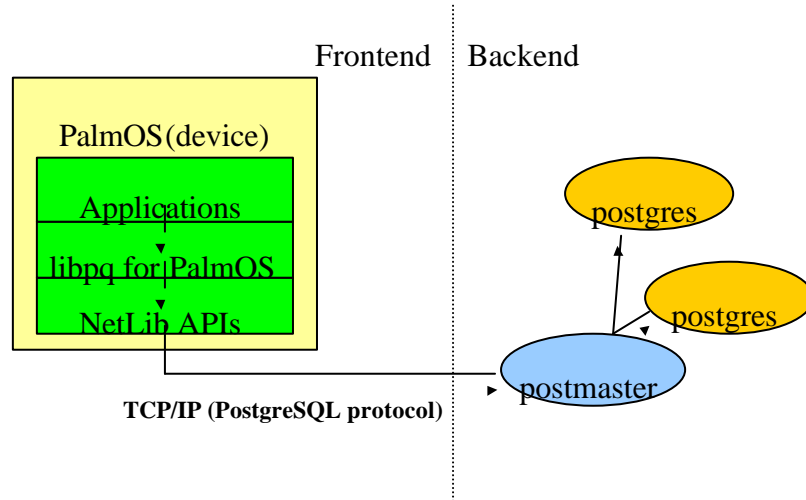
2 · PostgreSQL Conference in Tokyo 2002 (2002/06/15)



PostgreSQL と libpq



libpq for PalmOS 概要



libpq for PalmOS概要

■ 標準libpqのサブセット

- PalmOSの共有ライブラリとして実装
- TCP/IPでPostgreSQLサーバと通信
- 互換関数群を提供(非同期関数を除く)
 - PQsetdb(), PQexec(), PQfn(), PQgetvalue(), ...
- ラージオブジェクトのサポート
 - lo_open(), lo_read(), lo_write(), lo_close(), ...
- SSL/暗号化パスワード認証はサポート予定
- ライセンスはPostgreSQL本体に準じる

5 - PostgreSQL Conference in Tokyo 2002 (2002/06/15)



libpq for PalmOS概要

■ 共有ライブラリのメリット

- 単独でメンテ/配布可能
- コードの共有、アプリケーションのダウンサイジング
- 32kB/64kBの壁を突破できる
 - サブルーチンコールの壁(32kB)
 - コードサイズの壁(64kB)



6 - PostgreSQL Conference in Tokyo 2002 (2002/06/15)



なぜlibpq for PalmOSなのか?

- 大容量データベースの外部化 共有
 - リアルタイム性、同期ロジックが不要
- Web v.s. PostgreSQL
 - 不要な通信、処理が必要ないPostgreSQL
 - SQL、トランザクションが使える
- カスタムバックエンドv.s. PostgreSQL
 - 信頼性、実績のあるPostgreSQL
 - バックエンドの開発コスト

7 · PostgreSQL Conference in Tokyo 2002 (2002/06/15)



配布セットに含まれるもの

- ライブラリ本体 (libpq.prc) とヘッダファイル
- サンプルクライアント (psql.prc)
- ライブラリおよびサンプルのソースコード
 - ビルドにはCodeWarrior for Palmが必要
- ドキュメント
 - プログラミングチュートリアル

8 · PostgreSQL Conference in Tokyo 2002 (2002/06/15)



Palmwareの開発

■ 開発ツール

- gccを使う
 - gcc(m68k)、pilrc(リソースコンパイラ)
 - 環境を整えるのが大変(?)。特にWindows。
- CodeWarrior for PalmOSを使う
 - IDE、コンパイラ、Constructor(GUIビルダ)、SDKなど、全部入り
 - 有償(評価版あり)



Palmwareの開発

■ 開発の流れ (CodeWarriorの場合)

- ウィザードでテンプレート作成
- コーディング (CodeWarrior IDE)
- リソースの作成 (Constructor)
- エミュレータで実行、デバッグ
- 実機にインストール (HotSync)
- 動作確認、デバッグ



libpq for PalmOSの使い方

- ヘッダファイル(libpq.h)をインクルード
- ライブラリ(libpq.prc)をロード
- libpq関数を使う
- ライブラリをアンロード
- アプリ終了時にはPQclear(), PQfinish()
 - メモリリークを防ぐため



libpqプログラミング

■ libpqのロードとオープン

```
#include "libpq.h"  
...  
UInt libpqRefNum; /* as a global variable */  
...  
Err err = SysLibFind("libpq", &libpqRefNum);  
if (err) err = SysLibLoad('libr', 'PSQL', &libpqRefNum);  
ErrFatalDisplayIf(err, "Cannot load libpq!");  
err = PQOpen(libpqRefNum);
```



libpqプログラミング

■ データベースへの接続

```
PGconn *conn;
conn = PQsetdb(" dbhost ", " 5432", NULL, NULL, " dbname");
if ( PQstatus(conn)==CONNECTION_BAD ) { /* connect failed */
    PQfinish(conn); return;
}
PGresult *res = PQexec(conn, " set client_encoding to 'sjis'");
if ( PQresultStatus(res)!=PGRES_COMMAND_OK ) { /* error */
    PQclear(res); return;
}
PQclear(res);
```

13・PostgreSQL Conference in Tokyo 2002 (2002/06/15)



libpqプログラミング

■ 問い合わせの実行

```
res = PQexec(conn, " select * from pg_database");
if ( PQresultStatus(res)!=PGRES_TUPLES_OK ) { /* error */
    /* PQerrorMessage(conn); */
}
for ( int i=0 ; i<PQntuples(res) ; i++ ) {
    /* PQgetvalue(res, i, 0); */
}
PQclear(res);
```

14・PostgreSQL Conference in Tokyo 2002 (2002/06/15)



libpqプログラミング

■ ラージオブジェクトの利用

```
Oid pictoid;
PGresult *res = PQexec(conn, " begin"); PQclear(res);
int fd = lo_open(conn, pictoid, INV_READ);
MemPtr bufP = (MemPtr)MemPtrNew( 2048 );
int len = lo_read(conn, fd, bufP, MemPtrSize(bufP) );
lo_close(conn, fd);
/* do something using bufP */
MemPtrFree(bufP);
res = PQexec(conn, " rollback"); PQclear(res);
```

15・PostgreSQL Conference in Tokyo 2002 (2002/06/15)



libpqプログラミング

■ PostgreSQLへの接続の切断

```
PQfinish(conn);
```

■ ライブラリのクローズとアンロード

```
UInt nApps; // ライブラリを参照しているアプリケーションの数
Err err = PQClose(libpqRefNum, &nApps);
if ( nApps==0 ) {
    SysLibRemove(libpqRefNum);
}
```

16・PostgreSQL Conference in Tokyo 2002 (2002/06/15)



libpqプログラミング

■ 注意点

- メモリの容量
 - 大量の結果がある(ことが予想される)場合、offsetやlimitで制限するのが望ましい。
- メモリの開放
 - 無意識のうちにアプリケーションを終了する可能性
 - 終了時にはAppStop()関数が呼ばれる
 - 終了までに、PQclear()なりPQfinish()を行わないとメモリリークが起こる



サンプルプログラム

- psql
 - 標準psqlコマンドと同等
 - サーバに接続してクエリを実行
- デモ



サンプルプログラム

■ SO-RI

- Search Operator for Relational Image
 - Special thanks: 鈴木良介(慶應義塾大学)
- 人物の印象 (語) で画像を検索
- 画像にラージオブジェクトを使用
 - PalmのBitmapTypeでPostgreSQL上に保存



サンプルプログラム

■ Palmで使うビットマップ

- netpbmのpnmtopalm/palmtopnmコマンドで作ることができる(cygwinでコンパイル可)
 - `jpegtrans -grayscale foo.jpg | jpegtopnm | pnmtopalm > foo.img`
- インポートする
 - `select lo_import('foo.img');`



サンプルプログラム

■ 画像(ラージオブジェクト)の扱い方

```
Oid pict = StrATol( PQgetvalue(res, 0, 2) ); // 画像の Oid を取得
PGresult *tmp = PQexec(conn, "begin"); PQclear(tmp);
int fd = lo_open(conn, pict, INV_READ); // lobjオープン
char *bufP = (char *)MemPtrNew(2048); // メモリブロック確保
int len = lo_read(conn, fd, bufP, 2048); // lobj読み込み
BitmapType *bmpP = (BitmapType *)bufP; // Bitmap型にキャストして
WinPaintBitmap(bmpP, 2, 16); // 描画
lo_close(conn, fd); // lobjクローズ
MemPtrFree(bufP); // メモリ開放
PGresult *tmp = PQexec(conn, "rollback"); PQclear(tmp);
```

21・PostgreSQL Conference in Tokyo 2002 (2002/06/15)



SO-RI

■ デモ

22・PostgreSQL Conference in Tokyo 2002 (2002/06/15)



libpqプログラミングまとめ

- libpq関数の使い方は標準libpqと同じ
 - とても簡単に使える!
 - UNIX上で開発したロジックをそのまま移植可
- 共有ライブラリのロードオープン、クローズ/アンロードは必要
 - でも定石なので難しくない!
- 「気軽にデータベースアプリを作ろう!」



今後の予定

- 未実装の機能の追加
 - 暗号化 (md5 認証のサポート(2002/Q3))
 - SSLのサポート(PalmOS 5~)
 - 非同期関数のサポート(未定)
 - 圧縮プロトコルのサポート(新機能、未定)
- 他機種への移植
 - WindowsCEへの移植(未定)



募集中！

- ハックする人
 - SSLの実装、WindowsCE への移植、etc...
- 使う人
 - 開発 (利用) 事例
- おうえんする人
 - スポンサーとか...



リソース

- フロントエンドバックエンドプロトコル
 - PostgreSQL 7.1.3 開発者用ガイド
- Palm共有ライブラリ
 - Shared libraries on the Palm Pilot
 - <http://www.isaac.cs.berkeley.edu/pilot/shlib.html>
 - <http://snaga.org/palm/shlib-ja.html> (対訳)
 - Designing Non-Simple Applications
 - <http://www.palmos-japan.com/palmsource/slides.html>



libpq for PalmOSへのコンタクト

- libpq for PalmOS
 - ライブラリ、ドキュメントの配布など
 - <http://snaga.org/palm/libpq/>
 - 質問などもお気軽に
 - snaga@snaga.org



おしまい

