# Firebird on PHP

# Integrate or Abstract

# Firebird on PHP – Integrate or Abstract

What is PHP? Probably the best kept secret on the internet. OK it is not a secret, but as with MySQL, the bulk of users don't even know that they are using it.

# Firebird on PHP – Integrate or Abstract

What is PHP? Probably the best kept secret on the internet. OK it is not a secret, but as with MySQL, the bulk of users don't even know that they are using it. One of the major hurdles for Firebird is the fact that everybody supports Apache/MySQL/PHP  and almost every major PHP information site will not include a project unless it includes MySQL support.

# Firebird on PHP – Integrate or Abstract

What is PHP? Probably the best kept secret on the internet. OK it is not a secret, but as with MySQL, the bulk of users don't even know that they are using it. One of the major hurdles for Firebird is the fact that everybody supports Apache/MySQL/PHP  and almost every major PHP information site will not include a project unless it includes MySQL support.

We will skip the reasons why MySQL is the wrong solution to the problem, and why developers prefer PostgreSQL for PHP development. That is a subject for another paper.

# Firebird on PHP – Integrate or Abstract

What is PHP? Probably the best kept secret on the internet. OK it is not a secret, but as with MySQL, the bulk of users don't even know that they are using it.

One of the major hurdles for Firebird is the fact that everybody supports Apache/MySQL/PHP  and almost every major PHP information site will not include a project unless it includes MySQL support.

We will skip the reasons why MySQL is the wrong solution to the problem, and why developers prefer PostgreSQL for PHP development. That is a subject for another paper. Since Firebird ***IS*** the best database, how do we use it with PHP.

# php_interbase

While there are several methods of building PHP, they all build on a modular approach, and the Interbase module works fine with Firebird, but already has hooks to allow a split for a Firebird specific version.

# Windows Install

The zip install versions of PHP include all of the additional modules, and php_interbase is one of those packages, and just needs to be un-commented in the php.ini configuration file.

```
extension=php_interbase.dll
```

Then just restart PHP.

# Linux Install

Linux install is not quite as easy as windows, but a simple PHP reconfigure is provided on most distributions of Linux. Just add the location of your Firebird install to the conf file, and hit build.

I have not had any problems with adding the correct library in the past, but I have not had to do it recently!

Just contact the firebird-php list for help!

# Interbase Support

- ibase_connect
- ibase_pconnect
- ibase_close
- ibase_trans
- ibase_commit
- ibase_rollback
- ibase_prepare
- ibase_execute
- ibase_query
- ibase_field_info
- ibase_num_fields
- ibase_fetch_***
- ibase_blob_***
- ibase_timefmt
- ibase_errmsg

# Interbase Support in PHP5

- ibase_add_user
- ibase_delete_user
- ibase_modify_user

- ibase_affected_rows
- ibase_commit_ret
- ibase_rollback_ret
- ibase_errcode
- ibase_num_params

- ibase_param_info
- ibase_drop_db
- ibase_gen_id

- ibase_free_event_handler
- ibase_set_event_handler
- ibase_wait_even

- fbird_***

# Other Options

- ADOdb
- PEAR:DB
- Metabase
- MDB
- Creole
- PDO
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB
- Metabase
- MDB
- Creole
- PDO
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase
- MDB
- Creole
- PDO
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB
- Creole
- PDO
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB - Combines Metabase and PEAR:DB
- Creole
- PDO
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB - Combines Metabase and PEAR:DB
- Creole - New for PHP5 – but reported to be slow
- PDO
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB - Combines Metabase and PEAR:DB
- Creole - New for PHP5 – but reported to be slow
- PDO - Another attempt from PHP but lacking
- PHPLib
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB - Combines Metabase and PEAR:DB
- Creole - New for PHP5 – but reported to be slow
- PDO - Another attempt from PHP but lacking
- PHPLib - Still distributed but out of date
- ADOdbLite
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB - Combines Metabase and PEAR:DB
- Creole - New for PHP5 – but reported to be slow
- PDO - Another attempt from PHP but lacking
- PHPLib - Still distributed but out of date
- ADOdbLite – Cut down 'faster' ADOdb
- EzSQL

# Other Options

- ADOdb - I'm Biased – I like this one
- PEAR:DB - One of many attempts to copy ADOdb
- Metabase - Comprehensive but slow package
- MDB - Combines Metabase and PEAR:DB
- Creole - New for PHP5 – but reported to be slow
- PDO - Another attempt from PHP but lacking
- PHPLib - Still distributed but out of date
- ADOdbLite – Cut down 'faster' ADOdb
- EzSQL - Popular light weight abstraction

# PDO – SQL Smothing

The main problem with this approach is that every database engine has it's own set of functions, and while PHP5.1 will include a first attempt at an integrated package, with it's PDO module, the differences between how these engines work makes a common interface difficult.

# PDO Extract not Abstract

The main problem with PDO and some of the other Abstraction Layers is that they aren't!

They provide a compatible set of functions, but do nothing for the underlying SQL.

So while they all execute an SQL script in the same way, they don't bother about the differences in SQL between engines.

# ADOdb – Heavyweight Abstraction

ADOdb is one of a number of add-on's that allows a more generic design of SQL interface, and PHP applications that have been built using it can easily be ported to Firebird if that has not already been implemented.

ADOdb includes a PHP extension module that will speed some of the functions in the same way a fully integrated module would do, but since we only need to distribute the drivers we need we can trim the distribution as required.

# Light Weight ADOdb

The ADOdbLite package has achived many of the reductions in size by stripping functions that then require that the SQL is tailored to the engine you are working with.

There is not a Firebird drive for the package and while it would not be difficult to implement, the missing functions make it less attractive.

# Sites using ADOdb

Many popular web applications such as ACID, PostNuke, Xaraya, phpWiki, Mambo, PHP GACL, TikiWiki, eGroupWare and phpLens App Server are using ADOdb as their database abstraction layer.

Some reasons why ADOdb is popular include:

Single Schema design for all engines

All major differences managed

Easy to modify driver modules for new functions

Built in result caching

# ADOdb Connect to database

```
<?php
include('adodb/adodb.inc.php');
$db = ADONewConnection('firebird15');
$db->debug = true;
$db->Connect($server, $user, $password, $database);
$rs = $db->Execute('select * from some_small_table');
print "<pre>";
print_r($rs->GetRows());
print "</pre>";
?>
```

# ADOdb Print a Grid

# ADOdb Print a Grid

```
<? $ticketpager->RenderBody(); ?>
```

# ADOdb Add Navigation

```
$sql = "select * from adoxyz ";
$pager = new ADODB_Pager($db,$sql);
$pager->Render($rows_per_page=5);
```

# ADOdb Add Navigation

```
$sql = "select * from adoxyz ";
$pager = new ADODB_Pager($db,$sql);
$pager->Render($rows_per_page=5);
```

```
SelectLimit($sql,
$numrows=-1,$offset=-1,$inputarr=false);
```

# Generate or Sequence

**CreateSequence($seqName = 'adodbseq',$startID=1)**
Create a sequence. The next time GenID( ) is called, the value returned will be $startID. Added in 2.60.

**DropSequence($seqName = 'adodbseq')**
Delete a sequence. Added in 2.60.

**GenID($seqName = 'adodbseq',$startID=1)**
Generate a sequence number . Works for interbase, mysql, postgresql, oci8, oci8po, mssql, ODBC based (access,vfp,db2,etc) drivers currently. Uses $seqName as the name of the sequence. GenID() will automatically create the sequence for you if it does not exist (provided the userid has permission to do so). Otherwise you will have to create the sequence yourself.

# Creating Metadata

- ADOdb - PHP

```
'tiki_pages' => "
  page_id I4 AUTO PRIMARY,
  user C(40),
  creator C(40) NOTNULL,
  page_name C(160) NOTNULL,
  hits I4,
  data X,
  description C(200),
  last_modified I8,
  comment C(200),
  version I4 NOTNULL,
  ip C(15),
  flag C(1),
  points I4,
  votes I4,
  cache X,
  wiki_cache I8 DEFAULT 0,
  cache_timestamp I8,
  page_rank N(4,3),
  page_size I8 DEFAULT 0
",
```

# Abstract field types

C: Varchar, capped to 255 characters.

X: Larger varchar, capped to 4000 characters (to be compatible with Oracle).

XL: For Oracle, returns CLOB, otherwise the largest varchar size.

C2: Multibyte varchar

X2: Multibyte varchar (largest size)

B: BLOB (binary large object)

D: Date (some databases do not support this, and we return a datetime type)

T: Datetime or Timestamp

L: Integer field suitable for storing booleans (0 or 1)

I: Integer (mapped to I4)

I1: 1-byte integer

I2: 2-byte integer

I4: 4-byte integer

I8: 8-byte integer

F: Floating point number

N: Numeric or decimal number

# Links to PHP tools

- PHP - http://www.php.net/
- Firebird Function - http://uk.php.net/manual/en/ref.ibase.php
- PDO - http://uk.php.net/manual/en/ref.pdo.php
- ADOdb - http://adodb.sourceforge.net/
- ezSQL - http://justinvincent.com/home/docs/ezsql/readme.txt
- ADOdbLite - http://adodblite.sourceforge.net

# For all your SQL translation needs

*Lester Caine*
*L.S.Caine Electronic Services*
*25 Smallbrook Road*
*Broadway*
*Worcestershire*
*WR12 7EP*
*United Kingdom*