Database interfaces for open source languages suck

They're all limited

They're all different

They're all duplicating development effort

We can fix this

We need a common database driver
API for Parrot

So database drivers...

Can be implemented in
any Parrot language

Can be used by
any Parrot language

Mature, Stable, Functional,
Object Oriented

Well documented,
ideally with a test suite

Well known to a wide user base

Well known to driver developers

JDBC

No, not Java, just the JDBC API

The class and method names,
the semantics

Actually, not plain JDBC

Sanitize the worst influences of Java

Easier to use for dynamic languages

Each Parrot language...

could have adaptation interface
layered over the raw JDBC API

For Perl6...

DBI

What's happening currently

JDBC.pm

PDBC.pm

DBI42.pm

http://svn.perl.org/modules/dbi2/trunk/

Assume Perl 6 will be able to use
Perl 5 DBI v1 efficiently

Ignoring DBI v2 API for now,
radically refactoring the internals

Exploring impedance mismatches
between JDBC and DBI

A DBI v2 API will evolve in time

Interested?

Tim.Bunce@pobox.com