

SQL database access through PDO.

[#aura](#) [#database](#) [#profiler](#) [#php](#) [#pdo-instance](#)

637 commits

4 branches

20 releases

27 contributors

MIT

Branch: 3.x ▾















New pull request

Create new file

Upload files

Find file

Clone or download ▾

 pmjones committed on Jun 28, 2017 update changelog for release	Latest commit 596fc78 on Jun 28, 2017
 docs	Create _bookdown.json 11 months ago
 src	Removed sqlsrv hack 8 months ago
 tests	kill the test 8 months ago
 .gitignore	first attempt at Psr\Log profiling 2 years ago
 .scrutinizer.yml	update coverage config for travis and scrutinizer 4 years ago
 .travis.yml	specify minor versions of PHP 8 months ago
 CHANGELOG.md	update changelog for release 8 months ago
 CONTRIBUTING.md	add a contributing doc 3 years ago
 LICENSE	update license years a year ago
 README.md	update README to remove sqlsrv warning note 8 months ago
 composer.json	update phpunit requirement 11 months ago
 phpunit.php	composer/phpunit changes stolen from aura.di to fix travis failure 2 years ago
 phpunit.xml.dist	add phpunit whitelist 2 years ago

 [README.md](#)

Aura.Sql

Provides an extension to the native [PDO](#) along with a profiler and connection locator. Because *ExtendedPdo* is an extension of the native *PDO*, code already using the native *PDO* or typehinted to the native *PDO* can use *ExtendedPdo* without any changes.

Added functionality in *Aura.Sql* over the native *PDO* includes:

- **Lazy connection.** *ExtendedPdo* connects to the database only on method calls that require a connection. This means you can create an instance and not incur the cost of a connection if you never make a query.
- **Decoration.** *DecoratedPdo* can be used to decorate an existing PDO instance. This means that a PDO instance can be "extended" **at runtime** to provide the *ExtendedPdo* behaviors.
- **Array quoting.** The `quote()` method will accept an array as input, and return a string of comma-separated quoted values.
- **New `perform()` method.** The `perform()` method acts just like `query()`, but binds values to a prepared statement as part of the call. In addition, placeholders that represent array values will be replaced with comma-separated quoted values. This means you can bind an array of values to a placeholder used with an `IN (...)` condition when using `perform()`.
- **New `fetch*()` methods.** The new `fetch*()` methods provide for commonly-used fetch actions. For example, you can call `fetchAll()` directly on the instance instead of having to prepare a statement, bind values, execute, and then fetch from the prepared statement. All of the `fetch*()` methods take an array of values to bind to to the query statement, and use the new `perform()` method internally.
- **New `yield*()` methods.** These are complements to the `fetch*()` methods that `yield` results instead of `return`ing them.

- **Exceptions by default.** *ExtendedPdo* starts in the `ERRMODE_EXCEPTION` mode for error reporting instead of the `ERRMODE_SILENT` mode.
- **Profiler.** An optional query profiler is provided, along with an interface for other implementations, that logs to any PSR-3 interface.
- **Connection locator.** A optional lazy-loading service locator is provided for picking different database connections (default, read, and write).

Installation and Autoloading

This package is installable and PSR-4 autoloadable via Composer as [aura/sql](#).

Alternatively, [download a release](#), or clone this repository, then map the `Aura\Sq1\` namespace to the package `src/` directory.

Dependencies

This package requires PHP 5.6 or later; it has also been tested on PHP 7 and HHVM. We recommend using the latest available version of PHP as a matter of principle.

Aura library packages may sometimes depend on external interfaces, but never on external implementations. This allows compliance with community standards without compromising flexibility. For specifics, please examine the package [composer.json](#) file.

Quality

Scrutinizer `unknown` coverage `unknown` build `passing` pds `skeleton`

This project adheres to [Semantic Versioning](#).

To run the unit tests at the command line, issue `composer install` and then `./vendor/bin/phpunit` at the package root. (This requires [Composer](#) to be available as `composer`.)

This package attempts to comply with [PSR-1](#), [PSR-2](#), and [PSR-4](#). If you notice compliance oversights, please send a patch via pull request.

Community

To ask questions, provide feedback, or otherwise communicate with other Aura users, please join our [Google Group](#), follow [@auraphp](#), or chat with us on Freenode in the `#auraphp` channel.

Documentation

This package is fully documented [here](#).