Query Builder

Introduction

HUBzero offers a query builder to help in the abstraction of language-specific SQL syntax, as well as making code more readable. While the query builder itself takes many of its structural queues from SQL, it is not syntax specific. To get started, simply:

```
$query = new HubzeroDatabaseQuery;

// Sample select statement
$users = $query->select('*')
            ->from('#__users')
            ->whereEquals('job', 'programmer')
            ->fetch();
```

Functionally speaking, the query builder serves as a nice intermediary between the feature-rich abstraction that is the ORM, and the execution of raw language specific queries against the database driver. The query builder is actually available on ORM models, and the models will filter all applicable method calls down from the model itself to the underlying query as needed.

Fetching Results

Caching

By default, when fetching results using the query builder, query results will be cached. To disable this behavior, you can pass TRUE as the second argument of the fetch method call.

```
$query->fetch('rows', true);
```

Alternatively, instead of disabling the cache for a single fetch, you can clear the entire cache.

```
Query::purgeCache();
```

Inserting, Updating, and Deleting

When it comes to adding, modifying, or removing records, you have two options for going about this. You can manually build the appropriate query, or you can use one of the shortcut methods. To exemplify this behavior, check out the following two examples.

```php
// Full insert
$query = new Query;

$query->insert('users')
      ->values(['name' => 'me', 'email' => 'you@me.com'])
      ->execute();

// Shortcut method
$query->push('users', ['name' => 'me', 'email' => 'you@me.com']);
```

The same principle also applies to updates and deletes:

```php
// Full update
$query = new Query;

$query->update('users')
      ->set(['name' => 'you'])
      ->whereEquals('id', 1)
      ->execute();

// Shortcut method
$query->alter('users', 'id', 1, ['name' => 'you']);
```

```php
// Full delete
$query = new Query;

$query->delete('users')
      ->whereEquals('id', 1)
      ->execute();

// Shortcut method
$query->remove('users', 'id', 1);
```