

Upgrading

Upgrading

Anonymous · 08/10/2026

Upgrading

This page lists the changes that can break code written against a previous version of derafu/etl. Nothing here is deprecated first: the old API is gone.

Requirements

Package	Before	Now
doctrine/dbal	^4.4	^4.5
phpoffice/phpspreadsheet	^5.4	^5.10
symfony/console	^8.0	^8.1
derafu/translation	not used	dev-main

DBAL 4.5 deprecates `Index::getFlags()` and `Table::addIndex()` / `addUniqueIndex()`. The package no longer uses them: it reads `Index::getType()` and `isClustered()`, and builds indexes with `IndexEditor` and `TableEditor`.

Indexes: `type` and `clustered` instead of `unique` and `flags`

Derafu\ETL\Schema\Index and IndexInterface changed. See [Schema Model](#) for the new model.

Before	Now
<code>new Index(\$name, \$columns, bool \$unique, array \$flags)</code>	<code>new Index(\$name, \$columns, IndexType \$type, bool \$clustered)</code>
<code>setUnique(bool)</code>	<code>setType(IndexType::UNIQUE)</code> (or another type)
<code>getFlags()</code> / <code>setFlags(array)</code>	removed
<code>isUnique()</code>	kept, derived from the type
–	<code>getType()</code> , <code>setType()</code> , <code>isClustered()</code> , <code>setClustered()</code>

```
// Before.  
$index = new Index('idx_email', ['email'], true);
```

```
$index->setFlags(['clustered']);

// Now.
use Derafu\ETL\Schema\Enum\IndexType;

$index = new Index('idx_email', ['email'], IndexType::UNIQUE, true);
```

The old `flags` could hold any string; the model now only has the three things DBAL itself distinguishes: the type (regular, unique, fulltext, spatial) and clustered.

If you have your own `IndexInterface` implementation, add the four new methods and remove `getFlags()` / `setFlags()` / `setUnique()`.

Spreadsheet format

The schema sheet now stores `type` and `clustered` for each index:

```
{ "columns": ["invoice_number"], "type": "unique", "clustered": false }
```

- **Reading old files keeps working.** `unique` and `flags` (`clustered`, `fulltext`, `spatial`) are still understood. Any other flag, or a `type` that is not one of the four values, throws an exception instead of being ignored.
- **Old versions cannot read the new files correctly.** They look for `unique`, so a unique index would load as a regular one. Upgrade every consumer of the files before writing them with the new version.

Output of the text targets

- **Markdown:** the last column of the index table is now `Clustered` (it was `Flags`), and the *Type* column can also say `FULLTEXT` or `SPATIAL`.
- **Text:** the `FLAGS: ...` line under an index is gone; a clustered index gets a `CLUSTERED` line, and the type can be `FULLTEXT INDEX` or `SPATIAL INDEX`.
- **D2:** the type can also be `FULLTEXT` or `SPATIAL`.

Anything that compares or parses that output needs the new text.

Exceptions

Exceptions thrown by the package are now translatable (`TranslatableRuntimeException`, `TranslatableInvalidArgumentException`, `ETLException`). They still extend `RuntimeException` and `InvalidArgumentException`, so catch blocks keep working, and `getMessage()` returns the same English text. A few messages now have named placeholders and the pipeline messages are one string instead of a concatenation, so code that matches on the exact text of a message should be checked.

See [Translations](#).

Subclasses of `AbstractDatabase`

The `$schema` property is now `?SchemaInterface = null` (it was an uninitialized `SchemaInterface`). A subclass that invalidates the cached schema sets `$this->schema = null` instead of `unset($this->schema)`, and checks `=== null` instead of `isset()`.

Last updated on 08/10/2026