

Migration to PHP \geq 8.3

- [What's new in PHP 8.3+](#)

What's new in PHP 8.3+

Minimum PHP version updated

The application now requires **PHP 8.3** as the minimum version.

Database updates

Starting from version 26.01 of vtenext, some important database changes took place, such as:

Charset and Collation

Database is now using by default the charset **utf8mb4** and collation **utf8mb4_general_ci**, which allows to store and display all possible UTF-8 characters, including emojis and special symbols.

BIGINT columns

All columns containing crmids are now BIGINT (for new installations). When creating columns that store crmid, always use BIGINT (or **18** when using adodb datadict). This is necessary to support a large number of records (more than ~2 billion).

TINYINT columns

All columns containing only boolean integers (e.g., 0/1 for checkbox fields uitype 56, deleted=0/1, or various status columns with only small integers) are now TINYINT (or **11** when using adodb datadict). This change will slightly reduce the database size and query execution time in certain cases.

During the update (step 3019 - 3020) a script to automatically convert the charset and the integer columns is executed (modules/Update/changes/bigint_utf8mb4_alter.php). Since this change can take a very long time, bigger tables (modules with more than 100k records, or tables bigger than 5GB) are excluded and should be converted manually at the end of the update (or at an appropriate time to minimize downtime, like at night). The script will output if this is necessary or if all tables have been converted. In case the manual conversion is necessary, it can be done in 2 ways:

- By running the queries in file: modules/Update/changes/bigint_alter_big.sql
- For extremely large tables, by executing the script: modules/Update/changes/percona_alter.sh, to use the pt-online-schema-change tool (installation to be done separately) to avoid locking the tables. It is advised to review this script and manually add the necessary command line option to the Percona command.

Licence tag

A new SPDX license tag is now **mandatory in all vtenext core files** (php, js and tpl) that are not external libraries

```
/*  
 * SPDX-FileCopyrightText: 2009-present Vtenext S.r.l. Società Benefit  
 * SPDX-License-Identifier: LicenseRef-vtenext-business-license  
***/
```

New configuration files

All new custom configuration files (e.g., custom api token) must be placed in the `/config` folder. If some configuration is different from dev/prod environments, use the corresponding config file in `/config/`.

- `config.$envType.php`: Overrides of PHP configuration and global variables for a specific environment type
- `phpstan/*`: Configuration to use when running phpstan
- `smarty/*`: Configuration directory for Smarty library
- `logging.php`: Configuration for the legacy logs (old log4php)
- `request.config.php`: See [Configuration - RequestHandler](#)

New config variables

The following global config variables have been added to `config.inc.php`:

`$enableLegacyLogs`

Activate the legacy logs (the old log4php), which logs a lot of stuff in `logs/vtenext.log`. The usefulness of this log is not certain. Configuration of this log is in `/config/logging.php`.

`$envType`

Specify the environment type. Possible values:

- `"prod"` (default)
- `"preprod"`
- `"dev"`
- `"debug"`

Each level activates more error messages. The default configuration is in `config.inc.php`, variable `$php_config`, overridden by the specific `/config/config.$envType.php` file.

`$php_config`

Used to specify the default PHP configuration used by `$envType`.

`$smarty_warn_superglobals`

A global configuration flag that enables warnings when PHP superglobals are used in Smarty templates. When enabled, the system registers a Smarty POST filter that scans compiled template code for superglobal usage (`$_GET`, `$_POST`, `$_REQUEST`, `$_COOKIE`, `$_SERVER`, `$_ENV`). By default it is enabled only in development environments.

New functions and methods

RequestHandler (RH)

`RequestHandler` provides secure, centralized access to HTTP request data. It automatically sanitizes input using `vtlib_purify()` and applies type-safe filters from `F::` enum. This replaces direct access to superglobals like `$_REQUEST`, `$_GET`, `$_POST`, etc.

For details, see: [RequestHandler](#)

BaseAction

We have moved the route resolution logic to the new `IndexRouter` class and introduced the `BaseAction` class to handle actions.

For details, see: [Routing system](#).

CRMEntity methods update

The old `retrieve_entity_info` has been renamed to `retrieve_html` and `retrieve_entity_info_no_html` to `retrieve`. This is to make more explicit that there is a HTML conversion going on. The old functions are deprecated, but still working.

- `CRMEntity::retrieve_html()`: Retrieves entity with HTML conversion
- `CRMEntity::retrieve()`: Retrieves entity without HTML conversion

Translation aliases

New convenient aliases for translation functions:

- `trans()`: Alias for `getTranslatedString()`
- `trans_app()`: Alias for `getTranslatedString($str, "APP_STRINGS")`
- `trans_js()`: Alias for `getTranslatedString($str, "ALERT_ARR")`

New database query methods

New database query methods are available. For details, see: [Database Best Practices](#)

Autoescaping of all Smarty variable

All variables outputted by Smarty templates (for example `{${VARIABLE}}`) by default have all applicable characters are converted to html `&...;` notation.

The reason of this change is to reduce as much as possible exposure to XSS attacks, since it was extremely difficult to track every possible variable in templates and ensure it was properly escaped.

For details, see: [Escape](#).

New tools

`tools/change-env`

Allows to easily change the `$envType` variable in `config.inc.php` and automatically execute scripts afterwards. Scripts to execute are located in `plugins/envs/make.$env/*`.

`tools/activate`

Prefix the absolute path of `tools/` to `PATH`.

`tools/analyze`

Install and run phpstan on the vte root directory, or a custom one.

`tools/tests`

Install and run phpunit with vte unit and functional tests.

`tools/change-color`

Simple script to change color of vtenext using some standard values. The purpose is to quickly change color of an environment to differentiate it from the production.

tools/compile-themes

Compile scss sources of all (or some) known vte themes. Can monitor for file changes and trigger a compilation automatically (`inotifywait` must be installed).

Important files and folders

cache_local/

This folder should contain cache files to be stored on the same host as the webserver. It is useful in case of cluster deployment, where the `cache/` folder is normally shared, but `cache_local/` is only local.

plugins/envs/

Contains the scripts to be executed when changing the environment type (e.g., after cloning the prod in the dev host).

GDPR portal update

The GDPR portal has been significantly updated with the following improvements:

- Migration of libraries to Composer for better dependency management
- API calls switched to vtenext RestClient
- Smarty templating engine upgraded to the latest version (^5.4)

Libraries

All third party libraries now must be imported via composer (living in the `/vendor/` folder), or placed in the `/vendor-extra/` folder. Libraries in vendor-extra are automatically included in the class path.

Important note

Whenever a new library is added, either via composer or manually, the following command must be executed to refresh the classmap: `composer optimize` → alias of `composer dump-autoload -o`

Libraries updated

Library	Previous version	New version
phpmailer/phpmailer	6.6.5	6.9.3
ezyang/htmlpurifier	4.13.0	4.18.0
mpdf/mpdf	8.0.12	8.2.5
league/iso3166	2.1.5	4.3.2
league/oauth2-client	2.7.0	2.8.1
league/oauth2-google	3.0.4	4.0.1
spomky-labs/otphp	10.0.3	11.3.0
joypixels/emoji-toolkit	6.6.0	9.0.1
phpoffice/phpspreadsheet	1.8.2	4.1.0
phpseclib/phpseclib	3.0.42	3.0.43
jaybizzle/crawler-detect	1.3.0	1.3.4
hubspot/hubspot-php	2.0	5.3.0
slinec/shopify-api-php	2.5	3.1.0
zendesk/zendesk_api_client_php	2.29	4.1.0
lesstif/php-jira-rest-client	1.41	5.9.0

Libraries added

- web-token/jwt-library 4.0.4
- php-amqplib/php-amqplib 3.7.3
- econea/nusoap 0.9.18
- dragonmantank/cron-expression 3.4.0
- qualityunit/tnef-decoder 1.2.9
- sabre/vobject 4.5.6
- flipboxdigital/oauth2-hubspot 1.0.1
- stevenmaguire/oauth2-salesforce 2.0.1
- stevenmaguire/oauth2-zendesk 2.1.0
- automattic/woocommerce 3.1.0

- `monolog` (replaces log4php) 3.8.1

Libraries moved to vendor

The following libraries have been moved from various locations to `/vendor/` or `/vendor-extra/`:

- `include/nusoap` → moved to vendor
- `modules/Settings/ProcessMaker/thirdparty/cron-expression` → moved to vendor
- `modules/PDFMaker/classes/simple_html_dom.php` → moved to vendor-extra
- `include/magpierss` → moved to vendor-extra
- `modules/Morphsuit/utils/RSA` → replaced with phpseclib in vendor
- `modules/Newsletter/bounce_driver.class.php` → moved to vendor-extra
- `modules/Messages/src/attachment_tnef` → moved to vendor
- `modules/Messages/src/Squirrelmail.php` → moved to vendor-extra
- `smartoptimizer` → moved to vendor-extra (updated minifier, added brotli and zstandard support)
- `portal/nusoap` → moved to vendor
- `portal/include/htmlpurifier` → removed
- `include/pChart` → patched and moved to vendor-extra
- `include/freetag` → fixed and moved to vendor-extra
- `modules/VteSync/vendor` → moved to main vendor
- `modules/VteSync/VteSyncLib/src/Connector/Jira/vendor` → moved to main vendor

Removed libraries

- `log4php` - Replaced with monolog
- `vtlib/SimplePie` - No longer used
- `include/antlr` - Not working, grammar unknown and never used
- `modules/SDK/examples/intellisense` - Removed
- `modules/Calendar/iCal/ical-parser-class.php` - Not used
- `modules/Calendar/iCal/iCalendar_*.php` (Bennu) - Replaced with sabre/vobject
- `modules/Calendar/iCal/iCalcreator` - Replaced with sabre/vobject
- `include/Zend` - Simplified
- `gdpr/include/vtwsclib/lib/Zend` - Removed
- `portal/include/Zend` - Removed
- `videlalvaro/php-amqplib` - Replaced with php-amqplib/php-amqplib
- `web-token/jwt-easy`, `web-token/jwt-key-mgmt`, `web-token/signature-pack` - Replaced with web-token/jwt-library

Zend framework replaced with Laminas

The Zend framework (a discontinued project) has been completely removed and replaced with **Laminas**, its official successor. All Zend dependencies have been migrated to the Laminas equivalents.

Removed config variables

The following config variables have been removed:

`$display_empty_home_blocks`

Never used.

`$php_max_execution_time`

Misleading, since it was applied only in a few cases, and it was always 0, setting the limit to unlimited. Replaced by the `$php_config` and specific overrides when needed (e.g., Report export, PDF generation).

Removed files and folders

`modules/Dashboard`

Removed this module, already deprecated and inactive.

`PEAR.php`

Used only by Dashboard and by obsolete libraries.

`Image/*`

Used only by Dashboard module.

`include/db_backup`

Not used anymore (and also a bad idea to make a full backup on admin logout).

`Smarty/templates_c`

Moved to `cache_local/smarty` to keep the number of writable folders limited.

`plugins/erpconnectorDir`

This folder was used in the past to perform one/two-way synchronizations with external systems. It has now been replaced by the [Data Import](#).

Old portal (v1)

The previous version of the portal is **no longer supported**. You can choose one of the following options:

1. **Switch to the new Business Portal**

Go to *Settings* → *Business Portal* to start using the updated version.

2. **Restore the portal from the upgrade backup**

You may restore the previous version from the upgrade backup and update it to ensure compatibility with the new system. Please note that the old portal is **not compatible with PHP 8**. For more information and assistance, contact support.