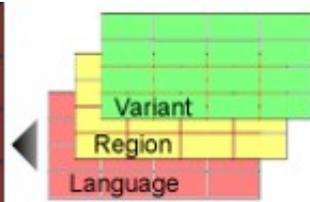
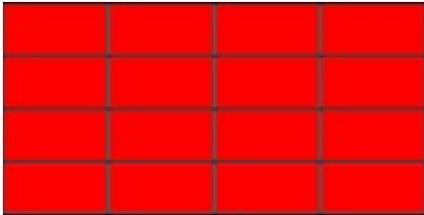


Klasse Translate Version 0.0.2

Diese Klasse stellt das komplette Handling der Übersetzungstabellen für den Kern und auch alle Zusatzmodule zur Verfügung.
Abhängigkeiten: Es werden zusätzlich die Klassen TranslateLanguage, TranslationException sowie ein passender Adapter benötigt.

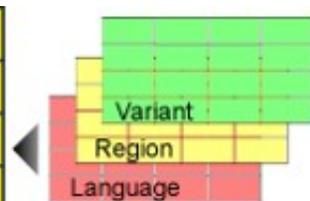
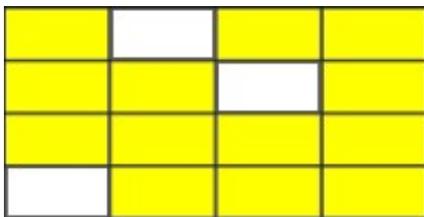
Schichtenmodell der Sprachquellen

Sprache 1 (Hardcoded System-Default)



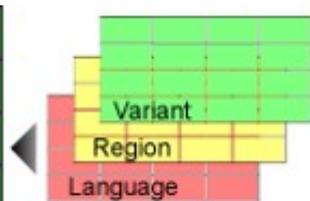
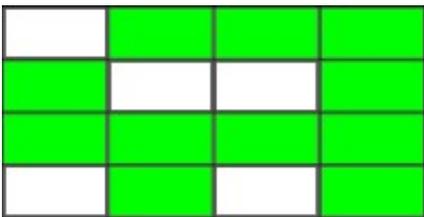
In der Regel ist das englisch
Diese Sprachdatei muss zwingend alle verfügbaren Texte enthalten
Sprachdateien setzen sich bei Bedarf überlagernd aus der Hauptsprache, der regionalen Sprache sowie einer Variation (Dialekt) zusammen.

Sprache 2 (DEFAULT_LANGUAGE)



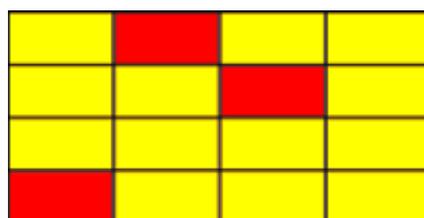
Die in den Systemeinstellungen vorgegebene, allgemeine Sprache. Ist sie identisch zum System-Default, wird sie ignoriert werden.

Sprache 3 (LANGUAGE)

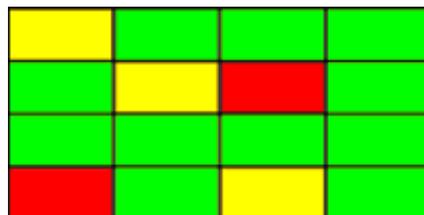


Die eingestellte Benutzersprache. Ist sie identisch zu DEFAULT_LANGUAGE, wird sie ignoriert.
Ist sie identisch zum System-Default, wird sie, ebenso wie DEFAULT_LANGUAGE, ignoriert und nur noch System-Default verwendet.

Die geladenen Sprachmodule werden dann in der entsprechenden Reihenfolge nacheinander überlagert, woraus sich dann die endgültig verfügbare Texttabelle ergibt.



1. Überlagerung
Sprache 2 über Sprache 1 gelegt.



2.Überlagerung und Endergebnis
Sprache 3 über das Ergebnis der 1.Überlagerung gelegt.

Durch die Überlagerungstechnik ergibt sich ein Fallback-Mechanismus der dafür sorgt, dass Lücken durch fehlende Übersetzungen von Inhalten der zuvor geladenen Tabellen aufgefüllt werden.

Die Klasse Translate ist so aufgebaut, dass für jede Sprache Länderspezifische Anpassungen und sogar einzelne Dialekte definiert werden können. Für `de_AT_tir`, also österreichisches deutsch mit Tiroler Dialekt werden daher folgende Übersetzungstabellen benötigt: `de` → deutsche Grundsprachen + `de_AT` → allgemeine österreichische Abwandlungen von deutsch + `de_AT_tir` → Tiroler Dialektabweichungen.
`en_US_TEX` → amerikanisches englisch mit texanischen Dialekt.

Klasse **Translate** Version 0.0.2

Methoden der Klasse 'Translate'

static getInstance() : object Der zentrale Zugriffspunkt, um die Klasse an jeder Stelle der Applikation aufzurufen.

__isset(\$sLanguageKey) : boolean

@param \$sLanguageKey TranslationCode
@description überprüft, ob für einen Schlüssel eine Übersetzung existiert.

__get(\$sLanguageKey) : string

@param \$sLanguageKey TranslationCode
@description gibt die Übersetzung zu einem Schlüssel zurück

initialize(\$sDefaultLanguage, \$sUserLanguage, \$sAdaptor)

@param \$sDefaultLanguage die aktuell eingestellte Default-Sprache
@param \$sUserLanguage die eingestellte Benutzersprache (default = 'sDefaultLanguage').
@param \$sAdaptor zu verwendender Adapter. (default = 'WbOldStyle')
@description Initialisiert die Klasse und setzt die gewünschten Sprachen

addAddon(\$sAddon = "")

@param \$sAddon Bezeichnung des Addons, dessen Sprachdateien zu laden sind. Die Bezeichnung entspricht der üblichen Namespace-Syntax
Beispiel: `modulemyModule` für alte Module oder `admin\pages` für alte Backendaddons. Im Prinzip ist die Benennung nichts anderes als die Pfadangabe, ausgehend von `sAppPath` jedoch mit Backslash anstatt Slash.
@description Lädt die gewünschten Sprachdefinitionen eines Addons unter Nutzung des Schichtenmodells.

enableAddon(\$sAddon)

@param \$sAddon Bezeichnung des Addons
@description aktiviert ein Addon. Bei Bedarf wird es nachgeladen.

disableAddon()

@description deaktiviert ein evt. aktiviertes Addon.

getLangArray() : array (deprecated)

gibt die komplette Übersetzungstabelle von Kern + aktivem Addon zurück.

Achtung: Die Nutzung dieser Methode ist unerwünscht und wird kurzfristig in einer der Folgeversionen wieder entfernt. Sie dient nur für eine kurze Übergangszeit zur Versorgung der **phplib** Templateengine.

Die Klasse ist in der Lage, zusammengesetzte ISO-Sprachcodes(RFC4234) nach dem Muster

`2*3ALPHA (ISO639) language`

`2ALPHA (ISO3166) region`

`3*5alphanum (registered Variants) variant`

zu verarbeiten. Beispiel: `de_CH_URI`. Hierbei werden die Sprachen von links nach rechts geladen, wobei die nachfolgende immer die vorhergehende überlagert.

`de.php <== de_CH.php <== de_CH_URI.php`. Dadurch genügt es, in der Region-Datei und in der Variant-Datei jeweils nur die Differenzen zum aktuellen Zustand gespeichert werden müssen.

Nutzungshinweise

Die Klasse wird direkt beim Start der Applikation automatisch initialisiert und die Übersetzungstabelle des Kerns geladen.

Für die bisherigen Module von WebsiteBaker gilt: Für jede Funktion(Datei), die Übersetzungstabellen benötigt, muss die Modul-Sprachtable mit

Translate::getInstance()->enableAddon('modules\Addon'); geladen werden. Beim Ende des Modulcodes (bevor der Rückfall zum Kern-Code erfolgt) muss die Addon - Übersetzungs-Tabelle wieder mit

Translate::getInstance()->disableAddon(); neutralisiert werden.

Zusätzlich zu den Kern-Übersetzungstabellen ist zu jedem Zeitpunkt nur jeweils eine einzige zusätzliche Addon-Übersetzungstabelle im direkten Zugriff.