

Von 5.3.33 auf 5.3.57

Das ist kein Extension-Update, das ist ein

Plattformwechsel. 5.3.33 läuft auf TYPO3 12.4 und 13.4, 5.3.57 verlangt `^14.3`. Die Extension lässt sich erst installieren, wenn der Core steht.

Dazwischen liegen 24 Versionen: neun Upgrade-Wizards, rund vierzig neue Datenbankspalten, der Umzug von FontAwesome auf `EXT:iconpack`, ein Hauptversionssprung bei `b13/container` und das eigene SCSS, das aus Dateien in den Konfigurations-Datensatz gewandert ist.

Die Reihenfolge ist nicht beliebig. Wer die Wizards vor der Datenbankanalyse startet, bekommt Fehler; wer den Datensatz vor den Wizards speichert, schreibt halbe Konstanten. Diese Anleitung geht sie in der Reihenfolge durch, in der sie funktioniert.

VON → AUF

5.3.33 → 5.3.57

TYPO3

14.3

PHP

8.2 – 8.5

Zuerst der Core	TYPO3 14.3 ist Pflicht. 5.3.33 lief auf 12.4/13.4 — es gibt keine Zwischenversion von t3sbootstrap, die beides kann.
Abhängigkeiten	<code>b13/container</code> springt von <code>^3.1</code> auf <code>^4.1</code> . <code>rte_ckeditor</code> und <code>fluid_styled_content</code> sind jetzt ausdrückliche Anforderungen. <code>ichhabrecht/content-defender</code> ist keine mehr.
Datenbank	Rund vierzig neue Spalten, dazu entfallene wie <code>jumbotron_rounded</code> , <code>general_rootline</code> , <code>compress</code> , <code>tx_t3sbootstrap_flexbox_*</code> . Datenbankanalyse ist Pflicht, nicht Kür.
Neun Wizards	Backend-Layouts, drei für Iconpack, Anker, Asset-Pfade, Navbar-Logo und zwei RTE-Einstellungen. Abschnitt 5 nennt sie einzeln.
FontAwesome	Die Felder <code>tx_t3sbootstrap_fontawesome_icon</code> und <code>tx_t3sbootstrap_header_fontawesome</code> sind weg. Wer Icons gesetzt hat, braucht <code>EXT:iconpack</code> und <code>EXT:iconpack_fontawesome</code> — vor den Wizards.
Eigenes SCSS	Quelle ist jetzt der Konfigurations-Datensatz (<code>custom_scss</code> , <code>custom_variables_scss</code>), nicht mehr die Datei in <code>typo3temp</code> . Die Schalter dafür sind aus der Extension-Konfiguration in die Site Settings gezogen.
Konfigurationstyp	Der Datensatz hat Typen bekommen (<code>record_type</code>). Nach dem Update einmal öffnen und speichern — sonst fehlen Konstanten.
Neue Schalter	Überzeile, senkrechte Überschrift, Speaking ID, Titelfarbe, Seiten-Opt-out, Inhaltseinzug — alle ab Werk aus . Bestehende Seiten ändern sich dadurch nicht.
Zeitbedarf	Rechne mit einem halben Tag auf einer Kopie, nicht mit einer Mittagspause auf Live.

Nur die Punkte, die beim Upgrade etwas von dir verlangen. Reine Bugfix-Releases sind weggelassen.

VERSION	WAS ZU TUN IST
5.3.40 05.05.2026	Support für TYPO3 v14. Ab hier ist der Core-Sprung überhaupt möglich.
5.3.42 17.05.2026	Das Site Set „T3S Bootstrap — OTHER SETTINGS“ wird obligatorisch. Fehlt es in der Site-Konfiguration, fehlen Einstellungen. Dazu neu: Icon statt Bild, Option <i>Titelfarbe</i> , deutsche Backend-Übersetzung.
5.3.46 20.06.2026	Der <i>BackendlayoutUpgradeWizard</i> kommt dazu — der erste Wizard in Abschnitt 5.
5.3.48 26.07.2026	Speaking ID wird eingeführt (Schalter, ab Werk aus) und die Card mit Icon.

VERSION	WAS ZU TUN IST
5.3.50 22.08.2026	EXT:t3sb_package wird nicht mehr gebraucht. Die erzeugten Dateien liegen jetzt unter <code>typo3temp/assets/t3sbootstrap/</code> ; der Wizard <i>Move generated assets out of EXT:t3sb_package</i> holt vorhandene einmalig herüber. Ebenfalls hier: b13/container ab <code>^4.1</code> , Export & Import der Konfiguration im Backend-Modul, neue Einstellung <code>flexformDir</code> .
5.3.51 01.09.2026	Drei neue Spalten <code>navbar_image_width/height/alt</code> plus der Navbar-Wizard; die alten Site-Settings heißen im Editor jetzt DEPRECATED – Badges bekommen ein eigenes RTE-Werkzeug (<code>rteStyleBadges</code> → <code>rteBadge</code> , mit Wizard), dazu das neue Preset <code>t3sbootstrap_extended</code> . Geändertes Verhalten: die Lightbox beachtet jetzt den Bildzuschnitt.
5.3.52 02.10.2026	Jumbotron-Hintergrundvideo — neue Spalten, DB-Update nötig.
5.3.55 14.10.2026	Inhaltseinzug und senkrechte Überschrift — neue Spalten, DB-Update nötig. (5.3.54 trägt denselben Inhalt; der Tag wurde nachträglich umgehängt.)
5.3.56 15.10.2026	Werkzeug <i>Klassen</i> im RTE, das Backend-Layout wird an Unterseiten vererbt.
5.3.57 19.10.2026	<code>div</code> als Block im RTE, <code>h1</code> kann den Navigationstitel nehmen — neue Spalte, DB-Update nötig.

Zu 5.3.34 bis 5.3.39 gibt es weder Tags noch Log-Einträge — die Nummern sind nie einzeln veröffentlicht worden.

1

Reihenfolge: zuerst TYPO3, dann die Extension

5.3.57 fordert in `ext_emconf.php` und `composer.json` ausdrücklich TYPO3 **14.3**. Ein Versuch, die Extension auf einer 12er oder 13er Installation zu aktualisieren, scheitert bereits bei Composer. Der Weg ist deshalb immer:

1. Core	TYPO3 auf 14.3 bringen — bei 12.4 über 13.4 als Zwischenstufe, das ist der übliche Core-Pfad und hat mit t3sbootstrap nichts zu tun.
2. Abhängigkeiten	b13/container auf <code>^4.1</code> . Das ist ein Hauptversionssprung mit eigenen Änderungen — die Release Notes von b13 gehören dazugelesen.
3. t3sbootstrap	Erst jetzt auf 5.3.57.
3a. Site Set	T3S Bootstrap — OTHER SETTINGS in der Site-Konfiguration eintragen, falls noch nicht geschehen. Seit 5.3.42 obligatorisch.
4. Datenbank	Analyse ausführen, bevor irgendein Wizard startet.
5. Wizards	In der Reihenfolge aus Abschnitt 5.
6. Datensatz	Konfigurations-Datensatz öffnen, speichern, Cache leeren.

⚠ NICHT AUF LIVE

Dieser Sprung lässt sich nicht in Einzelschritten zurücknehmen. Arbeite auf einer Kopie der Instanz und spiele sie erst live, wenn das Frontend geprüft ist. Datenbank-Dump und `fileadmin` vorher sichern.

Was sich bei den Anforderungen geändert hat: `php` 8.2–8.5, `typo3` 14.3, neu als harte Abhängigkeit `fluid_styled_content` und `rte_ckeditor`, `container` 4.1. Entfallen ist `ichhabrecht/content-defender` — wer es weiter nutzt, muss es selbst in die `composer.json` des Projekts aufnehmen.

2 Vorbereitung

Drei Dinge, die nach dem Update nicht mehr auslesbar sind und die man dann vermisst:

- **Die Extension-Konfiguration abschreiben.** Mehrere Optionen sind verschoben oder entfallen (Abschnitt 6). Ein Screenshot der Tabs genügt.
- **Eigenes SCSS sichern.** Alles aus `public/typo3temp/assets/t3sbootstrap/`, besonders `custom-*.scss` und `custom-variables-*.scss`. `typo3temp` ist nach Core-Definition Wegwerfware — jedes Update darf es leeren.
- **Den Konfigurations-Datensatz sichern.** Seit 5.3.50 geht das direkt im Backend-Modul *T3SB Config* über *Export* — das ist der bequemste Weg zurück, falls beim Speichern etwas schiefgeht. Auf einer 5.3.33 gibt es das noch nicht; dort die Seite mit dem Datensatz exportieren oder die Werte abschreiben.

Wer FontAwesome-Icons in Inhalten, Überschriften oder Seiteneigenschaften gesetzt hat, installiert jetzt `EXT:iconpack` und `EXT:iconpack_fontawesome`. Die drei Iconpack-Wizards in Abschnitt 5 brauchen sie; ohne sie migrieren sie ins Leere.

3 Das Paket aktualisieren

```
composer require t3sbs/t3sbootstrap:^5.3.57
vendor/bin/typo3 extension:setup
vendor/bin/typo3 cache:flush
```

Meldet Composer einen Konflikt, liegt es fast immer an einem der drei Punkte aus Abschnitt 1: Core noch nicht auf 14.3, `b13/container` noch auf 3.x, oder `content-defender` hängt noch als transitive Abhängigkeit im Projekt.

4 Datenbank

```
vendor/bin/typo3 database:updateschema
```

Oder im Installtool unter *Analyze Database Structure*. Beides muss laufen, **bevor** ein Wizard startet — die Wizards schreiben in Spalten, die es sonst noch nicht gibt.

Was dazukommt

Konfigurations-Datensatz	<code>record_type</code> (die neuen Typen), <code>custom_scss</code> und <code>custom_variables_scss</code> (das SCSS wandert in den Datensatz), <code>page_titlenavtitle</code> , <code>navbar_image_width/height/alt</code> , die <code>jumbotron_bgvideo_*</code> -Felder.
Inhaltselemente	<code>tx_t3sbootstrap_anchor</code> (Speaking ID), <code>tx_t3sbootstrap_supraheader</code> und <code>_class</code> (Überzeile), <code>tx_t3sbootstrap_header_vertical</code> , <code>tx_t3sbootstrap_no_indent</code> .
Seiten	<code>tx_t3sbootstrap_indent</code> sowie die Opt-out-Schalter <code>tx_t3sbootstrap_navbar_disable</code> , <code>_breadcrumb_disable</code> , <code>_meta_disable</code> , <code>_jumbotron_disable</code> , <code>_pagetitle_disable</code> , <code>_footer_disable</code> .

Was verschwindet

`jumbotron_rounded`, `general_rootline`, `general_override`, `compress`, `container_error`, `tx_t3sbootstrap_flexbox_columns` und `_prefix`, `tx_t3sbootstrap_chapter`, `tx_t3sbootstrap_picture_variants`, `tx_t3sbootstrap_media_width` sowie die beiden FontAwesome-Felder `tx_t3sbootstrap_fontawesome_icon` (Seiten) und `tx_t3sbootstrap_header_fontawesome` (Inhalte).

⚠ SPALTEN ERST AM ENDE LÖSCHEN

Die Datenbankanalyse bietet das Löschen der alten Spalten an. Lass es stehen, bis die Iconpack-Wizards durch sind und das Frontend geprüft ist — die beiden FontAwesome-Felder sind die **Quelle** der Migration. Einmal gelöscht, sind die Icons nicht wiederherstellbar.

5 Die neun Upgrade-Wizards

Installtool → *Upgrade* → *Upgrade Wizard*, oder einzeln auf der Kommandozeile. Die Reihenfolge unten ist die empfohlene: erst Struktur, dann Icons, dann Einstellungen.

```
vendor/bin/typo3 upgrade:list
vendor/bin/typo3 upgrade:run t3sbootstrap_backendlayoutUpgradeWizard
```

WIZARD

WAS ER TUT

backendlayout

t3sbootstrap_backendlayoutUpgradeWizard

Entfernt den Suffix `_Extra` aus `backend_layout` und `backend_layout_next_level` der Seiten. **Zuerst ausführen** — ohne ihn rendern Seiten mit altem Layout-Schlüssel ins Leere.

legacyAssetLocation

t3sbootstrap_legacyAssetLocationUpgradeWizard

Holt die erzeugten Dateien einmalig aus `EXT:t3sb_package` nach `typo3temp/assets/t3sbootstrap/`. Seit 5.3.50 wird das Sitepackage nicht mehr gebraucht — **danach kann `EXT:t3sb_package` aus dem Projekt**. Fehlendes TypoScript und SCSS schreibt die Extension beim nächsten Frontend-Aufruf aus der Datenbank neu.

iconpackTitle

t3sbootstrap_iconpackTitleUpgradeWizard

FontAwesome-Icons aus `pages:tx_t3sbootstrap_fontawesome_icon` nach Iconpack.

iconpackHeader

t3sbootstrap_iconpackHeaderUpgradeWizard

Dasselbe für `tt_content:tx_t3sbootstrap_header_fontawesome`.**iconpackBodytext**

t3sbootstrap_iconpackBodytextUpgradeWizard

Dasselbe für Icons im `bodytext` — inklusive `fa-fw`, Größe, `fa-rotate` und Dekoration.**navbarBrandImage**

t3sbootstrap_navbarBrandImageUpgradeWizard

Zieht `settings.bootstrap.navbar.image.defaultPath/width/height/altText` aus jeder Site-Konfiguration in den Konfigurations-Datensatz.**rteBadgeSetting**

t3sbootstrap_rteBadgeSettingUpgradeWizard

Benennt `rteStyleBadges` in `rteBadge` um — die Badges sind aus dem Styles-Menü in ein eigenes Werkzeug gewandert.**rteColorSetting**

t3sbootstrap_rteColorSettingUpgradeWizard

Dasselbe für `rteStyleColors` → `rteColor`.**anchor**

t3sbootstrap_anchorUpgradeWizard

Erzeugt die fehlenden Speaking IDs aus den Überschriften. **Läuft nur, wenn der Schalter `speakingID` vorher eingeschaltet ist** — sonst meldet er „nothing to generate“. Als Zuletzt-Schritt gedacht.

Alternativ zum Anker-Wizard der Befehl `vendor/bin/typo3 t3sbootstrap:anchor`. Mit `--mode clear` räumt er gespeicherte Anker wieder weg, falls du die Speaking ID doch nicht willst.

6

Extension-Konfiguration

Entfallen — jetzt woanders

customScssJetzt Site Setting `bootstrap.cdn.customScss`.**bootswatch**Jetzt Site Setting `bootstrap.cdn.bootswatch`.**editScss**

Entfallen — bearbeitet wird im Konfigurations-Datensatz.

keepVariables

Mit 5.3.57 ersatzlos gestrichen. Es hat sein Versprechen nie gehalten: Der Datensatz hat die Datei bei jedem Speichern überschrieben.

Neu — alle ab Werk aus

Diese Schalter blenden nur Felder ein. Solange sie aus sind, ändert sich an bestehenden Seiten nichts:

Inhalt`supraheader` (Überzeile), `headerVertical` (senkrechte Überschrift, nur im Background-Wrapper), `speakingID` (sprechende Anker), `allowReferences` (Referenzen in Card- und Carousel-Wrapper).**Seite**`titlecolor` (Titel- und Untertitelfarbe), `pageOptOut` (Tab „T3SB“ mit den Abschalt-Schaltern je Seite), `contentIndent` und `contentIndentInherit` (Inhaltseinzug in Grid-Spalten).**Vorschau**`pageInfo` („Page Info“ im Seitenmodul, ab Werk **1**).**FlexForm**`flexformDir` (seit 5.3.50) — Verzeichnis für eigene FlexForm-Erweiterungen.

Nach jeder Änderung an der Extension-Konfiguration: Cache leeren. Die Schalter greifen ins TCA.

Die wichtigste Änderung für alle, die eigenes SCSS pflegen: **Quelle ist der Konfigurations-Datensatz**, nicht mehr die Datei. Die Felder `custom_scss` und `custom_variables_scss` sind dafür neu dazugekommen; die Datei in `typo3temp` ist nur noch das Erzeugnis.

Nach dem Update einmal durchlaufen lassen:

```
vendor/bin/typo3 t3sbootstrap:cdnToLocal
vendor/bin/typo3 t3sbootstrap:customScss <Rootseiten-ID>
```

Beides gibt es auch als Scheduler-Aufgaben („T3SB CDN to local“, „T3SB Custom SCSS“). `cdnToLocal` braucht dafür einen **ausgehenden Internetzugang des Webserver**s — nicht des Browsers — zu `github.com`, `cdn.jsdelivr.net`, `cdnjs.cloudflare.com`, `unpkg.com`, `code.jquery.com`, `gwfh.mranftl.com` und `bootswatch.com`. In abgeschotteten Umgebungen läuft er sonst ins Leere und die Seite rendert ohne Stylesheet.

▲ TYPO3TEMP IST WEGWERFWARE

Jedes Update, jedes Deployment und jedes Cache-Leeren darf `typo3temp/assets/t3sbootstrap/` ausräumen. Was daraus neu gebaut werden kann, kommt wieder; die heruntergeladenen Bootstrap-Quellen kommen nur mit Internetzugang wieder. Deshalb die beiden Befehle oben nach jedem Deployment.

Der Datensatz hat Typen bekommen (`record_type`) — statt eines langen Formulars wählst du oben die Bootstrap-Komponente und bekommst nur deren Felder. Bestehende Datensätze landen im Typ *general*.

Zum Abschluss, in dieser Reihenfolge:

Öffnen	Den Datensatz auf der Rootseite im Modul <i>Liste</i> aufrufen.
Prüfen	Navbar-Logo (der Wizard hat es aus der Site-Konfiguration geholt), Seitentitel, Jumbotron, Footer.
Speichern	Einmal speichern, auch ohne Änderung. Erst dabei werden die Konstanten nach <code>t3sbconstants.typoscript</code> geschrieben.
Cache	<code>vendor/bin/typo3 cache:flush</code>

- **Eine Seite je Backend-Layout** — der Layout-Wizard hat Schlüssel umgeschrieben.
- **Eine Seite mit Icons** in Seitentitel, Überschrift und Fließtext — die drei Iconpack-Wizards.
- **Navbar mit Logo** — Pfad, Breite, Höhe, Alt-Text sitzen jetzt im Datensatz.
- **Ein Text im RTE** öffnen und speichern, danach `bodytext` vergleichen. Badges und Textfarben haben eigene Werkzeuge bekommen.

- **Container-Elemente** — `b13/container` 4.x ist ein eigener Hauptversionssprung.
- **Eine Lightbox mit zugeschnittenem Bild** — seit 5.3.51 öffnet sie den Ausschnitt statt des Originals. Das volle Bild gibt es über die Option *Originalbild für Lightbox verwenden*. Das ist geändertes Verhalten, kein Fehler.
- **Das Stylesheet** — wenn die Seite unformatiert ist, fehlen die Dateien aus Abschnitt 7.

10

Checkliste

• Pflicht

- ☐ **Kopie der Instanz** — Datenbank-Dump und `fileadmin` gesichert, Arbeit nicht auf Live
- ☐ **Extension-Konfiguration und SCSS gesichert** — Screenshot der Tabs, Dateien aus `typo3temp/assets/t3sbootstrap/`
- ☐ **TYPO3 auf 14.3** — vorher geht nichts
- ☐ **`b13/container` auf ^4.1** — Hauptversionssprung, eigene Release Notes lesen
- ☐ **Paket aktualisieren** — `composer require t3sbs/t3sbootstrap:^5.3.57`, dann `extension:setup`
- ☐ **Site Set „T3S Bootstrap — OTHER SETTINGS“ eintragen** — seit 5.3.42 obligatorisch, ohne es fehlen Einstellungen
- ☐ **Datenbank analysieren** — rund vierzig neue Spalten. Alte Spalten *noch nicht* löschen
- ☐ **Backend-Layout-Wizard zuerst** — `t3sbootstrap_backendlayoutUpgradeWizard`
- ☐ **Restliche Wizards** — in der Reihenfolge aus Abschnitt 5
- ☐ **SCSS und CDN-Dateien erzeugen** — `t3sbootstrap:cdnToLocal` und `t3sbootstrap:customScss`
- ☐ **Konfigurations-Datensatz speichern** — einmal, auch ohne Änderung, danach Cache leeren

- **Nur wenn es dich betrifft**

- ☐ **FontAwesome-Icons im Einsatz** — `EXT:iconpack` und `EXT:iconpack_fontawesome` installieren, **bevor** die drei Iconpack-Wizards laufen
- ☐ **content-defender** wird weiter gebraucht — selbst in die `composer.json` des Projekts aufnehmen, es ist keine Abhängigkeit mehr
- ☐ **EXT:t3sb_package** im Einsatz — nach dem Asset-Wizard aus dem Projekt nehmen; seit 5.3.50 wird es nicht mehr gebraucht
- ☐ **Eigenes RTE-Preset gesetzt** — seit 5.3.51 gibt es zusätzlich `t3sbootstrap_extended` mit breiterer Klassen-Whitelist
- ☐ **Navbar-Logo über die Site-Konfiguration gesetzt** — nach dem Wizard im Datensatz prüfen und die alten `bootstrap.navbar.image.*` entfernen — sie heißen im Settings-Editor seit 5.3.51 `DEPRECATED – ...`
- ☐ **Eigenes SCSS** — Inhalt im Konfigurations-Datensatz prüfen, `bootstrap.cdn.keepVariables` aus der Site-Konfiguration entfernen
- ☐ **Speaking ID gewünscht** — `speakingID` einschalten, danach den Anker-Wizard laufen lassen
- ☐ **Webserver ohne Internetzugang** — die Dateien aus Abschnitt 7 einmal auf einer Maschine mit Zugang erzeugen und mitdeployen

- **Zum Prüfen**

- ☐ **Je eine Seite pro Backend-Layout** im Frontend
- ☐ **Icons** in Seitentitel, Überschrift und Fließtext
- ☐ **RTE** — Text öffnen, speichern, `bodytext` vergleichen
- ☐ **Container-Elemente** nach dem Sprung auf b13/container 4
- ☐ **Erst danach** die alten Datenbankspalten löschen

✓ **FERTIG**

Der Sprung ist groß, aber er hat nur eine echte Einbahnstraße: die gelöschten FontAwesome-Spalten. Solange die stehen, lässt sich jeder Schritt wiederholen.