

### Master-Slave Extension (1.0.1) : weitere Funktionen für Master/Slave VEYTON®-Plugin mit nützlichen Erweiterungen für Detailansicht und Listing

#### I. Allgemeines

Das Plugin ermöglicht einstellbar bestimmte Daten des Masters für die Slaves zu übernehmen und stellt wahlweise auch im Produktlisting die Slaves komplett mit ihren Optionen zur Verfügung.

Damit ist je nach Anforderung erhebliche Arbeitersparnis im Backend möglich und im Listing kann eine Produktauswahl inkl. Warenkorb über die Slave-Attribute realisiert werden. Slaves und Master mit Nullbestand können wahlweise automatisch deaktiviert werden.

**NEU:** Cross-Sell-Artikel des Masters können verwendet werden.

**NEU:** Eigene Felder in products und products\_description können verwendet werden.

**Durch das Aufsummieren der Orders bei den Mastern erscheinen diese jetzt auch bei den Bestsellern !**

Die Lizenzierung bezieht sich auf den/die Domain(s) und localhost für offline-Benutzung.

#### II. Installation

Wie jedes andere Plugin für VEYTON®, ist die Installation leicht und bequem: Plugin ins Plugin-Verzeichnis des Shops kopieren und unter Inhalte->Plugins->deinstallierte Plugins aufrufen und installieren. Danach unter „installierte Plugins“ aktiv setzen.

#### III. Plugin-Einstellungen

Das Plugin wird mit dem Häkchen bei „Plugin Status“ für den gesamten Shop aktiviert. Pro Mandant muss es ebenfalls aktiviert werden.

#### IV. Einstellungen im Backend

Standard	
Name:	iti_ms_extension Plugin for VEYTON
Plugin Status:	<input checked="" type="checkbox"/>
Code:	iti_ms_extension

  

Main Store	
Slaves und Attribute im Produktlisting:	true
Master/Slaves deaktivieren bei Nullbestand:	false
Downloads des Masters verwenden:	true
Cross-Sell-Produkte des Masters verwenden:	true
Kurz-Beschreibung des Masters verwenden:	true
Beschreibung des Masters verwenden:	true
weitere Bilder des Masters verwenden:	true
Hauptbild des Masters verwenden:	true
Orders addieren Master:	true
aktiv:	true
Bestandsführung Master:	true
eigene Felder aus Produkt uebernehmen ( Liste mit Komma getrennt ):	products_addon_cats,products_addon_infos
eigene Felder aus Beschreibung uebernehmen ( Liste mit Komma getrennt ):	products_order_limit_hinweis

Die Einstellungen sind selbsterklärend: Wichtig ist der Punkt - Deaktivieren bei Nullbestand - hier ist abzuwägen, ob sich dies evt. negativ auf die SEO auswirkt. Slaves und Attribute im Produktlisting erlaubt unter anderem, bereits dort z.B. Icons der Slaves inkl. Warenkorb-Button unterzubringen.

Wenn selbst angelegte Felder aus der xt\_products oder der xt\_products\_description ebenfalls vom Master auf die Slaves übertragen werden sollen, können diese als Liste mit Komma getrennt in den entsprechenden Feldern definiert werden.

## V. Initialisierung

Im Backend erscheint ein neuer Eintrag in der Rubrik „Shop“:



Damit werden die Bestände und Bestellungen erstmalig aufsummiert und bei entsprechender Einstellung Master und Slaves mit Nullbestand deaktiviert:



### Bestände/Bestellungen werden auf Master summiert

#### Master und Slaves mit Null-Bestand werden deaktiviert !

Master: master Slave: Slave rot-large Bestand: 0.00 Orders: 96 **deaktiviert**  
Master: master Slave: Slave gelb large Bestand: 10.00 Orders: 4  
Master: master Slave: Slave grün Bestand: 10.00 Orders: 0  
Master: master Slave: Slave rot-small Bestand: 20.00 Orders: 0  
Master: master Slave: Slave rot-medium Bestand: 20.00 Orders: 0  
Master: master Slave: Slave gelb small Bestand: 10.00 Orders: 0  
Master: master Slave: Slave gelb medium Bestand: 10.00 Orders: 1  
Bestand Master: **80** Orders Master: **101**

### Bestände/Bestellungen wurden auf Master summiert

## VI. Variable im Produkt-Listing

Im Produktlisting wird bei gewählter Einstellung: Slaves im Produktlisting == true eine zusätzliche Tabelle „slaves“ mit ihrerseits identischem Aufbau wie im normalen Listing ausgegeben, Auf die Slaves kann mit einem „foreach“ zugegriffen werden , z.B. im Template product\_listing\_v1.html:

```
{foreach name=aussen item=module_data from=$product_listing}
.....
    {foreach name=innen item=slave_data from=$module_data.slaves}
    .....
    {foreach}
    .....
{foreach}
```