

# Joomla.asset.json - Dokumentation (Joomla Template)

---

Dieses Dokument beschreibt Aufbau, Zweck und empfohlene Struktur der Template-Datei *joomla.asset.json* des Joomla-Templates **joomla-tpl-prosoft** und dient als technische Referenz für Registrierung, Versionierung und Einbindung der CSS- und JavaScript-Assets über den Joomla WebAssetManager.

## 1. Dateien-Überblick

---

joomla.asset.json

## 2. Schema und Daten

---

Die Datei verwendet das offizielle Joomla-Web-Assets-Schema zur Validierung der Asset-Struktur und zur Sicherstellung der Joomla-5-Konformität. Die enthaltenen Metadaten dienen der eindeutigen Zuordnung, Versionierung und zentralen Verwaltung der CSS- und JavaScript-Assets des Templates.

### 2.1 JSON-Schema

Diese Angabe definiert das offizielle Joomla-Web-Assets-Schema:

*„\$schema“: „https://developer.joomla.org/schemas/json-schema/web\_assets.json*

#### Zweck:

- Validierung der Datei-Struktur
- Autovervollständigung & Fehlermeldungen in Editoren (z.B. VS Code, IntelliJ, usw.)
- Absicherung gegen Tippfehler und ungültige Asset-Definitionen

### 2.2 Name des Asset-Pakets

*„name“: „joomla-tpl-prosoft“*

- Eindeutiger Bezeichner des Asset-Pakets
- Interne Verwendung durch Joomla
- Sollte mit dem Namen des Templates übereinstimmen

## 2.3 Versionsangabe

„version“: „1.0.0“

- Version der Asset-Definitionen
- Nutzung für Wartung, Releases und Debugging

## 2.4 Beschreibung

„description“: „This file contains details oft he assets used by joomla-tpl-prosoft, the template used by ProSoft Krippner GmbH.“

- Freitext-Beschreibung der Datei
- Rein dokumentarisch
- Hilfreich für Teamarbeit und Übergaben

## 2.5 Einstiegspunkt

„assets“: [ ... ]

- Zentrales Array für alle registrierten CSS- und JavaScript-Dateien
- Jedes Objekt innerhalb des Arrays beschreibt genau ein Asset
- Definierten Namen werden in der *index.php* genutzt

### 2.5.1 Beispiele

```
„assets“: [  
    { „name“: „joomla-tpl-prosoft.styles“, ... },  
    { „name“: „joomla-tpl-prosoft.custom“, ... },  
    { „name“: „joomla-tpl-prosoft.hero“, ... },  
    { „name“: „joomla-tpl-prosoft.popup“, ... },  
    { „name“: „joomla-tpl-prosoft-template.script“, ... },  
    { „name“: „joomla-tpl-prosoft-font.script“, ... },  
    { „name“: „joomla-tpl-prosoft-darkmode.script“, ... },  
    { „name“: „joomla-tpl-prosoft-mobile.script“, ... }  
]
```

### 3. Asset-Eintrag-Struktur

---

Jeder Eintrag im Array „*assets*“: [ ... ] beschreibt **ein einzelnes Asset**, das über den Joomla *WebAssetManager* geladen werden kann.

#### 3.1 name (Handle / Schlüssel)

„*name*“: „*joomla-tpl-prosoft.styles*“

- Eindeutiger Name, der später in der *index.php* aufgerufen wird via `useStyle('joomla-tpl-prosoft.styles')`
- Muss **exakt** übereinstimmen, damit das Asset korrekt geladen wird

#### 3.2 description (Dokumentation)

„*description*“: „*The file containing the css script for this template.*“

- Rein beschreibend, keinerlei technische Funktion
- Dokumentation für bessere Zusammenarbeit

#### 3.3 type (Asset-Typ)

„*type*“: „*style*“ oder „*type*“: „*script*“

- Style → CSS-Datei
- Script → JavaScript-Datei

#### 3.4 uri (Dateipfad)

„*uri*“: „*media/templates/site/joomla-tpl-prosoft/css/template.css*“

- Pfad relativ zum Joomla-Root
- Bei Templates üblich: `media/templates/site/<template>/...`

#### 3.5 version (Cache-Busting)

„*version*“: „*auto*“

- Joomla hängt automatisch eine Version an die URL an, sodass der Browser die Caches korrekt validieren kann
- „*auto*“ liefert im Template-Kontext eine automatische Versionierung

### 3.6 dependencies (Abhängigkeiten)

`„dependencies“: [ „bootstrap.css“ ]`

- Sorgt dafür, dass Joomla Assets in der richtigen Reihenfolge lädt
- Wichtig, falls das CSS / JavaScript auf anderen Assets aufbaut

### 3.7 attributes (Skript-Attribute)

`„attributes“: { „defer“: true }`

- Zusätzliche Attribute für das Script-Tag
- defer → JavaScript wird erst nach dem Parsen des HTML ausgeführt (Performanceoptimierung)