

Reiter „Spalten“

Die Spaltenliste ist die ganze Schnittstelle einer Definition: **eine Spalte, die hier nicht steht, wird nicht geliefert**, lässt sich nicht filtern, sortieren oder durchsuchen und wird beim Anlegen und Ändern abgewiesen – gleich, was in der SQL-Tabelle steht.

Links die Liste der deklarierten Spalten, rechts das Formular zur ausgewählten.

Die Leiste über der Liste

Knopf	Wirkung
+ Spalte	hängt eine leere Spalte vom Typ <code>text</code> an
+ aus Tabelle ...	übernimmt Spalten aus dem gelesenen Tabellenaufbau (siehe unten)
↑ / ↓	verschiebt die ausgewählte Spalte; die Reihenfolge ist die der Anzeige
Kopie	legt eine Kopie mit dem Zusatz <code>_Kopie</code> direkt darunter an
Entfernen	nimmt die Spalte aus der Definition – die SQL-Tabelle bleibt unberührt

In der Liste steht hinter dem Namen der `meta.type`. Ein `key` kennzeichnet den Primärschlüssel. Eine hervorgehobene Zeile heißt: diese Spalte gibt es in der SQL-Tabelle nicht – der Router lässt sie aus jeder Antwort weg.

Ein **leerer Eintrag** entsteht durch ein Loch in der JSON-Liste (`[ { ... }, , { ... } ]`). Der Router übergeht ihn, das Speichern entfernt ihn.

+ aus Tabelle ...

Zeigt alle Spalten der SQL-Tabelle, die noch nicht deklariert sind, mit SQL-Typ, Schlüssel- und Pflichtkennzeichen. Angehakte werden übernommen, und der Editor schlägt dabei gleich etwas vor:

- Primärschlüssel → `meta.type = key`, nicht sichtbar
- Spalte `Licence` → `meta.type = licence`, nicht sichtbar
- `bit` → `checkbox`, `int` → `int`, `decimal` → `number`, `date` → `date`, `datetime` → `datetime`, alles andere → `text` mit Suchfeld

- Datums- und Zeitspalten mit Standardwert sowie hochgezählte Spalten werden beim Anlegen nicht änderbar
- `Principal` und `CompanyCode` bekommen die **Firmenliste** aus der Tabelle `principal` als Werteliste vorgeschlagen – bei `CompanyCode` mit dem Standardwert `ANY`. Liegt die Definition nicht selbst in `DEX_MasterData`, wird die Verbindung dorthin gleich mit eingetragen

Alle Vorschläge lassen sich danach ändern.

Das Formular einer Spalte

Ist der Tabellenaufbau gelesen, steht oben eine Zeile dazu, was die Datenbank über diese Spalte sagt: Typ und Länge, Primärschlüssel, hochgezählt, NULL oder NOT NULL, Standardwert und ob sie beim Anlegen Pflicht ist.

Feld	Eigenschaft	Bedeutung
Name	<code>name</code>	Der SQL-Spaltenname. Er ist der Schlüssel in den Zeilen und beim Schreiben. Pflichtfeld
Beschriftung	<code>label</code>	Fehlt sie, zeigt der Client den Namen
SQL-Typ (<i>nur View</i>)	<code>type</code>	Views werden nicht ausgelesen – der Typ kommt von hier. Ohne ihn meldet die API <code>string</code>
Typ	<code>meta.type</code>	<code>text</code> , <code>int</code> , <code>number</code> , <code>date</code> , <code>datetime</code> , <code>checkbox</code> , <code>list</code> , <code>multiplelist</code> , <code>combobox</code> , <code>comboboxText</code> , <code>popup</code> , <code>iconorlist</code> , <code>key</code> , <code>licence</code> , <code>hidden</code> . <code>key</code> und <code>licence</code> sind die technischen Spalten, <code>hidden</code> lässt die Spalte in der API, aber nicht im Formular
Sichtbar	<code>meta.show</code>	<i>nein</i> blendet die Spalte in der Liste aus
Standardwert	<code>meta.defaultValue</code>	Wird beim Anlegen vorbelegt. Als Text, Zahl, <i>ja</i> , <i>nein</i> oder <i>leer (null)</i>
Suchfeld	<code>meta.searchAutocomplete</code>	Nimmt die Spalte in die Suche der Autovervollständigung auf
Werteliste	-	Woher die Auswahl kommt (siehe unten)

Werteliste

Vier Wege, die einander ausschließen. Der Client wertet sie in dieser Reihenfolge aus:

`possibleValues` vor `meta.listScript` vor `meta.listItems`.

Auswahl	Was passiert
keine	freies Feld
feste Liste	Die Einträge stehen als <code>meta.listItems</code> in der Definition selbst – der Weg, den der DEX-Client für eine feste Auswahl kennt. Je Eintrag <code>key</code> (gespeicherter Wert) und <code>label</code> (angezeigter Text). Einträge ohne beides lässt der Client weg
aus einem Skript	<code>meta.listScript</code> nennt ein Portalskript; der Client ruft es über <code>dex.callScript</code> auf und erwartet eine JSON-Zeichenkette mit <code>{ value, label }</code> . Darunter stehen die <code>meta.listScriptParams</code> als Name-Wert-Liste – DOCUMENTS übergibt sie als Text
aus anderer Tabelle	Nachgeschlagen in einer Tabelle, geliefert unter <code>/table/<Definition>/col/<Spalte>/possibleValues</code>
<i>feste Liste in</i> <code>possibleValues</code>	Erscheint nur, wenn eine Definition sie schon mitbringt. Die API liefert sie, der DEX-Client zeigt sie nicht . Der Knopf <i>In feste Liste übernehmen</i> macht <code>meta.listItems</code> daraus

Für *aus anderer Tabelle*:

Feld	Bedeutung
Verbindung der Liste	leer lassen, wenn die Tabelle in derselben Datenbank liegt wie die Definition
Tabelle der Liste	Pflichtfeld – fehlt sie, endet der Aufruf mit 404
Wertespalte	die Spalte, deren Werte angeboten werden. Pflichtfeld
Anzeige	Vorlage für den angezeigten Text, <code>{{Spaltenname}}</code> wird je Zeile ersetzt
Schema / Katalog (erweitert)	MySQL: <code>schema</code> = Datenbankname, <code>catalog</code> = <code>def</code> . MS SQL: <code>schema</code> = <code>dbo</code> , <code>catalog</code> = Datenbankname
WHERE-Bedingung der Liste (erweitert)	schränkt die Liste ein
Wertespalte darf der Client wählen (erweitert)	<code>valueColumnByClient</code>

Übersetzungsfilter

`meta.translationFilter` sagt, wo die Übersetzungen der Liste stehen: **Application**, **PropType** und **PropTypeSubCategory** bilden einen festen Filter auf die Übersetzungstabelle.

Aneinandergehängt sind sie zugleich der Namensraum, unter dem der Client die geladene Liste behält – zwei Spalten mit demselben Filter teilen sie sich. Ohne Filter bleiben die Beschriftungen unübersetzt.

Der Filter gilt für alle drei Listenarten: feste Einträge, Einträge aus einem Skript und die Beschriftungen einer nachgeschlagenen Liste.

Erweitert

Feld	Eigenschaft	Bedeutung
Pflichtfeld	<code>mandatory</code>	Vorgabe entscheidet die Datenbank (NOT NULL, kein Standardwert, keine Identity)
Beim Anlegen änderbar	<code>editableOnInsert</code>	<i>nein</i> blendet das Feld schreibgeschützt ein; Pflichtspalten bleiben änderbar
Beim Ändern änderbar	<code>editableOnUpdate</code>	Primärschlüssel sind <i>nein</i> , solange hier nichts anderes steht
meta.label	<code>meta.label</code>	überschreibt die Beschriftung im DEX-Tabellenclient
Breite	<code>meta.layout</code>	Breite des Eingabefelds im Raster, 1 bis 12
Änderbar / Pflicht im Client / „ANY“ anbieten	<code>meta.editable</code> , <code>meta.mandatory</code> , <code>meta.addAny</code>	Verhalten im Client
Listenformat	<code>meta.listFormat</code>	z. B. <code>%value% (%label%)</code>
Unbekannten Wert zeigen	<code>meta.showValueNotFound</code>	zeigt einen gespeicherten Wert, den es in der Liste nicht mehr gibt
Technischen Wert zeigen	<code>meta.showTechnicalNameInList</code>	
Übersetzungsvorsatz	<code>meta.translationPrefix</code>	wird den nachgeschlagenen Schlüsseln vorangestellt; ohne Übersetzungsfiler wirkungslos
Liste immer neu laden	<code>meta.alwaysLoadList</code>	nötig, wenn zwei Spalten denselben Filter, aber verschiedene Werte haben
Weitere meta-Eigenschaften (JSON)	-	z. B. <code>listFilter</code> , <code>iconName</code> , <code>width</code> - bleiben erhalten
Weitere Eigenschaften der Spalte (JSON)	-	bleiben erhalten

Revision #1

Created 2026-09-17 11:03:54 UTC by Helge Czerwinski

Updated 2026-09-17 11:03:54 UTC by Helge Czerwinski