

Title: Wstrzykiwanie kolumn

Subject: eDokumenty - elektroniczny system obiegu dokumentów, workflow i CRM - DeployerGuide/Customization/ColumnInjection

Version: 42

Date: 07/03/24 14:15:34

## Table of Contents

<i>Wstrzykiwanie kolumn</i>	3
<i>UWAGI</i>	3

## Wstrzykiwanie kolumn

Aby dodać do listy np. klientów nowe kolumny wyświetlające informacje których standardowo tam nie ma (np. graficzny status klienta, kwotę faktur sprzedaży itp) należy plik `custom_list_fields.xml` z definicją kolumny umieścić w katalogu `apps/edokumenty/var/tpl`

Przykładowy plik jest umieszczony w katalogu `apps/edokumenty/var/tpl_default/custom_list_fields.xml_default` - można go skopiować do `tpl` jednocześnie usuwając z nazwy rozszerzenia słowo `_default`.

### UWAGI

Parametr `searchable` nie jest na chwilę obecną obsługiwany. Jego definicji powinna wyglądać jako:

```
<searchable>1</searchable>
```

Patrz na wykaz list, które obsługują wyszukiwanie (na dół).

### Wstrzykiwanie kolumn do listy klientów

Edycja pliku `custom_list_fields.xml`.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <ContactsList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 1 AND tblnam = 'contacts' AND tbl_id = contid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </ContactsList>
</data>
```

### Wstrzykiwanie kolumn do listy uczestników kampanii

Edycja pliku `custom_list_fields.xml`.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <CampaignParticipantsList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 1 AND tblnam = 'contacts' AND tbl_id = contid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </CampaignParticipantsList>
</data>
```

### Wstrzykiwanie kolumn do listy dokumentów

Postępujemy podobnie jak w przypadku listy klientów. Klucz główny na liście to `doc_id`. Dostępne listy to:

Nazwa listy (tagu XML)	Lista	Uwagi
------------------------	-------	-------

DocumentsList	lista dokumentów w module Dokumenty	Alias na kolumny z głównego zapytania to doc czyli np. wstawiając kolumnę z vatnote robimy SELECT v.doc_id FROM vatnote AS v WHERE v.doc_id = doc.doc_id
ProcessDocumentsList	lista dokumentów na kartotece sprawy	Alias na kolumny z głównego zapytania to doc czyli np. wstawiając kolumnę z vatnote robimy SELECT v.doc_id FROM vatnote AS v WHERE v.doc_id = doc.doc_id
ClientDocumentsList	lista dokumentów na kartotece kontrahenta	Alias na kolumny z głównego zapytania to doc czyli np. wstawiając kolumnę z vatnote robimy SELECT v.doc_id FROM vatnote AS v WHERE v.doc_id = doc.doc_id

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <DocumentsList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 1 AND tblnam = 'depository' AND tbl_id = depoid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
    </cecha_1>
  </DocumentsList>
</data>
```

Od wersji 5.0.162 dostępna jest rozszerzona konfiguracja. ID kolumny (w przykładzie poniżej kto\_dodal2) nie może mieć mniej niż 7 znaków (aby uniknąć konfliktów z wbudowanymi kolumnami).

```
<DocumentsList>
  <join_definition id="aqq1">
    <sql>select usrn as kto_dodal, usr_id as aqq1_usr_id from users</sql>
    <join_statement>doc.adduid = aqq1_usr_id</join_statement>
  </join_definition>

  <kto_dodal join_definition="aqq1">
    <name>Kto dodał</name>
    <css>width:100px;</css>
    <tooltip/>
    <hidden />
  </kto_dodal>
</DocumentsList>
```

### Wstrzykiwanie kolumn do listy produktów

Postępujemy podobnie jak w przypadku listy klientów.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <ProductsList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 1 AND tblnam = 'depository' AND tbl_id = depoid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
    </cecha_1>
  </ProductsList>
</data>
```

```

        <searchable />
    </cecha_1>
</ProductsList>
</data>

```

\*Od wersji 5.0.241, 5.2.31 obsługuje parametr searchable.

Lista prosta (występująca przy wyszukiwaniu np. dodaj zbiorczo) produktów

```

<?xml version="1.0" encoding="UTF-8"?>
<data>
  <SimpleProductsList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 1 AND tblnam = 'depository' AND tbl_id = depoid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </SimpleProductsList>
</data>

```

#### Wstrzykiwanie kolumn do listy urządzeń

Postępujemy podobnie jak w przypadku listy klientów.

Przykład:

```

<?xml version="1.0" encoding="UTF-8"?>
<data>
  <DevicesList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 1 AND tblnam = 'depository' AND tbl_id = depoid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </DevicesList>
</data>

```

#### Wstrzykiwanie kolumn do listy spraw

Edycja pliku custom\_list\_fields.xml.

Przykład:

```

<?xml version="1.0" encoding="UTF-8"?>
<data>
  <ProcessesList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT ftopnm FROM features_opt_view WHERE featid = 17 AND tblnam = 'processes' AND tbl_id = prc_id</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </ProcessesList>

```

```
</data>
```

### Wstrzykiwanie kolumn do listy osób kontaktowych

Edycja pliku custom\_list\_fields.xml.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <ContactPersonsList>
    <cecha_1>
      <name>Cecha 1</name>
      <sql>SELECT copeid||'-'||bcrdid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </ContactPersonsList>
</data>
```

- zapytanie korzysta z widoku business\_card\_view toteż nasze podzapytanie może korzystać z wszystkich kolumn z tego widoku

### Wstrzykiwanie kolumn do listy spraw windykacyjnych

Edycja pliku custom\_list\_fields.xml.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <VindicationList>
    <cecha_2>
      <name>Cecha 2</name>
      <sql>SELECT 666</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_2>
  </VindicationList>
</data>
```

### Wstrzykiwanie kolumn do listy kosztów podróży w dokumencie delegacji

Edycja pliku custom\_list\_fields.xml.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <DelegationTravelCostsList>
    <cecha_2>
      <name>Cecha 1</name>
      <sql>SELECT 1</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_2>
  </DelegationTravelCostsList>
</data>
```

```
</DelegationTravelCostsList>
</data>
```

**Wstrzykiwanie kolumn do listy kosztów na fakturze** Kontekst wykonania listy to widok vatnote\_costs\_view

Edycja pliku custom\_list\_fields.xml.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <VatCostsList>
    <cecha_2>
      <name>Cecha kosztu</name>
      <sql>SELECT data__ FROM
        vatnote_costs vc
        INNER JOIN features_text_view ftv ON ftv.tbl_id = vc.vtcoid AND ftv.feaid = 12
        WHERE vc.vtcoid = vatnote_costs_view.vtcoid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_2>
  </VatCostsList>
</data>
```

**W przypadku listy produktów na formularzach dokumentów lub w sprawie widokiem na którym bazuje wyświetlana lista to fk\_elements\_view. Wszystkie przykłady dotyczące listy produktów wyświetlają dodatkową kolumnę Opis produktu pobraną z cennika (depository).**

Dostępne listy to:

Nazwa listy (tagu XML)	Lista	Uwagi
FKElementsProcessList	lista produktów w sprawie	możliwość zdefiniowania wielu list w zależności od atrybutów dos_id lub dsexid. Dla przykładu - aby wyświetlić dodatkowe kolumny w liście pozycji w sprawie tylko dla określonej teczki (dsexid) definicja taga powinna wyglądać następująco <FKElementsProcessList dsexid="3". W przypadku atrybutu dos_id należy podmienić w przykładzie dsexid na dos_id i podać odpowiednią wartość. Obecnie obsługiwane są tylko te 2 atrybuty.
FKElementsVatNoteList	lista produktów na fakturze	
FKElementsWZList	lista produktów na wydaniu zewnętrznym	
FKElementsPZList	lista produktów na przyjęciu zewnętrznym	
FKElementsOrderList	lista produktów na zamówieniu (ZW oraz ZO)	
FKElementsOfferList	lista produktów na ofercie	
FKElementsCustomDocumentList	lista produktów na dokumencie customowych (dotyczy wszystkich)	
FKElementsDemandList	lista produktów na zapotrzebowaniu	
FKElementsOrderCompleteList	lista produktów na kompletacji w fakturze	Obsługuje parametr searchable

Przykład: **Wstrzykiwanie kolumn do listy produktów w sprawie**

Edycja pliku custom\_list\_fields.xml.

Przykład:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <FKElementsProcessList>
    <cecha_1>
      <name>Opis produktu</name>
      <sql>SELECT dscrpt FROM depository WHERE depoid = fk_elements_view.depoid</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </cecha_1>
  </FKElementsProcessList>
</data>
```

Dla pozostałych typów list produktów należy zmienić nazwę tagu FKElementsProcessList na wybraną listę.

### Wstrzykiwanie kolumn do listy w dziennikach

Obsługiwane typy dzienników w których można wstrzyknąć dodatkowe kolumny to:

Nazwa listy (tagu XML)	Lista	Uwagi	Obsługuje wyszukiwanie
ARegOfPapersList	Dziennik korespondencji	Nazwa tabeli w bazie to regofpapers (kontekst: ppr)	Tak
ARegOfVatNotesList	Dziennik faktur	Nazwa tabeli w bazie to regofvatnotes	Tak
ProcessesList	Moduł Sprawy	Kontekst: processes_view	Tak

### Wstrzykiwanie kolumn do przychodów na fakturze sprzedaży

Nazwa kodowa listy: VatNoteProfitList

Bezpośrednio dostępne tabele:

Tabela	Nazwa	Alias
vatnote_profits	Pozycje przychodu dla faktury przychodzącej	vp
types_of_incomes	Rodzaje przychodu (słownik)	toi
users	Użytkownicy (dane użytkownika dodającego wpis)	u

Przykład:

wyświetlenie kolumny z dostępnej tabeli - add\_date (data dodania pozycji)

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <VatNoteProfitList>
    <add_date>
      <name>Data dod.</name>
      <sql>vp.adddat::date</sql>
      <css>width:100px;</css>
      <tooltip />
      <hidden />
      <searchable />
    </add_date>
  </VatNoteProfitList>
</data>
```



```
</VatNoteProfitList>  
</data>
```

pobranie danych z innej tabeli - lm\_user (nazwa użytkownika ostatnio zmieniającego pozycję)

```
<?xml version="1.0" encoding="UTF-8"?>  
<data>  
  <VatNoteProfitList>  
    <lm_user join_definition="lmu">  
      <name>Zmienił</name>  
      <css>width:100px;</css>  
      <tooltip />  
      <hidden />  
      <searchable />  
    </lm_user>  
    <join_definition id="lmu">  
      <sql>select usrn as lm_user, usr_id as lmu_usr_id from users</sql>  
      <join_statement>vp.lm_uid = lmu_usr_id</join_statement>  
    </join_definition>  
  </VatNoteProfitList>  
</data>
```