

Конфигуратор: Фильтры

В этом разделе рассказывается как настраивать фильтры. Раздел предназначен только для специалистов со знанием основ языка SQL, а так же структуры базы данных НЭП.

К картинок здесь не будет...

Ссылка: [Описание работы с Фильтрами для пользователей](#)

Определения:

- Грид - табличная часть справочников или журналов, для которых настраивается фильтр;
- Результирующий запрос (или просто запрос) - SQL запрос для вывода грида;
- Результат запроса - то что выводится в грид;
- Отфильтрованный результат запроса - результат запроса с применением фильтра;
- Фильтровать, значит - пропускать. Т.е. отфильтровать запись, значит вывести ее в грид;
- Наименование поля = Наименование колонки;
- Поле фильтра = Колонка фильтра;
- Фильтр всегда привязывается к конкретной колонке грида, и включается галкой в выпадающем меню фильтров, которое появляется при нажатии на заголовок колонки (значок активации фильтра);

Помимо системных фильтров, которые создаются автоматически для основных полей возможно настроить пользовательские фильтры.

Доступны три типа фильтров:

Фильтр по значению

Самый простой вариант фильтра, для которого указывается только значение фильтра, по которому будут отфильтровываться записи.

Создадим новый «Фильтр по значению». Заполним следующие свойства фильтра:

- «Наименование фильтра» - с этим именем фильтр будет отображаться в выпадающих списках фильтров колонок;
- «Поле для отображения фильтра» - указывается наименование колонки, к которой будет привязан фильтр. Привязан - значит то, что фильтр теперь появиться в выпадающем списке фильтров колонки при нажатии на ее заголовок (значок активации фильтра). Причем фильтр появиться в меню у всех одноименных колонок во всех гридах.

Если для таких колонок ранее не было определено ни одного фильтра, то после привязки нового фильтра у соответствующих одноименных колонок появится кнопка, которая вызывает меню с фильтрами.

ВАЖНО! Фильтры появятся у пользователей только после перезагрузки Nep.exe

- «Поле для проверки условий» - указывается наименование поля, по значениям которого

будет вестись выборка. Может быть отличным от «Поля фильтра». Если данный параметр оставить пустым, то в качестве «Поля для проверки условий», будет взято «Поле для отображения фильтра»;

- Значение - сюда вводится значение фильтра. Можно указать несколько значений через запятую

При активации фильтра такого типа с одним значением к запросу в секцию where будет добавлено следующее условие:

```
where FIELD_NAME=FILTER_VALUE
```

где FIELD_NAME - Поле для проверки условий,
а FILTER_VALUE - Значение фильтра

При активации фильтра такого типа с несколькими значениями через запятую к запросу в секцию where будет добавлено следующее условие:

```
where FIELD_NAME in (FILTER_VALUE)
```

где FIELD_NAME - Поле для проверки условий,
а FILTER_VALUE - Значение фильтра (несколько через запятую)

Если условие where уже присутствует в запросе, то в этом случае будет добавлен оператор and

Например, для отбора Своих фирм по ID в качестве «Поле для отображения фильтра» можно указать FIRM_ALIAS (поле «Краткое наименование»), а в качестве «Поле для проверки условий» нужно указать FIRM_ID.

В окно со Значениями в этом случае нужно вбить ID Своей фирмы по которой должна вестись выборка для данного фильтра. Возможно указать несколько значений через запятую, в следующе виде: 20,100,12

Если необходимо фильтровать по текстовому наименованию Своей фирмы можно в качестве «Поля для проверки условий» указать FIRM_NAME (или тот же FIRM_ALIAS)

Рассмотрим примеры, где «Поле для отображения фильтра» заполнено, как FIRM_ALIAS, а «Поле для проверки условий» заполнено как FIRM_NAME

Для текстовых полей в окне со Значениями можно указать теги , которые позволяют вместо строгого списка задать условия:

```
<:Not> - тег для негативного отбора;  
<:Like> - тег для отбора "похожих" по тексту значений;  
Например, если в окне "Значение" написать <:Like>"ИП%",  
то будут отбираться Свои фирмы, наименование которых  
начинается с фразы "ИП";  
<:Containing> - тег для отбора строк, включающих  
необходимую фразу. Например, если в окне "Значение"  
написать <:Containing>"лтд" - будут отбираться Свои фирмы,  
в наименовании которых содержится текст "лтд";
```

`<:is distinct from>` - тег для условного отбора по значениям, отличающихся от указанного. Условие отбора в этом случае похоже на условный оператор `<>`, с тем отличием, что сравнение можно делать с `null`.
Например, если в окне "Значение" написать
`<:is distinct from null>` - будут отбираться Свои фирмы наименования, которых не пустые (`null`).
Если указать дополнительно тег негативного отбора в виде `<:not><:is distinct from>null`, то будут наоборот отбираться Свои фирмы с пустым полем наименования

Фильтр с SQL условием

Для выборки используется произвольное SQL условие, которое программируется на оригинальном языке SQL Firebird, что позволяет существенно расширить возможности фильтрации в отличие от «Фильтров по значению».

Возможно использовать вложенные подзапросы и встроенные функции языка SQL, SQL условие при включении фильтра подключается в секцию `where` основного запроса.

Для этого типа фильтра, так же необходимо заполнить «Поле для отображения фильтра» и заполнить «Текст SQL условия»

- «Текст SQL условия» - заполняется условием на языке SQL

Рассмотрим это условие подробнее:

Например, предыдущие фильтры можно усовершенствовать с помощью SQL условия

```
Upper(<:Alias>.FIRM_NAME) containing "ЛТД"
```

данное условие в отличие от фильтра «по Значению» решает проблему чувствительности к регистру. Условия проверяются с заглавными буквами.

Важно всегда перед полем по которому проверяется условия указывать тег:

```
<:Alias>
```

- он нужен для ссылки на основной SQL запрос в котором это поле должно быть доступно.

Еще примеры SQL условий:

Пусть так же фильтры будут в выпадающем списке у поля `FIRM_NAME`:

```
(select Count(*) from FIRMS where FIRM_NAME=  
<:Alias>.FIRM_NAME)>1
```

- основной запрос будет отфильтровывать Свои фирмы, у которых дублируется наименование;

```
<:Alias>.FIRM_ID in (select distinct FIRM_ID from
```

```
ChecksView where DOC_ID=0  
and KIND=0 and TAKEN<>"V")
```

- основной запрос буде отфильтровывать Свои фирмы, на которых весят нереализованные Документы резерва;

Список фильтров из SQL запроса

С помощью SQL запроса формируется список фильтров, на подобии системных фильтров.

Результатом такого SQL запроса будет набор фильтров, который будет отображаться в выпадающем списке фильтров у колонок, которые можно пометить аналогично системным фильтрам.

SQL запрос необходимо написать с учетом определенных правил. А именно, SQL запрос должен возвращать заполненными свойства, которые были необходимы для определения фильтра «По значению»:

- «Наименованием фильтра», которое будет отображаться в выпадающем меню фильтров;
- «Значение фильтра» - по которым непосредственно будет осуществляться отбор.

Соответственно SQL запрос должен возвращать как минимум эти два поля.

После того, как запрос списка фильтров сформирован, нужно определить правило по которому будет осуществляется выборка по «Значениям фильтра».

Тут два варианта: Первый - простой, выборка будет осуществляться по простой схеме с «по значению»

```
FILED_NAME in (A,B,C)
```

где A,B,C - «Значения фильтров» через запятую (например ID) , соответствующие включенным в выпадающем меню фильтрам.

Пример:

```
for  
select t.TARGET_ID, t.TARGET_ID, t.TARGET_NAME  
from CLASSES_TARGETS t  
where t.CLASS_ID=122  
Order by t.TARGET_NAME  
into :FILTER_ID, :FILTER_VALUE, :FILTER_NAME
```

- выводит список фильтров в качестве которых выступают Страны из таблицы Атрибутов. В качестве «Поле для отображения фильтра» укажем COUNTRY.

Соответственно этот список фильтров будет доступен у всех таблиц, где есть поле COUNTRY (Страна). Например «Партии» (PARTS) и «Складские документы» (CHECKS)

Пример фильтра по секциям прайс-листа

Поле фильтра: PRICELIST_NAME

SQL запрос списка:

```
select t.TARGET_ID, t.TARGET_NAME  
from CLASSES_TARGETS t  
where t.CLASS_ID=109  
Order by t.TARGET_NAME
```

SQL условие:

```
<:ALIAS>.PRICELIST_ID=<:FILTER_VALUE>
```

ВАЖНО!

- Фильтры разного типа (По значениям, SQL условие и текстовый фильтр) добавляют условия к результирующему SQL запросу через AND
- Внутри одного типа через условия добавляются через OR

From:

<http://docs.comsystems.ru/> - **Руководство пользователя НЭП**

Permanent link:

http://docs.comsystems.ru/docs/settings/conf_filters

Last update: **2023-06-27 00:25**

