

Коллеги, доброе время суток.

Вот и подошло время для очередного обновления плагина к NotePad++

Ключевые изменения:

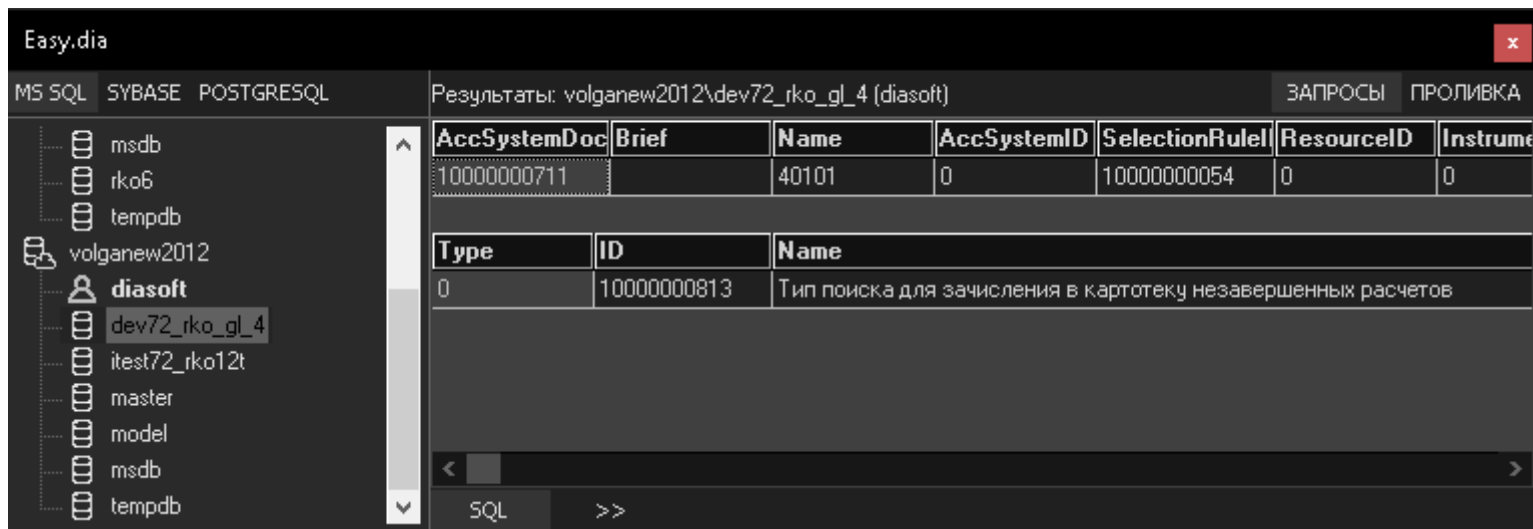
- Поддержка работы плагина на последней версии NotePad++ v8.4.6 (64-bit)
- Появилась возможность подключения к любому серверу по имени/порту
- Поддержка подключений к серверам PostgreSQL
  - Возможность запуска и остановки записи RTI, выгрузка записанных серверных RTI в файл
  - Работают\* команды `sp_help`, `sp_helpindex`, `sp_helptext` (\* в качестве команд плагина)
  - Возможность конвертации “пролитого” t01 в синтаксис PostgreSQL
  - Возможность посмотреть текст плана запроса PostgreSQL
- Появилась возможность “запоминать” подключения к БД и использовать их позднее. Теперь подключиться можно в два клика мышкой 😊
- Поддержка темной темы NotePad++. Работать с темной темой считается более комфортно, глазки меньше устают. Реализованы две цветковые схемы подсветки синтаксиса – в стиле SQL Notepad++, и в стиле Visual Source Code
- Интерфейс минимизирован. (минимализм – наше все)



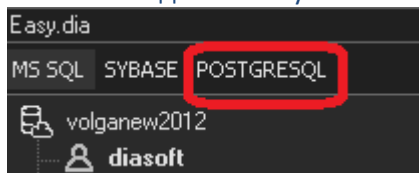
Далее более подробно:

### **1. Сначала общие изменения.**

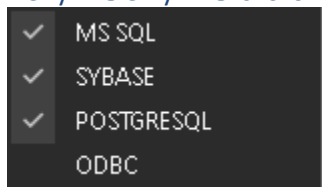
Окно плагина изменилась мало, немного поменялись кнопки, режим добавления вкладки в окне SQL, вот собственно и все.



1.1 Появился отдельный пункт меню – POSTGRESQL

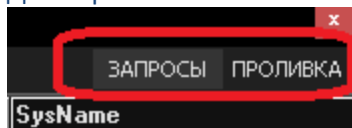


1.2 Ненужные пункты этого меню можно скрыть с помощью выпадающего списка:

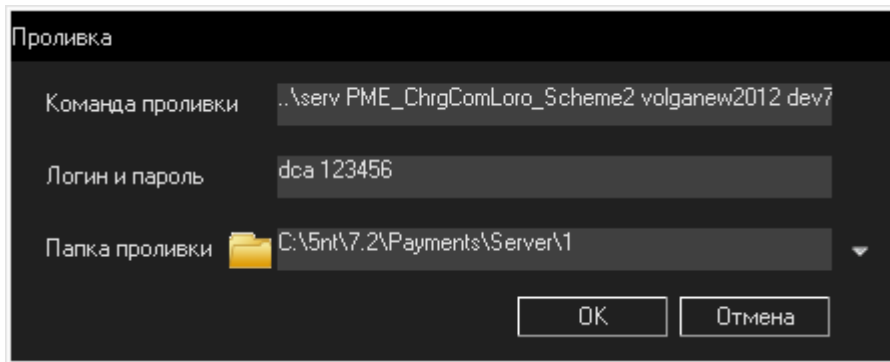


1.3 “Проливка” скриптов теперь производится **в том же окне**, где и выполняются запросы SQL, никаких дополнительных вкладок нет

Для переключения из окна результатов проливки в окно с результатами запросов SQL справа имеются две кнопки:



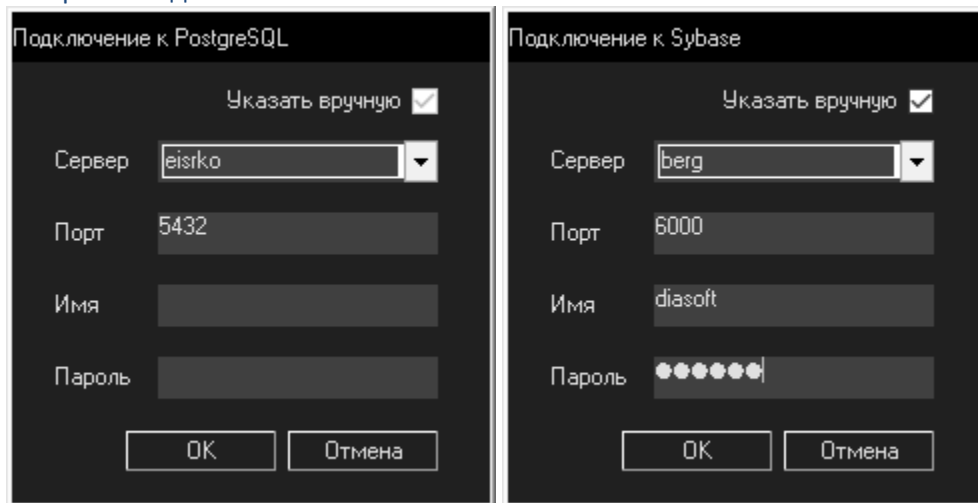
1.4 Окно проливки теперь выглядит так:



Общий смысл остался тот же – формируется консольная команда, которая проливает скрипт, создавая результирующий файл t01. Поскольку и проливка, и исполнение SQL запросов теперь, по сути – один интерфейс, то пролить скрипт можно только на тот сервер и базу, которая в настоящий момент выбрана. Проливка на Postgres напрямую не поддерживается (но ничто не мешает нам сконвертировать скрипт в синтаксис PostgreSQL и затем его выполнить на базе PostgreSQL)

## 2. Окно подключения

теперь выглядит так:



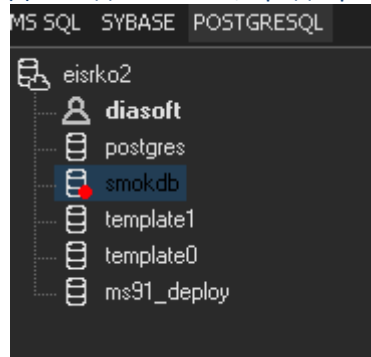
С помощью крыжика “Указать вручную” можно отказаться от предлагаемого списка и задать подключение самостоятельно.

Подключение сохраняется для последующего выбора в этом окне.

### 3. Работа с PostgreSQL

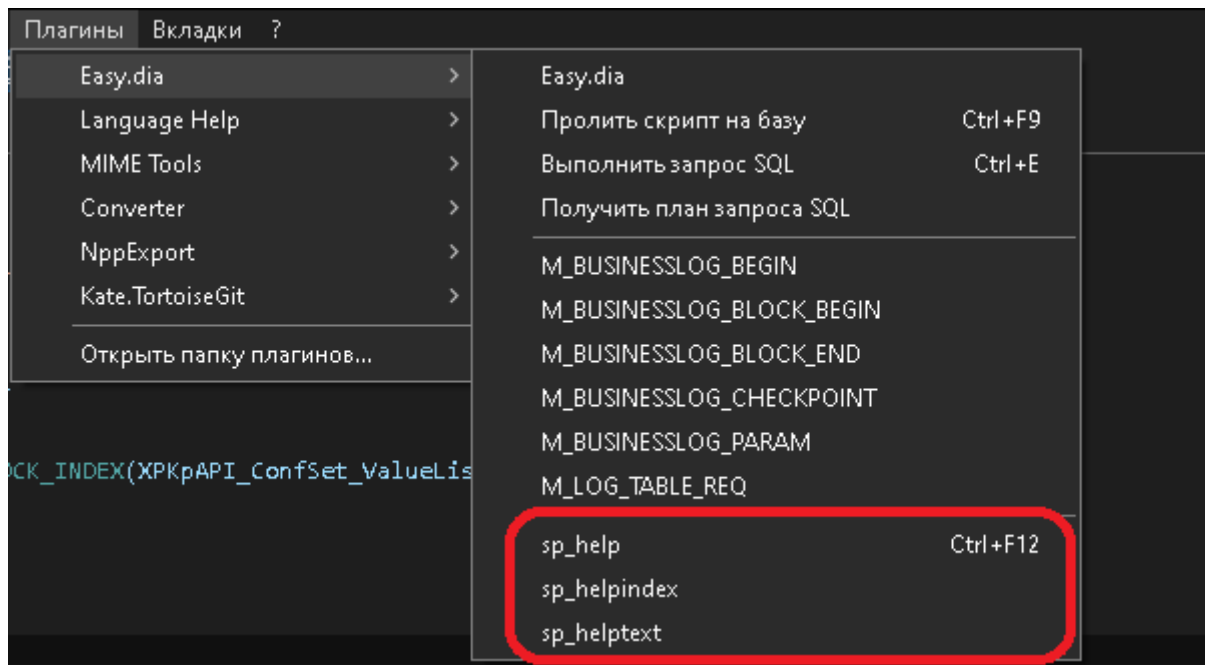
в контекстном меню ничем не отличается от работы с другими серверами данных.

Для подключения, предварительно требуется установить на компьютере драйвер ODBC «PostgreSQL Unicode». Без него каменный цветок не выйдет.



С помощью контекстного меню можно запускать серверное логирование для данного пользователя, останавливать его и выгружать в файл с расширением \*.rti

3.1 Также, как и для других видов подключения, для PostgreSQL работают команды плагина (они же есть и в контекстном меню):



Надо понимать, что эти команды напрямую не поддерживаются в PostgreSQL, они эмулируются.

Например, `sp_helptext` подменяется на:

```
SELECT
  pg_get_functiondef((
    SELECT
      oid FROM pg_proc
    WHERE
      proname = '%s' ));
```

Соответственно, результат будет выдан одной строкой, которую можно скопировать в буфер обмена и вставить в новое окно NotePad++

Выделив текст в редакторе, например `dealtransact_insert` и выполнив команду плагина `sp_helptext`, мы получим вот такой результат (не забывайте про регистр наименований!):

Результаты: eiskol\smokdb (diasoft)

**pg\_get\_functiondef**

```
CREATE OR REPLACE PROCEDURE public.dealtransact_insert(
  INOUT var_implicitretval integer,
  IN var_implicitresultset refcursor,
  INOUT var_aggregate_param jsonb)
```

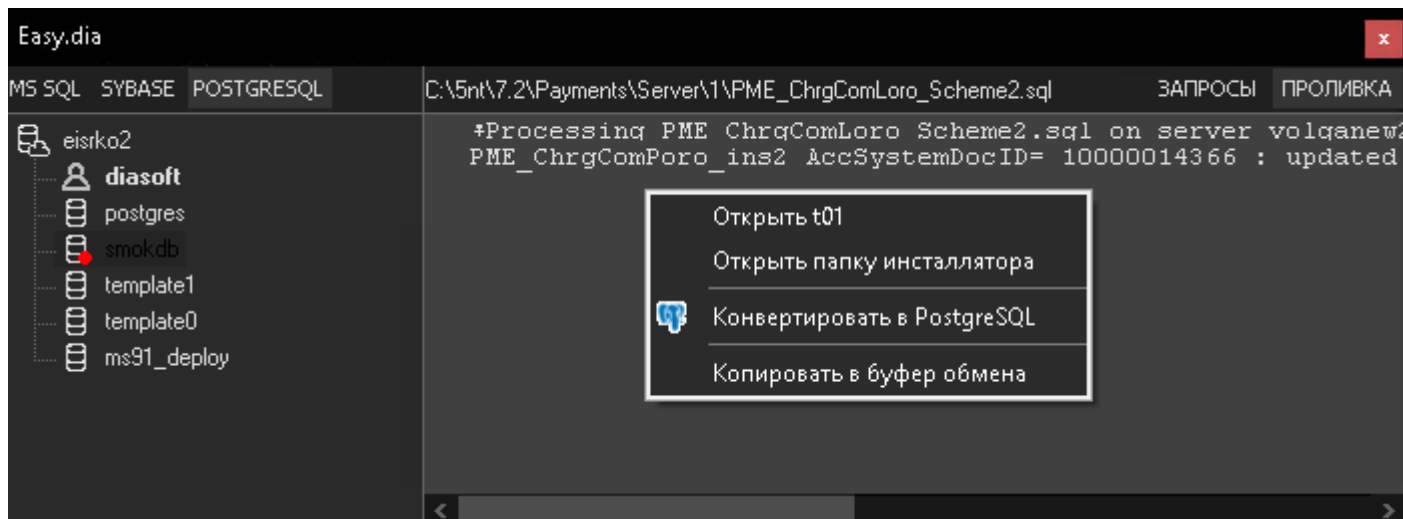
Копируем эту строку в буфер обмена, вставляем в новое окно редактора и получаем нормальный текст процедуры:

```



1 CREATE OR REPLACE PROCEDURE public.dealtransact_insert(INOUT var_implicitretval integer, IN var_implicitresultset refcursor, INOUT var_aggreg
2 LANGUAGE plpgsql
3 AS $procedure$
4 DECLARE
5     var__UserID DSIDENTIFIER;
6     var__ProcessID DSIDENTIFIER;
7     var__Number DSIDENTIFIER;
8     var__UserAccess DSINT_KEY;
9     var__TraceProc DSINT_KEY;
10    var__TraceProcParam DSINT_KEY;
11    var__TraceTrace DSINT_KEY;
12    var__TraceDebug DSINT_KEY;
13    var__TraceProFile DSINT_KEY;
14    var__TraceInfo DSINT_KEY;
15    var__TraceError DSINT_KEY;
16    var__TraceTable DSINT_KEY;
17    var__TraceGlobal DSINT_KEY;
18    var__TraceBusiness DSINT_KEY;
19    var__TraceObjectListID DSCOMMENT300;
20    var__TraceVars varchar (2000);
21    var__TraceOneVar varchar (2000);
22    var__TraceOneValue1 varchar (2000);
23    var__TraceOneValue2 varchar (2000);
24    var__TraceOneValue3 varchar (2000);
25    var__TraceProcName varchar (32);
26    var__TraceRetVal int;
27    var__TraceStartDT timestamp (3);
28    var__TraceHostName DSCOMMENT;
29    var__TraceUserBrief DSUSERNAME;
30    var__CheckContextInfo2 DSTINYINT;
31    var__ImplicitCallerRetVal int;
32    var__Param_DSINT_KEY_ActivateClientFlag DSINT_KEY;

```

3.2 Конвертация «пролитого» скрипта, а точнее, его развернутый в t01 вариант, производится через контекстное меню в окне результатов проливки:



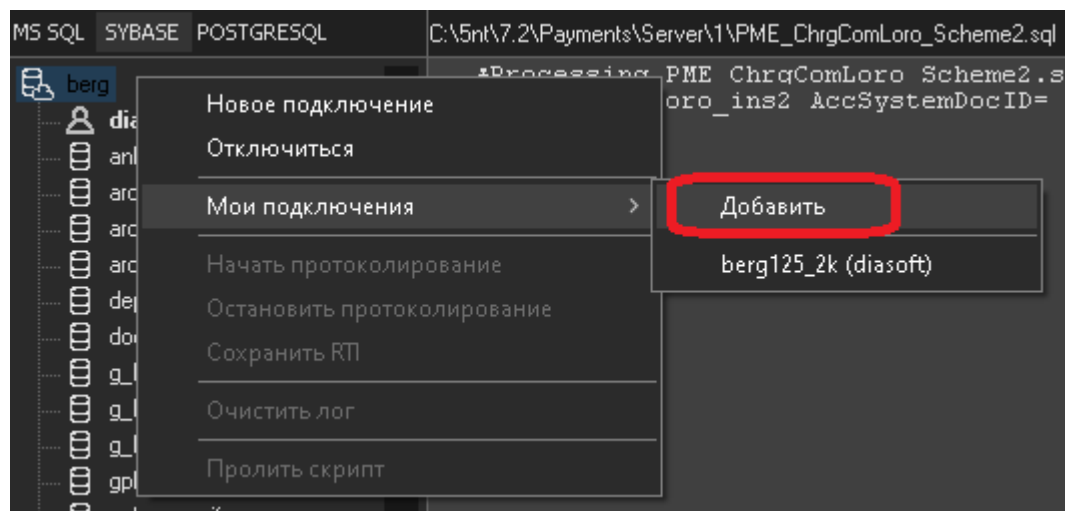
Конвертация сейчас производится с помощью сервиса «<http://postgrefagl:8089/v1/batch>», с поддержкой преобразования кириллицы. Строка подключения к сервису хранится в реестре, если что – можно будет поменять. Результат конвертации открывается в новом окне редактора.

3.3 Текст плана запроса в PostgreSQL можно получить как и раньше – выделив в редакторе запрос и нажав кнопку панели инструментов  (для темной темы),  (для светлой)

QUERY PLAN
Result (cost=6.43..6.44 rows=1 width=64)
InitPlan 1 (returns \$0)
-> Index Scan using pg_proc_proname_args_nsp_index on pg_proc (cost=0.41..6.43 rows=1 width=4)
Index Cond: (proname = 'accsystemdoc_search':name)

#### 4. Появилась возможность запоминать текущее подключение в «избранные».

Тут все просто:



Таким же способом подключение можно удалить из сохраненных.

## 5 Поддержка темной темы

В меню редактора теперь три темы:

- **Diasoft** – светлая тема
- **Diasoft Dark Mode** – темная тема в соответствии со стандартной темой SQL самого Notepad++
- **Diasoft Dark Mode 2** – темная тема в стиле Visual Source Code

Никто не мешает вам завести свои излюбленные цветовые сочетания – в Notepad++ это делается просто.

Скриншотов приводить много не буду, выглядит как то так:

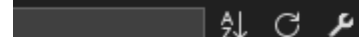


C:\FA-Payments-EXT\Payments\_Ext\SERVER\ChrgComLoro\PME\_ChrgComLoro\_Term.sql - Notepad++ [Administrator]

Файл Правка Поиск Вид Кодировки Синтаксисы Опции Инструменты Макросы Запуск Плагины Вкладки ?



Список функций



▼ PME\_ChrgComLoro\_Term.sql

- define M\_NR\_SYSNAME
- define M\_NR\_BRIEF
- define M\_NR\_NAME
- define M\_NS\_SYSNAME
- define M\_NS\_BRIEF
- define M\_NS\_NAME
- define M\_GET\_CONFMID
- define M\_INSERT\_DOCUMENTS
- define M\_NOERRORS
- PME\_ChrgComLoro\_Term

```
129 Возвращаемые значения:
130 0 - Ошибка нет
131 -----
132 __BEGIN_PROCEDURE__(PME_ChrgComLoro_Term)
133 declare @DateB DSOPERDAY ,
134 @DateE DSOPERDAY ,
135 @DealTransactID DSIDENTIFIER ,
136 @Incflag DSTINYINT ,
137 @NR_DocTypeID DSIDENTIFIER, --Тип поиска для зачисления в картотеку незавершенных расч
138 @NS_DocTypeID DSIDENTIFIER, --Тип поиска для зачисления в картотеку невыясненных сумм
139 @Purpose DSCOMMENT ,
140 @DealProtocolID DSIDENTIFIER,
141 @CompleteID DSIDENTIFIER, -- Идентификатор документа списания
142 @CompleteCommID DSIDENTIFIER, -- Идентификатор документа списания (комиссия)
143 @NotificationID DSIDENTIFIER,
144 @NTFMessage DSCOMMENT ,
145 @Hit DSTINYINT ,
146 @Rc DSINT_KEY
147
148 M_BUSINESSLOG_BEGIN
149 M_BUSINESSLOG_BLOCK_BEGIN('PME_ChrgComLoro_Term')
150
151 select @DateB = convert(smалldatetime,convert(varchar(8), @SDate, 112), 112)
152 select @DateE = convert(smалldatetime,convert(varchar(8), @EDate, 112), 112),
153 @ByMark = isnull(@ByMark,0)
154
155
156 -----
157 if @ByMark = 1
158 if not exists(select 1
159 from tDocMark dm M_NOLOCK_INDEX(XPKtDocMark)
160 where dm.spid = @@spid
161 and dm.Type = MARKTYPE_DOCINT
162 )
163 begin
164 select @NTFMessage = 'Не выделено ни одного документа.'
165
166 insert pRKOErrorLine M_P_WITH_ROWLOCK (SPID, Number, Tag, TextLine)
167 values (@@spid, 1, '', @NTFMessage)
```

Спасибо что дочитали до конца. Удачи!  
**Буду рад вашим «спасибо на INSPIDER»**