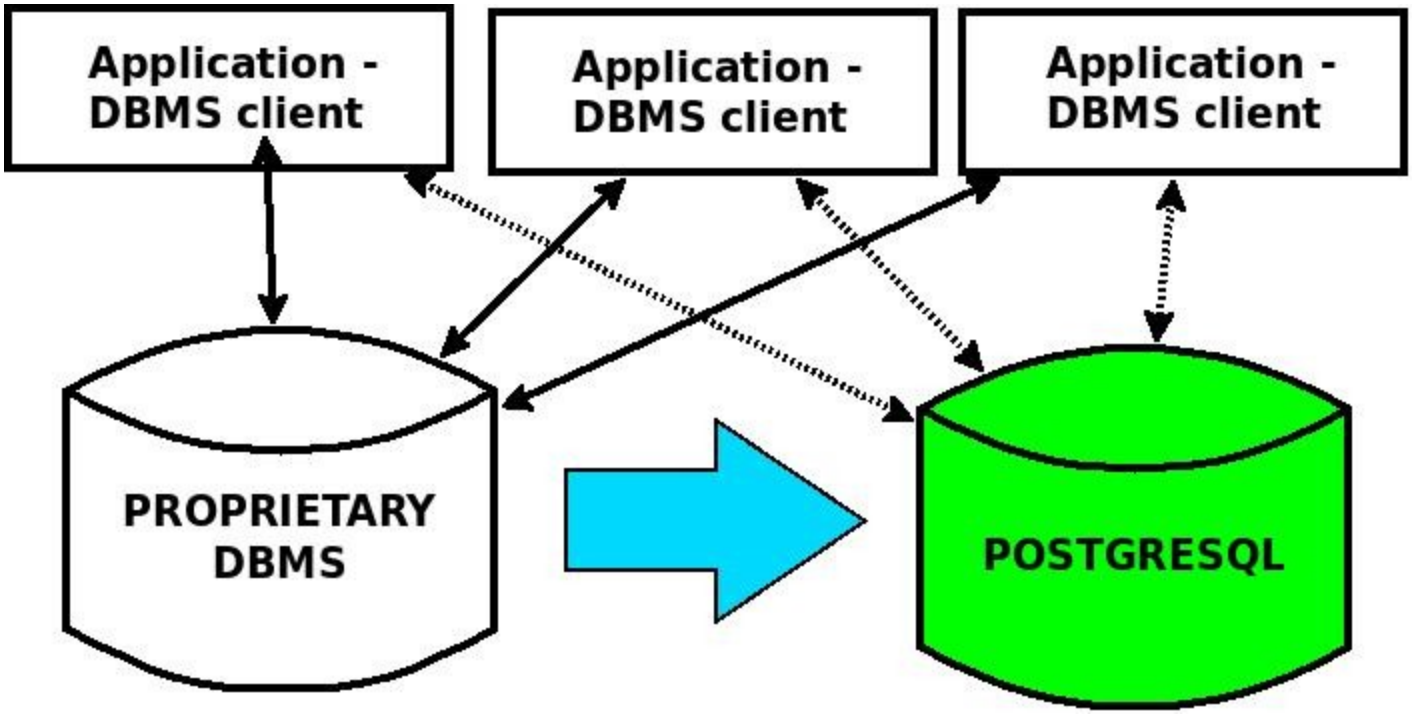
Диасофт
Платформа

Автоматизированная миграция приложений с проприетарных СУБД на PostgreSQL

Верюгин Борис Викторович,
руководитель управления развития платформы «**Diasoft Framework**»
boris.veryugin@diasoft-platform.ru

ЦЕЛИ И ЗАДАЧИ МИГРАЦИИ

- Цели:
  - Импортозамещение
  - Снижение стоимости владения
  - Повышение гибкости системы
- Задачи:
  - Адаптировать существующие системы на PostgreSQL в кратчайшие сроки
  - Разработать универсальную технологию миграции приложений на PostgreSQL и другие СУБД



ОБЗОР СУЩЕСТВУЮЩИХ ТЕХНОЛОГИЙ

Миграция Oracle → PostgreSQL:

- приложение «**ora2pg**»
  - не конвертирует хранимую логику (пакеты, функции, процедуры)
- расширение «**orafce**»
  - нет поддержки (или неполная поддержка) многих нативных пакетов Oracle, например:
    - DBMS_SQL
    - DBMS_LOB
    - DBMS_LOCK
    - DBMS_TYPES
    - …

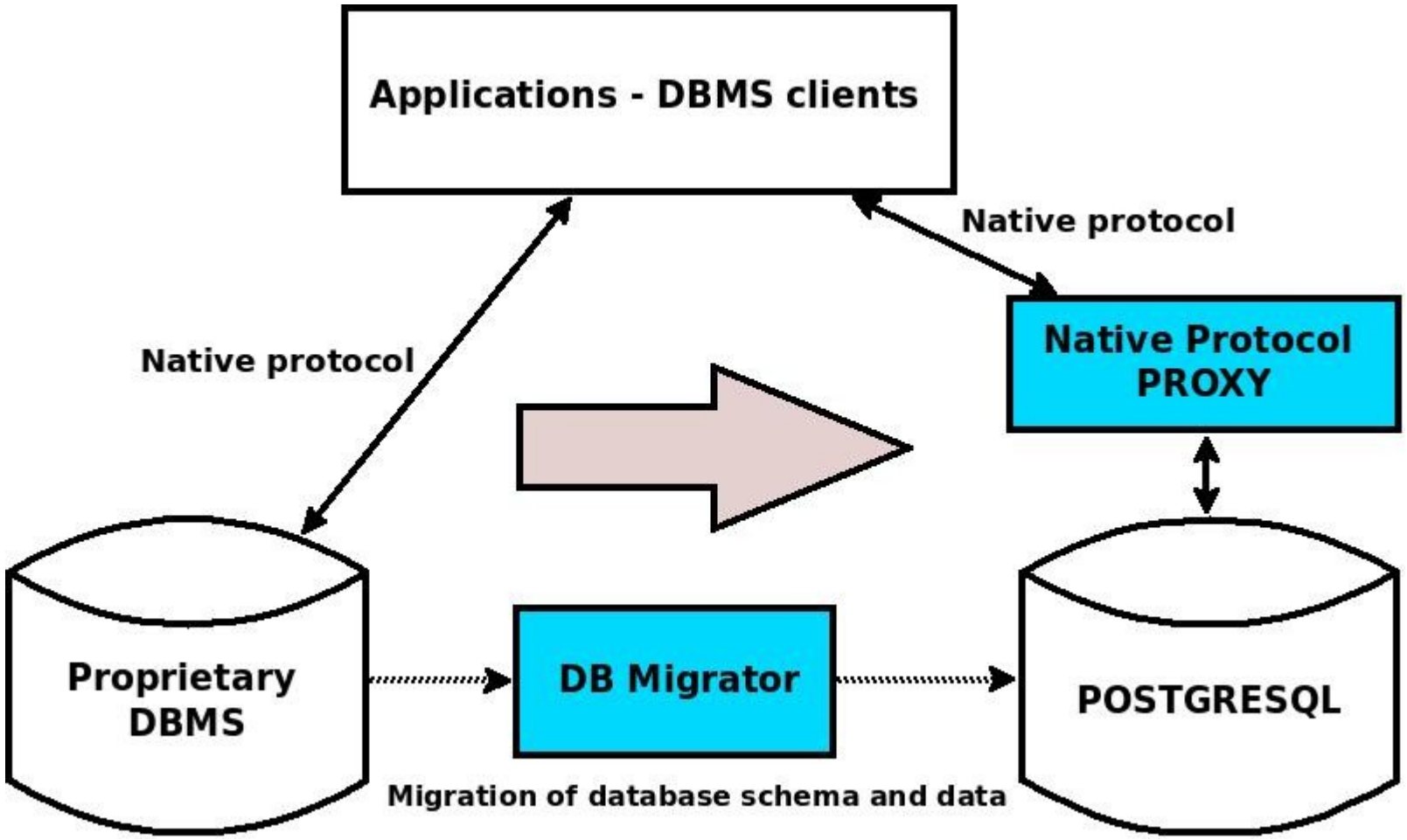
Миграция Microsoft SQL Server → PostgreSQL:

- приложение «**sqlserver2pgsql**»
  - не конвертирует хранимую логику (пакеты, функции, процедуры)

Не найдено инструментов, позволяющих адаптировать существующие клиентские приложения без изменения их кода.

DIASOFT DATABASE ADAPTER

Общая схема миграции



DIASOFT DATABASE ADAPTER

Diasoft Database Adapter - продукт, позволяющий адаптировать приложения, созданные в расчете на использование СУБД Oracle или Microsoft SQL Server, на СУБД PostgreSQL (в следующих версиях будут поддержаны и другие СУБД) без изменения их исходного кода.

Компоненты продукта

DB Migrator

- Обеспечивает миграцию баз данных (схем и самих данных)
- Конвертирует логику, написанную на PL/SQL (Oracle) или Transact SQL (Microsoft SQL Server) в PL/PgSQL.

Native Protocol Proxy

- Работает по нативному протоколу с клиентским приложением;
- Конвертирует поступающие на вход запросы в грамматику PostgreSQL;
- Позволяет переключить существующие приложения на PostgreSQL без их изменения.

DSQLProxy

- Написанная на C функция для PostgreSQL, предназначенная для конвертации динамического SQL посредством обращения к экземпляру Native Protocol Proxy.

DIASOFT DATABASE ADAPTER

- Миграция схемы
- Миграция данных
- Трансляция кода хранимой прикладной логики (функции, процедуры, пакеты, триггеры)
  - Oracle PL/SQL => PL/PgSQL
  - MS.SQL Server Transact SQL => PL/PgSQL
- Адаптация существующих клиентских приложений через поддержку нативного протокола обмена данными:
  - Oracle Transparent Network Substrate (TNS)
  - Microsoft Tabular Data Stream (TDS)

DATABASE MIGRATOR

Миграция схемы БД и данных

DATABASE MIGRATOR

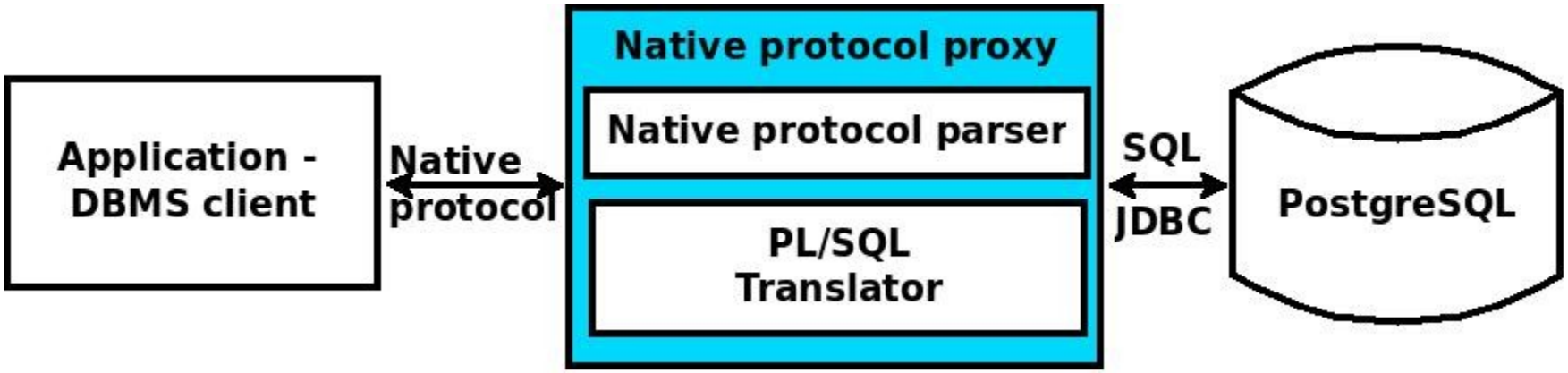
Миграция схемы БД и данных

При работе DB Migrator осуществляется:

- Миграция схемы БД
- Миграция существующих данных
- Автоматизированная трансляция текстов запросов
- Автоматизированная трансляция объектов БД, таких как:
  - Таблицы
  - индексы
  - последовательности
  - ограничения, ключи,
  - представления
  - Пакеты, функции, хранимые процедуры, агрегатные функции, триггеры
  - Пользовательские типы, в том числе объектные и табличные
  - Ссылки на внешние базы данных, связанные таблицы во внешних БД
  - Объекты безопасности: разрешения(Grant), методы аутентификации

Осуществляется трансляция специфичных для СУБД Oracle или MS.SQL Server конструкций, например:
Иерархические запросы, **Merge** , **(Oracle) нотации для внешних JOIN (+), Курсоры**, **Подпроцедуры**

NATIVE PROTOCOL PROXY



NATIVE PROTOCOL PROXY

Native Protocol Proxy выполняет следующие функции:

- Аутентификация в приложениях по нативному протоколу;
- Передача и преобразование SQL-запросов в грамматику PostgreSQL;
- Обратная передача ответов от PostgreSQL в приложение.

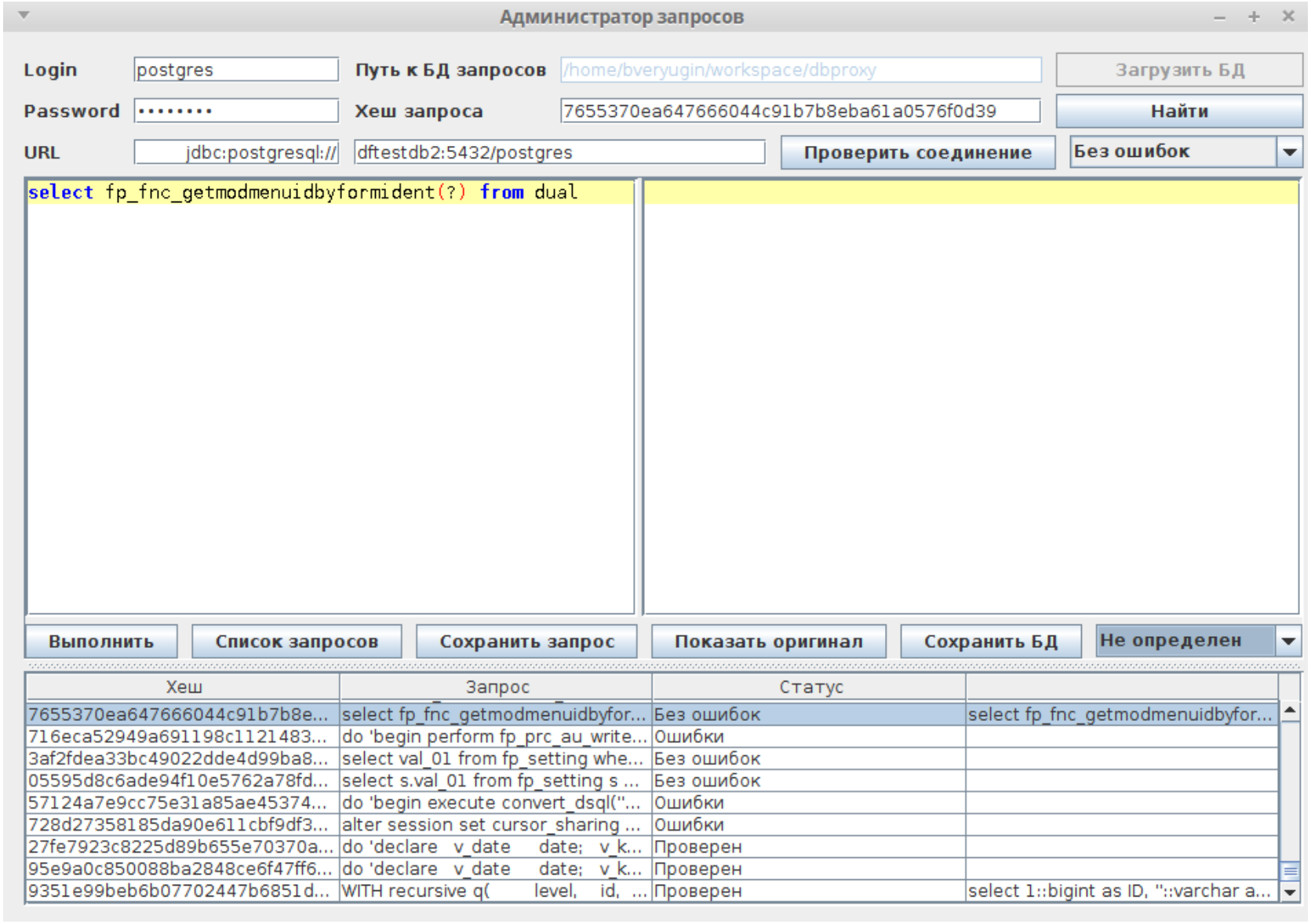
Трансляция SQL-запросов осуществляется компонентом «PL/SQL Translator» несколькими возможными способами:

- непосредственная трансляция;
- трансляция по словарю.

PL/SQL Translator - самообучаемый компонент, допускающий ручную коррекцию: при первой трансляции запроса заданного типа, его хэш с результатом трансляции пишется в словарь (H2-базу). При повторном исполнении того же запроса, соответствующий PostgreSQL запрос извлекается из словаря.

Среднее время обработки запроса на стороне Native Protocol Proxy - 1,5 мс.

NATIVE PROTOCOL PROXY: РУЧНАЯ КОРРЕКЦИЯ ЗАПРОСОВ



ИСПОЛЬЗУЕМЫЕ ТЕХНОЛОГИИ

ANother Tool for Language Recognition **http://www.antlr.org/**

Грамматика для pl/sql: **https://github.com/porcelli/plsql-parser**

Используется лексический, синтаксический и контекстный анализ PL/SQL кода и последующая трансляция в код на PL/pgSQL.

http://netty.io - используется для реализации нативных протоколов

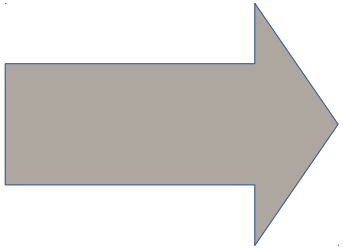
Расширения PostgreSQL

- **orafce** https://github.com/orafce/orafce
  Реализует многие функции Oracle

- **oracle_fdw** https://github.com/laurenz/oracle_fdw
  Позволяет реализовать подключение из PostgreSQL к внешним БД Oracle. Является аналогом Oracle database link.

ПОДДЕРЖИВАЕМЫЕ СУБД

- Oracle
- Microsoft SQL Server
- Sybase
- IBM DB2



- PostgreSQL
- Oracle
- SAP HANA
- MySQL
- Линтер
- Ред База данных

- поддерживается в текущей версии

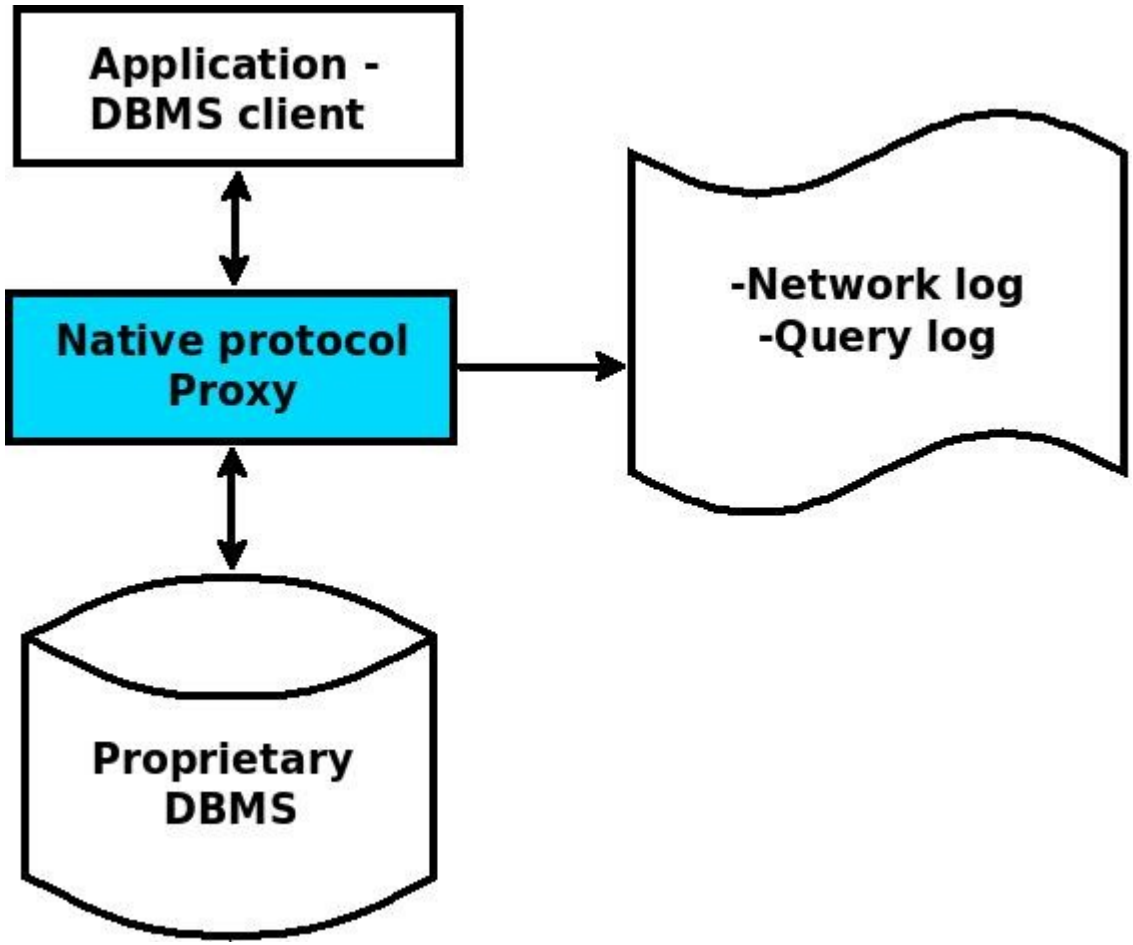
- ведутся работы

- планируется поддержать в следующих версиях

Приложения

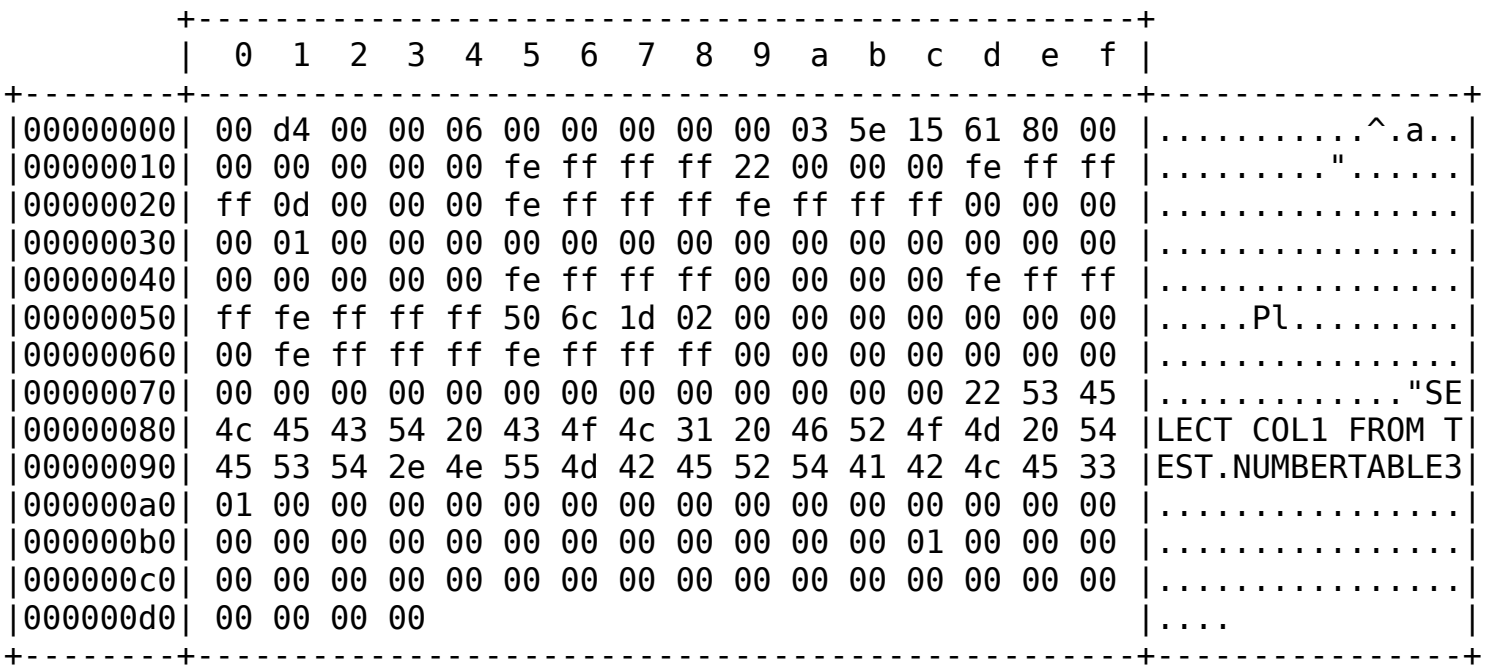
Исследование нативного протокола (на примере SELECT-запроса к Oracle)

ИССЛЕДОВАНИЕ НАТИВНОГО ПРОТОКОЛА



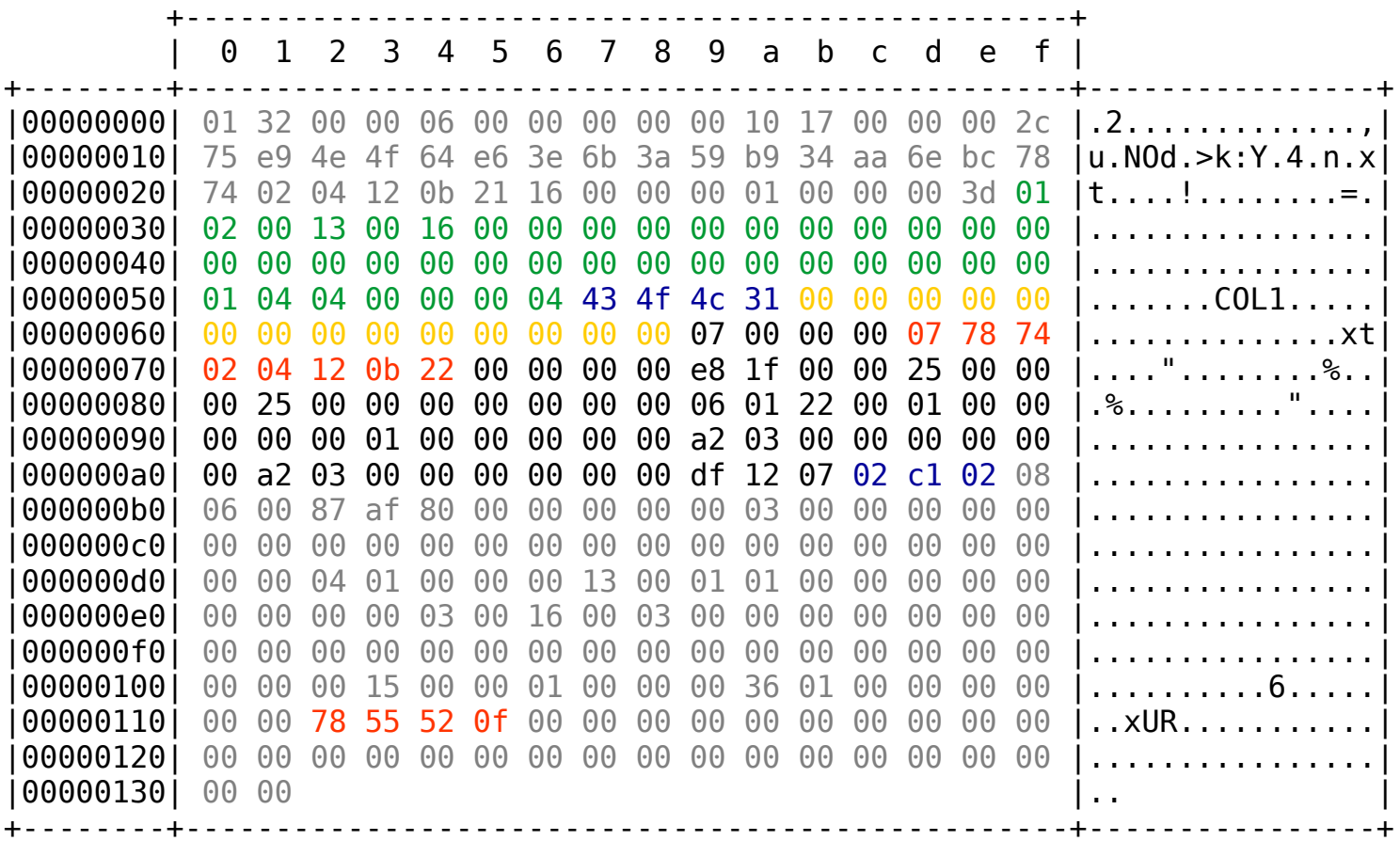
- Анализ пакетов нативного протокола.
- Анализ запросов, поступающих из приложения в базу данных.

Запрос:



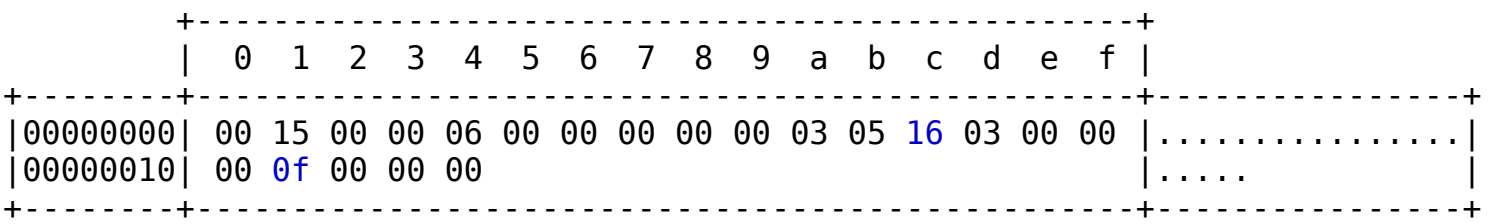
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 d4 00 00 06 00 00 00 00 00 03 5e 15 61 80 00 |...........^.a..|
|00000010| 00 00 00 00 00 fe ff ff ff 22 00 00 00 fe ff ff |.........".......|
|00000020| ff 0d 00 00 00 fe ff ff ff fe ff ff ff 00 00 00 |................|
|00000030| 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000040| 00 00 00 00 00 fe ff ff ff 00 00 00 00 fe ff ff |................|
|00000050| ff fe ff ff ff 50 6c 1d 02 00 00 00 00 00 00 00 |.....Pl.........|
|00000060| 00 fe ff ff ff fe ff ff ff 00 00 00 00 00 00 00 |................|
|00000070| 00 00 00 00 00 00 00 00 00 00 00 00 00 22 53 45 |.............."SE|
|00000080| 4c 45 43 54 20 43 4f 4c 31 20 46 52 4f 4d 20 54 |LECT COL1 FROM T|
|00000090| 45 53 54 2e 4e 55 4d 42 45 52 54 41 42 4c 45 33 |EST.NUMBERTABLE3|
|000000a0| 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000b0| 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 |................|
|000000c0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000d0| 00 00 00 00                                     |....            |
+--------+-------------------------------------------------+----------------+
```

Ответ:



```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 01 32 00 00 06 00 00 00 00 00 10 17 00 00 00 2c |.2.............,|
|00000010| 75 e9 4e 4f 64 e6 3e 6b 3a 59 b9 34 aa 6e bc 78 |u.NOd.>k:Y.4.n.x|
|00000020| 74 02 04 12 0b 21 16 00 00 00 01 00 00 00 3d 01 |t....!........=.|
|00000030| 02 00 13 00 16 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000040| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000050| 01 04 04 00 00 00 04 43 4f 4c 31 00 00 00 00 00 |.......COL1.....|
|00000060| 00 00 00 00 00 00 00 00 00 07 00 00 00 07 78 74 |..............xt|
|00000070| 02 04 12 0b 22 00 00 00 00 e8 1f 00 00 25 00 00 |....".........%..|
|00000080| 00 25 00 00 00 00 00 00 00 06 01 22 00 01 00 00 |.%.........".....|
|00000090| 00 00 00 01 00 00 00 00 00 a2 03 00 00 00 00 00 |................|
|000000a0| 00 a2 03 00 00 00 00 00 00 df 12 07 02 c1 02 08 |................|
|000000b0| 06 00 87 af 80 00 00 00 00 00 03 00 00 00 00 00 |................|
|000000c0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000d0| 00 00 04 01 00 00 00 13 00 01 01 00 00 00 00 00 |................|
|000000e0| 00 00 00 00 03 00 16 00 03 00 00 00 00 00 00 00 |................|
|000000f0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000100| 00 00 00 15 00 00 01 00 00 00 36 01 00 00 00 00 |..........6.....|
|00000110| 00 00 78 55 52 0f 00 00 00 00 00 00 00 00 00 00 |..xUR...........|
|00000120| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000130| 00 00                                           |..              |
+--------+-------------------------------------------------+----------------+
```

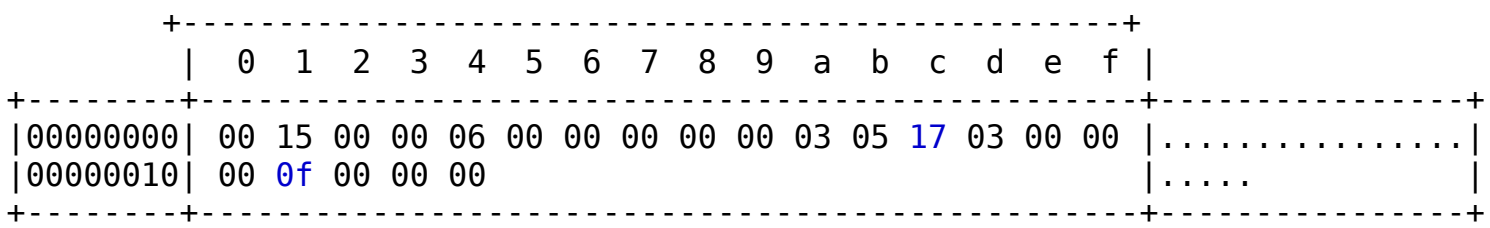
Запрос:


0 1 2 3 4 5 6 7 8 9 a b c d e f
00000000 00 15 00 00 06 00 00 00 00 00 03 05 16 03 00 00
00000010 00 0f 00 00 00

Ответ:

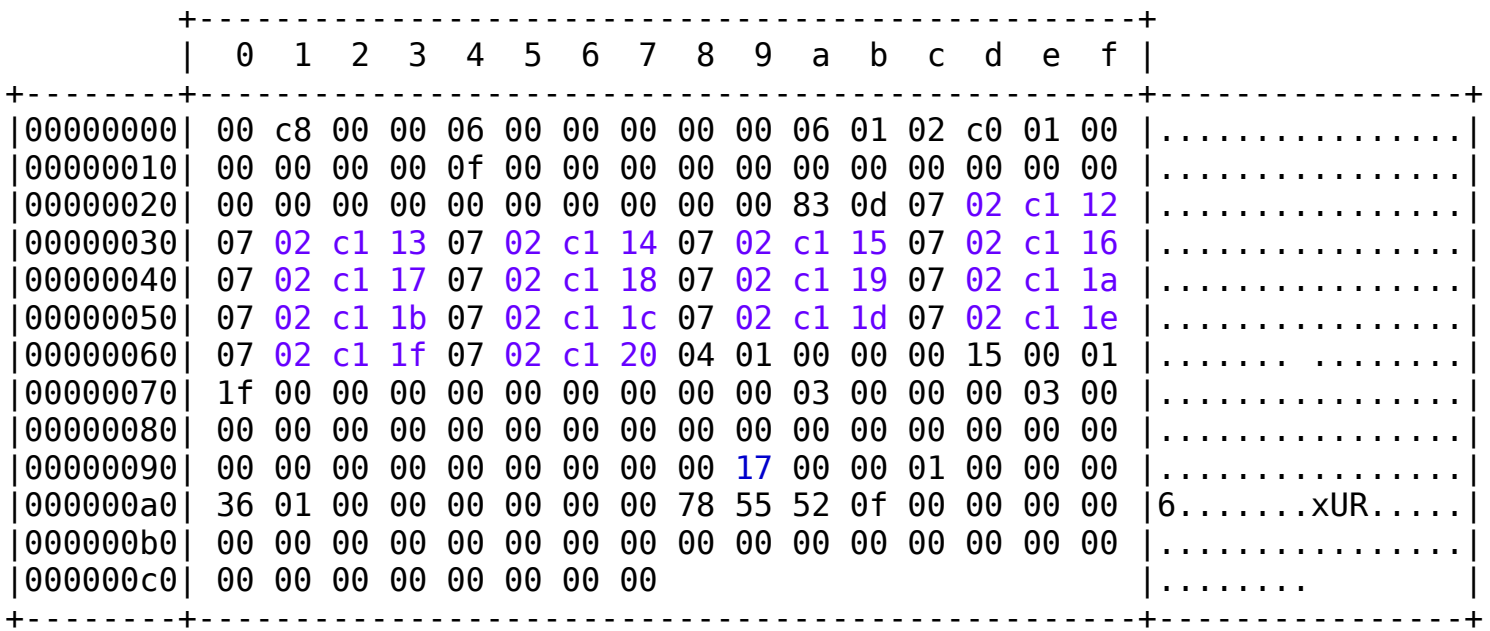
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 01 00 00 00 06 00 00 00 00 00 06 01 02 c0 01 00 |................|
|00000010| 00 00 00 00 0f 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000020| 00 00 00 00 00 00 00 00 00 00 83 0d 07 02 c1 03 |................|
|00000030| 15 01 00 01 07 02 c1 04 15 01 00 01 07 02 c1 05 |................|
|00000040| 15 01 00 01 07 02 c1 06 15 01 00 01 07 02 c1 07 |................|
|00000050| 15 01 00 01 07 02 c1 08 15 01 00 01 07 02 c1 09 |................|
|00000060| 15 01 00 01 07 02 c1 0a 15 01 00 01 07 02 c1 0b |................|
|00000070| 15 01 00 01 07 02 c1 0c 15 01 00 01 07 02 c1 0d |................|
|00000080| 15 01 00 01 07 02 c1 0e 15 01 00 01 07 02 c1 0f |................|
|00000090| 15 01 00 01 07 02 c1 10 15 01 00 01 07 02 c1 11 |................|
|000000a0| 04 01 00 00 00 14 00 01 10 00 00 00 00 00 00 00 |................|
|000000b0| 00 00 03 00 00 00 03 00 20 00 00 00 00 00 00 00 |........ .......|
|000000c0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000d0| 00 16 00 00 01 00 00 00 36 01 00 00 00 00 00 00 |........6.......|
|000000e0| 78 55 52 0f 00 00 00 00 00 00 00 00 00 00 00 00 |xUR.............|
|000000f0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
+--------+-------------------------------------------------+----------------+
```

Запрос:



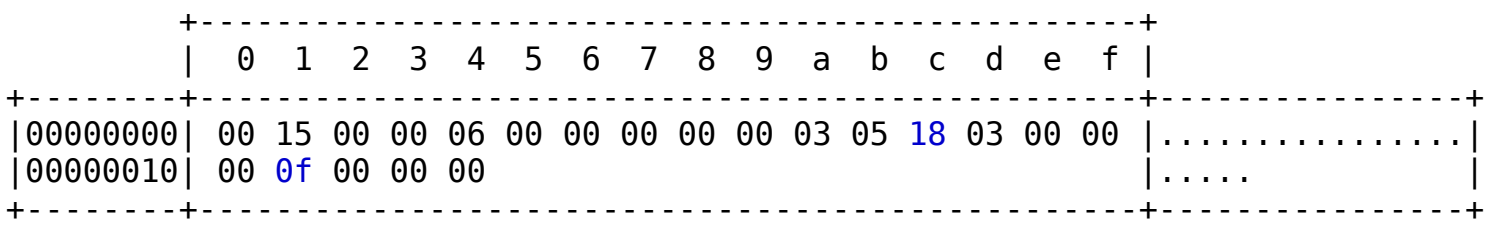
```
         +-------------------------------------------------+
          |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+---------+-------------------------------------------------+-----------------+
|00000000| 00 15 00 00 06 00 00 00 00 00 03 05 17 03 00 00 |.................|
|00000010| 00 0f 00 00 00                                  |.....            |
+---------+-------------------------------------------------+-----------------+
```

Ответ:



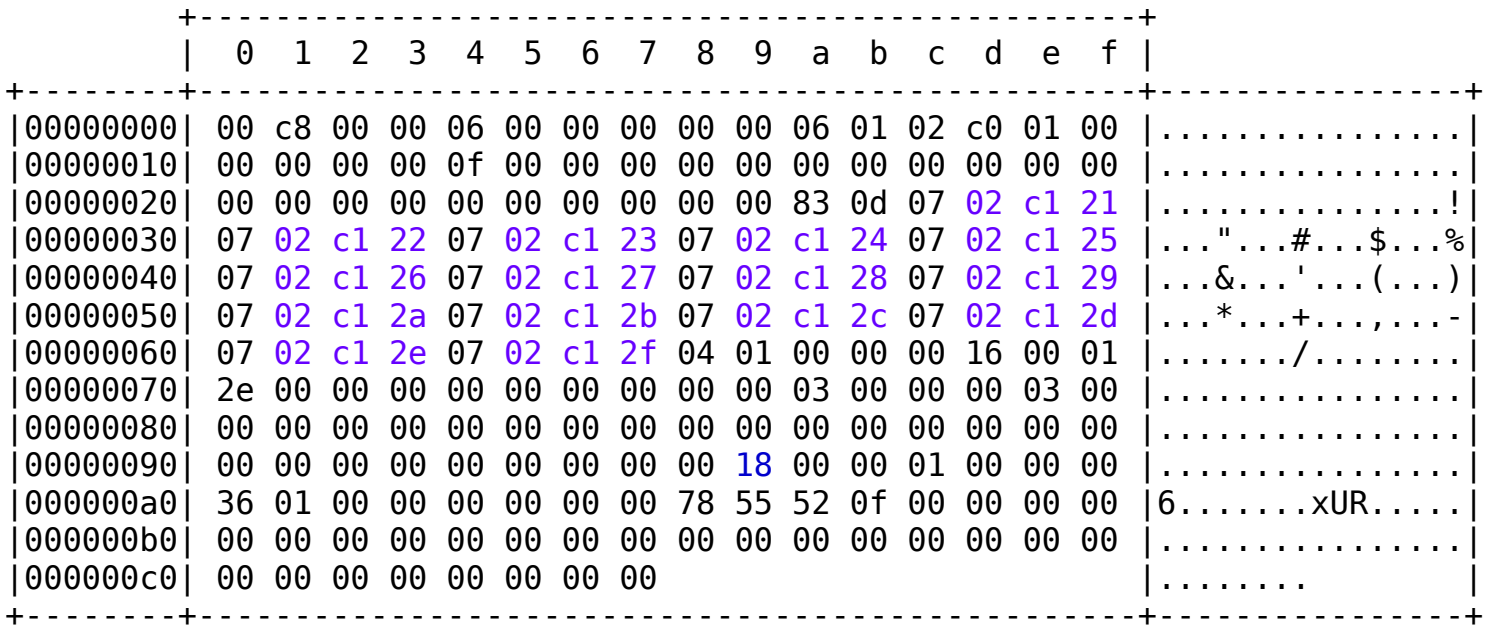
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 c8 00 00 06 00 00 00 00 00 06 01 02 c0 01 00 |................|
|00000010| 00 00 00 00 0f 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000020| 00 00 00 00 00 00 00 00 00 00 83 0d 07 02 c1 12 |................|
|00000030| 07 02 c1 13 07 02 c1 14 07 02 c1 15 07 02 c1 16 |................|
|00000040| 07 02 c1 17 07 02 c1 18 07 02 c1 19 07 02 c1 1a |................|
|00000050| 07 02 c1 1b 07 02 c1 1c 07 02 c1 1d 07 02 c1 1e |................|
|00000060| 07 02 c1 1f 07 02 c1 20 04 01 00 00 00 15 00 01 |....... ........|
|00000070| 1f 00 00 00 00 00 00 00 00 00 03 00 00 00 03 00 |................|
|00000080| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000090| 00 00 00 00 00 00 00 00 00 17 00 00 01 00 00 00 |................|
|000000a0| 36 01 00 00 00 00 00 00 78 55 52 0f 00 00 00 00 |6.......xUR.....|
|000000b0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000c0| 00 00 00 00 00 00 00 00                         |........        |
+--------+-------------------------------------------------+----------------+
```

Запрос:



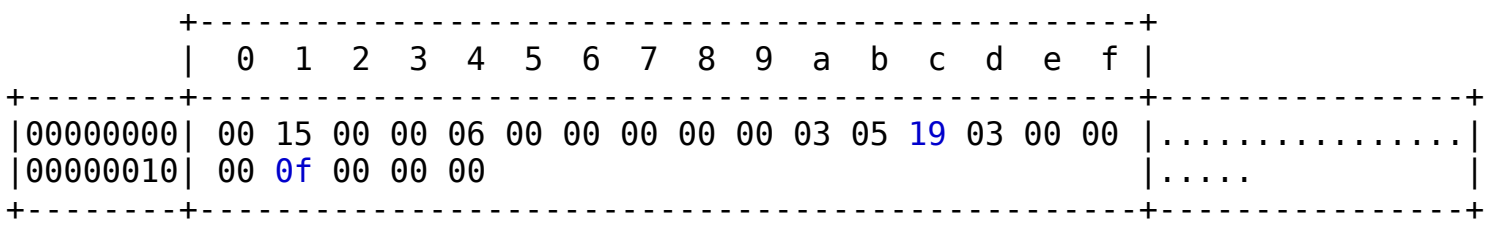
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+-----------------+
|00000000| 00 15 00 00 06 00 00 00 00 00 03 05 18 03 00 00 |................|
|00000010| 00 0f 00 00 00                                  |.....           |
+--------+-------------------------------------------------+-----------------+
```

Ответ:



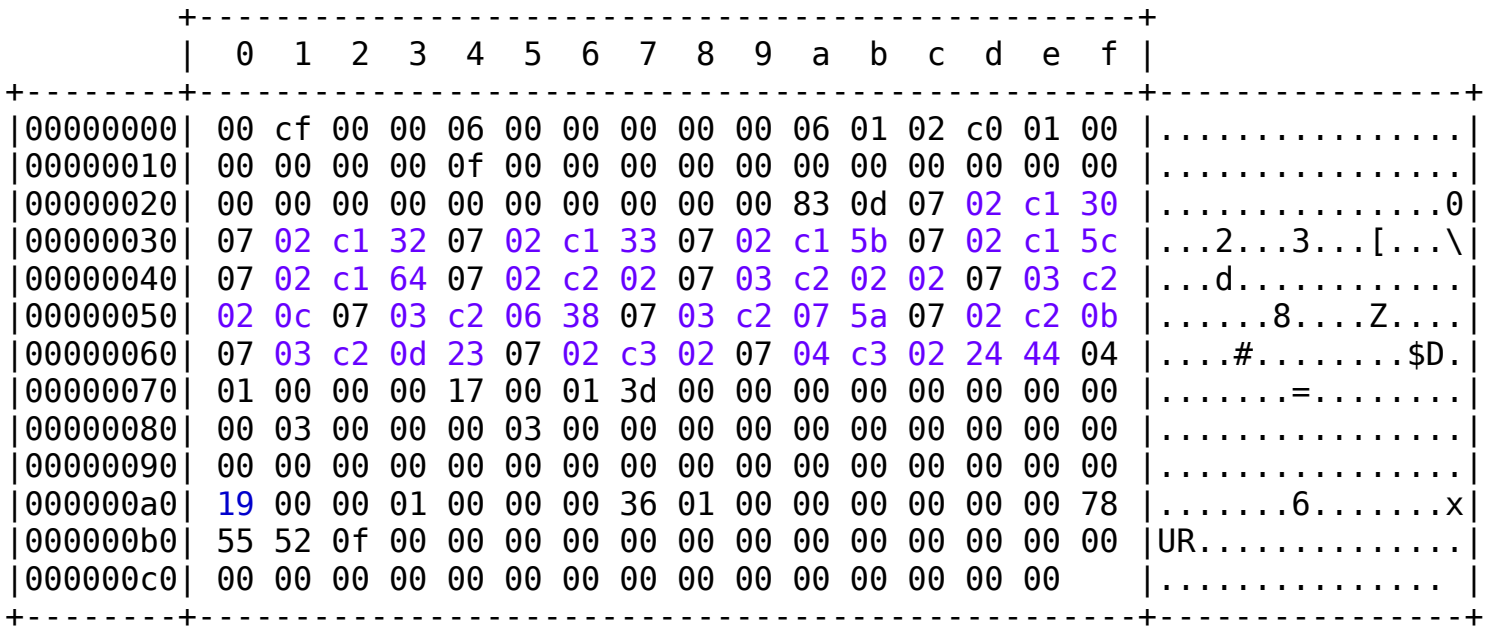
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 c8 00 00 06 00 00 00 00 00 06 01 02 c0 01 00 |................|
|00000010| 00 00 00 00 0f 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000020| 00 00 00 00 00 00 00 00 00 00 83 0d 07 02 c1 21 |...............!|
|00000030| 07 02 c1 22 07 02 c1 23 07 02 c1 24 07 02 c1 25 |...".../#...$...%|
|00000040| 07 02 c1 26 07 02 c1 27 07 02 c1 28 07 02 c1 29 |...&...'...(...)|
|00000050| 07 02 c1 2a 07 02 c1 2b 07 02 c1 2c 07 02 c1 2d |...*...+...,...-|
|00000060| 07 02 c1 2e 07 02 c1 2f 04 01 00 00 00 16 00 01 |......./........|
|00000070| 2e 00 00 00 00 00 00 00 00 00 03 00 00 00 03 00 |................|
|00000080| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000090| 00 00 00 00 00 00 00 00 00 18 00 00 01 00 00 00 |................|
|000000a0| 36 01 00 00 00 00 00 00 78 55 52 0f 00 00 00 00 |6.......xUR.....|
|000000b0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000c0| 00 00 00 00 00 00 00 00                         |........        |
+--------+-------------------------------------------------+----------------+
```

Запрос:



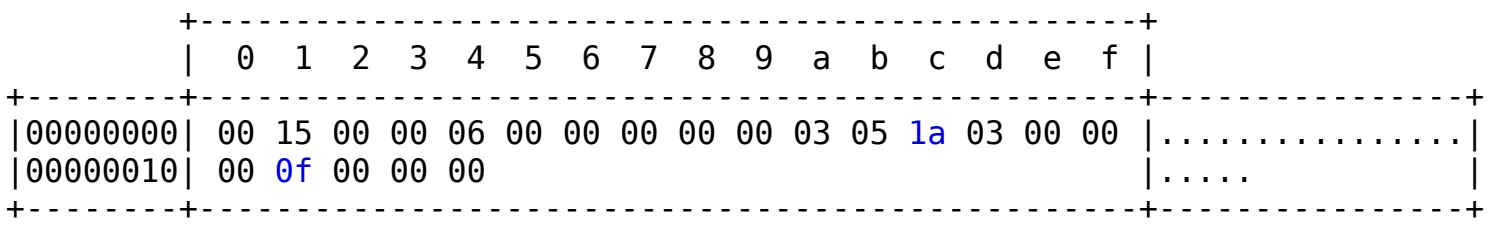
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 15 00 00 06 00 00 00 00 00 03 05 19 03 00 00 |................|
|00000010| 00 0f 00 00 00                                  |.....           |
+--------+-------------------------------------------------+----------------+
```

Ответ:



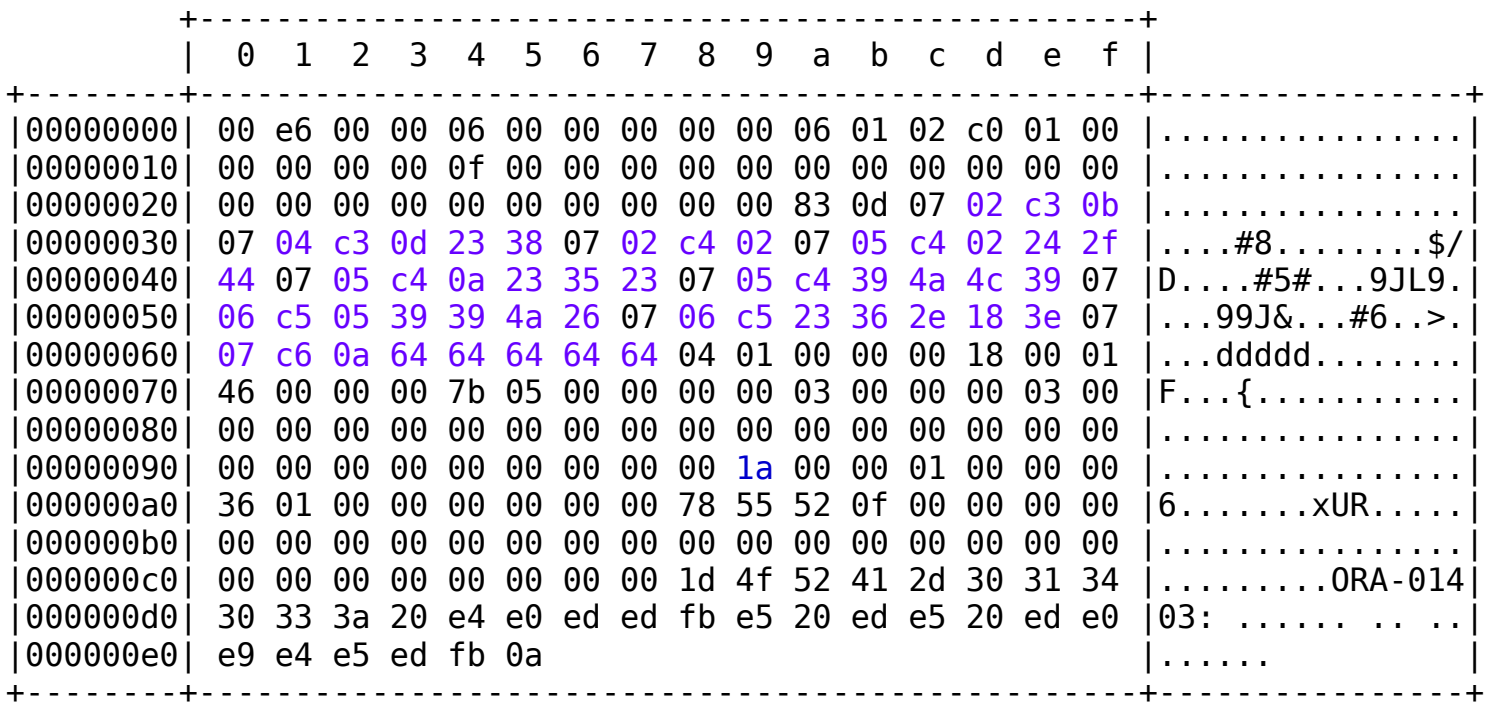
```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 cf 00 00 06 00 00 00 00 00 06 01 02 c0 01 00 |................|
|00000010| 00 00 00 00 0f 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000020| 00 00 00 00 00 00 00 00 00 00 83 0d 07 02 c1 30 |...............0|
|00000030| 07 02 c1 32 07 02 c1 33 07 02 c1 5b 07 02 c1 5c |...2...3...[...\|
|00000040| 07 02 c1 64 07 02 c2 02 07 03 c2 02 02 07 03 c2 |...d............|
|00000050| 02 0c 07 03 c2 06 38 07 03 c2 07 5a 07 02 c2 0b |......8....Z....|
|00000060| 07 03 c2 0d 23 07 02 c3 02 07 04 c3 02 24 44 04 |....#........$D.|
|00000070| 01 00 00 00 17 00 01 3d 00 00 00 00 00 00 00 00 |.......=........|
|00000080| 00 03 00 00 00 03 00 00 00 00 00 00 00 00 00 00 |................|
|00000090| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000a0| 19 00 00 01 00 00 00 36 01 00 00 00 00 00 00 78 |.......6.......x|
|000000b0| 55 52 0f 00 00 00 00 00 00 00 00 00 00 00 00 00 |UR..............|
|000000c0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................ |
+--------+-------------------------------------------------+----------------+
```

Запрос:



```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 15 00 00 06 00 00 00 00 00 03 05 1a 03 00 00 |................|
|00000010| 00 0f 00 00 00                                  |.....           |
+--------+-------------------------------------------------+----------------+
```

Ответ:



```
         +-------------------------------------------------+
         |  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f |
+--------+-------------------------------------------------+----------------+
|00000000| 00 e6 00 00 06 00 00 00 00 00 06 01 02 c0 01 00 |................|
|00000010| 00 00 00 00 0f 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000020| 00 00 00 00 00 00 00 00 00 00 83 0d 07 02 c3 0b |................|
|00000030| 07 04 c3 0d 23 38 07 02 c4 02 07 05 c4 02 24 2f |....#8........$/|
|00000040| 44 07 05 c4 0a 23 35 23 07 05 c4 39 4a 4c 39 07 |D....#5#...9JL9.|
|00000050| 06 c5 05 39 39 4a 26 07 06 c5 23 36 2e 18 3e 07 |...99J&...#6..>.|
|00000060| 07 c6 0a 64 64 64 64 64 04 01 00 00 00 18 00 01 |...ddddd........|
|00000070| 46 00 00 00 7b 05 00 00 00 00 03 00 00 00 03 00 |F...{...........|
|00000080| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|00000090| 00 00 00 00 00 00 00 00 00 1a 00 00 01 00 00 00 |................|
|000000a0| 36 01 00 00 00 00 00 00 78 55 52 0f 00 00 00 00 |6.......xUR.....|
|000000b0| 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
|000000c0| 00 00 00 00 00 00 00 00 1d 4f 52 41 2d 30 31 34 |.........ORA-014|
|000000d0| 30 33 3a 20 e4 e0 ed ed fb e5 20 ed e5 20 ed e0 |03: ...... .. ..|
|000000e0| e9 e4 e5 ed fb 0a                               |......          |
+--------+-------------------------------------------------+----------------+
```

Разработка транслятора при помощи ANTLR

Разработка транслятора при помощи ANTLR

- ANTLR - инструмент для построения LL(k) - анализаторов.
- Для записи грамматики используется расширенная форма Бакуса-Науэра (РБНФ)
- Открытый исходный код инструмента (https://github.com/antlr)
- Существуют доступные готовые грамматики для:
  - Oracle PL/SQL:
    https://github.com/porcelli/plsql-parser
  - MS.SQL Server Transact SQL:
    https://github.com/antlr/grammars-v4/tree/master/tsq

Лексический анализ

```
public Token nextToken() {
    while (true) {
        if (tokenBuffer.size() == 0) {
            state.token = null;
            state.channel = Token.DEFAULT_CHANNEL;
            state.tokenStartCharIndex = input.index();
            state.tokenStartCharPositionInLine = input.getCharPositionInLine();
            state.tokenStartLine = input.getLine();
            state.text = null;
            if (input.LA(1) == CharStream.EOF) {
                return getEOFToken();
            }
            try {
                int m = input.mark();
                state.backtracking = 1;
                state.failed = false;
                mTokens();
                state.backtracking = 0;

                if (state.failed) {
                    input.rewind(m);
                    input.consume();
                } else {
                    emit();
                }
            } catch (RecognitionException re) {
                // shouldn't happen in backtracking mode, but...
                reportError(re);
                recover(re);
            }
        } else {
            Token result = tokenBuffer.poll();
            if (result == Token.SKIP_TOKEN || result.getType() == Token.INVALID_TOKEN_TYPE || result == null)
            {
                // discard
                // SKIP & INVALID
                // tokens
                continue;
            }
            return result;
        }
    }
```

Лексический анализ

```
SQL92_RESERVED_DELETE
    :    'delete'
    ;

SQL92_RESERVED_DESC
    :    'desc'
    ;

SQL92_RESERVED_DISTINCT
    :    'distinct'
    ;

SQL92_RESERVED_DROP
    :    'drop'
    ;

SQL92_RESERVED_ELSE
    :    'else'
    ;

SQL92_RESERVED_END
    :    'end'
    ;
```

Синтаксический анализ

```
statement
options{
backtrack=true;
}
    :    create_key swallow_to_semi (SEMICOLON|EOF)
    |    alter_key swallow_to_semi  (SEMICOLON|EOF)
    |    grant_key swallow_to_semi  (SEMICOLON|EOF)
    |    truncate_key swallow_to_semi  (SEMICOLON|EOF)
    |    (begin_key) => body
    |    (declare_key) => block
    |    type_delete_statement
    |    assignment_statement
    |    continue_statement
    |    exit_statement
    |    goto_statement
    |    if_statement
    |    loop_statement
    |    forall_statement
    |    null_statement
    |    raise_statement
    |    return_statement
    |    case_statement[true]
    |    sql_statement
    |    pipe_statement
    |    function_call
    ;
```

Синтаксический анализ

```
sql_statement
options{
backtrack=true;
}    :   {GrammarTreeReader.enterScope(); } et=execute_immediate
{ExecuteImmediateReader.setExecuteImmediateStatement($et.text, $et.stop);} {GrammarTreeReader.leaveScope();}
    |    execute_immediate_pg
    |    data_manipulation_language_statements
    |    cursor_manipulation_statements
    |    transaction_control_statements
    ;

execute_immediate
    :    execute_key immediate_key {ExecuteImmediateReader.setImmediateToken($immediate_key.start);} expression
    (    x1=into_clause {ExecuteImmediateReader.setFistTokenAfterStatement($x1.start);
ExecuteImmediateReader.hasInto();} using_clause?
    |    x2=using_clause {ExecuteImmediateReader.setFistTokenAfterStatement($x2.start);
ExecuteImmediateReader.hasUsing();} dynamic_returning_clause?
    |    x3=dynamic_returning_clause {ExecuteImmediateReader.setFistTokenAfterStatement($x3.start);}
    )?
    ;
```

СПАСИБО ЗА ВНИМАНИЕ!

Жду ваши вопросы на email: **boris.veryugin@diasoft-platform.ru**
Либо по телефонам:
+7 (967) 292-24-28
+7 (495) 780-75-75 доб. 7623