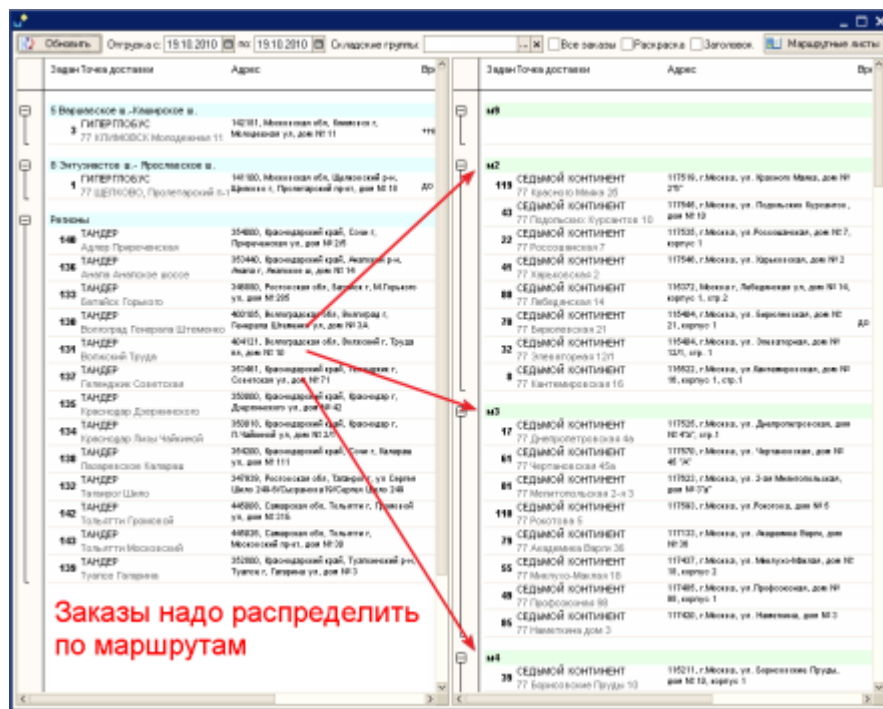


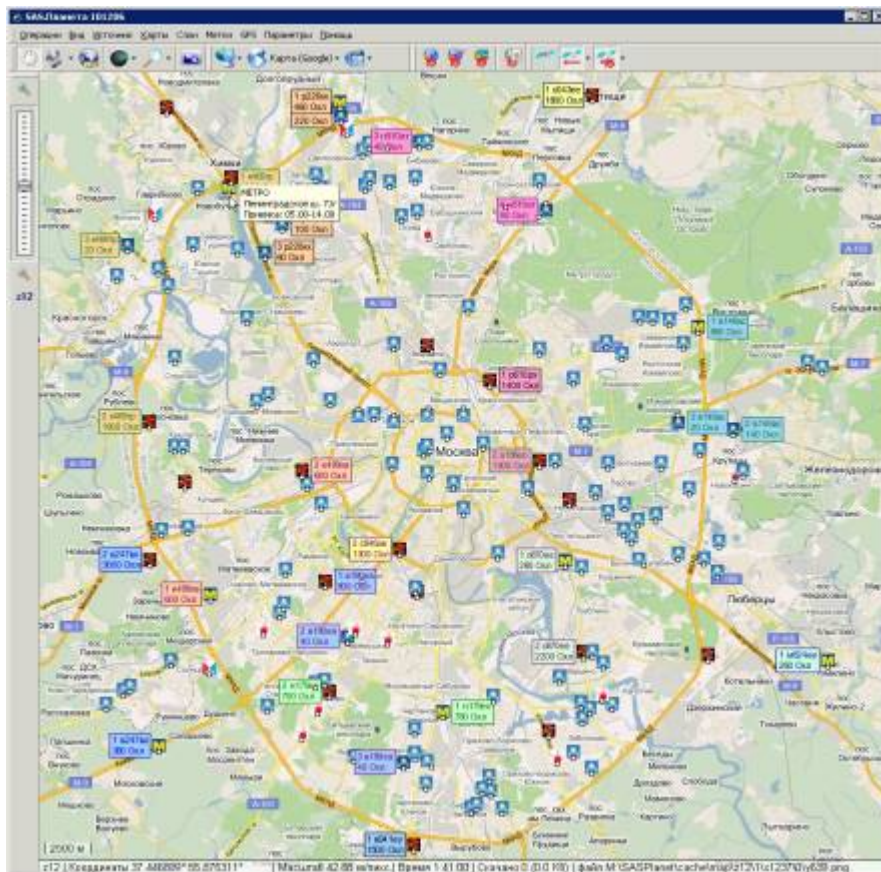
В данной статье рассматривается примеи организация связи между программами 1С и SAS.Планета.

Типичной задачей, где нужны карты, является работа логиста по планированию развоза. То есть: имеем заказы от покупателей (адреса доставки и объемы (веса)). Из них надо составить маршруты, по которым поедут машины.

Примерно вот так:



Разумеется, что при наличии карты, на которой все это наглядно отображено, выполнять такую работу намного проще, особенно когда заказов много.

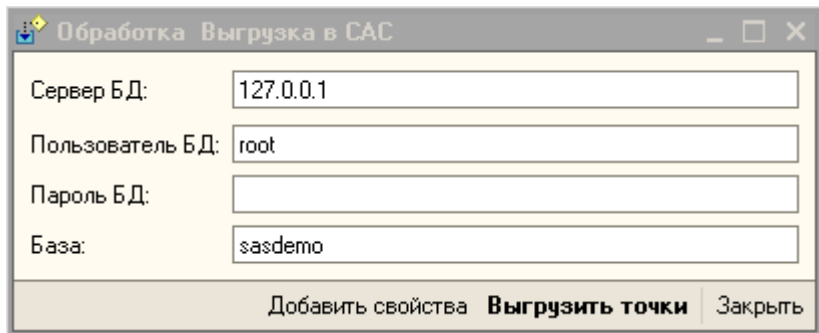


Информацию из 1С мы будем выводить с помощью технологии, описанной [здесь](#). Если вкратце - к SAS Планета подключается дополнительный слой, который посредством специального PHP скрипта выбирает из БД информацию о выводимых точках и отдает ее в SAS.Планету в виде картинок и/или слоя KML.

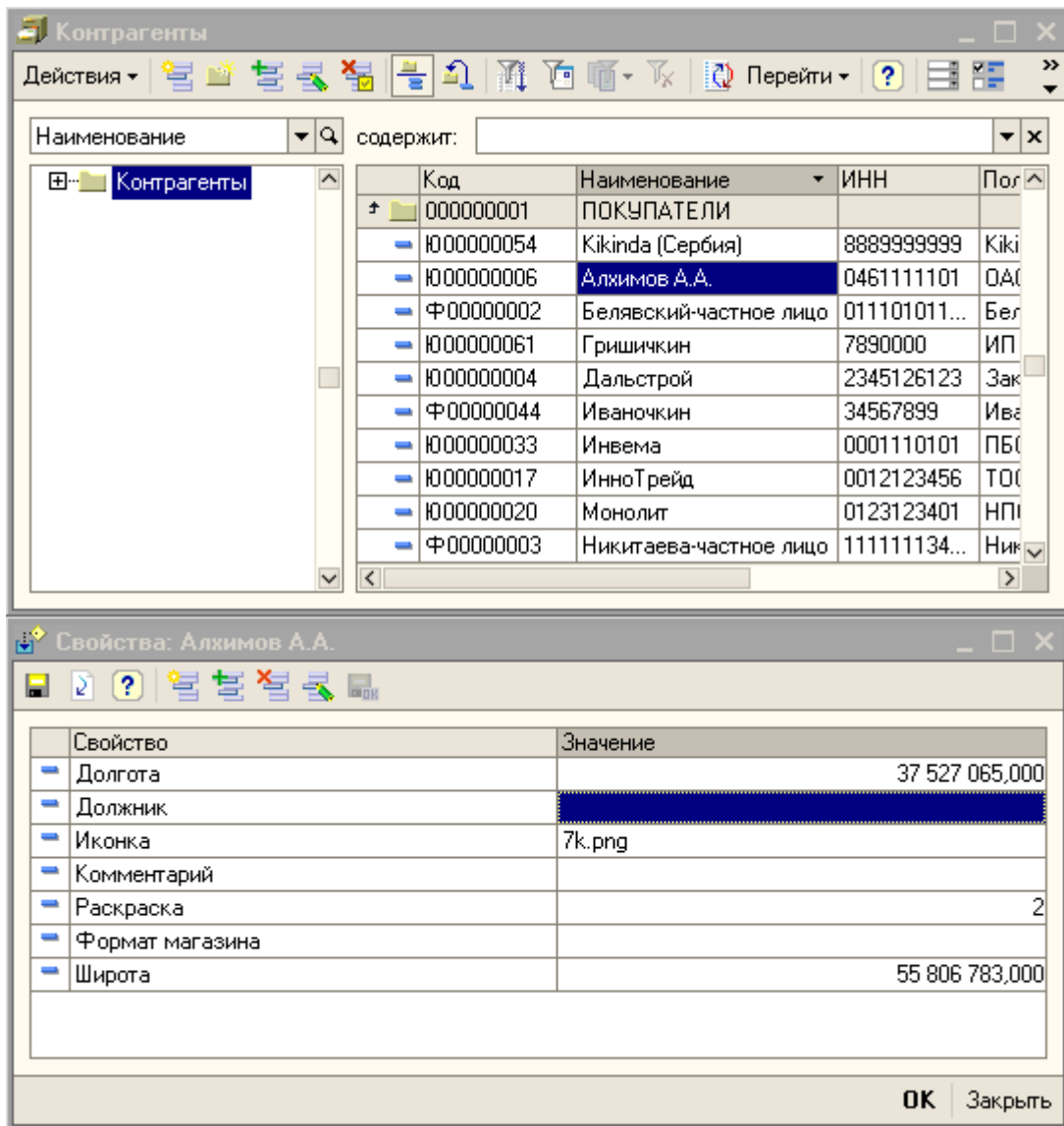
Обращение к БД реализуется по-разному, в зависимости того, в каком режиме используется 1С - файловом или серверном. При использовании 1С в файловом варианте необходима выгрузка информации на сервер БД. (Есть еще вариант с веб-компонентой, его рассмотрим в другой раз). Пример выгрузки данных из конфигурации Управление торговлей, ред 10.3

1. Установите у себя пример базы данных в соответствии с инструкцией
2. Установите [MySQL ODBC Connector 5.1](#)
3. Скачайте обработку выгрузки: Выгрузка в SAS.erf
4. Откройте обработку в режиме «1С:Предприятие». (Я использовал «демонстрационную» базу «Управление торговлей»).

Основная форма обработки:



1. Нажмите кнопку «Добавить характеристики». После этого для справочника «Контрагенты» будут добавлены дополнительные свойства:



Широта, Долгота - географические координаты. ВНИМАНИЕ: поскольку в конфигурации «Управление Торговлей ред. 10.3» для дополнительных свойств типа число всегда установлена точность 15.3, то десятичная точка в координатах сдвинута на 6 знаков вправо. Вместо 55.123456 вводим 55123456. Разумеется, в реальном применении можно добавить реквизиты с нужной точностью. В демонстрационном примере это сделано только для того, чтобы обеспечить его работу без внесения изменений в конфигурацию БД.

Комментарий - строка, до 50 символов.

Иконка - название файла с иконкой. В демо-примере есть следующие файлы: (7k.png, auchan.png, metro.png, x5.png, tande.png, victoria.png, sweetlife.png, spar.png). Если иконка не задана, или такого файле нет, то используется иконка noicon.png

Раскраска - цвет комментария, указывается число в интервале 0-19

2. В справочнике «Контрагенты» укажите соответствующие значения для нескольких контрагентов

3. В форме обработки укажите параметры подключения к MySQL серверу

4. Нажмите кнопку «Выгрузить».

5. Обновите слои с точками в SAS.Планете (Масштаб увеличить - подождать 4 секунды - масштаб уменьшить).

Более удобен для работы с SAS.Планета серверный вариант использования 1С. То есть сервер уже есть и можно читать информацию можно непосредственно из БД. 1С, не заводя промежуточной базы Благодаря этому на карте будет всегда актуальная информация, причем никаких действий со стороны 1С выполнять не потребуется.

1С 8 версии может работать с различными серверами БД, однако наиболее широкое распространение получила связка 1С + MS SQL, а в качестве операционной системы - Windows Server различных версий.

PHP (я использовал версию 5.2) имеет встроенные средства для работы с MS SQL. Однако, поскольку PHP + MS SQL (не путать с MySQL) используется не очень часто, то модуль MSSQL по умолчанию в PHP выключен. Его надо включить - раскомментировать строчку

```
extension=php_mssql.dll
```

в разделе extensions файла php.ini А также из каталога windows\system32 MSSQL сервера взять файл ntwdblib.dll и скопировать его в соответствующий каталог сервера, где стоит Apache (и еще можно в каталог PHP). Если этого не сделать, то в логге Апача будет сообщение о проблемах в модуле mssql и ни слова об отсутствующей dll ке. Функции PHP для работы с MSSQL аналогичны функциям для работы с MySQL.

Осталось найти нужные поля в SQL базе 1С. Дело в том, что 1С встроенными средствами названия полей не показывает. Проще всего будет воспользоваться готовой обработкой, например: (совсем простая) или <http://infostart.ru/public/16282/> (посложнее) <http://infostart.ru/public/19821/> (совсем навороченная)

В итоге у Вас должно получиться что-то вроде такого:

```
function gettotallabel($x1, $y1, $x2, $y2, $begdate, $enddate) {
    $arr = array();
    $bcolor1 = array(255, 255, 255);
    $bcolor2 = array(224, 224, 224);

    $t_sk = '_Reference6896'; // склады контрагентов
    $t_sk_x = '_Fld6917'; // долгота
```

```

$t_sk_y = '_Fld6916'; // широта

$t_ml      = '_Document6904'; //
$t_mlm     = '_Document6904_VT7017'; // маршрутный лист
маршрут
$t_mlm_id  = '_Document6904_IDRRef';
$t_mlm_zadanie = '_Fld7019RRef';

$t_orders  = '_Document6903'; // заказы
$t_orders_weight = '_Fld6961'; // вес
$t_orders_sk = '_Fld6952RRef'; // всего количество
$t_orders_datefinal = '_Fld7329'; // окончательная дата

$$BegdateStart = date('Ymd', $begdate);
$$BegdateEnd = date('Ymd', $begdate + 24 * 3600);
$$EnddateEnd = date('Ymd', $enddate + 24 * 3600);

// заказы на текущий день НЕ на маршруте
$query = "
    SELECT
        COUNT(*) AS kolvoOrders,
        COUNT(DISTINCT orders.$t_orders_sk) AS kolvoPoints,
        SUM(orders.$t_orders_weight) AS weight
    FROM
        $t_orders AS orders
    WHERE
        orders.$t_orders_datefinal='$$BegdateStart'
        AND orders._IDRRef IN
            (SELECT orders_IDRRef FROM #ordersnoroute)";
$result = my_mysql_query($query);
$rowFree = mssql_fetch_assoc($result);

```

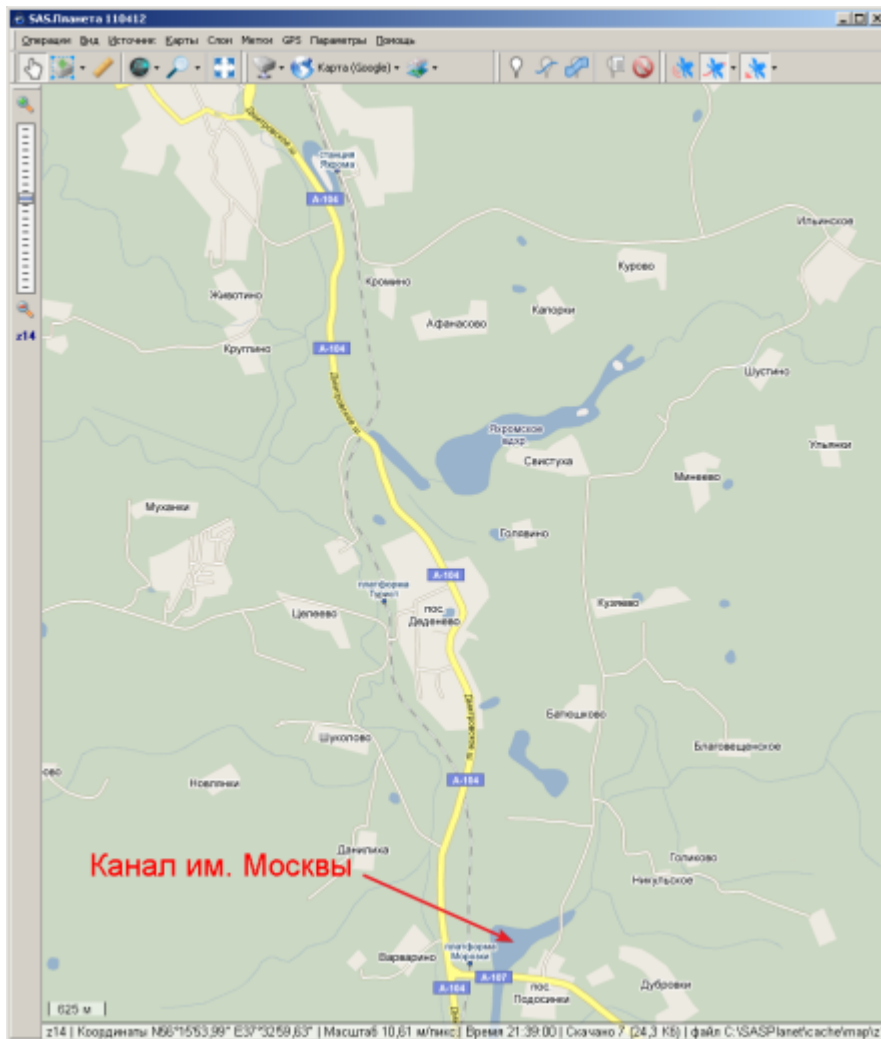
Необходимо иметь в виду, что идентификаторы таблиц и полей в 1С не являются постоянным, т.е. если выгрузить конфигурацию, где добавлены таблицы/поля (не всю БД) из одной базы и загрузить в другую, то эти добавленные таблицы/поля могут получить уже другие идентификаторы.

Поскольку в наиболее распространенных типовых конфигурациях 1С готовых полей для хранения координат нет, уже настроенный скрипт привести не могу. Но надеюсь, что на основании изложенного здесь принципа и примера скрипта тут, который можно взять за основу написать скрипт под конкретную базу 1С проблем не составит.

Что мы получаем на выходе

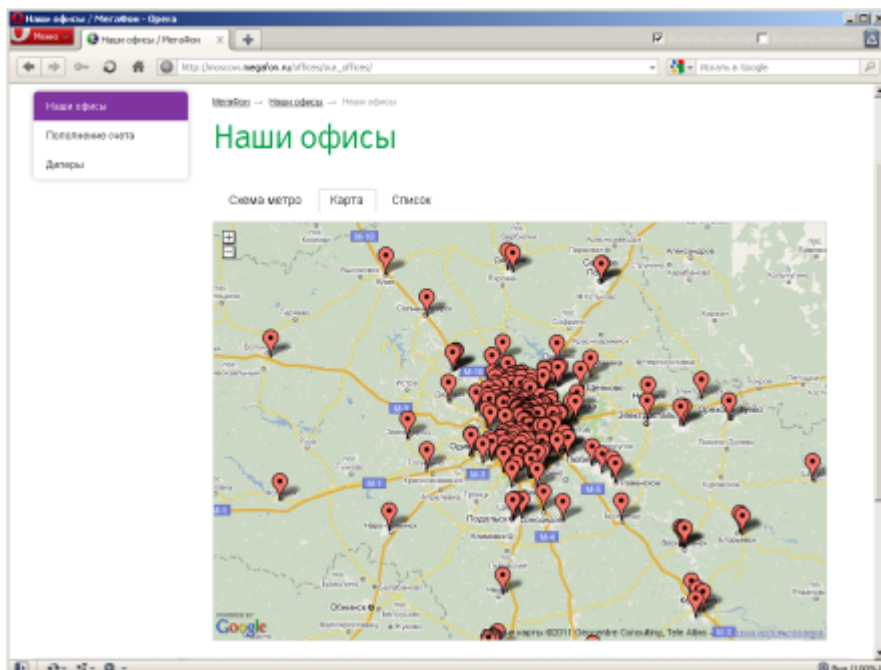
Красивую картинку на карте - это понятно. Но не только ее

1. Возможность использования для каждого места лучшей карты Преимущество использования программы SASПланета в данном случае в том, что мы не привязаны к какой-либо конкретной карте. И чем дальше от столицы, тем это важнее. Ведь качество наиболее популярных в интернете карт, обратно пропорционально удалению от Москвы. Если уж на часто используемой карты от Google канал им. Москвы (а это, прямо скажем, не деревенская улица и даже не МКАД) заканчивается у платформы Морозки



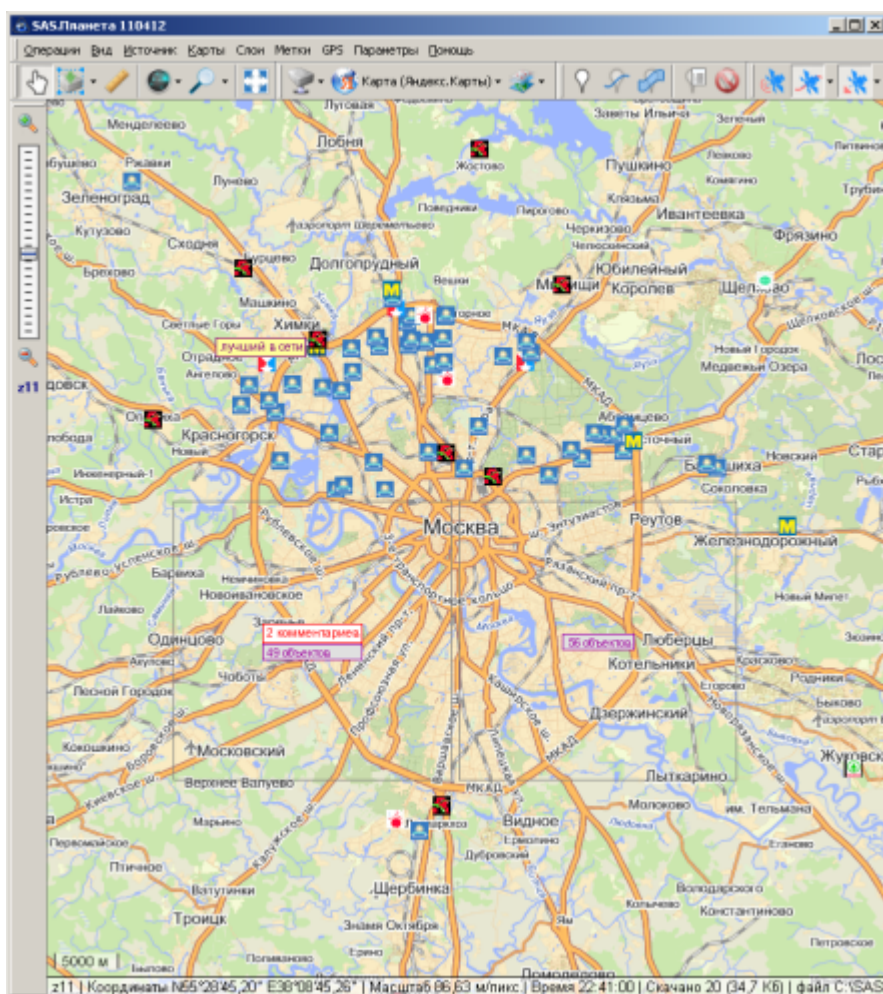
То говорить о каком-либо качестве карт километров хотя бы за 100 от столицы уже не приходится.

2. Полный контроль над формируемой картой (слоем). Например в ситуации: что делать, когда точек в тайле слишком много. Если выводить все, то имеем мешанину:



На мой взгляд, это неудобно.

Другой вариант, который применяется например в слое Раnogaтiо - выводить часть точек. Но он тоже не дает полной картины. Поэтому я лично в таком случае просто вывожу сообщение и большом количестве объектов в тайле:



3. Возможность автономной (без интернета) работы (благодаря умению SAS Планеты работать с кэшем)

From: <http://www.sasgis.org/wikisasiya/> - SAS.Wiki

Permanent link: http://www.sasgis.org/wikisasiya/doku.php/playground:1c_%D0%B8_sas.%D0%BF%D0%BB%D0%B0%D0%BD%D0%B5%D1%82%D0%B0?rev=1306674713

Last update: 04/03/2014 05:15

