GeoKettle

www.geokettle.org
Лицензия: GPL v2

Программа GeoKettle в терминах бизнес-анализа (business intelligence) относится к инструментам ETL (Extract, Transform, Load) и, соответственно, предназначена для загрузки данных из разрозненных источников в единый банк данных.

Применим ее возможности для повседневной работы с данными ГИС. Программа позволяет загружать данные самых различных форматов, преобразовывать их и сохранять результат обратно в файлы и базы данных. Поскольку GeoKettle основана на ETL общего назначения Kettle [1], то помимо специфичных функций ГИС (см. ниже) доступны и базовые операции — алгебраические функции, фильтры (в том числе по RegExp), обработка текстовых строк, пользовательские скрипты на языках JavaScript и SQL, и т. п.

Как правило это не разовые операции, а рутинные процессы. Например, вы регулярно получаете файлы от смежных организаций, загружаете материалы с публичных интернет-ресурсов, производите агрегацию данных двух независимых департаментов и т. п. В таком случае составляется цепочка действий, сохраняется и вызывается в нужный момент. Думаю, многие специалисты ГИС имеют в своем запасе несколько командных файлов (скриптов), для преобразования форматов, проекций, объединения файлов. GeoKettle выполняет тоже самое, но в графическом виде.

В статье не описываются специфичные для бизнес-анализа задачи (составление кубов Spatial OLAP), а затрагиваются только возможности ГИС.

Итак, программа умеет работать с различными форматами данных:

- Базы данных: PostGIS, Oracle Spatial, MySQL, Microsoft SQL Server и т. д.;
- Файлы ESRI Shapefile, GML, KML, форматы GDAL/OGR (только векторные);
- Сервисы OGC: SOS, CSW.

Выполняет преобразования:

- Построение буферов и центроидов, вычисление длин и площадей;
- Пространственная алгебра — объединение, пересечение и т. п.;
- Преобразование линий в полигоны и обратно, упрощение и сглаживание;
- Триангуляция Делоне.

Существует проект BeETLe [2], ставящий своей целью интеграцию в GeoKettle библиотеки SEXTANTE [3] и возможность работы с растровыми форматами. Приложение было представлено на FOSS4G 2010, но состояние развития проекта неизвестно.

По функциональности программа близка к приложению ArcGIS ModelBuilder.

Установка программы

Программа работает в Windows, Linux, Mac OS X и др. Инсталлятор единый — geokettle-XXX-installer.jar [4] (около 160 Мб). В системе уже должен быть установлен Java SDK. При работе в Linux файл требуется сделать запускаемым (chmod -x). Во время установки создается ярлык на рабочем столе, но, если этого не произошло, то программа запускается файлом geokettle.bat (geokettle.sh).

Простой пример

В качестве примера рассмотрим загрузку в базу данных PostGIS слоя границ субъектов РФ [5]: файл в формате ESRI Shapefile, кодировка windows-1251, проекция Albers-Siberia на эллипсоиде WGS84. Требуемая проекция — epsg:4326 (WGS84).

Запустим программу и выберем вариант "No repository". (Репозиторий используется для хранения документов и настроек в базе данных и для совместной работы.)

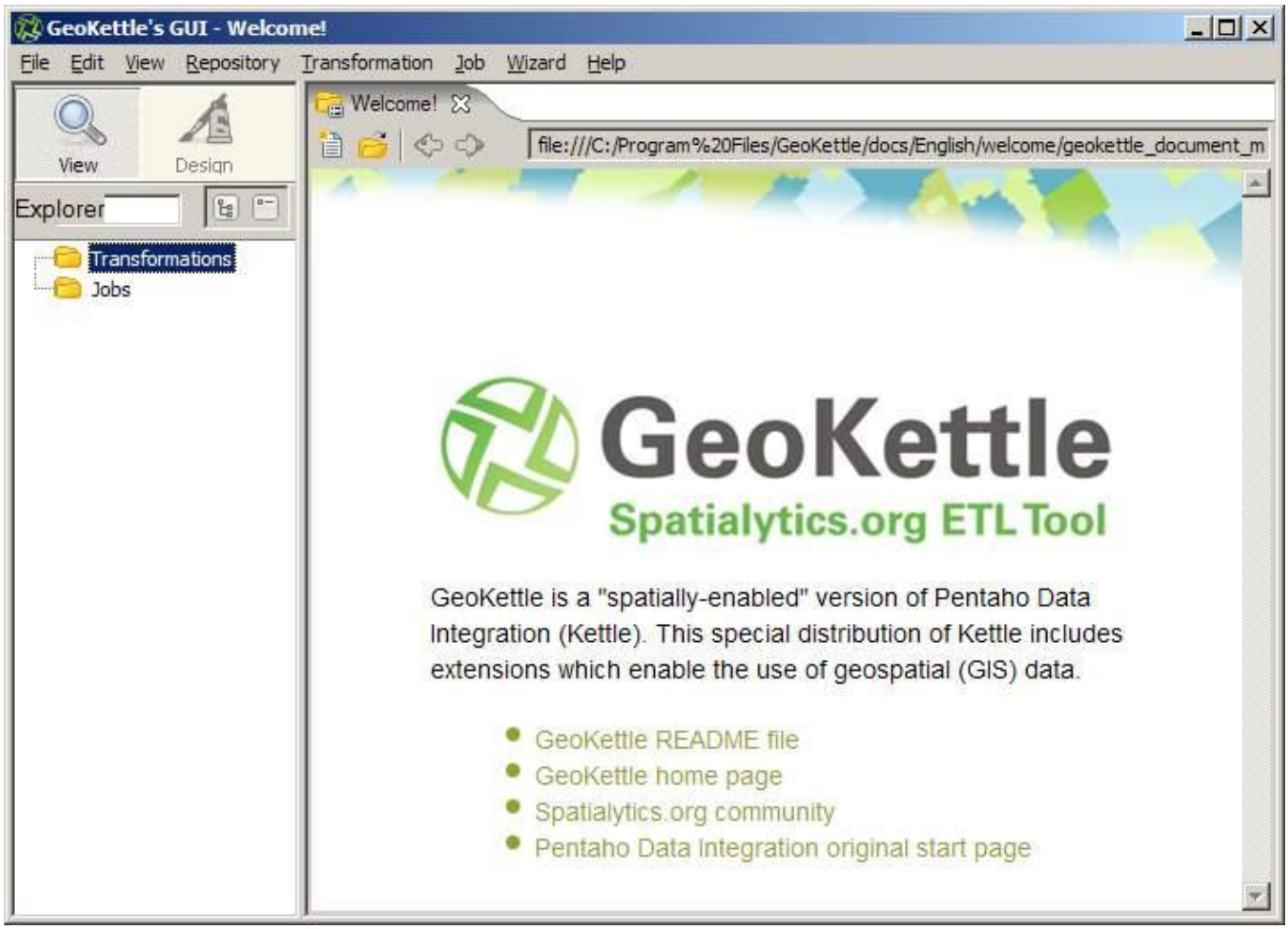

Рис. 1. Главное окно программы

В открывшемся главном окне программы (рис. 1) нужно создать документ. Возможны два типа — *Transformation* (преобразование) и *Job* (задача). Первый предназначен для работы с данными, а второй манипулирует файлами, запускает внешние скрипты, отправляет уведомления по почте. Transformation работает внутри Job.

Для одного файла только Transformation. Создадим новый документ (File|New|Transformation). Из списка в левой части перенесем блоки:

- Input|Shapefile File Input
- Transform|SRS Transformation
- Output|Table output

Обратите внимание, что в разделе Input доступны еще два "подходящих" варианта: **ESRI Shapefile Reader** и **OGR File Input**. Второй вариант не подходит потому, что не позволяет задать кодировку текстовых полей, а первый — просто не работает.

Соединим все три блока последовательно. Операция выполняется средней кнопкой мышки (колесо прокрутки) — нажать на первом блоке и протащить на второй.

Рис. 2. Цепочка операций.

Перейдем к настройке отдельных блоков. Двойным щелчок открываем блок **Shapefile File Input**, задаем имя файла и кодировку.

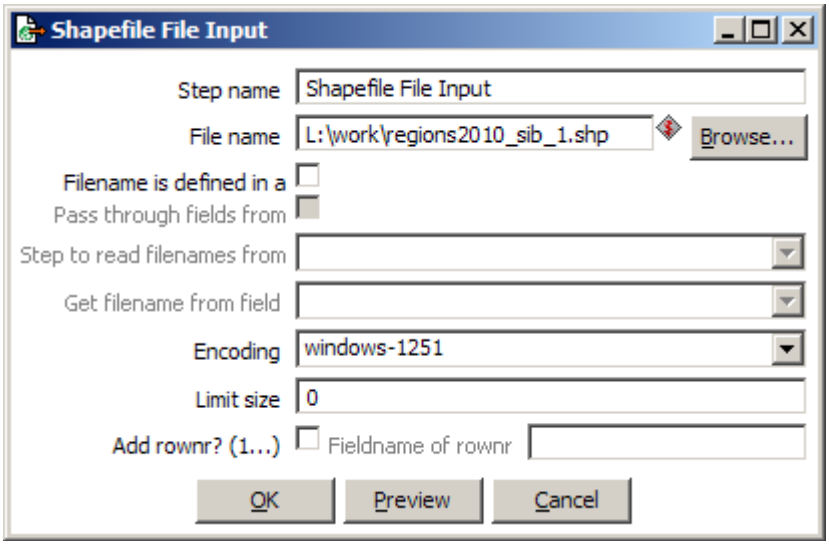

Рис. 3. Настройки импорта ESRI Shapefile

Рис. 4. Предварительный просмотр

По кнопке Preview можно посмотреть содержимое файла, его графическое представление и убедиться что кодировка выбрана правильно.

В блоке **SRS Transformation** укажем поле, содержащее геометрию объектов (the_geom), поставим галочку в поле *"Auto-detect spatial reference system from source"* для автоматического определения исходной проекции и в списке в правой части выберем требуемую проекцию (WGS 84).

Блок **Table output** предназначен для экспорта результата в базу данных. В графе *Connection* по кнопке *New...* создадим подключение к PostGIS. Выберем тип БД — PostgreSQL и укажем параметры подключения (рис. 6). Кнопка *Test* служит для проверки доступа к БД, ответ должен быть "[Test] is OK". Создание подключения закончено, возвращаемся в блок Table output. Осталось указать только название таблицы (*Target table*) — пусть будет "regions".

Попробуем запустить процесс. Для этого сохраняем документ (File|Save), выбираем в меню Transformation|Run, в открывшемся окне нажимаем кнопку *Launch*. Процесс завершится ошибкой в блоке **Table output** (красный значок на схеме, рис. 6).

Рис. 5. Ошибка при выполнении операции

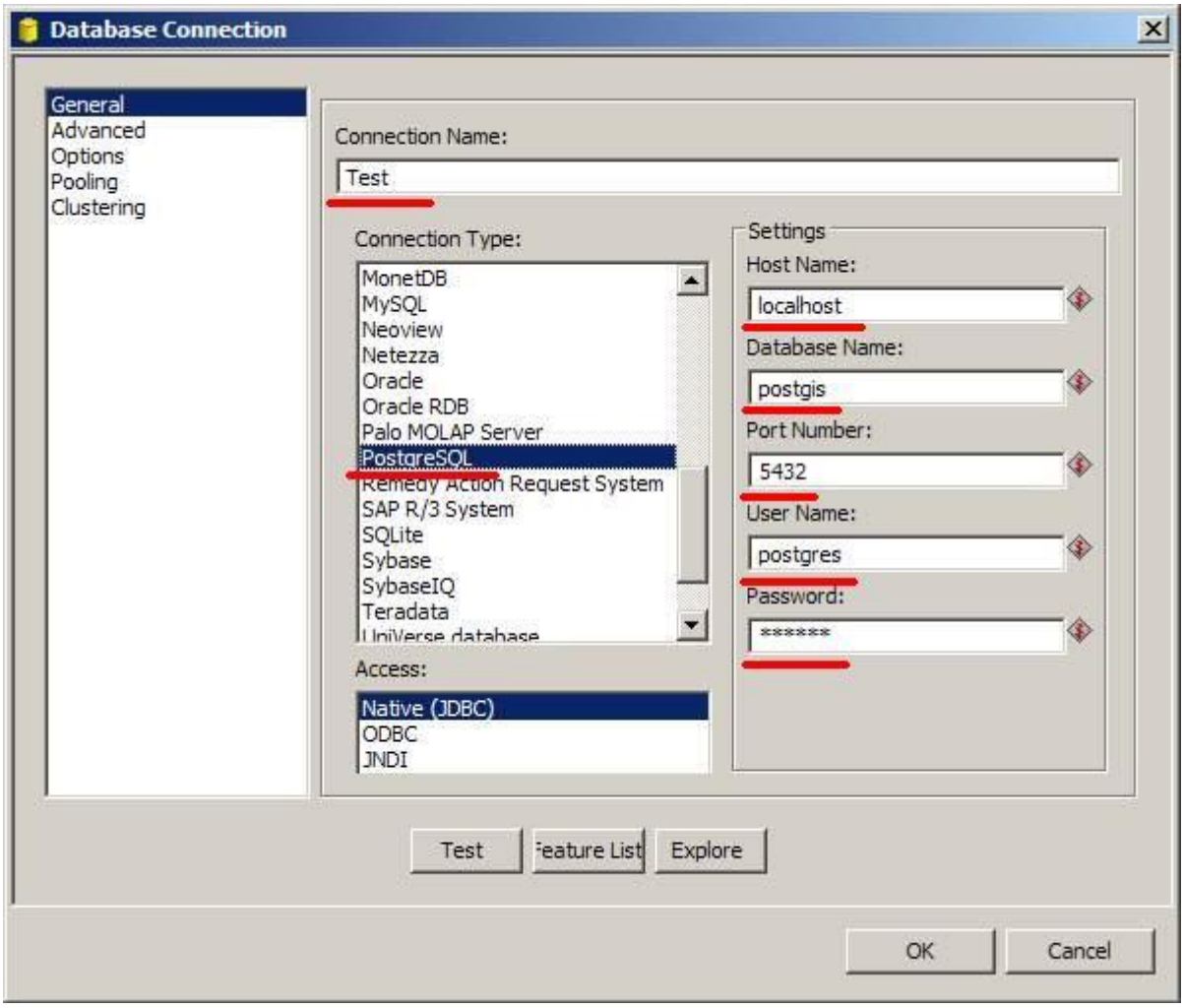

Рис. 6. Создание подключения к БД

Ошибка возникла потому, что в базе данных отсутствует таблица "regions" и ее требуется предварительно создать. Для этого опять откроем блок **Table output** и нажмем кнопку *SQL*. Предложенный sql-запрос можно сразу запустить (кнопка *Execute*) или изменить его и добавить в код создания таблицы дополнительные поля. Например, введем колонку первичного ключа (id).

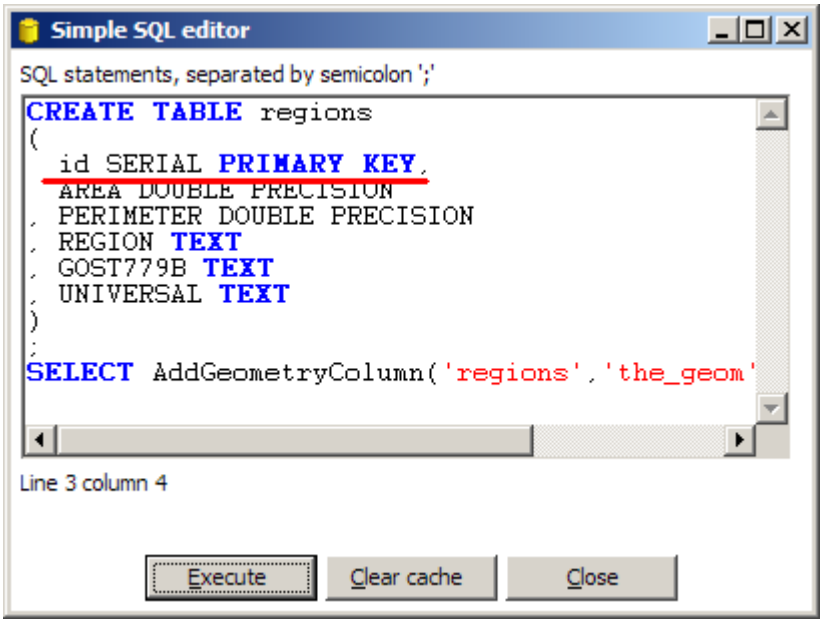

Рис. 7. Запрос SQL для создания таблицы

Если таблица с указанным именем в БД уже существует, то программа может предложить добавить поля (ALTER TABLE... ADD COLUMN) или изменить тип данных.

Повторим процедуру запуска трансформации, которая теперь пройдет без ошибок, и убедимся что в БД создана новая таблица, данные загружены и внесена запись в системную таблицу "geometry_columns".

Обработка множества файлов

Поставим задачу преобразования набора файлов для MapInfo из проекта GeoSample [6] в формат ESRI Shapefile. Для этого потребуется составить список файлов с раширением "*.tab" в каталоге и потом для каждого элемента списка выполнить процедуру считывания (ввода) и сохранения (вывода).

К статье приложен готовый проект, рекомендуется обращаться к нему для выяснения деталей настройки отдельных блоков.

1. Список файлов

Создадим новый документ типа *Transformation* и составим цепочку действий (рис. 8).

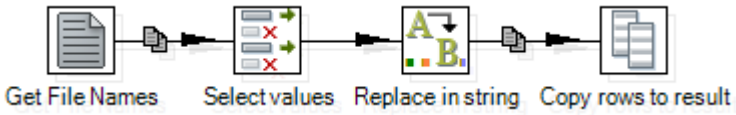

Рис. 8. Формирование списка файлов

Первый блок **Get File Names** (рис. 9) формирует список файлов в каталоге. В настройках блока можно указать как абсолютный путь, так и относительно расположения скрипта GeoKettle:

```
${Internal.Transformation.Filename.Directory}/tab
```

Список доступных переменных можно посмотреть нажав сочетание клавиш Ctrl+Пробел.

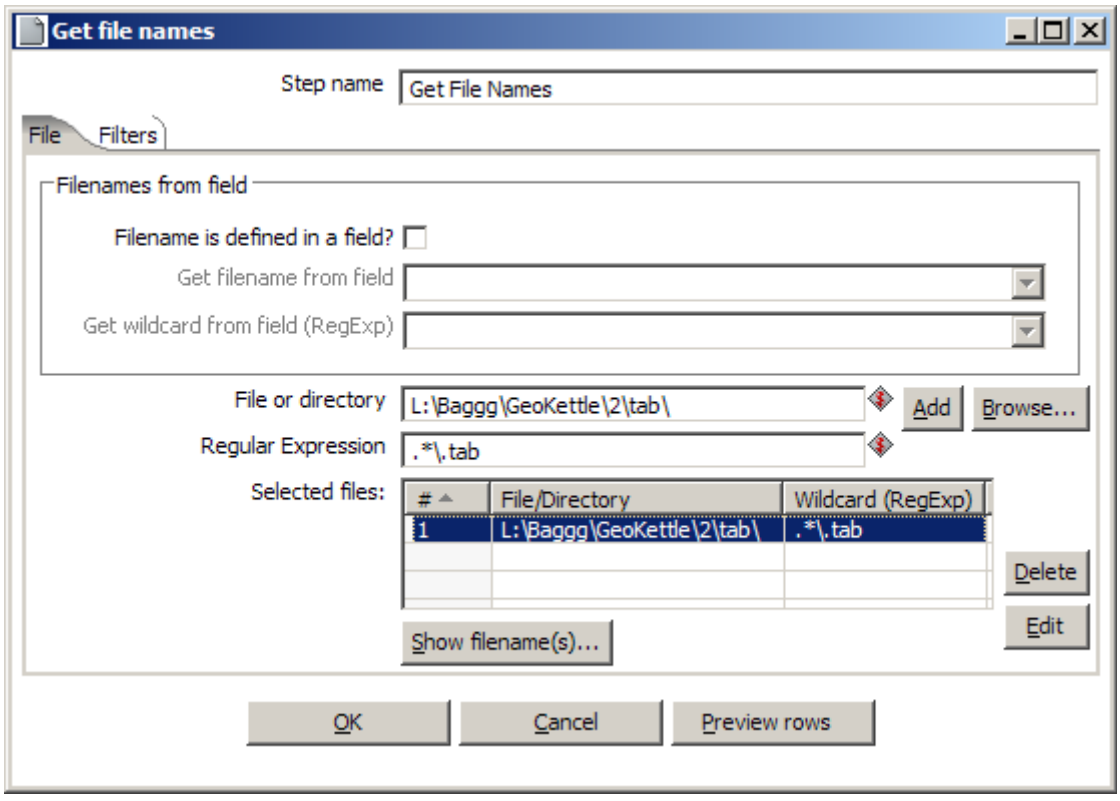

Рис. 9. Get File Names. Параметры настройки

Результатом работы **Get File Names** будет список, содержащий полное и сокращенное имена файлов, размер файла, атрибуты "скрытый" и "защита от записи" (read only) и др. Поэтому следующие блоки выбирают нужное поле (filename) в двух экземплярах и заменяют в одном из них "tab" на "shp".

Последний блок завершает процедуру и передает результат работы на следующий шаг — список, каждая строка которого содержит поля filenameTAB и filenameSHP.

2. Обработка отдельного файла

Следующим этапом идет преобразование формата MapInfo в ESRI Shapefile (рис. 10). Оно оформляется отдельно и главной задачей здесь стоит использование не абсолютных путей к файлам, а генерируемых на предыдущем шаге.

Рис. 10. Пример обработки отдельного файла

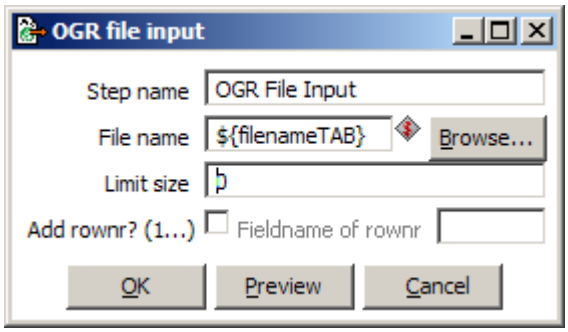

Рис. 11. Импорт файла

Для этого в графе "*File name"* в блоках **OGR file input** и **OGR file output** указываются переменные. **SRS Transformation** использован только как пример обработки файла и может быть заменен на любой другой или исключен из схемы.

3. Создание переменных

Переменные filenameTAB и filenameSHP формируются из полей списка файлов, полученного на первом шаге. Для этого служит блок **Job|Set Variables**. Особенность его работы заключается в том, что созданные переменные будут доступны только на следующем этапе и их нельзя использовать в том же документе *Tranformation*. Это ограничение происходит из условия **определенности** всех данных на момент запуска скрипта и используется для распараллеливания процесса.

Поэтому создается отдельный, третий по счету, документ типа *Tranformation* (рис. 12), который должен будет отработать до этапа "Обработка отдельного файла".

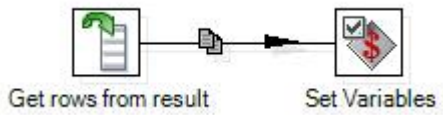

Рис. 12. Создание переменных

Примечание. В версии 2.0 появилась функция "Filename is defined in a field", которая позволила бы объединить этапы 2 и 3 в один. Но на текущий момент она реализована не во всех блоках ввода. Например, смотрите **Shapefile File Input**.

4. Конвейер

Осталось собрать все трансформации в единую схему. Создадим документ типа *Job* (рис. 13), добавим трансформацию "Список файлов" (filelist) и задачу "every file job", в параметрах которой

укажем *"Execute for every input row"*. Это означает, что задача "every file job" будет вызвана столько раз, сколько строк сформировано в списке файлов.

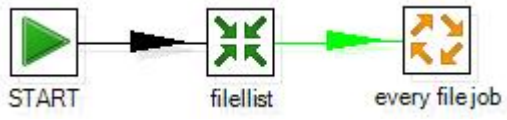

Рис. 13. Главный конвейер

Второй документ (every file job) типа *Job* (рис. 14) содержит трансформации этапов 2 и 3 в упомянутом порядке.

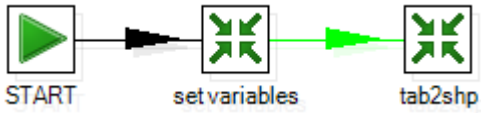

Рис. 14. Содержимое "every file job"

Запуск всей процедуры обработки файлов выполняется из документа, показанного на рис. 13 (файл "root.kjb" в приложении)..

Заключение

Программа GeoKettle обладает большими возможностями и высокой производительностью. Создание простых трансформаций является очень простой и наглядной операцией. Но работа со сложными схемами с вложенными задачами сильно запутана и представляет сложности при отладке с появлением неявных и плохо задокументированных ошибок. Тем не менее вторая версия программы идет по пути облегчения работы и добавляет более простые механизмы.

В комплект дистрибутива включены хорошие примеры работы с геометрией (используется диалект JavaScript для библиотеки JTS). Расположены в каталоге установленной программы "samples\transformations\geokettle\".

Аналоги:

- Talend Open Studio [7], лицензия GPL v2.
- FME [8] (Safe Software), коммерческое приложение.
- ArcGIS ModelBuilder [9], коммерческое приложение.
- Список программ в Wikipedia // http://en.wikipedia.org/wiki/Spatial_ETL#Spatial_ETL_tools

---

[1] Программа Kettle из пакета бизнес-анализа Pentaho Data Integration // http://kettle.pentaho.com/
[2] BeETLe Project // http://beetle-project.blogspot.com/
[3] Sextante project // http://www.sextantegis.com/
[4] Версия программы на момент написания статьи — http://sourceforge.net/projects/geokettle/files/geokettle-2.x/2.0-RC1/
[5] Дубинин М.Ю., Генерализованные слои границ субъектов РФ // http://gis-lab.info/qa/rusbounds-rosreestr-gen.html
[6] Geosample: Открытый набор геоданных для различного ПО ГИС // http://gis-lab.info/qa/geosample.html#get
[7] Talend Open Studio // http://www.talendforge.org/
Spatial module // http://www.talendforge.org/wiki/doku.php?id=sdi:MainPage
[8] FME Spatial Data Transformation Platform // http://safe.com/
[9] Дубинин М.Ю., Использование скриптов-посредников на Python в моделях ArcGIS // http://gis-lab.info/qa/mb-python.html

Mavka, 29.07.2011