

Глава 16. Автоматизация обновления решений в Power Query

Это продолжение перевода книги Кен Пульс и Мигель Эскобар. Язык M для Power Query. Главы не являются независимыми, поэтому рекомендую читать последовательно.

[Предыдущая глава](#) [Содержание](#) [Следующая глава](#)

По мере того, как вы создаете все больше и больше решений на основе Power Query, и начинаете понимать, сколько времени это экономит, вы захотите еще большей автоматизации. Да, вы можете просто щелкнуть правой кнопкой мыши таблицу, которая была импортирована с помощью Power Query, но даже это ощущается как... ручная операции. Нельзя ли запланировать обновление или, возможно, контролировать порядок обновления?

На самом деле вы можете использовать несколько различных методов автоматизации обновления решений Power Query:

- Обновление при открытии книги
- Обновление каждые X минут
- Обновить соединение по требованию с помощью VBA
- Обновить все соединения по требованию с помощью VBA
- Использовать расписание обновлений в сторонней надстройке

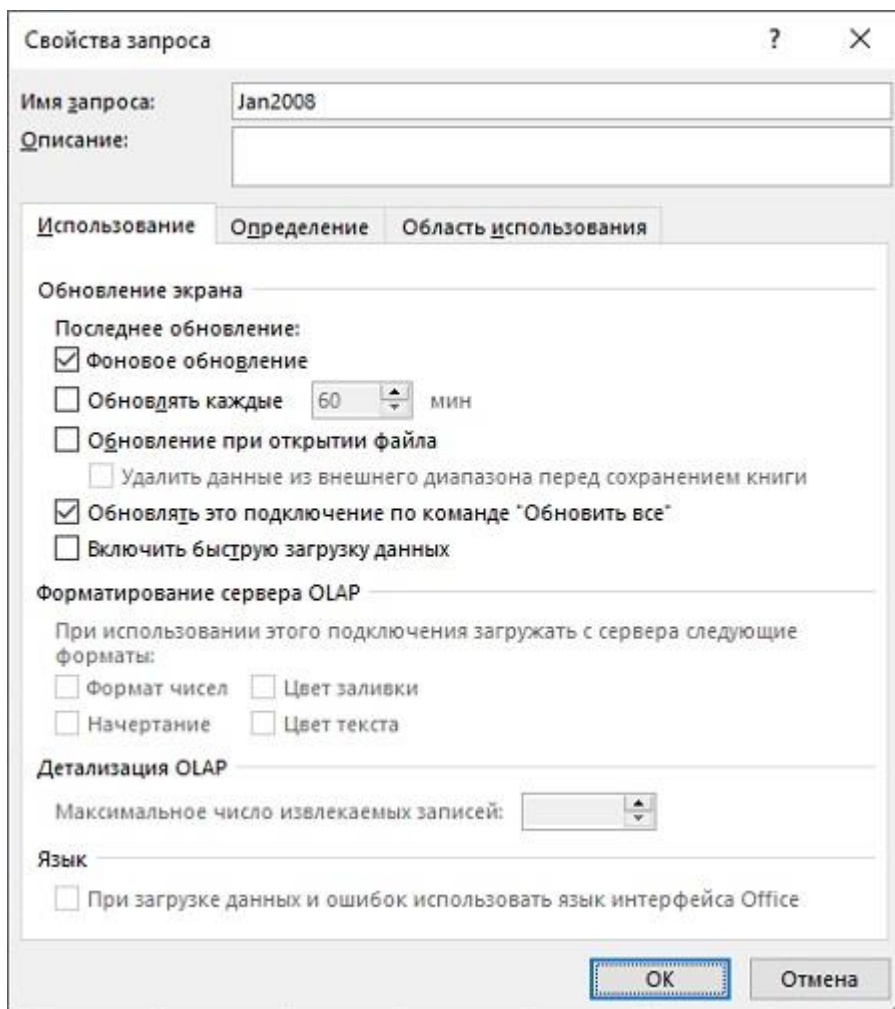


Рис. 16.1. Настройка параметров подключения

Планирование обновлений без использования кода

Первые два метода работают через пользовательский интерфейс и не требуют кода VBA. Они могут быть настроены на основе подключения, и вы даже можете автоматизировать обновление вплоть до Power Pivot. Откройте файл Automating Refresh.xlsx. Перейдите на вкладку *Данные* → *Запросы и подключения*. В правой части окна Excel откроется область *Запросы и подключения*. Кликните правой кнопкой мыши имя запроса → *Свойства*. Откроется окно *Свойства запроса* (рис. 1).

Обратите внимание, что флажок *Фоновое обновление* установлен по умолчанию. Этот параметр позволяет указать, следует ли продолжать работать в Excel во время обновления данных. Если вы снимите этот флажок, вы можете потенциально уменьшить время, необходимое для обновления вашего решения, но вы также заблокируете пользовательский интерфейс, что означает, что вы не сможете делать другие вещи, пока обновление запроса не будет завершено. Желательно оставить этот параметр включенным.

Следующий параметр – *Обновлять каждые XX минут*. Этот параметр очень пригодится, если вы извлекаете данные из веб-источника, который постоянно меняется, или если вы ориентируетесь на базу данных, которая регулярно обновляется. Для того, чтобы это обновление произошло книга Excel должна быть открыта. Допустимые значения для этого параметра – от 1 до 32 767.

Обновлять при открытии книги. Вторым параметром в этом разделе – *Удалять данные из внешнего диапазона перед сохранением книги*. Установите этот флажок, если вы хотите сохранить книгу с определением запроса, но без использования внешних данных. Снимите этот флажок, чтобы сохранить, и определение запроса, и данные. Этот флажок, будет доступен только после установления флажка *Обновлять при открытии книги*.

Автоматизация обновления запросов с помощью VBA

Приемы, описанные выше, и не использующие макросы, обходятся без каких-либо окон, предупреждающих о безопасности. Кроме того, такие книги легче переносить в Power BI, так как они не вызывают никаких проблем с блокировкой. Однако если вы работаете исключительно в настольном экземпляре Excel, иногда может потребоваться предоставить пользователю удобный и очевидный способ обновления решений Power Query. Это может быть сделано с помощью записи кода VBA.

Можно создать макрос для обновления одного подключения Power Query. В файле Automating Refresh.xlsx и перейдите на лист *Transactions*. На листе расположена *Таблица Transactions*, а также сводная таблица. Допустим, вы хотите создать макрос для обновления их обеих. Перейдите на вкладку *Разработчик*. Если вы не видите вкладку *Разработчик*, щелкните правой кнопкой мыши любую вкладку на ленте и выберите *Настроить ленту*. В правом окне установите флажок рядом с вкладкой *Разработчик* и нажмите *Ок*. На вкладке *Разработчик* нажмите кнопку *Запись макроса*. Также можно нажать иконку в левом нижнем углу окна Excel:

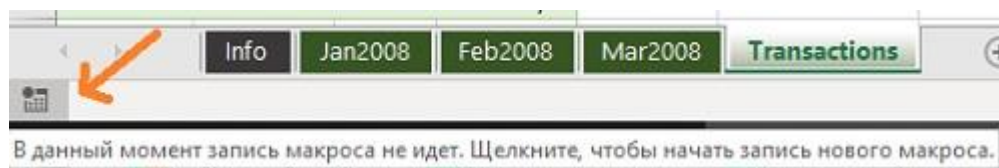


Рис. 16.2. Кнопка *Начать запись макроса*

Как только вы нажмете эту кнопку, Excel начнет записывать каждый щелчок листа, каждое нажатие клавиши и... каждую ошибку, которую вы делаете. Выполните в точности следующие действия. Дайте макрос *Refresh* и сохраните его в текущей книге. Пройдите по меню *Данные* → *Запросы и подключения* → В правой части окна Excel в области *Запросы и подключения* кликните правой кнопкой мыши на запросе *Transactions* → *Обновить*. Щелкните правой кнопкой мыши любую ячейку сводную таблицу → *Обновить*. Повторно кликните кнопку, как на рис. 16.2. Запись макроса будет остановлена. Проверьте код. Пройдите по меню *Вид* → *Макросы*:

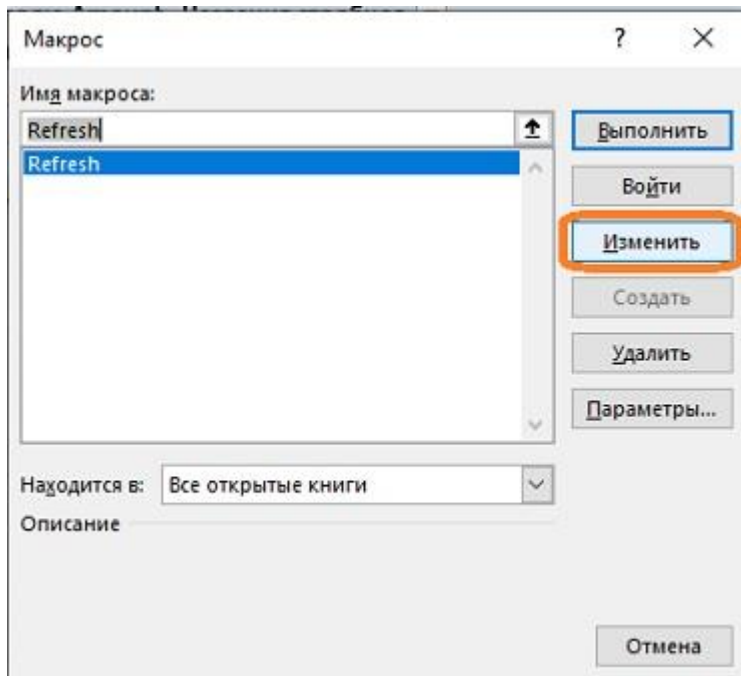


Рис. 16.3. Макросы

Нажмите *Изменить*. Откроется окно редактора VBA

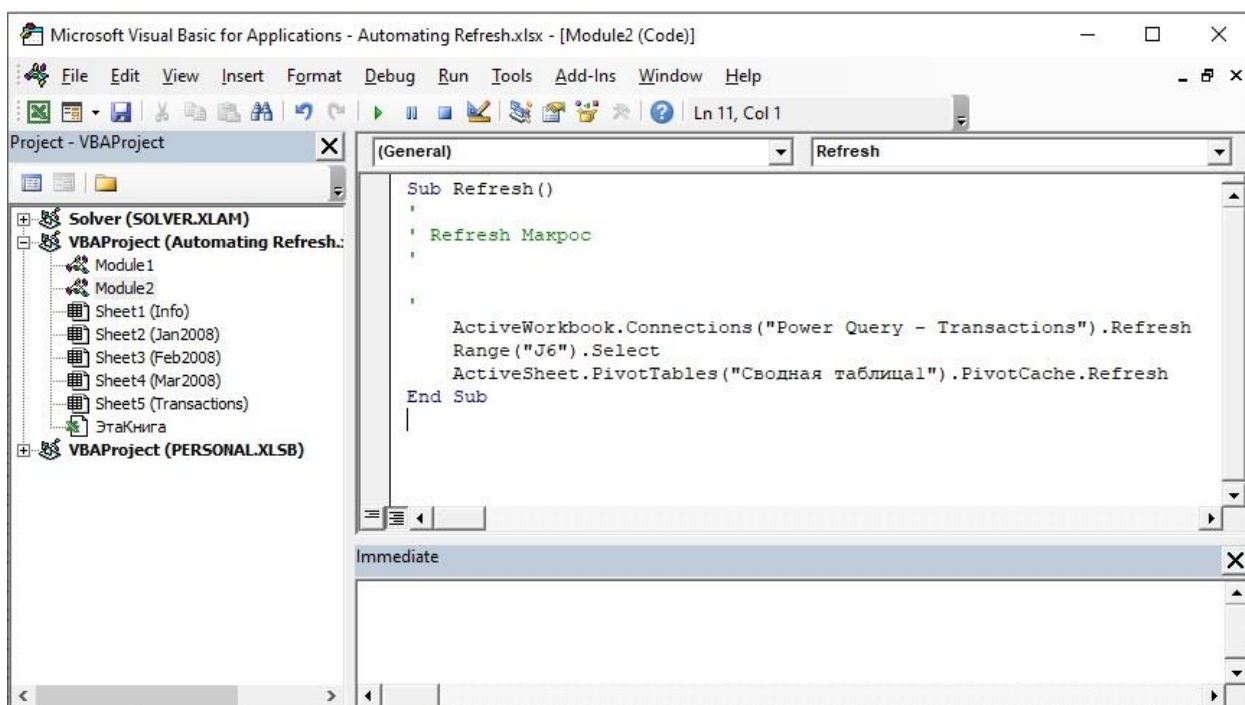


Рис. 16.4. Код VBA

Добавьте кнопку для обновления макроса. Перейдите на вкладку *Разработчик* → *Вставить* → *Кнопку*:

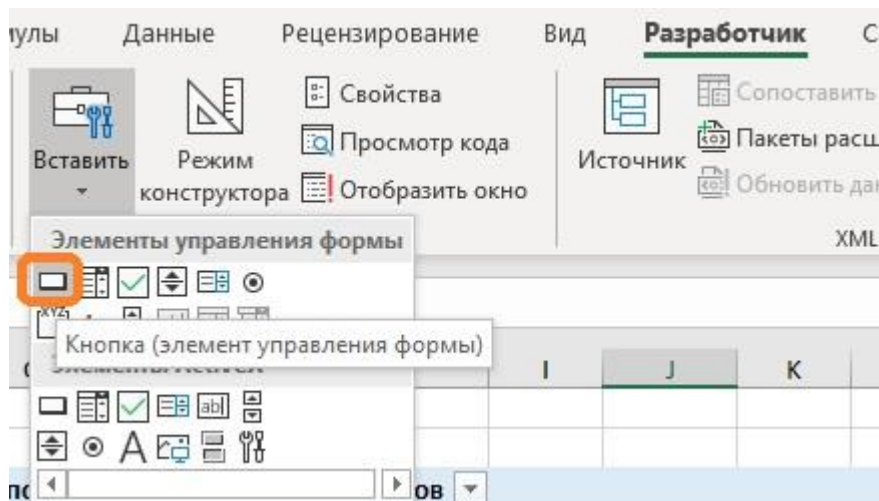


Рис. 16.5. Вставить элемент управления *Кнопку*

Выберите место на листе, и удерживая левую кнопку мыши, нарисуйте прямоугольник для Кнопки. Отпустите левую кнопку мыши. Появится окно с предложением поставить в соответствие Кнопке макрос. Выберите *Refresh*. Нажмите *Ок*. Щелкните правой кнопкой мыши по *Кнопке* → *Изменить текст*. Введите *Обновить*. Щелкните любую ячейку на листе, чтобы выйти из режима редактирования названия кнопки.

	D	E	F	G	H
	Amount				
50	-2603				
50	-5507			Сумма по полю Amount	Названия столбцов
50	-106			Названия строк	1
50	-29			январь	-893 5
50	-4802			февраль	-3 246 4
50	-135			март	-4 227 3
50	-7704			Общий итог	-8 367 4
50	-30595				
50	-9515				
50	879			Обновить	
20	-5674				
20	-4122				

Рис. 16.6. Кнопка, подключенная к макросу

Обновление нескольких запросов

Теперь можно добавить иные запросы в макрос. Пройдите по меню *Разработчик* → *Макросы* → *Refresh* → *Изменить*. На данный момент код имеет вид (см. также рис. 16.4):

```
Sub Refresh()
'
' Refresh Макрос
'
ActiveWorkbook.Connections("Power Query - Transactions").Refresh
Range("J6").Select
ActiveSheet.PivotTables("Сводная таблица1").PivotCache.Refresh
End Sub
```

Первые четыре строки после `Sub Refresh()` – это просто комментарии. Сейчас в них нет смысла, так что можно их удалить. Строка, начинающаяся с `ActiveWorkbook`, обновляет запрос. В следующей строке выбирается ячейка на активном листе. Предпоследняя строка обновляет сводную таблицу на активном листе.

Вы можете внести некоторые изменения в этот макрос, чтобы не только контролировать порядок обновления всех соединений, но и сделать код немного более надежным. (Сейчас код потерпел бы неудачу, если бы кто-то попытался запустить его из другого рабочего листа, так как на нем не было бы сводной таблицы.) Вот как должен выглядеть усовершенствованный код:

Sub Refresh()

```
ActiveWorkbook.Connections("Power Query - Jan2008").Refresh
ActiveWorkbook.Connections("Power Query - Feb2008").Refresh
ActiveWorkbook.Connections("Power Query - Mar2008").Refresh
ActiveWorkbook.Connections("Power Query - Transactions").Refresh
Worksheets("Transactions").PivotTables("PivotTable1")._
    PivotCache.Refresh
```

End Sub

Символы пробела и подчеркивания используются для обозначения разрыва строки в коде VBA. Код будет работать, если PivotCache.Refresh останется в той же строке (без переноса). Вот только длинные строки плохо размещать на картинках)) Использование Worksheets("Transactions") вместо ActiveSheet более универсально.

Имена подключений должны совпадать с именами, которые можно увидеть в диалоговом окне *Существующие подключения*:

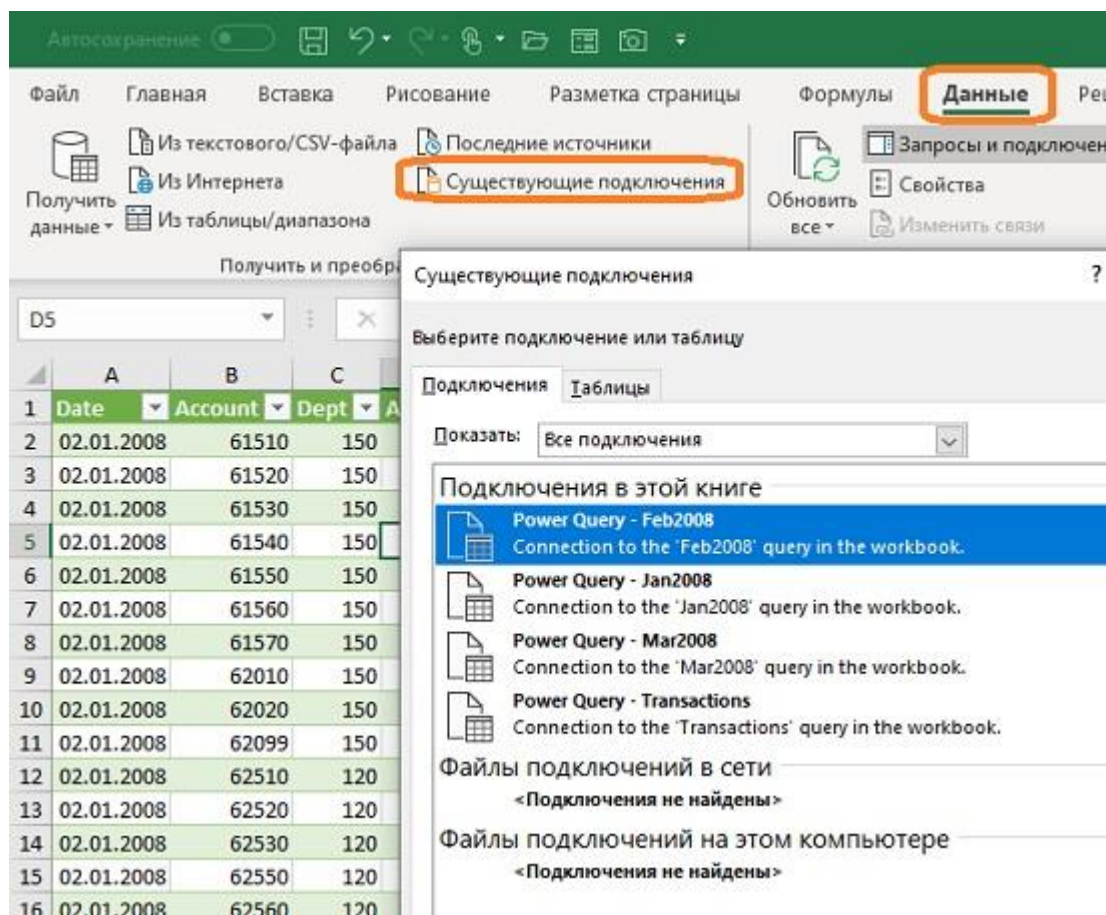


Рис. 16.7. *Существующие подключения*

Поскольку теперь книга включает макрос, вы не можете сохранить ее в формате XLSX. Вместо этого сохранить книгу в формате XLSM. При открытии книги пользователи получают предупреждение системы безопасности.

Обновление всех запросов

Чтобы обновить все запросы Power Query, необходимо использовать несколько иной код. Следующий макрос будет просматривать все подключения в книге и обновлять те из них, что созданы Power Query (и игнорировать остальные):

```
Public Sub UpdatePowerQueriesOnly()
```

```

Dim lTest As Long, cn As WorkbookConnection
On Error Resume Next
For Each cn In ThisWorkbook.Connections
    lTest = InStr(1, cn.OLEDBConnection.Connection, _
        "Provider=Microsoft.Mashup.OleDb.1")
    If Err.Number <> 0 Then
        Err.Clear
        Exit For
    End If
    If lTest > 0 Then cn.Refresh
Next cn
End Sub

```

Этот макрос может быть сохранен в той же книге, что и созданный в предыдущем разделе, или он может заменить предыдущий код (в этом случае вам нужно будет повторно связать кнопку с новым макросом).

Имейте в виду, что этот код не обязательно обновит запросы в том порядке, в котором они должны быть обновлены, так как Excel обновляет запросы в алфавитном порядке. Если порядок обновления важен, пройдите по меню в Данные → *Существующие подключения*. Откроется окно *Существующие подключения* (см. рис. 16.7). Кликните на подключении правой кнопкой мыши → *Изменить свойства подключения*. Откроется окно *Свойства запроса*. Последовательно измените имена запросов, например, на такие 01-Jan2008, 02-Feb2008, 03-Mar2008, 99-Transactions. Теперь запросы будут обновляться в нужном порядке.