**Совместная работа в Power Query**

Это фрагмент книги [Гил Равив. Power Query в Excel и Power BI: сбор, объединение и преобразование данных](https://baguzin.ru/wp/?p=24451).

[Предыдущий раздел](https://baguzin.ru/wp/?p=24620) [К содержанию](https://baguzin.ru/wp/?p=24451#content)  Следующий раздел

#### Локальные файлы, параметры и шаблоны

Одна из наиболее распространенных проблем, с которыми сталкиваются при совместном написании отчета в Excel, связана с ситуацией, когда источники данных являются локальными файлами Excel или текстовыми файлами. Можно проиллюстрировать эту проблему очень распространенной историей двух авторов отчета: Алисой и Бобом.

Загрузите рабочую книгу Алисы, C08E01 - Alice.xlsx. В качестве источника данных она использовала файл C08E01.xlsx. Когда Боб получает отчет Алисы, при попытке обновления он получает ошибку:



Рис. 1. Ошибка, связанная с неверным путем к источнику данных

Для устранения этой проблемы Боб выполняет несколько операций, а потом сохраняет данные Алисы, C08E01.xlsx, в своей локальной папке, C:\Users\Bob\Documents \C08\C08E01.xlsx. Он открывает рабочую книгу и для каждого из трех запросов (*Revenues, Categories и Colors*) Боб выполняет следующие действия:

* Выбирает шаг *Source* на панели *Примененные шаги*.
* В строке формул заменяет путь в строке с

= Excel.Workbook(File.Contents("C:\Users\Alice\Documents\C08\C08E01.xlsx"), null, true)

на:

= Excel.Workbook(File.Contents("C:\Users\Bob\Documents\C08\C08E01.xlsx"), null, true)

Это исправляет ошибку обновления. После этого Боб может приступить к работе над отчетом. Несколько часов спустя Боб посылает отчет назад Алисе, которая обнаруживает, что отчет не обновляется. Следуя инструкциям Боба, направленным по электронной почте, она изменяет путь в трех запросах.

Поскольку Алиса и Боб продолжают обмениваться отчетами, со временем они постепенно увеличивают число запросов, которые импортируют данные из локальных файлов. В конце концов всякий раз, когда им нужно работать над запросами и обмениваться рабочей книгой или отчетом, они вынуждены менять наименование пути в десятках запросов при переключении с одного на другой. Все становится более трудоемким по мере разбухания команды и присоединения к ней других разработчиков. И вот однажды они узнают, что имеется удобный способ для разрешения проблемы: применение параметра.

#### Применение параметра для имени пути

Power Query позволяет создавать параметры и включать их в М-формулы. Используя данный подход, можно располагать несколькими запросами, которые ссылаются на один параметр. Также можно проконтролировать значение одного параметра и тем самым повлиять на все запросы. В нашей истории Боб и Алиса могут применить параметр для наименования пути, это избавит их от всех проблем.

Загрузите исходные данные C08E01.xlsx и рабочую книгу Алисы, C08E01 - Alice.xlsx. Откройте файл C08E01 - Alice.xlsx. Пройдите *Данные* –> *Получить данные* –> *Запустить редактор запросов*. В окне редактора Power Query пройдите *Главная* –> *Управление параметрами* –> *Создать* *параметр*. Настройте окно *Управление параметрами*:



Рис. 2. Настройка окна *Управление параметрами*

Щелкните Ok.

На панели *Запросы* выбирайте по очереди запросы *Revenues*, *Colors* и *Categories* и для каждого запроса выполняйте следующие шаги. Выберите шаг *Source*. В строке формул замените:

= Excel.Workbook(File.Contents("C:\Users\Alice\Documents\C08\C08E01.xlsx"), null, true)

на

= Excel.Workbook(File.Contents(Path & "\C08E01.xlsx"), null, true)

Проверьте, что на панели предварительного просмотра нет ошибок. Можно загрузить файл решения C08E01 - Solution.xlsx

#### Применение параметров в Excel

В Excel существует возможность загружать таблицу параметров в Power Query из внешнего источника, например листа Excel или файла конфигурации. Подобные параметры могут определять некие пороговые значения на этапах преобразования данных, переключаться между несколькими предварительно определенными источниками данных (такими, как тестовая и боевая базы данных), а также ограничивать число строк или диапазон дат, которые необходимо извлекать из базы данных.

Откройте книгу C08E01 - Solution.xlsx. Щелкните *Новый лист* и переименуйте его в *Start Here*. В ячейку A1 введите Path. В ячейку A2 введите C:\Data\C08. Выберите ячейки A1 и A2 и нажмите Ctrl+T (англ.) для создания таблицы. В окне *Создать таблицу* поставьте галочку *Таблица с заголовками*. В области А1:А2 появится Таблица. Выберите ячейку A1 или ячейку A2 и на вкладке *Конструктор таблиц* в поле *Имя таблицы* введите *Parameters*:



Рис. 3. Таблица с параметрами

Путь располагается на месте первой ячейки таблицы *Parameters* и может легко изменяться соавторами книги без необходимости запуска Power Query и поиска соответствующего параметра на панели *Запросы*. Если поделиться такой рабочей книгой с соавторами, которым необходимо обновить рабочую книгу, можно упростить их работу; кроме того, инструкции, которые, возможно, понадобятся им (для изменения пути), будут намного проще.

Если необходимо передать несколько параметров, можно добавить столбцы, каждый с соответствующим именем параметра в строке 1 и его значениями в строке 2.

Выберите ячейку A1 или A2 и пройдите *Данные* –> *Из таблицы/диапазона*. После открытия редактора Power Query обратите внимание, что на панели *Запросы* появился новый запрос *Parameters*. Щелкните на нем правой кнопкой мыши и выберите *Ссылка*. Переименуйте новый запрос в *Path2*. (Пока еще присутствует старый параметр Path из предыдущего упражнения, но скоро он будет удален.)

Выберите запрос *Path2.* Щелкните правой кнопкой мыши на ячейке C:\Path\C08 и выберите *Детализация*. Таблица преобразуется в текст со значением пути.



Рис. 4. Окно предварительного просмотра после команды *Детализация*

На запрос *Path2* можно ссылаться для получения значения пути из любого места в ваших запросах. В строке формул вы увидите:

= Источник{0}[Path]

Поскольку идентификатор *Источник* идентичен *Parameters*, можно не делать два предыдущих шага, ссылаясь на следующий код всякий раз, когда необходимо получать доступ к значению *Path*:

= Parameters{0}[Path]

Если в будущем появятся новые параметры, можете просмотреть значение нового параметра, сославшись на имя его столбца вместо Path. Например, если имеется новый столбец для имени файла, в следующей строке можно получить доступ к новому параметру Filename:

= Parameters{0}[Filename]

Обновим все запросы, что позволит загрузить путь из запроса *Path2* вместо считывания его из собственного параметра запроса *Path*. Для каждого из запросов *Revenues*, *Colors* и *Categories* измените формулу на шаге *Source* с

= Excel.Workbook(File.Contents(Path & "\C08E01.xlsx"), null, true)

на

= Excel.Workbook(File.Contents(Path2 & "\C08E01.xlsx"), null, true)

#### Ошибка конфиденциальности

К сожалению, после этого шага для запросов Revenues, Colors и Categories появляется ошибка:



Рис. 5. Ошибка конфиденциальности

Для устранения ошибки измените параметры конфиденциальности этой книги. В редакторе Power Query пройдите *Файл* –> *Параметры и настройки* –> *Параметры запроса*. В окне диалога *Параметры запроса* перейдите в раздел *Текущая книга*, выберите на левой панели *Конфиденциальность*. Установите переключатель в положение *Игнорировать уровни конфиденциальности для возможного улучшения производительности*. Щелкните Ok.

Игнорирование уровней конфиденциальности должно выполняться с некоторой осторожностью. Если вы не доверяете владельцу книги или видите, что запросы связаны с неизвестными внешними источниками, не рекомендуется применять этот параметр. Однако в некоторых случаях Power Query не удается поддерживать обращение к нескольким источникам данных, тогда отображается ошибка Formula.Firewall. Ниже будет показана усовершенствованная методика, которая позволит устранять ошибки без необходимости игнорировать уровни конфиденциальности.

Удалите параметр *Path* и переименуйте *Path2* в *Path*. Переименование *Path2* автоматически изменяет все ссылки на новое имя запроса в других запросах. Выберите запрос *Path* пройдите Главная –> Закрыть и загрузить в… Оставьте активной опцию *Только создать подключение*. Это гарантирует, что запрос Path не будет загружен на лист Excel.

#### Метод, преодолевающий ошибку конфиденциальности

Загрузите книгу C08E03 - Solution.xlsx. При открытии редактора PQ вы столкнетесь с ошибкой Formula.Firewall (если только не установлено игнорирование уровней конфиденциальности для всех рабочих книг). Ошибки Formula.Firewall можно избежать, перестроив запросы, чтобы все ссылки на внешние источники выполнялись в одном запросе.

В редакторе Power Query пройдите *Просмотр* –> *Зависимости запроса*. Рассматривая картинку и изучая код в строках *Source* различных запросов, можно увидеть, что каждый из запросов *Revenues, Colors и Categories* зависит как от текущей книги, так и от книги c:\data\c08\c08e01.xslx. Но если книга c08e01.xslx ссылается непосредственно на запросы *Revenues, Colors и Categories*, то текущая книга не ссылается непосредственно на эти запросы, а реализует это посредством параметров *Path* и *Parameters*.



Рис. 6. Зависимости запроса

Объединяя в одном запросе ссылки на зависимые внешние источники, можно избежать ошибки. В этом примере консолидация произойдет, если каждый из запросов *Revenues, Colors и Categories* непосредственно ссылается как на текущую рабочую книгу, так и на источник данных c:\data\c08\c08e01.xslx. Чтобы это реализовать, выберите каждый из запросов *Revenues, Colors, и Categories* и в расширенном редакторе добавьте следующую строку кода после слова let:

Path = Excel.CurrentWorkbook(){[Name="Parameters"]}[Content]{0}[Path],

Эта строка объединит в одну строку все шаги преобразования запроса *Parameters*. Теперь можно удалить запрос *Parameters*, поскольку на него не ссылаются ни в одном из запросов.

Еще один подход заключается в создании нового запроса, который возвращает двоичный файл рабочей книги, и тогда запросы *Revenues, Colors и Categories* ссылаются на запрос рабочей книги. Затем в запросе книги можно получить ссылки на внешний файл Excel и текущую книгу. Для реализации описанного подхода откройте файл C08E03 - Solution.xlsx и пройдите *Данные* –> *Получить данные* –> *Запустить редактор Power Query*.

Проверьте, что в окне *Параметры запроса* в разделе *Текущая книга* для уровней конфиденциальности выбран переключатель *Совмещать данные в соответствии с настройками уровней производительности для каждого источника*.

На панели *Запросы* щелкните правой кнопкой мыши на *Colors* и выберите *Дублировать*. Новый запрос переименуйте в *Workbook*, удалите все шаги на панели *Примененные шаги*, кроме первого шага *Source*. Пройдите *Главная* –> *Расширенный редактор* и добавьте код:

Path = Excel.CurrentWorkbook(){[Name="Parameters"]}[Content]{0}[Path],

между строками

let

и

Source = Excel.Workbook(File.Contents(Path & "\C08E01.xlsx"), null, true)

Щелкните Готово. Теперь запрос *Workbook* возвращает содержимое таблицы книги C08E01.xlsx.

На панели *Запросы* выберите последовательно запросы *Revenues, Colors и Categories*. Для каждого из них выберите шаг *Source* и измените код в строке формул на:

= Workbook

Удалите запросы *Path* и *Parameters*. Выделите запрос *Workbook*, пройдите Главная –> Закрыть и загрузить… Установите переключатель *Только создать подключение*, чтобы не загружать запрос *Workbook* на лист Excel.

Выполнив эти шаги, вы объединили все ссылки на внешние источники в запросе *Workbook* и избежали ошибки Formula.Firewall. Вернитесь в редактор PQ и пройдите *Просмотр* –> *Зависимости запроса*:



Рис. 7. Зависимости запроса изменились

Теперь только запрос *Workbook* имеет доступ к двум внешним источникам данных. Можно загрузить книгу решения C08E03 - Solution 2.xlsx. Благодаря предпринятым мерам всем пользователям файла не потребуется изменять или отключать собственные уровни конфиденциальности.

Изменение ваших запросов для предотвращения ошибок брандмауэра — не всегда простая задача. При объединении нескольких запросов в один теряется некоторая гибкость в плане поддержки и повторного использования запросов. Следует оценить все за и против, прежде чем приступать к консолидации. Как показано в следующем разделе, иногда имеются лучшие решения для полного исключения необходимости обращения к двум источникам данных.

При совместном использовании более гибким является использование Power BI. Служба Power BI может подключаться к локальным файлам (или к любому иному локальному источнику данных) с помощью локальных шлюзов данных. Подробнее см. <https://powerbi.microsoft.com/ru-ru/gateway/>

Еще более интересным представляется размещение общих файлов на OneDrive for Business или на сайте SharePoint Online. Поскольку для выполнения упражнений понадобится учетная запись OneDrive for Business здесь эта часть книги опущена.

#### Вопросы безопасности

Power Query разработан с учетом мер безопасности. Если создается отчет в Power Query, который подключается к базе данных организации или любому другому набору данных, требующему аутентификации, ваши учетные данные хранятся на вашем компьютере, тогда можно обновлять отчет по требованию и не вводить учетные данные для каждого обновления. Если файл отчета передается другим пользователям, учетные данные остаются на вашем компьютере. Ваши соавторы должны предоставить свои собственные учетные данные для подключения к соответствующему источнику данных. Этот подход защищает ваши учетные данные и гарантирует, что посторонний автор отчета не сможет получить доступ к вашим учетным данным.

В некоторых случаях можно поделиться запросами, моделью и визуализациями, но убедитесь, что данные в отчете очищены перед тем, как предоставить отчет другим пользователям. Для сохранения книги Excel в качестве шаблона и удаления данных из модели или таблиц пройдите Файл –> Сохранить как. Выберите тип файла Шаблон Excel (\*.xltx). Перед сохранением файла отобразится предупреждение:



Рис. 8. Предупреждение при сохранении шаблона

Щелкните Да для удаления данных. Если ваш коллега откроет отчет, то Power Query запустит обновление и предложит пользователю предоставить свои учетные данные, если это необходимо. Ваши данные не будут сохраняться в отчете.

#### Удаление всех запросов в Excel с помощью инспектора документов

В некоторых случаях может возникнуть необходимость поделиться преобразованными данными или статическими диаграммами в рабочих листах, но удалить запросы и модель из рабочей книги. И хотя можно вручную удалить каждый запрос и сохранить без изменений отдельные таблицы, имеется более быстрый способ. Для мгновенного удаления всех запросов в вашей книге можно воспользоваться Инспектором документов.

В файле Excel пройдите *Файл* –> *Сведения* –> *Найти проблем* –> *Проверить документ*. В окне *Инспектор документов* кликните *Проверить.* После проверки прокрутите вниз до раздела *Настраиваемые XML-данные*. Именно здесь Power Query сохраняет запросы. Эта область в файле также используется сторонними надстройками. Поэтому в качестве меры предосторожности всегда следует запускать *Инспектор документов* с копией рабочей книги на тот случай, если в файле хранятся дополнительные данные для надстроек, которые будут удалены Инспектором вместе с запросами. Щелкните *Удалить все* рядом со строкой *Настраиваемые XML-данные*, затем щелкните *Закрыть*.

На вкладке *Данные* выберите команду *Запросы и подключения*. Все запросы удалены. Если вы загрузили запросы в модель данных, они также будут удалены.