**Обнаруженные проблемы**

* **Диалоговые окна при экспорте**. При запуске Dynamo-скрипта через Batch Revit Processor (RBP) для печати листов в PDF всплывают системные диалоги, требующие вмешательства. В частности, появляются окна **«Save File»** (сохранение PDF-файла) и **«Confirm Save Location»** (подтверждение пути сохранения) от Revit. Эти диалоги блокируют автоматическое выполнение: например, RBP запрашивает «Do you want to save changes to XXX.rvt?» («Сохранить изменения в файле XXX.rvt?»), и пользователю приходится вручную нажимать «No» для каждого файла [forum.dynamobim.com](https://forum.dynamobim.com/t/revit-batch-processor-save-file-dialog-box/62945#:~:text=I%20want%20to%20batch%20export,I%20didn%E2%80%99t%20do%20any%20modification). Такие всплывающие окна не позволяют выполнить экспорт полностью без участия пользователя, нарушая требование автоматизации.
* **Несохранение файла и зависание процесса**. Batch Revit Processor по умолчанию пытается автоматически закрыть файл Revit без сохранения, однако при закрытии Revit всё равно выводит запрос на сохранение изменений. В англоязычной версии RBP автоматически нажимает «No» (не сохранять), но в русской локализации этого не происходит[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=scripts%20do%20not%20suffer%20this,It%20is). Дело в том, что **RBP распознаёт и обрабатывает диалоговые окна только на английском языке** [github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=scripts%20do%20not%20suffer%20this,It%20is). Если Revit или Windows используются на русском, RBP не «узнает» кнопки «Да/Нет» на кириллице. Из-за этого пакетный процесс подвисает в ожидании закрытия диалога, и появляются ошибки, связанные с невозможностью нажать локализованные кнопки.
* **Раздельные PDF вместо одного файла**. Текущая конфигурация экспорта, судя по описанию, не объединяет листы в один PDF-файл. Пользователь, вероятно, получает отдельные PDF для каждого листа или неполный объединённый файл. Это противоречит требованию №2 (Combine = true). Например, ранее при попытке объединения через Dynamo без правильных настроек PDFExportOptions многие сталкивались с тем, что PDF-файлы перезаписывались или создавался только первый лист [forum.dynamobim.com](https://forum.dynamobim.com/t/dynamo-exporting-pdf-by-adobe-pdf/18706#:~:text=Hi%2C%20The%20option%20%E2%80%9CCombinedFile%E2%80%9D%20does,with%20boolean%20False%20on%20%E2%80%9CCombinedfile%E2%80%9D). Это указывает на неверную настройку опций экспорта PDF: если свойство **Combine** не включено или не задано имя выходного файла, Revit не сможет сохранить все листы в единый PDF.
* **Некорректная конфигурация PDFExportOptions**. Даже используя встроенный экспорт PDF (появившийся в Revit 2022+), нужно правильно задать параметры. Если этого не сделать, автоматический экспорт может прерываться. В частности, для объединения нескольких видов/листов в одном файле **требуется явно включить опцию Combine и задать имя итогового PDF**. По умолчанию же имя файла не задано (пустое), когда Combine=true [revitapidocs.com](https://www.revitapidocs.com/2023/f5d51aa8-71ae-0526-1668-e0d97eb8315e.htm#:~:text=Since%20the%20%20,PDF%20file). Если оставить его пустым, API экспорта выдаст ошибку или вызовет диалог. Неполная настройка PDFExportOptions могла быть причиной, почему скрипт останавливался с запросом пути сохранения или генерировал несколько файлов вместо одного.

**Решения**

Чтобы обеспечить полностью автоматический экспорт листов в единый PDF без ошибок, необходимо внести следующие коррективы:

1. **Настройка PDFExportOptions для объединённого PDF**. В Dynamo-скрипте (например, в Python-ноде) нужно создать и настроить объект PDFExportOptions перед вызовом экспорта. Основные настройки:
   * Включить режим объединения: opts.Combine = True. Это укажет Revit экспортировать все указанные листы в один файл[forum.dynamobim.com](https://forum.dynamobim.com/t/merge-combine-pdf-export/70789#:~:text=They%20need%20to%20go%20before,that%20you%20want%20to%20set).
   * Задать имя итогового файла: opts.FileName = "ИмяФайла.pdf". Название можно сформировать по шаблону (например, проект + дата). Важно указать строку с расширением .pdf. Согласно документации, если Combine=true, Revit использует свойство FileName для названия единого PDF-файла[revitapidocs.com](https://www.revitapidocs.com/2026/65f97585-8c92-b52e-93dd-8a6b4bfc5a1a.htm#:~:text=If%20%20%20%20%2C,NamingRule%5D). По умолчанию FileName пустой, поэтому его **необходимо установить вручную перед вызовом Document.Export**, иначе экспорт «в один файл» не состоится[revitapidocs.com](https://www.revitapidocs.com/2023/f5d51aa8-71ae-0526-1668-e0d97eb8315e.htm#:~:text=Since%20the%20%20,PDF%20file).
   * При необходимости установить другие свойства качества и графики (разрешение DPI, цветность и т.д.), но они не влияют на объединение. Главное – Combine и FileName.
   * Пример кода настройки (как советуют на форуме Dynamo):

python

КопироватьРедактировать

opts = PDFExportOptions()

opts.Combine = True

opts.FileName = "Combined.pdf"

Этот код должен выполняться **до** вызова метода экспорта, чтобы опции применились[forum.dynamobim.com](https://forum.dynamobim.com/t/merge-combine-pdf-export/70789#:~:text=They%20need%20to%20go%20before,that%20you%20want%20to%20set).

* + Вызывать сам экспорт через API Revit: doc.Export(folderPath, viewIdList, opts), где folderPath – существующая папка для сохранения. Путь должен существовать, иначе экспорт выдаст ошибку пути[revitapidocs.com](https://www.revitapidocs.com/2023/93d66d57-c20e-a103-39a1-77bc2ea05183.htm#:~:text=folder%20Type%3A%20%20System%20String,The%20folder%20must%20exist). viewIdList – список ElementId выбранных листов (или видов), полученных в Dynamo.
  + Такой подход использует встроенный механизм печати PDF Revit 2022+ и не вызывает диалогов сохранения файла, т.к. путь и имя уже заданы программно. Все листы сохранятся сразу в один PDF-файл с заданным именем в указанной папке.

1. **Избежание диалогов "Save File" при печати**. Если ранее скрипт полагался на виртуальный PDF-принтер (Adobe PDF, CutePDF и т.п.), то именно драйвер принтера вызывал окно "Save As". Решение – перейти на вышеописанный прямой экспорт PDF через API Revit. Он полностью тихий (silent) и не открывает GUI. При использовании Document.Export никаких диалогов "Save PDF" не появляется – процесс происходит в фоне[revitapidocs.com](https://www.revitapidocs.com/2023/93d66d57-c20e-a103-39a1-77bc2ea05183.htm#:~:text=folder%20Type%3A%20%20System%20String,The%20folder%20must%20exist).

*Для сценариев, где нужно использовать печать на PDF-принтер (например, в старых версиях Revit без PDFExport):* можно программно подавить диалог, указав свойства печати:

python

КопироватьРедактировать

printMgr = doc.PrintManager

printMgr.SelectNewPrintDriver("Ваш PDF принтер")

printMgr.PrintToFile = True

printMgr.PrintToFileName = r"C:\Folder\Output.pdf"

Таким образом, Revit не спросит путь (он уже задан). Однако при этом сложнее добиться объединения нескольких листов в один PDF – это зависит от возможностей драйвера. Поэтому предпочтителен вариант с PDFExportOptions при наличии Revit 2022+.

1. **Предотвращение диалога "Confirm Save Location" / запроса сохранения изменений**. Этот диалог возникает, когда Revit считает, что модель изменилась, либо при работе с отсоединёнными файлами. Решения:
   * **Открытие без сохранения**. Убедиться, что RBP открывает модели в режиме, не требующем сохранения. Для проектов с рабочими наборами (central files) в настройках Batch Revit Processor следует включить опцию **«Detach from Central»** (отсоединить от центра)[forum.dynamobim.com](https://forum.dynamobim.com/t/revit-batch-processor-save-file-dialog-box/62945#:~:text=Hi%20Daniel%2C). RBP тогда будет открывать копию модели, не привязанную к центральному файлу, и не станет пытаться её сохранить обратно. Это предотвратит диалог о сохранении в центральный файл. По умолчанию RBP как раз использует Detach для центральных файлов, с опцией **Close All Worksets** (не загружать рабочие наборы)[forum.dynamobim.com](https://forum.dynamobim.com/t/revit-batch-processor-save-file-dialog-box/62945#:~:text=I%20thought%20that%20settings%20are,from%20Central%2C%20Close%20All%20Worksets).
   * **Отказ от сохранения изменений**. Удостоверьтесь, что скрипт не вносит изменений в документ Revit. Простое получение видов и экспорт в PDF обычно не помечает документ как “Modified”. Если модификаций нет, Revit не должен запрашивать сохранение. Однако на практике даже открытие файла или печать могут установить флаг изменений. В RBP нет отдельной настройки “не сохранять”, но инструмент пытается автоматически нажать "No" при запросе сохранения. Чтобы это сработало в наших условиях, можно **запустить Revit в английском интерфейсе**. Если установить для RBP/Revit англоязычные настройки (например, добавить ключ запуска "/language ENU" при наличии англоязычного Language Pack), то при закрытии файла RBP распознает диалог "Do you want to save?" и нажмет кнопку "No" автоматически[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=scripts%20do%20not%20suffer%20this,It%20is). Это обходной путь, позволяющий обойти проблему с кириллицей, не требуя пользовательского клика.
   * **Принудительное закрытие через скрипт**. Альтернативно, можно доработать Dynamo-скрипт, добавив финальный шаг: программно закрыть документ без сохранения. В Python-скрипте (если бы задача выполнялась на Python) можно вызвать doc.Close(False). В Dynamo же нет прямого узла для закрытия, но можно попробовать использовать Zero-Touch-ноды или сторонние пакеты (например, Orchid) для выполнения Document.Close. Однако закрывать файл прямо из Dynamo может быть рискованно, так как RBP ожидает закрыть файл самостоятельно после выполнения скрипта. Этот метод следует применять с осторожностью.
   * **Обработка резервных копий**. Диалог "Confirm Save Location" часто появляется при открытии резервной копии Revit (файл с расширением \*.0001.rvt и т.п.) и попытке сохранить его как основной файл. Решение – избегать включения backup-файлов в список на обработку. Если необходимо экспортировать из резервной копии, лучше заранее сохранить её как обычный .rvt, чтобы RBP работал с нормальным файлом и не вызывал этот диалог.
2. **Учёт языковых ограничений RBP**. Как отмечалось, Revit Batch Processor на данный момент обрабатывает только англоязычные диалоговые окна[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=scripts%20do%20not%20suffer%20this,It%20is). Если полностью устранить появление окон не удаётся, стоит обеспечить англоязычные надписи:
   * **Запуск Revit на английском**. Если среда позволяет, установить англ. язык для сеанса Revit. В результате кнопки диалога будут "Yes/No" вместо "Да/Нет", и RBP автоматически нажмёт "No" и продолжит работу без зависания[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=scripts%20do%20not%20suffer%20this,It%20is).
   * Либо, как временное решение, сменить язык интерфейса Windows для диалогов (но в случае Revit это привязано к языку программы). В общем, любые меры, позволяющие получить диалоги на английском, увеличат надёжность пакетной обработки.
   * На будущее можно предложить разработчикам RBP расширить поддержку локализаций или добавить в конфигурацию список строк для авто-нажатия на других языках.

**Изменения в коде**

На основании вышеописанного, рекомендуется скорректировать Dynamo-скрипт (или соответствующий Python-код), чтобы реализовать решения:

* **Использование API экспорта PDF**. Заменить существующую логику печати на вызов метода Document.Export с параметрами для PDF. В Dynamo это можно сделать через Python-ноду. Ниже приведён шаблон кода:

python

КопироватьРедактировать

import clr

clr.AddReference('RevitAPI')

from Autodesk.Revit.DB import PDFExportOptions

# Получаем документ и нужные листы (ElementId листов)

doc = DocumentManager.Instance.CurrentDBDocument # требует RevitServices

view\_ids = List[ElementId](лист\_с\_идентификаторами) # список ID листов для экспорта

# Настраиваем опции PDF

opts = PDFExportOptions()

opts.Combine = True # объединить все листы в один PDF

opts.FileName = "Export.pdf" # имя итогового файла (без пути)

# Запускаем экспорт в указанный каталог

output\_folder = r"C:\Temp\PDF" # пример пути папки

doc.Export(output\_folder, view\_ids, opts)

В этом фрагменте показано ключевое – установка Combine = True и имени файла перед экспортом. Как указано в документации, именно эти параметры ответственны за объединённый вывод[revitapidocs.com](https://www.revitapidocs.com/2026/65f97585-8c92-b52e-93dd-8a6b4bfc5a1a.htm#:~:text=If%20%20%20%20%2C,NamingRule%5D)[revitapidocs.com](https://www.revitapidocs.com/2023/f5d51aa8-71ae-0526-1668-e0d97eb8315e.htm#:~:text=Since%20the%20%20,PDF%20file). Убедитесь, что путь output\_folder существует на диске[revitapidocs.com](https://www.revitapidocs.com/2023/93d66d57-c20e-a103-39a1-77bc2ea05183.htm#:~:text=folder%20Type%3A%20%20System%20String,The%20folder%20must%20exist); при необходимости можно создать папку заранее средствами Python (модуль os). После выполнения doc.Export(...) Revit создаст PDF-файл Export.pdf в папке C:\Temp\PDF, содержащий все листы из списка view\_ids.

* **Удаление вызовов, ведущих к диалогам**. Если в скрипте были узлы/код, которые инициировали диалоговые окна, их нужно убрать или заменить. Например:
  + Исключить использование PrintManager без задания PrintToFileName (чтобы не появлялся "Save File"). При переходе на Document.Export этот пункт решается автоматически.
  + Не вызывать Document.SaveAs без необходимости – это могло приводить к диалогу подтверждения пути для отсоединённого файла. В контексте экспорта PDF сохранять проект не требуется, поэтому вызов Save или SaveAs в скрипте лишний. Его следует убрать, полагаясь на автоматическое закрытие без сохранения. (RBP сам нажмёт "No" при закрытии, если интерфейс англоязычный.)
* **Отладка признаков модификации документа**. После экспорта PDF можно проверить флаг изменений документа (например, doc.IsModified). В идеале он должен остаться False. Если же после вызова Export флаг становится True, возможно, Revit пометил документ изменённым (например, из-за обновления настроек печати в проекте). В таком случае, чтобы избежать диалога, можно принудительно снять флаг – например, вызовом doc.Save() в **новый временный файл** и тут же удалить его. Но это крайние меры. Скорее всего, при правильном использовании PDFExportOptions документ не будет считаться изменённым.
* **Логирование**. Добавьте вывод ключевых сообщений в консоль RBP (с помощью revit\_script\_util.Output или аналогов) – например, сообщайте о начале экспорта, о папке назначения и об успешном завершении. Это поможет в журнале RBP убедиться, что экспорт прошёл без ошибок и PDF сохранён. Например:

python

КопироватьРедактировать

import revit\_script\_util

from revit\_script\_util import Output

...

Output("Exporting sheets to PDF...")

result = doc.Export(output\_folder, view\_ids, opts)

if result:

Output("PDF экспортирован успешно.")

else:

Output("Ошибка при экспорте PDF!")

Такой лог будет виден в интерфейсе BatchRvt и запишется в лог-файл, что упростит дальнейший поиск проблем, если они возникнут.

При необходимости, если Dynamo-скрипт оказывается слишком сложным или ограниченным, можно рассмотреть перенос логики экспорта в отдельный **Python-скрипт** для RBP. Python-скрипт, запущенный через BatchRvt, имеет прямой доступ к Revit API и позволяет, например, самому закрыть документ без сохранения (метод Document.Close(False)) после экспорта, полностью устраняя диалоги. Однако это потребует переписать логику Dynamo на Python. Если текущий подход с Dynamo можно исправить описанными выше мерами, то смена на Python не обязательна.

**Настройки BatchRvt**

Чтобы Batch Revit Processor не зависал на диалогах и корректно обрабатывал список файлов, важно правильно настроить его перед запуском задачи:

* **Detach from Central**: Включите опцию **«Detach from Central»** (Отсоединить от центрального файла) для файлов проектов. Это особенно актуально, если в списке есть центральные модели (общие файлы с рабочими наборами). При включенном Detach RBP открывает копию файла, не пытаясь сохранить изменения в оригинал[forum.dynamobim.com](https://forum.dynamobim.com/t/revit-batch-processor-save-file-dialog-box/62945#:~:text=Hi%20Daniel%2C). Таким образом, исключается диалог с вопросом о сохранении в центральный файл или выдаче прав. Даже для обычных (локальных) файлов эта настройка не повредит – RBP обнаружит, что файл не central, и просто откроет его как обычно. *(Примечание: пользователь на форуме подтвердил, что оставил настройку Detach по умолчанию даже для независимых файлов*[*forum.dynamobim.com*](https://forum.dynamobim.com/t/revit-batch-processor-save-file-dialog-box/62945#:~:text=I%20thought%20that%20settings%20are,from%20Central%2C%20Close%20All%20Worksets)*.)*
* **Worksets (Рабочие наборы)**: Если используется Detach, можно задать **Close All Worksets** (не открывать рабочие наборы)[forum.dynamobim.com](https://forum.dynamobim.com/t/revit-batch-processor-save-file-dialog-box/62945#:~:text=I%20thought%20that%20settings%20are,from%20Central%2C%20Close%20All%20Worksets), чтобы ускорить загрузку модели и снизить нагрузку на память. Однако убедитесь, что это не помешает печати листов: если на листах есть виды с элементами из закрытых workset'ов, они могут не отобразиться. В большинстве случаев печать листов происходит корректно даже с закрытыми наборами, но стоит протестировать. Альтернативно, выберите **Open All Worksets**, если необходимо полностью загрузить модель для правильного отображения всех элементов на чертежах.
* **Automatic Run (Автоматический запуск Dynamo)**: Убедитесь, что Dynamo-скрипт (.dyn) сохранён в режиме **Automatic** (Автоматический) перед использованием в RBP[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=,Microsoft%20Office%20%2F%20Excel%20installed). Batch Revit Processor требует, чтобы Dynamo-скрипт автоматически выполнялся при открытии, иначе он не запустится внутри RBP. Откройте файл .dyn в Dynamo, выставьте **Run Type = Automatic**, затем сохраните. Это предотвратит ситуацию, когда RBP открыл Revit и Dynamo, но скрипт ждёт ручного нажатия **Run**.
* **Версия Dynamo**: По документации RBP, для каждого Revit должна быть установлена **только одна** версия Dynamo (соответствующая версии Revit)[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=,Microsoft%20Office%20%2F%20Excel%20installed). Например, для Revit 2023 должен стоять Dynamo Core 2.13 (или 2.12) единственной версии. Если у вас параллельно установлены несколько версий Dynamo для одной версии Revit, RBP может столкнуться с ошибкой загрузки скрипта. Проверьте и удалите лишние версии Dynamo, либо отключите ненужные пакеты, чтобы исключить конфликт.
* **Язык интерфейса Revit (UI)**: По возможности используйте англоязычную версию Revit при пакетном запуске. Как обсуждалось, это позволит RBP автоматически закрывать диалоги без зависания[github.com](https://github.com/bvn-architecture/RevitBatchProcessor#:~:text=scripts%20do%20not%20suffer%20this,It%20is). Если установка другого языка невозможна, попробуйте временно переключить язык интерфейса через пакет локализации или запустить RBP под пользователем с англоязычными настройками среды – главное, чтобы системные диалоги внутри Revit были на английском. Это сильно повысит надёжность автоматизации.
* **Логи и отладка**: В BatchRvt Settings (или через GUI) укажите папку для логов (опция **Log folder**). После прогона задачи ознакомьтесь с лог-файлом. В логе должно быть видно, как RBP обрабатывал каждый файл, и были ли попытки закрывать диалоги. Если всё настроено правильно, в логе вы увидите сообщение об успешном экспорте PDF и отсутствие ошибок диалогов. Например, RBP может сообщить: *“Sending button click to 'No' (Не сохранять изменения)”* – в идеале такого сообщения не должно быть, если диалог не появлялся или был устранён.