

# Руководство «Менеджер потоков»

## Оглавление

Предисловие .....	3
Взаимодействие с платформой 1С:Предприятие .....	4
Установка «Менеджера потоков» .....	5
Объекты, добавленные в конфигурацию: .....	7
Концепция решения .....	8
Параметры «Менеджера потоков» .....	9
Параметры «Общие» .....	10
ИдентификаторМенеджера .....	10
МодульОбработкиСобытийРазработчика .....	10
КоличествоПотоков .....	11
КоличествоПопытокОбработкиОбъекта .....	11
КоличествоОбъектовВПакете .....	11
Параметры «Разработчик» .....	12
Общие .....	12
ПоСобытиям .....	12
Параметры «Метод.ОбработатьКоллекцию» .....	13
КоличествоЭлементовКоллекцииНаОбъект .....	13
СобратьКоллекциюИзФрагментов .....	13
Параметры «Метод.ПолучитьГраф_Gerhi» .....	14
Каталог .....	14
Параметры «Граф» .....	15
ОграничениеРазмера .....	15
УчитыватьОбъектыБезРесурсов .....	15
Методы .....	16
Метод: ОбработатьКоллекцию() .....	17
Метод: ОбработатьОбъект() .....	19
Метод: ПолучитьГраф_Gerhi() .....	20
Метод: ПроизвольнаяОбработкаГрафа() .....	22
Метод: ПолучитьПараметрыИнициализацииМенеджераПотоков() .....	23
Метод: ИнициализироватьМенеджерПотоков() .....	24
Метод: ДождатьсяОстановкиМенеджераПотоков () .....	25
События .....	26
Событие: ПередЗапускомМенеджераПотоков .....	27
Событие: ПриЗапускеМенеджераПотоков .....	28

Событие: ПередЗапускомПотока .....	29
Событие: ПриЗапускеПотока .....	30
Событие: ПриРасчетеРесурсов .....	31
Событие: ПриОбработкеПотоком .....	32
Событие: ПриОбработкеОшибки .....	33
Событие: ПриОбработкеПропуска .....	34
Событие: ПриОбработкеРезультата.....	35
Событие: ПриПроизвольнойОбработкеГрафа .....	36
Событие: ПередЗавершениемМенеджераПотоков .....	37
Событие: ПриСохраненииГрафа.....	38
Событие: ПриСборкеФрагментовКоллекции.....	39
Событие: ПослеЗавершенияМенеджераПотоков .....	40
Методы и события (матрица вызовов).....	41
Ресурсы.....	42
Правила формирования ресурсов .....	43

## Предисловие

В рамках данного руководства будет достаточно часто встречаться название программы «Gephi» (<https://gephi.org/>), прошу ни считать за рекламу. Выбор мой пал на данную программу по следующим причинам:

- она бесплатна;
- может обрабатывать очень большие графы;
- поддерживает множество плагинов;
- проста в использовании;
- достаточно удобна.

Нашел данную программу благодаря статье на Хабре ([Визуализация больших графов для самых маленьких](#)), где автор рассматривает разные инструменты для работы с графами.

Есть правда один минус как отмечает автор:

*«Очень досадный факт в том, что Gephi уже несколько лет заброшен. Два основных разработчика, не обладают ресурсами чтобы передать свои знания, необходимые для дальнейшей разработки кому-то ещё, заявили что уже не могут больше активно поддерживать Gephi.»*

## Взаимодействие с платформой 1С:Предприятие.

Решение предназначено для встраивания в любой конфигурации 1С, работающие на платформе 1С:Предприятие 8.3. Позволяет выполнять алгоритмы в многопоточном режиме.

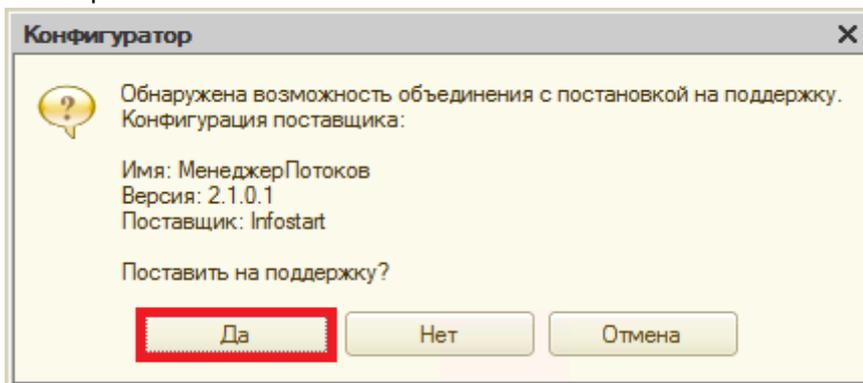
Требования:

- Работа в режиме «клиент-сервер».
- Платформа 1С:Предприятие для Windows конфигурация в режиме совместимости 8.2.12 или выше.
- Возможность внесения изменений (дополнений) в конфигурации, в которую встраивается решение "Менеджер потоков".

## Установка «Менеджера потоков»

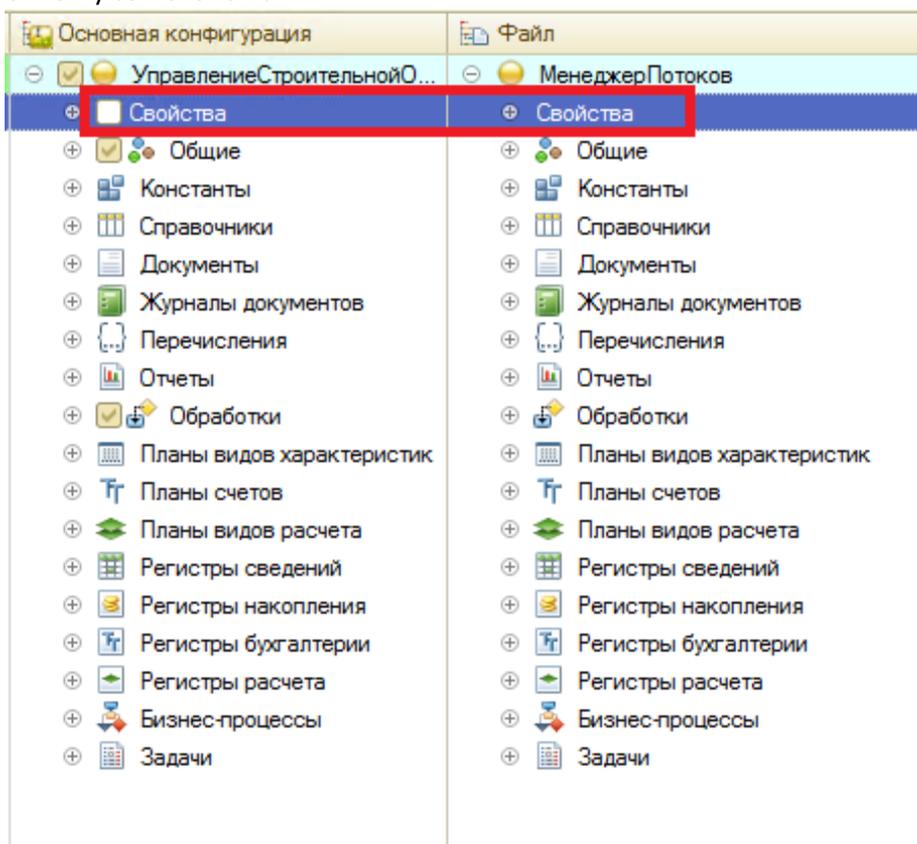
Установка дополнения к конфигурации выполняется через механизм «Сравнить, объединить с конфигурацией из файла...».

1. Создаем резервную копию базы данных (на всякий не предвиденный случай).
2. Запускаем конфигуратор.
3. Выбираем меню «Конфигурация» – «Сравнить, объединить с конфигурацией из файла...».
4. В диалоге выбора файла выбираем файл «1Cv8.cf» из поставки.
5. На вопрос:



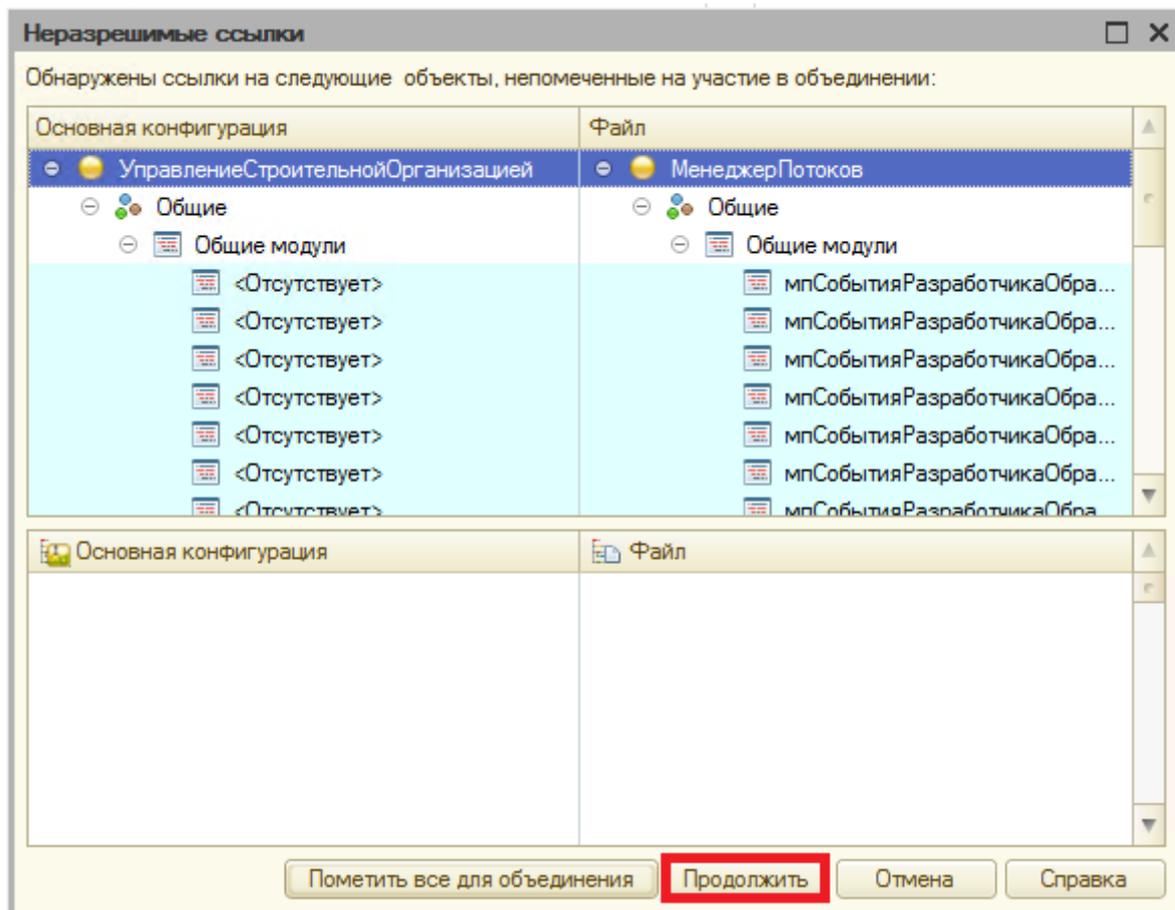
Рекомендуется ответить «Да».

6. В окне «Сравнение, объединение Основная конфигурация – Файл ...\1Cv8.cf», снимаем отметку со «Свойства»



Нажимаем кнопку «Выполнить».

7. На вопрос:

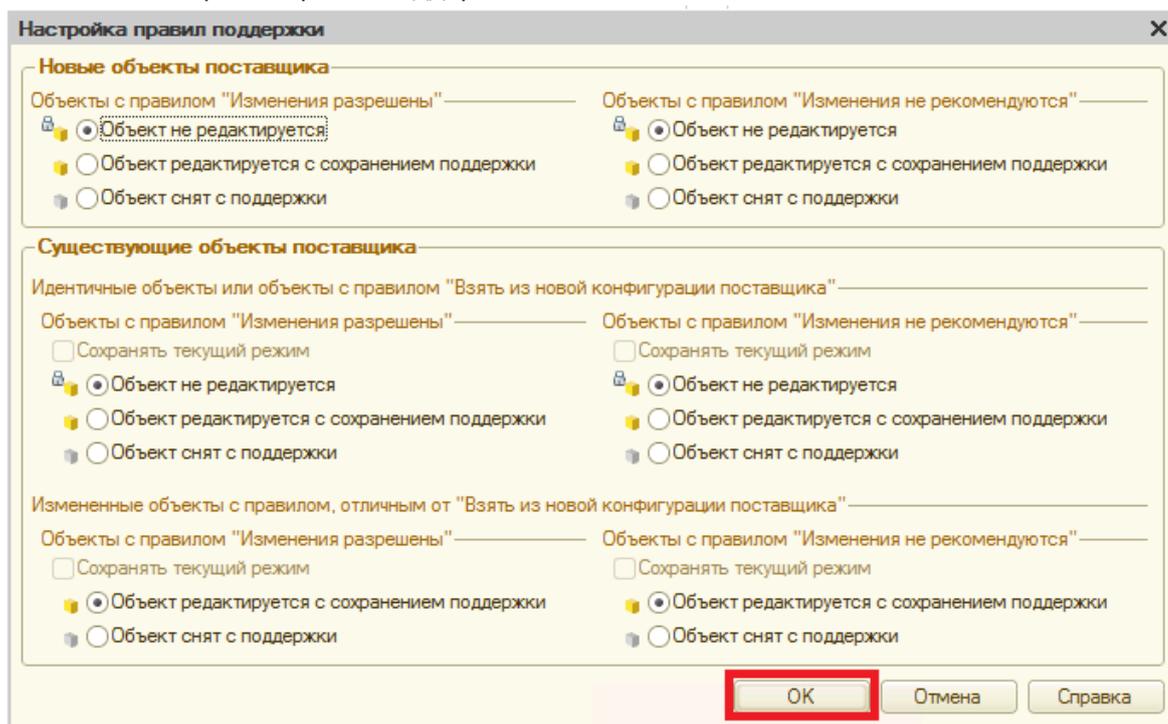


Нажимаем «Продолжить».

**Пояснения:** В конфигурации «Менеджер потоков», так же присутствуют демо обработки, картинки для справки этих обработок и общие модули для них. Вот о них и сообщается во всплывающем окне.

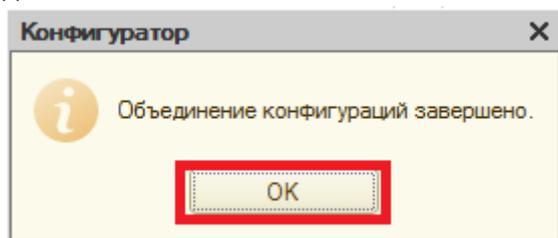
Если есть желание добавить эти объекты в «Основную конфигурацию» можно нажать «Пометить все для объединения»

8. В окне «Настройка правил поддержки»



Оставляем все без изменения и нажимаем «OK».

9. После чего происходит объединение конфигураций, по окончании которого высветится диалоговое окно:



Нажимаем «OK»

10. Обновляем конфигурацию базы данных:

меню «Конфигурация» - «Обновить конфигурацию базы данных».

11. Готово.

Объекты, добавленные в конфигурацию:

- ⊖ Общие
- ⊖ Подсистемы
  - ⊖ мпМенеджерПотоков
    - мпБиблиотека
    - мпСобытияРазработчика
- ⊖ Общие модули
  - мпМенеджерСервер
  - мпОбщееСервер
  - мпОсновнаяПрограммаВызовСервера
  - мпПотокиСервер
  - мпСобытияРазработчикаШаблон
- ⊖ Общие макеты
  - мпПолнаяСтруктураПараметров
- ⊖ Обработки
  - ⊕ мпИнтерактивнаяОстановкаМенеджеровПотоков

## Концепция решения

Менеджер потоков является решением позволяющим разработчику перевести исполнение длительного по времени алгоритма на обработку в потоках. Обычно к таким обработкам относятся переработка больших массивов (коллекций) информации, а именно:

- Выгрузка / загрузка / обновление списков (справочник «Номенклатура»);
- Формирование объектов (создание цепочек документов);
- Формирование отчетов (формирование «Расчетных листов» по каждому сотруднику);
- Обработка объектов (заполнение, перезапись, перепроведение, удаление);
- **Обработка сложно связанных объектов (графовая последовательность);**

Последний пункт можно выделить, как уникальную возможность, позволяющую решить проблемы блокировок объектов при массовых обработках, а также строгую обработку объектов в определенной последовательности (текущий объект не обрабатывается пока не будут обработаны все объекты от которых он зависит), к таким задачам можно отнести «**Восстановление последовательности партий**», «**Восстановление взаиморасчетов**» и т.д.

Менеджер потоков берет на себя весь функционал связанных с:

- Запуском потоков;
- Поддержанием жизнеспособности потоков (в случае «падения», поток перезапускается);
- Определением порции данных для каждого потока;
- Передачей данных между всеми участниками;
- Автоматической сборкой коллекций после обработки (если требуется);
- Выводом сообщений клиенту, выведенных при обработке в потоке;
- Созданием структуры взаимосвязей объектов (Граф);
- Контролем результатов обработки;
- Возможностью многократной обработки одного объекта;
- И многое другое.

Все взаимодействие разработчика с менеджером потоков происходит через «[События](#)».

## Параметры «Менеджера потоков»

Ниже будут перечислены основные параметры «Менеджера потоков» получаемые как результат функции [«ПолучитьПараметрыИнициализацииМенеджераПотоков \(\)»](#).

Пример:

```
ПараметрыМП = мпОсновнаяПрограммаВызовСервера. ПолучитьПараметрыИнициализацииМенеджераПотоков (  
    "ПростаяОбработкаКоллекции",  
    "мпСобытияРазработчикаОбработкаКоллекцииПростая");
```

Установка всех параметров должна быть произведена до вызова метода [«ИнициализироватьМенеджерПотоков \(\)»](#).

Параметры, отсутствующие в данном разделе считаются более не актуальны и не используются.

## Параметры «Общие»

Идентификатор Менеджера

**Тип:** Строка.

### Описание:

Хранит текстовый идентификатор «Менеджера потоков». Первый обязательный параметр функции [«ПолучитьПараметрыИнициализацииМенеджераПотоков \(\)»](#)

### Путь:

[ПараметрыМП.Общие.ИдентификаторМенеджера](#)

### Примечание:

Является разрезом хранения данных между всеми экземплярами «Менеджера потоков» запущенных под текущим пользователем.

Если в период действия одного «Менеджера потоков», под тем же пользователем, запустить еще один «Менеджер потоков» с таким же идентификатором (вызвать метод: [ИнициализироватьМенеджерПотоков \(\)](#)), то произойдет следующее:

- Остановится ранее запущенный «Менеджера потоков»;
- Остановятся его потоки;
- Произойдет очистка данных «Менеджера потоков».
- Запустится новый экземпляр «Менеджера потоков».

Под разными пользователями можно запускать одновременно «Менеджер потоков» с одинаковым идентификатором.

### Используется:

Доступен с версии 2.1.0

*Ранее назывался «Разрез Менеджера»*

МодульОбработкиСобытийРазработчика

**Тип:** Строка.

### Описание:

Хранит путь до модуля с событиями разработчика. Второй обязательный параметр функции [«ПолучитьПараметрыИнициализацииМенеджераПотоков \(\)»](#) Путь может быть указан до общего модуля или до модуля менеджера объекта.

### Путь:

[ПараметрыМП.Общие.МодульОбработкиСобытийРазработчика](#)

### Примечание:

В модуле **обязательно** должна располагаться

**Функция** [ОбработатьСобытиеРазработчика \(пПараметрыСобытия\)](#) **Экспорт**

Шаблон модуля можно взять в общем модуле «мпСобытияРазработчикаШаблон»

### Используется:

Доступен с версии 2.1.0

КоличествоПотоков

**Тип:** Число.

**Описание:**

Устанавливает сколько будет запущено потоков для обработки данных.

**Путь:**

[ПараметрыМП.Общие.КоличествоПотоков](#)

**Примечание:**

Значение по умолчанию: 10.

**Используется:**

Доступен с версии 1.0.1

КоличествоПопытокОбработкиОбъекта

**Тип:** Число.

**Описание:**

Определяет сколько раз будет производиться попытка обработать объект (так же влияет на расчет ресурсов). Во время работы «Менеджера потоков», могут возникнуть всякие внештатные ситуации, когда нет возможности обработать объект, например, превышено время ожидания захвата таблиц. В этом случае «Менеджера потоков», будет пытаться столько раз обработать объект, сколько указано в данном параметре.

**Путь:**

[ПараметрыМП.Общие.КоличествоПопытокОбработкиОбъекта](#)

**Примечание:**

Значение по умолчанию: 1.

**Используется:**

Доступен с версии 1.0.1

*Ранее назывался «ПределКоличествоПопытокОбработатьОбъект»*

КоличествоОбъектовВПакете

**Тип:** Число.

**Описание:**

Устанавливает максимальный размер пакета обмена данными

**Путь:**

[ПараметрыМП.Общие.КоличествоОбъектовВПакете](#)

**Примечание:**

Значение по умолчанию: 10.

Для метода: [«ОбработатьКоллекцию \(\)»](#) значение всегда: 1.

**Используется:**

Доступен с версии 2.1.0

## Параметры «Разработчик»

Общие

**Тип:** Структура.

**Описание:**

Структура, предназначенная для сохранения произвольных параметров разработчика.

**Путь:**

[ПараметрыМП](#) . [Разработчик](#) . [Общие](#)

**Используется:**

Доступен с версии 2.1.0

ПоСобытиям

**Тип:** Структура.

**Описание:**

Структура хранящая результат выполнения [событий](#).

**Путь:**

[ПараметрыМП](#) . [Разработчик](#) . [ПоСобытиям](#) . [<ИмяСобытия>](#)

**Примечание:**

Поддерживается автоматическое сохранение результатов из следующих событий:

- [ПередЗапускомМенеджераПотоков](#);
- [ПриЗапускеМенеджераПотоков](#);
- [ПриЗапускеПотока](#).

**Пример:**

Если в событии «[ПередЗапускомМенеджераПотоков](#)», вернуть значение, например, так:

```
Функция ПередЗапускомМенеджераПотоков (пПараметрыСобытия)  
    Возврат "Привет!";  
КонецФункции
```

То в событии «[ПередЗапускомПотока](#)», можно получить доступ к этому значению, так:

```
Функция ПередЗапускомПотока (пПараметрыСобытия)  
пПараметрыСобытия . Параметры . ПоСобытиям . ПередЗапускомМенеджераПотоков ;
```

Более подробно о передаче параметров

см. Демо базу, обработка: «Обработка коллекции: Автосборка и передача общих параметров»

**Используется:**

Доступен с версии 2.1.0

## Параметры «Метод.ОбработатьКоллекцию»

КоличествоЭлементовКоллекцииНаОбъект

**Тип:** Число.

**Описание:**

Определяет размер суб.коллекции, которая будет выделена из основной коллекции и отправлена в поток на обработку.

**Путь:**

[ПараметрыМП.Метод.ОбработатьКоллекцию.КоличествоЭлементовКоллекцииНаОбъект](#)

**Примечание:**

Значение по умолчанию: 0.

Если параметр = 0, то основная коллекция будет разделена пропорционально параметру «КоличествоПотоков».

**Используется:**

Доступен с версии 2.1.0

## СобратьКоллекциюИзФрагментов

**Тип:** Булево.

**Описание:**

Устанавливает признак необходимости собирать коллекцию после обработки.

**Путь:**

[ПараметрыМП.Метод.ОбработатьКоллекцию.СобратьКоллекциюИзФрагментов](#)

**Примечание:**

Значение по умолчанию: Ложь.

Если значение = Ложь – будет выполнена просто обработка коллекции

Если значение = Истина – разработчик должен в событии «[ПриСборкеФрагментовКоллекции](#)» описать алгоритм, как будут собираются фрагменты. Если разработчик не использует событие «[ПриСборкеФрагментовКоллекции](#)» или результат событие возвращает Неопределено. «Менеджер потоков» производит сборку коллекции автоматически. Поддерживаемые типы коллекций: Массив, СписокЗначений, ТаблицаЗначений.

Более подробно о сборке

см. Демо базу, обработка: «Обработка коллекции: Автосборка», «Обработка коллекции: Произвольная сборка»

**Используется:**

Доступен с версии 2.1.0

## Параметры «Метод.ПолучитьГраф\_Gephi»

Каталог

**Тип:** Строка.

**Описание:**

Задаёт путь куда будут сохранены файлы (\*.csv) для программы Gephi (<https://gephi.org/>)

**Путь:**

[ПараметрыМП.Метод.ПолучитьГраф\\_Gephi.Каталог](#)

**Примечание:**

Путь должен быть доступен со стороны сервера.

**Используется:**

Доступен с версии 2.1.0

## Параметры «Граф»

### ОграничениеРазмера

**Тип:** Число.

**Описание:**

Определяет максимальный размер графа, который оперирует «Менеджер потоков».

**Путь:**

[ПараметрыМП.Граф.ОграничениеРазмера](#)

**Примечание:**

Значение по умолчанию: 0.

Если значение = 0 – ограничений на размер графа – нет.

Если граф достигает значения данного параметра, то происходит следующее:

- Приостанавливается загрузка данных из основной программы (в основной программе объекты начнут копиться в очередь на обработку);
- Приостанавливается добавление новых узлов в граф.

**Используется:**

Доступен с версии 2.1.0

### УчитыватьОбъектыБезРесурсов

**Тип:** Булево.

**Описание:**

Изменяет правило добавления объектов в граф.

**Путь:**

[ПараметрыМП.Граф.УчитыватьОбъектыБезРесурсов](#)

**Примечание:**

Значение по умолчанию: Ложь.

Если значение = Ложь – Объекты без [ресурсов](#) в граф добавляться не будут.

Если значение = Истина – В граф добавляются все объекты.

**Используется:**

Доступен с версии 2.1.0

## Методы

Менеджер потоков обладает 2мя основными методами для обработки:

- [ОбработатьКоллекцию\(\)](#);
- [ОбработатьОбъект\(\)](#).

2мя дополнительными методами для анализа сложно зависимых объектов:

- [ПолучитьГраф\\_Gerhi\(\)](#);
- [ПроизвольнаяОбработкаГрафа\(\)](#).

И 3ми системными методами для запуска и завершения работы:

- [ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)](#);
- [ИнициализироватьМенеджерПотоков\(\)](#);
- [ДождатьсяОстановкиМенеджераПотоков\(\)](#);

## Метод: [ОбработатьКоллекцию\(\)](#)

### Синтаксис:

ОбработатьКоллекцию(<пПараметры>, <пКоллекция>, <пПараметрыДляКоллекции>);

### Параметры:

- <пПараметры> (обязательно)
  - Тип: Структура.
  - параметры «Менеджера потоков» См. метод [«ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)»](#);
- <пКоллекция> (обязательно)
  - Тип: Массив, СписокЗначений, ТаблицаЗначений.
  - Коллекция, которую необходимо обработать;
- <пПараметрыДляКоллекции>
  - Тип: Произвольно
  - Произвольные вспомогательные данные для обработки коллекции
  - Значение по умолчанию: Неопределено.

### Описание:

Данный метод может пригодиться для многопоточной обработки коллекции (Массив, Список значений, Таблица значений), при этом обработка коллекции может быть так же различная:

- Просто обработка коллекции (в демо конфигурации «*Обработка коллекции: Простая*»);
- Обработка коллекции и возврат такой же коллекции (в демо конфигурации «*Обработка коллекции: Автосборка*»);
- Обработка коллекции с возвратом произвольных данных (в демо конфигурации «*Обработка коллекции: Произвольная сборка*»).

### Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

### Примечание:

Данный метод является единственным, где не требуется вызов сервисных методов [«ИнициализироватьМенеджерПотоков\(\)»](#) и [«ДождатьсяОстановкиМенеджераПотоков\(\)»](#), т.к. они встроены уже внутри.

Обязательные [«события»](#):

- [«ПриОбработкеПотоком»](#);
- [«ПриСборкеФрагментовКоллекции»](#) - если требуется сборка в произвольный результат.

Принцип работы:

- Получаем параметры «Менеджера потоков»;
- Изменяем параметры «по умолчанию» (если требуется);
- Вызываем метод [«ОбработатьКоллекцию\(\)»](#).

### Пример (простая обработка)

```
ПараметрыМП = мОсновнаяПрограммаВызовСервера.ПолучитьСтруктуруПараметровИнициализацииМенеджераПотоков (  
    "ПростаяОбработкаКоллекции",  
    "мпСобытияРазработчикаОбработкаКоллекцииПростая");  
ПараметрыМП.Общие.КоличествоПотоков = КоличествоПотоков;  
мОсновнаяПрограммаВызовСервера.ОбработатьКоллекцию(ПараметрыМП, КоллекцияОбъектов);
```

### Пример (обработка с возвратом результата)

```
ПараметрыМП = мОсновнаяПрограммаВызовСервера.ПолучитьСтруктуруПараметровИнициализацииМенеджераПотоков (  
    "ОбработкаКоллекцииСАвтоборкой",  
    "мпСобытияРазработчикаОбработкаКоллекцииСАвтоборкой");  
ПараметрыМП.Общие.КоличествоПотоков = КоличествоПотоков;  
ПараметрыМП.Метода.ОбработатьКоллекцию.СобратьКоллекциюИзФрагментов = Истина;  
КоллекцияПриемник = мОсновнаяПрограммаВызовСервера.ОбработатьКоллекцию(ПараметрыМП, КоллекцияОбъектов);
```

## Метод: ОбработатьОбъект()

### Синтаксис:

ОбработатьОбъект(<пПараметры>, <пОбрабатываемыйОбъект>, <пПараметрыДляОбъекта>);

### Параметры:

- <пПараметры> (обязательно)
  - Тип: Структура.
  - параметры «Менеджера потоков» См. метод [«ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)»](#);
- <пОбрабатываемыйОбъект> (обязательно)
  - Тип: Произвольно.
  - Объект (ссылка) над которым проводится обработки;
- <пПараметрыДляОбъекта>
  - Тип: Произвольно
  - Произвольные вспомогательные данные для обработки объекта (ссылки)
  - Значение по умолчанию: Неопределено.

### Описание:

Использовать данный метод необходимо для обработки объектов со сложной зависимостью между собой, например, «Восстановление последовательности партий», «Восстановление взаиморасчетов» и т.д.

### Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

### Примечание:

Обязательные [«СОБЫТИЯ»](#):

- [«ПриРасчетеРесурсов»](#);
- [«ПриОбработкеПотоком»](#).

### Принцип работы:

- Получаем параметры «Менеджера потоков»;
- Изменяем параметры «по умолчанию» (если требуется);
- Запускаем «Менеджер потоков»;
- Запускаем цикл обхода коллекции;
  - В рамках цикла вызываем метод [«ОбработатьОбъект»](#).
- После цикла ожидаем завершения «Менеджера потоков»;

### Пример:

```
ПараметрыМП = мпОсновнаяПрограммаВызовСервера.ПолучитьСтруктуруПараметровИнициализацииМенеджераПотоков (
    "ОбработатьОбъект",
    "мпСобытияРазработчикаОбработатьОбъект");
ПараметрыМП.Графа.УчитыватьОбъектыВезРесурсов = ОбработатьОбъектыВезРесурса;
ПараметрыМП.Общие.КоличествоПотоков = КоличествоПотоков;

ПараметрыДляОбъекта = Новый Структура;
ПараметрыДляОбъекта.Вставить ("Алгоритм", Алгоритм);

мпОсновнаяПрограммаВызовСервера.ИнициализироватьМенеджерПотоков (ПараметрыМП);
Для Сч = 1 По РазмерКоллекции Цикл
    мпОсновнаяПрограммаВызовСервера.ОбработатьОбъект (ПараметрыМП, Сч, ПараметрыДляОбъекта);
КонецЦикла;

мпОсновнаяПрограммаВызовСервера.ДождатьсяОстановкиМенеджераПотоков (ПараметрыМП);
```

Метод: [ПолучитьГраф\\_Gephi\(\)](#)

**Синтаксис:**

ПолучитьГраф\_Gephi(<пПараметры>, <пОбрабатываемыйОбъект>, <пПараметрыДляОбъекта>);

**Параметры:**

- <пПараметры> (обязательно)
  - Тип: Структура.
  - параметры «Менеджера потоков» См. метод [«ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)»](#);
- <пОбрабатываемыйОбъект> (обязательно)
  - Тип: Произвольно.
  - Объект (ссылка) над которым проводится обработки;
- <пПараметрыДляОбъекта>
  - Тип: Произвольно
  - Произвольные вспомогательные данные для обработки объекта (ссылки)
  - Значение по умолчанию: Неопределено.

**Описание:**

Данный метод является взаимозаменяемым метода [«ОбработатьОбъект\(\)»](#).

Результатом вызова данного метода будут 2а сохраненных файла (\*.csv) для дальнейшей загрузки и анализа в программе Gephi (<https://gephi.org/>).

**Доступность:**

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

**Примечание:**

Обязательные [«события»](#):

- [«ПриРасчетеРесурсов»](#).

*Принцип работы:*

- Получаем параметры «Менеджера потоков»;
- Изменяем параметры «по умолчанию» (если требуется);
- Запускаем «Менеджер потоков»;
- Запускаем цикл обхода коллекции;
  - В рамках цикла вызываем метод [«ПолучитьГраф\\_Gephi»](#).
- После цикла ожидаем завершения «Менеджера потоков».

**Пример:**

```
ПараметрыМП = мпОсновнаяПрограммаВызовСервера.ПолучитьСтруктуруПараметровИнициализацииМенеджераПотоков (
    "ОбработатьОбъектфайлыГрафаGephi",
    "мпСобытияРазработчикаОбработатьОбъектфайлыГрафаGephi");

ПараметрыМП.Общие.КоличествоПотоков = КоличествоПотоков;
ПараметрыМП.Графа.УчитыватьОбъектыБезРесурсов = ОбработатьОбъектыБезРесурса;
ПараметрыМП.Метода.ПолучитьГраф_Gephi.Каталог = ПутьККаталогу;

ПараметрыДляОбъекта = Новый Структура;
ПараметрыДляОбъекта.Вставить ("Алгоритм", Алгоритм);

мпОсновнаяПрограммаВызовСервера.ИнициализироватьМенеджерПотоков (ПараметрыМП);
Для Сч = 1 По РазмерКоллекции Цикл
    мпОсновнаяПрограммаВызовСервера.ПолучитьГраф_Gephi (ПараметрыМП, Сч, ПараметрыДляОбъекта);
КонiecЦикла;

мпОсновнаяПрограммаВызовСервера.ДождатьсяОстановкиМенеджераПотоков (ПараметрыМП);
```

## Пример файла (ребра)

```
Source,Target,Type,Id,Label,timeset,Weight
380be57a-f96e-41cf-95cd-770952e386f7,967df2f2-38fd-4026-948a-ea94b1022957,Directed,20372579-137e-4a07-9799-a2ded62122b4,,,1
4bb3305c-cd95-4a08-b8fe-903c6f5e996d,4579e5f2-a786-4166-a73c-3dcb0ef47167,Directed,26e61762-f460-483c-9e06-b7b8748cb2e4,,,1
3752ae51-f59a-4ce3-9791-51310a288a91,95ccca22-9521-4457-8dd4-9eb5689a500b,Directed,ef49fabe-ab18-43ba-93a8-00bad5e871b7,,,1
0156d80b-f05b-46a1-9b18-7ba217fe5a6f,ffaadd22-a2c1-4456-b01e-0572f1e49cfa,Directed,3a1ccafa-7f58-4c43-bdac-00b9426f54b3,,,1
0a3f2f80-624e-43cd-8945-0afc31348d44,08a81caf-9703-44a2-a148-6242f881a74c,Directed,e2e67409-c77e-4c68-9402-0464545cbc07,,,1
8a58c933-db35-498f-aa21-2edb2d71f5e9,1289a4ef-e82e-491c-ac83-a41c19153c5b,Directed,912c91e6-4ea4-4186-8624-7de35ccc31d4,,,1
fb319b2e-19f9-463d-a9bd-f8c9e21eeef83,1289a4ef-e82e-491c-ac83-a41c19153c5b,Directed,7c372d24-dd1d-4fbb-b1ee-ea1723c9d311,,,1
560074fc-b730-450e-9ac7-314d43041264,1289a4ef-e82e-491c-ac83-a41c19153c5b,Directed,4434b967-e163-4c1a-b0f5-5c9f6b8325ea,,,1
cddcf4e1-a976-42d2-b353-c8e7294f1471,dc1ed32f-8c92-446b-a191-3fb4b947df08,Directed,b9033b9e-4a52-47a6-b50c-b9eb26f1f6fc,,,1
dc28003e-ea80-408e-8a03-dd053320f6fa,dc1ed32f-8c92-446b-a191-3fb4b947df08,Directed,09b1b44d-fef6-4020-91e8-fa6a3eb7768a...1
```

## Пример файла (узлы)

```
Id,Label,timeset
967df2f2-38fd-4026-948a-ea94b1022957,76,
4579e5f2-a786-4166-a73c-3dcb0ef47167,9,
95ccca22-9521-4457-8dd4-9eb5689a500b,66,
ffaadd22-a2c1-4456-b01e-0572f1e49cfa,4,
08a81caf-9703-44a2-a148-6242f881a74c,96,
1289a4ef-e82e-491c-ac83-a41c19153c5b,84,
dc1ed32f-8c92-446b-a191-3fb4b947df08,54,
6af05f2f-ad29-4d52-a6fc-847667f5b472,5,
39b710ef-eca1-4735-b18d-07c98b4f8cd3,7,
```

## Метод: ПроизвольнаяОбработкаГрафа()

### Синтаксис:

ПроизвольнаяОбработкаГрафа(<пПараметры>, <пОбрабатываемыйОбъект>, <пПараметрыДляОбъекта>);

### Параметры:

- <пПараметры> (обязательно)
  - Тип: Структура.
  - параметры «Менеджера потоков» См. метод [«ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)»](#);
- <пОбрабатываемыйОбъект> (обязательно)
  - Тип: Произвольно.
  - Объект (ссылка) над которым проводится обработки;
- <пПараметрыДляОбъекта> (не обязательно)
  - Тип: Произвольно
  - Произвольные вспомогательные данные для обработки объекта (ссылки)
  - Значение по умолчанию: Неопределено.

### Описание:

Данный метод является взаимозаменяемым метода [«ОбработатьОбъект\(\)»](#).

Разработчику предоставляется возможность обработать структуру граф произвольным образом.

### Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

### Примечание:

Обязательные [«события»](#):

- [«ПриРасчетеРесурсов»](#);
- [«ПриПроизвольнойОбработкеГрафа»](#);
- [«ПриСохраненииГрафа»](#)

### Принцип работы:

- Получаем параметры «Менеджера потоков»;
- Изменяем параметры «по умолчанию» (если требуется);
- Запускаем «Менеджер потоков»;
- Иницилируем цикл обхода коллекции;
  - В рамках цикла вызываем метод [«ПроизвольнаяОбработкаГрафа\(\)»](#).
- После цикла ожидаем завершения «Менеджера потоков».

## Метод: ПолучитьПараметрыИнициализацииМенеджераПотоков()

### Синтаксис:

ПолучитьПараметрыИнициализацииМенеджераПотоков(<пИдентификаторМенеджер>, <пМодульОбработкиСобытийРазработчика>)

### Параметры:

- <пИдентификаторМенеджер> (обязательно)
  - Тип: Строка.
  - Произвольная строка, идентифицирующая «Менеджер потоков».
- <пМодульОбработкиСобытийРазработчика> (обязательно)
  - Тип: Строка.
  - Путь до модуля содержащий экспортную процедуру «ОбработатьСобытиеРазработчика».

### Возвращаемое значение:

Тип: Структура – основные параметры «Менеджера потоков» используемые во всех остальных методах. Значения параметров в данной структуре разработчик может менять под свои потребности.

### Описание:

Данный метод является основным для начала работы, т.к. значение, полученное как результат данного метода используется при вызове всех остальных методов текущего экземпляра менеджера потоков.

### Доступность:

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

Метод: [ИнициализироватьМенеджерПотоков\(\)](#)

**Синтаксис:**

ИнициализироватьМенеджерПотоков(<пПараметры>)

**Параметры:**

- <пПараметры> (обязательно)
  - Структура – результат метода  
[«ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)»](#).

**Описание:**

Метод осуществляет запуск фонового задания «Менеджера потоков» с заданными параметрами.

**Доступность:**

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

**Примечание:**

Метод предназначен для работы с методами «[ОбработатьОбъект\(\)](#)», «[ПолучитьГраф\\_Graphi\(\)](#)», «[ПроизвольнаяОбработкаГрафа\(\)](#)».

Метод: [ДождатьсяОстановкиМенеджераПотоков \(\)](#)

**Синтаксис:**

ДождатьсяОстановкиМенеджераПотоков (<пПараметры>)

**Параметры:**

- <пПараметры> (обязательно)
  - Структура – результат метода  
[«ПолучитьПараметрыИнициализацииМенеджераПотоков\(\)»](#).

**Описание:**

Метод, предназначен для «досылки» данных скопившихся в очереди и ожидает окончания обработки данных «Менеджером потоков».

**Доступность:**

Тонкий клиент, веб-клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение(клиент), мобильное приложение(сервер).

**Примечание:**

Метод предназначен для работы с методами [«ОбработатьОбъект\(\)»](#), [«ПолучитьГраф\\_Graph\(\)»](#), [«ПроизвольнаяОбработкаГрафа\(\)»](#).

## События

События – это произвольный код разработчика находящийся в определенном контексте и выполняемый «Менеджером потоков» в разные моменты своего использования.

События используемые в текущей версии «Менеджера потоков» (2.1.0):

- [ПередЗапускомМенеджераПотоков](#) (Основная программа);
- [ПриЗапускеМенеджераПотоков](#) (ФЗ Менеджера потоков);
- [ПередЗапускомПотока](#) (ФЗ Менеджера потоков);
- [ПриЗапускеПотока](#) (ФЗ Потока);
- [ПриРасчетеРесурсов](#) (ФЗ Потока);
- [ПриОбработкеПотоком](#) (ФЗ Потока);
- [ПриОбработкеОшибки](#) (Основная программа);
- [ПриОбработкеПропуска](#) (Основная программа);
- [ПриОбработкеРезультата](#) (Основная программа);
- [ПриПроизвольнойОбработкеГрафа](#) (ФЗ Менеджера потоков);
- [ПередЗавершениемМенеджераПотоков](#) (ФЗ Менеджера потоков);
- [ПриСохраненииГрафа](#) (Основная программа);
- [ПриСборкеФрагментовКоллекции](#) (Основная программа);
- [ПослеЗавершенияМенеджераПотоков](#) (Основная программа);

События, отсутствующие в данном разделе считаются более не актуальны и не используются.

## Событие: ПередЗапускомМенеджераПотоков

### Место вызова:

Основная программа.

### Условие вызова:

Всегда.

### Частота вызова:

1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до вызова метода [«ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура. На текущий момент пустая.
- «Данные» - Неопределено.

### Описание:

#### Примечание:

Вызов события происходит еще до запуска фонового задания «Менеджера потоков».

Событие можно использовать для установки (корректировки) параметров «Менеджера потоков» или отмены его запуска.

#### Используется:

Доступен с версии 1.0.1

## Событие: ПриЗапускеМенеджераПотоков

### Место вызова:

Фоновое задание «Менеджера потоков»

### Условие вызова:

Всегда.

### Частота вызова:

1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до вызова метода [«ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура.
    - [«ПередЗапускомМенеджераПотоков»](#) - Значение возвращенной данного события.
- «Данные» - Неопределено.

### Описание:

Событие запускается сразу после запуска фонового задания «Менеджера потоков».

Событие можно использовать для разовой выборки данных в рамках фонового задания «Менеджера потоков».

### Примечание:

Фоновые задания «потоков» еще не запущены.

### Используется:

Доступен с версии 2.0.4

## Событие: ПередЗапускомПотока

### Место вызова:

Фоновое задание «Менеджера потоков».

### Условие вызова:

Всегда.

### Частота вызова:

1 раз перед запуском/перезапуском каждого фонового задания «потоков».

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до вызова метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура.
    - [ПередЗапускомМенеджераПотоков](#) – Значение, возвращенное данным событием;
    - [ПриЗапускеМенеджераПотоков](#) – Значение, возвращенное данным событием.
- «Данные» - Неопределено.

### Описание:

Событие используется для передачи произвольных данных в потоки.

### Примечание:

Более подробно см. Демо базу, обработка: «Обработка коллекции: Автосборка и передача общих параметров».

### Используется:

Доступен с версии 2.0.4

## Событие: ПриЗапускеПотока

### Место вызова:

Фоновое задание «потока».

### Условие вызова:

Всегда.

### Частота вызова:

1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до вызова метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура.
    - [«ПередЗапускомПотока»](#) – Значение, возвращенное данным событием.
- «Данные» - Неопределено.

### Описание:

Событие запускается сразу после запуска фонового задания «потока»

Событие можно использовать для разовой выборки данных в рамках фонового задания «потока».

### Примечание:

### Используется:

Доступен с версии 2.0.4

## Событие: ПриРасчетеРесурсов

### Место вызова:

Фоновое задание «потока».

### Условие вызова:

При вызове методов: «ОбработатьОбъект», «ПолучитьГраф\_Gerhi», «ПроизвольнаяОбработкаГрафа»

### Частота вызова:

Для каждого объекта, переданного в метод.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до вызова метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура.
    - [«ПередЗапускомПотока»](#) – Значение, возвращенное данным событием.
- «Данные» - Структура:
  - «КлючДанных» - УникальныйИдентификатор – (служебный)
  - «ОбрабатываемыйОбъект» - является 2ым параметром вызванного метода.
  - «ПараметрыДляОбъекта», является 3им параметром вызванного методов.

### Описание:

Событие можно использовать для разовой выборки данных в рамках фонового задания «потока».

### Примечание:

«ПараметрыДляОбъекта» - можно расширять и использовать на любом этапе:

- В момент ввода в «Менеджер потоков» (перед вызовом методов [«ОбработатьОбъект»](#), [«ПолучитьГраф\\_Gerhi»](#), [«ПроизвольнаяОбработкаГрафа»](#));
- В событии [«ПриРасчетеРесурсов»](#);
- В событии [«ПриОбработкеПотоком»](#);
- В событиях постобработки [«ПриОбработкеОшибки»](#), [«ПриОбработкеПропуска»](#) и [«ПриОбработкеРезультата»](#).

Более подробно см. Демо базу, обработка: «Обработать объект: Передача частных параметров».

Событие вызывается повторно для одного и того же объекта, если в прошлый раз во время исполнения произошла ошибка, например, «Превышено время ожидания захвата таблицы» (зависит от параметра [«КоличествоПопытокОбработкиОбъекта»](#)).

Более подробно о [ресурсах](#)

*Ранее называлось «ПриПолученииРесурсов»*

### Используется:

Доступен с версии 2.0.4

## Событие: ПриОбработкеПотоком

### Место вызова:

Фоновое задание «потока».

### Условие вызова:

При вызове методов: «[ОбработатьОбъект\(\)](#)», «[ОбработатьКоллекцию\(\)](#)»

### Частота вызова:

- Метод: «[ОбработатьОбъект\(\)](#)» - для каждого объекта, помещенного в граф;
- Метод: «[ОбработатьКоллекцию\(\)](#)» - для каждой части коллекции.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода «[ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура.
    - «[ПередЗапускомПотока](#)» – Значение, возвращенное данным событием.
- «Данные» - Структура.
  - «КлючДанных» - УникальныйИдентификатор – (служебный)
  - «ОбрабатываемыйОбъект» -
    - При вызове метода «[ОбработатьОбъект\(\)](#)», является 2ым параметром данного метода.
    - При вызове метода «[ОбработатьКоллекцию\(\)](#)», является 2ым параметром данного метода, а именно **часть** данной **коллекции** (тип коллекции и состав остается точно таким же).
  - «ПараметрыДляОбъекта», является 3им параметром вызываемых методов.

### Описание:

Является основным событием для обработки данных.

### Примечание:

«ПараметрыДляОбъекта» - можно расширять и использовать на любом этапе:

- В момент ввода в «Менеджер потоков» (перед вызовом методов «[ОбработатьОбъект](#)», «[ПолучитьГраф\\_Graphi](#)», «[ПроизвольнаяОбработкаГрафа](#)»);
- В событии «[ПриРасчетеРесурсов](#)»;
- В событии «[ПриОбработкеПотоком](#)»;
- В событиях постобработки «[ПриОбработкеОшибки](#)», «[ПриОбработкеПропуска](#)» и «[ПриОбработкеРезультата](#)».

Более подробно см. Демо базу, обработка: «Обработать объект: Передача частных параметров».

Событие вызывается повторно для одного и того же объекта, если в прошлый раз во время исполнения произошла ошибка, например, «Превышено время ожидания захвата таблицы» (зависит от параметра «[КоличествоПопытокОбработкиОбъекта](#)»).

*Ранее называлось «[ПриОбработкеДействияПотока](#)»*

### Используется:

Доступен с версии 2.0.4

## Событие: ПриОбработкеОшибки

### Место вызова:

Основная программа.

### Условие вызова:

При повторных попытках рассчитать ресурсы или обработать объект превышено значение параметра «[КоличествоПопытокОбработкиОбъекта](#)».

### Частота вызова:

По мере выявления таких объектов.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура:
    - [«ПередЗапускомМенеджераПотоков»](#) – Значение, возвращенное данным событием.
- «Данные» - Структура.
  - «ОбрабатываемыйОбъект» - является 2ым параметром вызываемого метода;
  - «ПараметрыДляОбъекта», является 3им параметром вызванного методов;
  - «РезультатОбработки» - Неопределено;
  - «СообщениеПользователю» - Строка, сообщения, выведенные во время выполнения события [«ПриОбработкеПотоком»](#);
  - «Статус» - Строка, «Ошибка».

### Описание:

Является признаком того, что в событиях [«ПриРасчетеРесурсов»](#) или [«ПриОбработкеПотоком»](#) произошли ошибки.

### Примечание:

«ПараметрыДляОбъекта» - можно расширять и использовать на любом этапе:

- В момент ввода в «Менеджер потоков» (перед вызовом методов [«ОбработатьОбъект»](#), [«ПолучитьГраф\\_Graphi»](#), [«ПроизвольнаяОбработкаГрафа»](#));
- В событии [«ПриРасчетеРесурсов»](#);
- В событии [«ПриОбработкеПотоком»](#);
- В событиях постобработки [«ПриОбработкеОшибки»](#), [«ПриОбработкеПропуска»](#) и [«ПриОбработкеРезультата»](#).

Более подробно см. Демо базу, обработка: «Обработать объект: Передача частных параметров».

### Используется:

Доступен с версии 2.0.4

## Событие: ПриОбработкеПропуска

### Место вызова:

Основная программа.

### Условие вызова:

При вызове методов: [«ОбработатьОбъект\(\)»](#), [«ПолучитьГраф Gephi\(\)»](#), [«ПроизвольнаяОбработкаГрафа\(\)»](#).

Объект не был помещен в граф, т.к. у объекта отсутствуют ресурсы И значение параметра [«УчитыватьОбъектыБезРесурсов»](#) = Ложь.

### Частота вызова:

По мере выявления таких объектов.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура:
    - [«ПередЗапускомМенеджераПотоков»](#) – Значение, возвращенное данным событием.
- «Данные» - Структура.
  - «ОбрабатываемыйОбъект» - является 2ым параметром вызываемого метода;
  - «ПараметрыДляОбъекта», является 3им параметром вызванного методов;
  - «РезультатОбработки» - Неопределено;
  - «СообщениеПользователю» - Строка, сообщения, выведенные во время выполнения события [«ПриОбработкеПотоком»](#);
  - «Статус» - Строка, «Пропуск».

### Описание:

Является признаком того, что объект не был обработан по причине отсутствия ресурсов.

### Примечание:

«ПараметрыДляОбъекта» - можно расширять и использовать на любом этапе:

- В момент ввода в «Менеджер потоков» (перед вызовом методов [«ОбработатьОбъект»](#), [«ПолучитьГраф Gephi»](#), [«ПроизвольнаяОбработкаГрафа»](#));
- В событии [«ПриРасчетеРесурсов»](#);
- В событии [«ПриОбработкеПотоком»](#);
- В событиях постобработки [«ПриОбработкеОшибки»](#), [«ПриОбработкеПропуска»](#) и [«ПриОбработкеРезультата»](#).

Более подробно см. Демо базу, обработка: «Обработать объект: Передача частных параметров».

### Используется:

Доступен с версии 2.0.4

## Событие: ПриОбработкеРезультата

### Место вызова:

Основная программа.

### Условие вызова:

Объект обработан событием «[ПриОбработкеПотоком](#)».

### Частота вызова:

По мере обработки объектов.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода «[ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура.
    - «[ПередЗапускомМенеджераПотоков](#)» – Значение, возвращенное данным событием.
- «Данные» - Структура.
  - «ОбрабатываемыйОбъект» -
    - При вызове метода «[ОбработатьОбъект\(\)](#)», является 2ым параметром данного метода.
    - При вызове метода «[ОбработатьКоллекцию\(\)](#)», является 2ым параметром данного метода, а именно **часть** данной **коллекции** (тип коллекции и состав остается точно таким же).
  - «ПараметрыДляОбъекта», является 3им параметром вызванного методов;
  - «РезультатОбработки» - Значение, возвращенное событием «[ПриОбработкеПотоком](#)».
  - «СообщениеПользователю» - Строка, сообщения, выведенные во время выполнения события «[ПриОбработкеПотоком](#)».
  - «Статус» - Строка, «Результат».

### Описание:

Является признаком того, что объект или часть объекта (метод «[ОбработатьКоллекцию\(\)](#)»), благополучно завершил обработку.

### Примечание:

«ПараметрыДляОбъекта» - можно расширять и использовать на любом этапе:

- В момент ввода в «Менеджер потоков» (перед вызовом методов «[ОбработатьОбъект](#)», «[ПолучитьГраф\\_Gerhi](#)», «[ПроизвольнаяОбработкаГрафа](#)»);
- В событии «[ПриРасчетеРесурсов](#)»;
- В событии «[ПриОбработкеПотоком](#)»;
- В событиях постобработки «[ПриОбработкеОшибки](#)», «[ПриОбработкеПропуска](#)» и «[ПриОбработкеРезультата](#)».

Более подробно см. Демо базу, обработка: «Обработать объект: Передача частных параметров».

*Ранее называлось «[ПриОбработкеОтвета](#)»*

### Используется:

Доступен с версии 1.0.1

## Событие: При Произвольной Обработке Графа

### Место вызова:

Фоновое задание «Менеджера потоков».

### Условие вызова:

При вызове метода: «[ПроизвольнаяОбработкаГрафа\(\)](#)»

### Частота вызова:

По умолчанию – 1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода «[ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура.
    - [ПередЗапускомМенеджераПотоков](#) – Значение, возвращенное данным событием;
    - [ПриЗапускеМенеджераПотоков](#) – Значение, возвращенное данным событием.
- «Данные» - Структура:
  - Схема – Соответствие:
    - Ключ – УникальныйИдентификатор (КлючДанных)
    - Значение – Структура:
      - Ведомые – Соответствие:
        - ✓ Ключ – УникальныйИдентификатор – Ключ ведомого узла;
        - ✓ Значение – Неопределено.
      - Ведущие – Соответствие (пустое);
      - ОбрабатываемыйОбъект – является 2ым параметром вызванного метода;
      - ПараметрыДляОбъекта - является 3им параметром вызванного метода.

### Описание:

### Примечание:

По умолчанию значению параметра «[ОграничениеРазмера](#)» = 0, в этом случае граф растет до «бесконечности».

Если задать значение параметра «[ОграничениеРазмера](#)» отличным от 0, то при достижении размерности графа этого показателя, будет вызвано данное событие, после чего граф будет полностью очищен. И продолжится его дальнейшее наполнение.

Более подробно см. Демо базу, обработка: «Обработать объект: файлы графа произвольная программа».

### Используется:

Доступен с версии 2.1.0

## Событие: [ПередЗавершениемМенеджераПотоков](#)

### Место вызова:

Фоновое задание «Менеджера потоков».

### Условие вызова:

Всегда.

### Частота вызова:

1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура.
    - [ПередЗапускомМенеджераПотоков](#) – Значение, возвращенное данным событием;
    - [ПриЗапускеМенеджераПотоков](#) – Значение, возвращенное данным событием.
- «Данные» - Неопределено.

### Описание:

### Примечание:

Последнее событие в рамках фонового задания «Менеджера потоков», после чего оно завершает свою работу.

### Используется:

Доступен с версии 1.0.1

## Событие: ПриСохраненииГрафа

### Место вызова:

Основная программа.

### Условие вызова:

При вызове метода: «[ПроизвольнаяОбработкаГрафа\(\)](#)»

### Частота вызова:

По умолчанию – 1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода [«ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура.
    - [ПередЗапускомМенеджераПотоков](#) – Значение, возвращенное данным событием.
- «Данные» - Значение, возвращенное событием [«ПриПроизвольнойОбработкеГрафа»](#)

### Описание:

### Примечание:

Событие вызывается только раз сколько раз было вызвано событие

[«ПриПроизвольнойОбработкеГрафа»](#).

Более подробно см. Демо базу, обработка: «Обработать объект: файлы графа произвольная программа».

### Используется:

Доступен с версии 2.1.0

## Событие: ПриСборкеФрагментовКоллекции

### Место вызова:

Основная программа.

### Условие вызова:

При вызове метода: [«ОбработатьКоллекцию\(\)»](#) И значение параметра [«СобратьКоллекциюИзФрагментов»](#) = Истина

### Частота вызова:

1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода [«ИнициализироватьМенеджерПотоков\(\)»](#);
  - «ПоСобытиям» - Структура.
    - [«ПередЗапускомМенеджераПотоков»](#) – Значение, возвращенное данным событием.
- «Данные» - Массив (элемент массива - Структура).
  - «ОбрабатываемыйОбъект» - является 2ым параметром вызываемого метода, а именно **часть** данной **коллекции** (тип коллекции и состав остается точно таким же);
  - «ПараметрыДляОбъекта» - является 3им параметром вызываемого метода;
  - «РезультатОбработки» - Неопределено;
  - «СообщениеПользователю» - Строка, сообщения, выведенные во время выполнения события [«ПриОбработкеПотоком»](#);
  - «Статус» - Строка: «Результат» / «Ошибка».

### Используется:

Доступен с версии 2.0.4

*Ранее назывался «ПриОбработкеМассиваОбработанныхФрагментовКоллекции»*

## Событие: ПослеЗавершенияМенеджераПотоков

### Место вызова:

Основная программа.

### Условие вызова:

Всегда.

### Частота вызова:

1 раз.

### Структура выходящих параметров события:

- «ИмяСобытия» - Строка. Имя текущего события;
- «Параметры» - Структура. ([ПараметрыМП.Разработчик](#));
  - «Общие» - Структура. Параметры, установленные разработчиком до запуска метода [«ИнициализироватьМенеджерПотоков\(\)](#)»;
  - «ПоСобытиям» - Структура.
    - [«ПередЗапускомМенеджераПотоков»](#) – Значение, возвращенное данным событием.
- «Данные» - Неопределено.

### Примечание:

Самое последнее событие.

### Используется:

Доступен с версии 1.0.1

## Методы и события (матрица вызовов)

Событие	Методы			
	Обработать Объект	Обработать Коллекцию	Получить Граф_Graph	Произвольная ОбработкаГрафа
1 ПередЗапускомМенеджераПотоков	+	+	+	+
2 ПриЗапускеМенеджераПотоков	+	+	+	+
3 ПередЗапускомПотока	+	+	+	+
4 ПриЗапускеПотока	+	+	+	+
5 ПриРасчетеРесурсов	+		+	+
6 ПриОбработкеПотоком	+	+		
7 ПриОбработкеОшибки	+	+		
8 ПриОбработкеПропуска	+			
9 ПриОбработкеРезультата	+	+		
10 ПриПроизвольнойОбработкеГрафа				+
11 ПередЗавершениемМенеджераПотоков	+	+	+	+
12 ПриСохраненииГрафа				+
13 ПриСборкеФрагментовКоллекции		+		
14 ПослеЗавершенияМенеджераПотоков	+	+	+	+
Событие не вызывается				
Событие вызывается	+			
Событие обязательное	+			
Событие обязательное если требуется произвольная сборка	+			

## Ресурсы

Ресурсы позволяют «Менеджеру потоков» связывать объекты между собой, по сколь угодно сложным алгоритмам. Объекты считаются связанными если они обладают хотя бы одним общим ресурсом. Каждый объект может либо не обладать ресурсами, либо иметь их неограниченное количество.

Каждый ресурс может быть составным. В состав ресурса может входить любое количество комбинаций примитивных типов и ссылок.

Для простоты понимания схем введем обозначения:

Объект (тот объект над которым мы хотим произвести обработку): 

Ресурс (разные цвета – разные ресурсы): 

Объект с ресурсами: 

Связь между объектами всегда однонаправленная «от более позднего к более раннему», т.е. первый объект переданный в методы: «[ОбработатьОбъект\(\)](#)» или «[ПолучитьГраф\\_Gephi\(\)](#)» или «[ПроизвольнаяОбработкаГрафа\(\)](#)» всегда будет ведущим для последующего(их) объекта(ов) по совпадающим ресурсам.

Пример:



Кроме того, объект может быть ведущим и для тех объектов с которыми он не пересекается по ресурсам на прямую, но связан через посредника(ов), посредников может быть сколь угодно много.



В данной схеме объект  ни при каких обстоятельствах не будет обработан перед объектом

, хотя по ресурсам они и не пересекаются.

## Правила формирования ресурсов

Ресурсы формируются в событии «[ПриРасчетеРесурсов](#)»

Ресурсы, с точки зрения типов данных 1С - это «массив массивов». На текущий момент в «Менеджере потоков» не предусмотрены методы для формирования ресурсов. Формирование ресурсов полностью возлагается на плечи разработчика и должны быть сформированы по следующей схеме.



Пример:

В демо обработке «Обработать объект» представлен в качестве примера алгоритм связи «Связь по делителям от 4 до 9»:

```

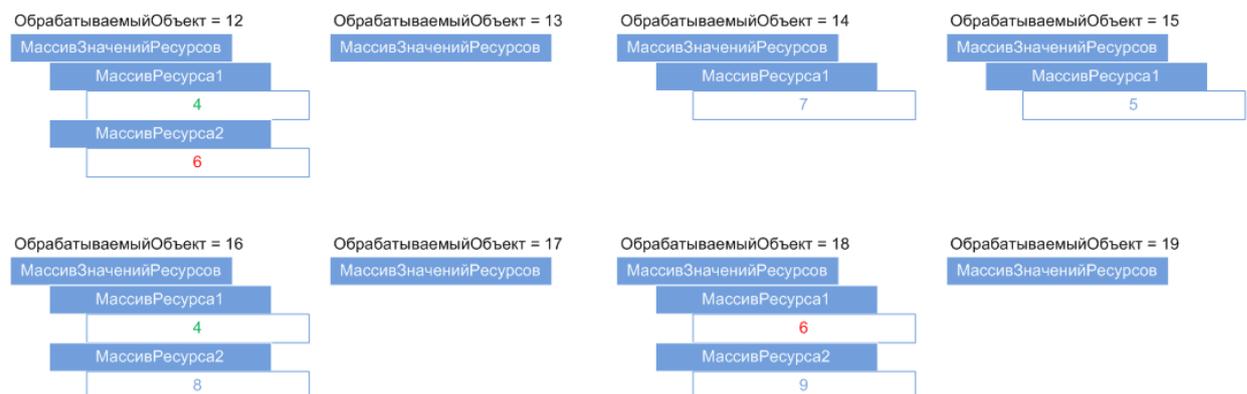
функция ПриРасчетеРесурсов (пПараметрыСобытия)
    МассивРесурсов = Новый Массив;

    ОбрабатываемыйОбъект = пПараметрыСобытия.Данные.ОбрабатываемыйОбъект;
    Для Сч = 4 По 9 Цикл
        Если ОбрабатываемыйОбъект % Сч = 0 Тогда
            МассивРесурса = Новый Массив;
            МассивРесурса.Добавить (Сч);
            МассивРесурсов.Добавить (МассивРесурса);
        КонецЕсли;
    КонецЦикла;

    Возврат МассивРесурсов;
Конецфункции
    
```

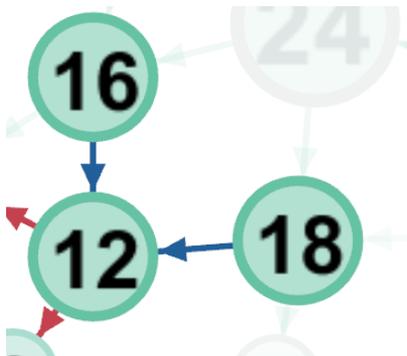
Где «ОбрабатываемыйОбъект» - это число.

По приведенному выше алгоритму для чисел от 12 до 19 будут сформированы такие ресурсы:

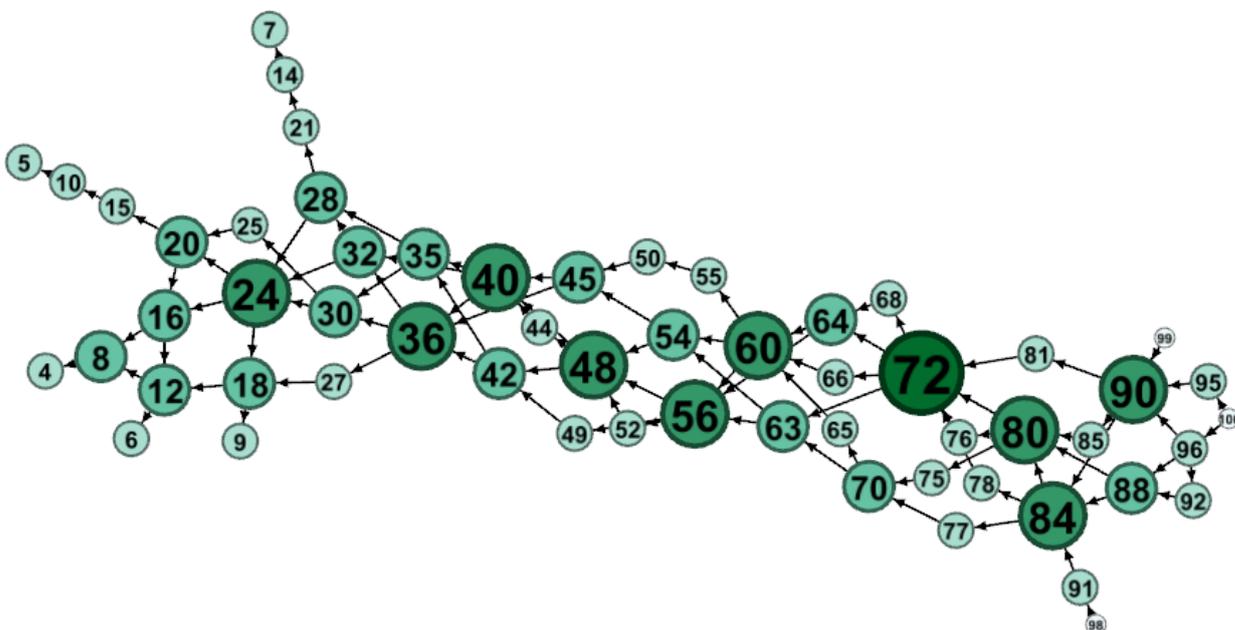


По таким ресурсам «Менеджер потоков» свяжет между собой объекты 12 и 16 (ресурс 4), а также 12 и 18 (ресурс 6).

Что собственно и видно на сформированном графе с помощью программы Gephi (<https://gephi.org/>)

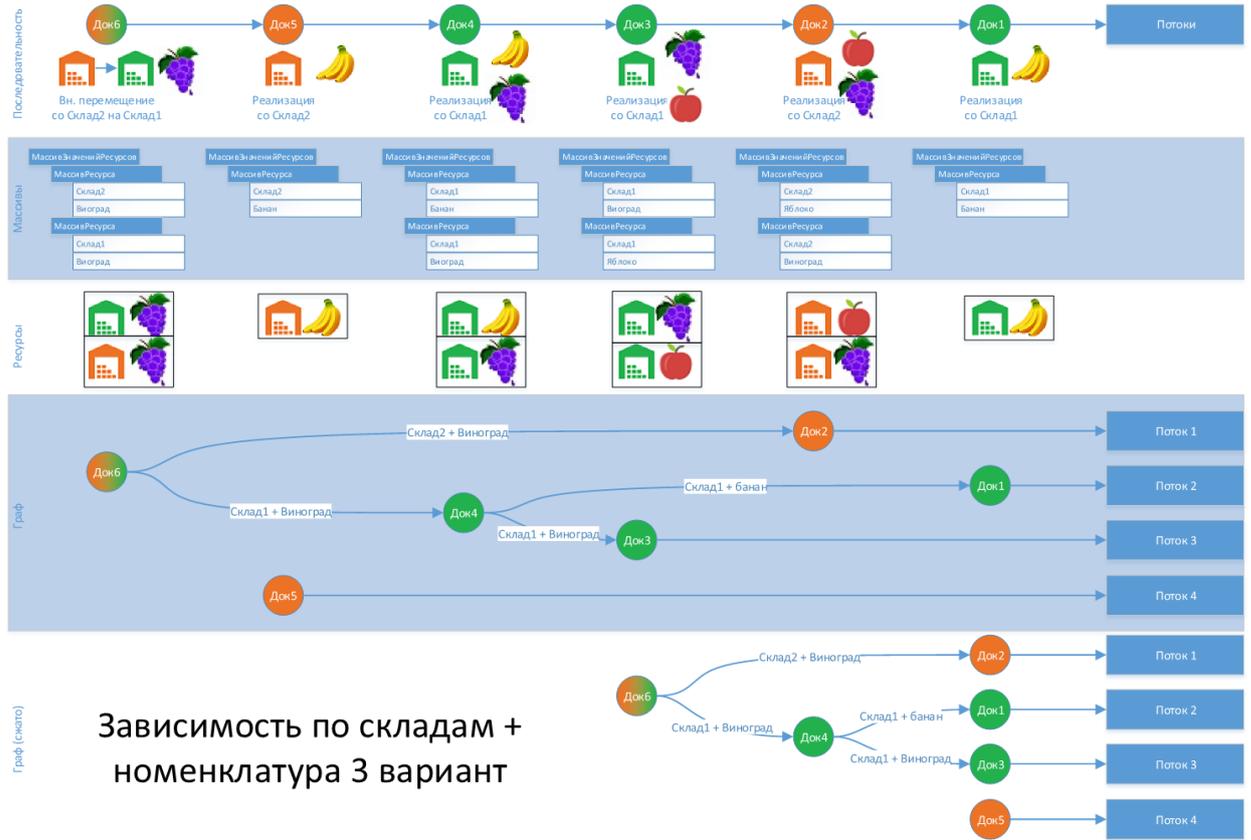


Полный же граф для чисел от 1 до 100 будет выглядеть так:



Файлы для получения такого граф с помощью программы Gephi (<https://gephi.org/>) можно сформировать с помощью обработки из демо примера «Обработать объект: файлы графа Gephi»

А вот пример с составным ресурсом:



Зависимость по складам + номенклатура 3 вариант