Общество с ограниченной ответственностью
"Современные бизнес-аналитические решения"
ИНН/КПП 7724373345/ 772401001
117405, г. Москва, ул. Дорожная, д. 60Б, этаж 0, офис 07
Тел+7(495) 540-46-94
e-mail: mail@modernsolution.ru
https://modernsolution.ru

Инструкция по созданию файла задания

Москва
2023

Содержание

1. Что такое файл задания

Задание – это последовательность шагов по обработке данных, записанная в формате XML в файл с расширением ".grf". Шагом является минимальный алгоритм обработки данных.

Созданием файла занимается пользователь системы – разработчик заданий.

2. Из чего состоит задание

Структура файла задания соответствует стандартной структуре XML документа. Файл задания должен быть синтаксически верным XML документом.

2.1. Структура заданий OneBridge

В системе определены некоторые элементы, которые стоит использовать для корректной передачи информации и отображения графов в инспекторе задач.

После декларации следует указать начальный тег корневого элемента документа <Graph>. В этот элемент помещается все описание алгоритма обработки данных, все используемые шаги, ребра и их метаданные.

За ним следуют строки, описывающие дочерние элементы корневого элемента. Два главных дочерних элемента это <Global> и <Phase>. В элементе <Global> описываются метаданные и параметры подключения.

Система OneBridge обрабатывает данные в виде записей. Каждая запись может состоять из нескольких полей разных типов. Метаданные хранят тип данных этих полей. Метаданные являются частью задания, они содержатся в файле задания и их нужно описывать в элементе <Metadata>, чтобы четко определить типы обрабатываемых данных.

Параметры подключения к базе данных, файлы с настройками, можно указать и подключить в элементе <GraphParameters>

В <Phase> задаются атрибуты шагов задания <Node> и описываются ребра <Edge>. Описание шагов может содержать в себе дочерние элементы <Attr>, в которых описывается метод преобразования записей данных.

Последняя строка файла определяет конец корневого элемента: </Graph>.

На схеме ниже представлена иерархия элементов в файле задания.

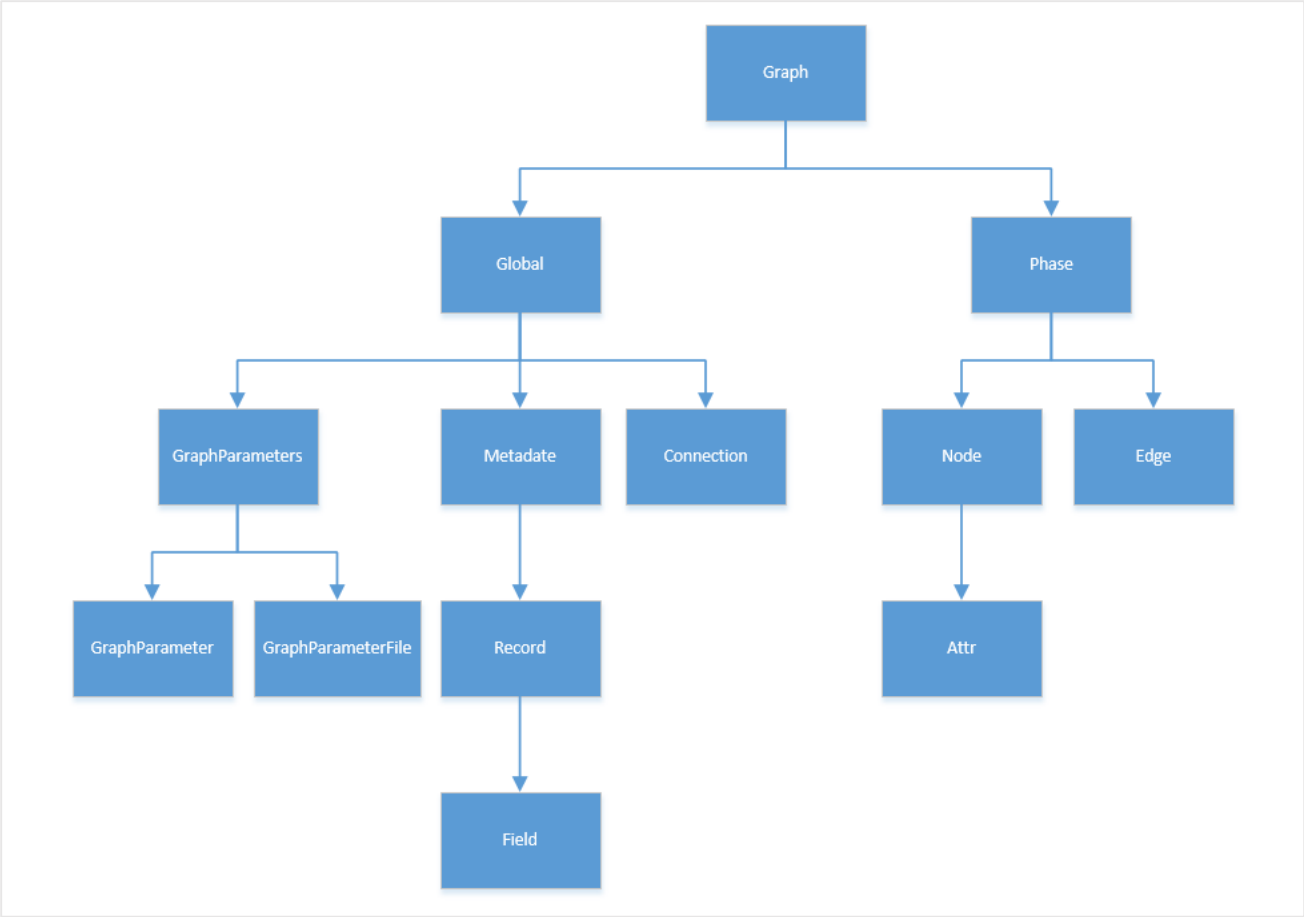


Рисунок 1. Схема вложенности элементов заданий в OneBridge.

3. Описание элементов и их атрибутов.

3.1. Элементы

Таблица 1. Описание элементов файла задания.

Элемент	Родительский элемент	Описание элемента
Graph	нет	Является главным элементом, определяющим граф. Содержит информацию о файле задания. Обязательный тег для отрисовки графа в инспекторе заданий.
Global	Graph	Содержит информацию о файле, не имеет атрибутов. Дочерние элементы:

Элемент	Родительский элемент	Описание элемента
		• Metadata - используемые метаданные; • GraphParameters – параметры графа; • Connection – подключения к базам данных.
Metadata	Global	Определяет тип данных записи
Record	Metadata	Используется для определения символов-разделителей полей и записей для шагов FlatFileReader и FlatFileWriter, которые читают и записывают данные из\в плоские файлы. По умолчанию разделитель полей — "," разделитель строк — "\n", если необходимо использовать другие разделители – нужно задать их в элементе Record. <Record fieldDelimiter=";" recordDelimiter="_">
Field	Record	Содержит имя поля и его тип. Если задан Record, то все Field должны идти внутри него. <Field name="y_coord" type="int"/>
GraphParameters	Global	Содержит элементы, в которых хранится информация для подключения к базам данных или путь к файлу для чтения. Может иметь атрибут `scopeNonce` - дополнительный параметр для secure параметров, например, пароля от базы данных.
GraphParameter	GraphParameters	Хранит параметры для используемых в файле шагов, например, путь к файлу для шага чтения данных. Атрибуты элемента описаны в таблице 2.
GraphParameterFile	GraphParameters	Подключает файл параметров.

Элемент	Родительский элемент	Описание элемента
		Атрибуты описаны в таблице 3.
Connection	Global	Хранит параметры подключения к базе данных.
Phase	Graph	Номер фазы присваивается шагам графа, если есть необходимость запускать часть шагов после завершения выполнения другой части шагов. Фаз в графе может быть несколько, так что им нужно присваивать атрибут number, указывающий очередность выполнения. Каждый граф выполняется параллельно в рамках одного и того же номера фазы; т. е. каждый шаг и каждое ребро с одинаковым номером фазы выполняются одновременно. Если процесс останавливается на какой-то фазе, более высокие фазы не запускаются. Только после успешного завершения всех процессов в рамках одной фазы начнется следующая фаза. Ребра графа, в которых описывается соединение шагов должны быть описаны в одной фазе с используемыми шагами. То есть нельзя объявлять шаги в одной фазе, а связывать их ребром - в другой.
Node	Phase	Описывает атрибуты шага. Атрибуты описаны в таблице 4.
Attr	Node	Описывает логическое выражение для фильтрации и сортировки или метод преобразования данных.
Edge	Phase	Описывает связь между шагами графа. Атрибуты описаны в таблице 5.

3.2. Атрибуты элементов

Таблица 2. Атрибуты тега <GraphParameter> для параметров графа

Название	Обязательный	Описание	Возможные значения
name	да	Имя параметра	name="READ_DIR"
value	нет	Значение параметра	value="test/files/generated"
public	нет	Публичность параметра	Значение по умолчанию: public="false"
required	нет	Обязательность указания значения параметра при запуске задания	Значение по умолчанию : required="false"
secure	нет	Параметр зашифрован.	Значение по умолчанию: secure="false"

- если public="true" и required="true", тогда value игнорируется;
- если public="true" и value не задан, тогда required устанавливается в "true";
- если public="false", то required игнорируется;
- если public="false", то value должно быть задано;
- значение name не может содержать в себе подпоследовательность "${".

Таблица 3. Атрибуты тега <GraphParameterFile> для параметров графа

Название	Обязательный	Описание	Возможные значения
fileURL	да	Путь к файлу с параметрами.	fileURL="${PRM_DIR}/db_01__full_conn.prm"

Таблица 4. Атрибуты тега <Node> для шагов графа

Название	Обязательный	Описание	Возможные значения
id	да	Удобное название шага для использования в атрибутах ребер графа.	id="reader"
guiName	да	Имя шага, отражаемое в инспекторе заданий. Обязательно должно быть	guiName="Copy"

Название	Обязательный	Описание	Возможные значения
		равно значению type текущего шага. С помощью этого атрибута происходит отрисовка шагов в инспекторе заданий.	
guiX	нет	Координата X левого верхнего угла шага для визуального отображения шага в инспекторе задач.	guiX="-132"
guiY	нет	Координата Y левого верхнего угла шага для визуального отображения шага в инспекторе задач.	guiY="212"
type	да	Тип шага. Определяет функциональность данного шага.	Все имеющиеся в системе типы шагов: type="FlatFileReader" type="DbReader" type="FlatFileWriter" type="DbWriter" type="PgDbWriter" type="Trash" type="Sort" type="Filter" type="Gather" type="Copy" type="Concat" type="Map" type="Dedup" type="HashJoin"

Таблица 5. Атрибуты тега <Edge> для ребер графа

Название	Обязательный	Описание	Возможные значения
id	нет	Уникальное название ребра в пределах графа.	id="edge0"
fromNode	нет*	Используется для описания направления ребра в теге <Edge>. Обозначает «из какого порта какого шага выходит ребро»	fromNode="filter_short:0"
toNode	нет*	Используется для описания направления ребра в теге <Edge>. Обозначает «в какой порт какого шага входит ребро»	toNode="sort_long:0"
chunkAmount	нет	кол-во chunk'ов которые могут одновременно находится внутри данного ребра	chunkAmount="42"
chunkSize	нет**	из какого количества записей состоит один chunk	chunkSize="5"
chunkMemSize	нет**	сколько памяти занимает один chunk	chunkMemSize="40_KB"
metadata	нет	Атрибут ребра, определяющий тип данных, передающихся по данному ребру	metadata="Purchase"

* - Хотя бы один из атрибутов должен быть указан.

** chunkSize и chunkMemSize не могут быть заданы одновременно.

4. Пример 1

4.1. Описание задачи

Создать файл с заданием для обработки данных следующим образом:

1) Прочитать данные из нескольких файлов.
2) Объединить считанные потоки данных.
3) Скопировать получившийся поток данных в несколько потоков.
4) Вывести один из потоков в файл.
5) Записи из второго потока отсортировать.
6) Затем отфильтровать.
7) Полученные данные записать в файл.

4.2. Заполнение файла задания

В начале создания файла нужно объявить, что его содержимое является заданием-графом, для этого открываем тег <Graph>.

Далее нужно открыть тег <Global>, в котором будут описаны метаданные и параметры графа.

Далее открываем элемент <Record>, атрибутами которого можно назначить fieldDelimiter – для объявления разделителя полей записи, и recordDelimiter – для указания символа, разделяющего записи.

Дочерним элементу <Record> является пустой элемент <Field>, у которого нет закрывающего тега. Его используем, чтобы указать имя (name) и тип данных (type) для полей записей.

Теперь поочередно закрываем теги </Record>, </Metadata> и </Global>.

Следующий открывающий тег - <Phase> с атрибутом number, указывающим номер фазы для определения очередности выполнения шагов. В этом примере фаза всего одна, ее номер можно не указывать.

Далее открываем три пустых элемента <Node>, атрибуты которых, описывают имя шага (id), входные и выходные файлы для шагов чтения и записи (file), типы шагов (type) и координаты для отрисовки шагов в инспекторе задач.

У элемента <Node> может быть дочерний элемент <Attr>, которому нужно задать имя name, а внутри записать логическое выражение для фильтрации или метод преобразования данных.

Еще нужно указать пустые теги <Edge> для описания ребер графа. В атрибутах этого элемента можно пояснить, какими ребрами какие шаги связать в этом графе и указать ссылку на метаданные, указанные в начале файла в теге <Metadata>.

В конце закрываем элементы </Phase> и </Graph>.

Получился файл задания:

```
<?xml version="1.0" encoding="UTF-8"?>
<Graph>
    <Global>
        <Metadata id="ObjectWithPos">
            <Record fieldDelimiter="|" recordDelimiter="\n">
                <Field name="a" type="string"/>
                <Field name="b" type="integer"/>
                <Field name="c" type="float"/>
            </Record>
        </Metadata>
    </Global>

    <Phase number="0">
        <Node id="reader0" guiX="50"   guiY="100" guiName="FlatFileReader" file="data-
in/others/example_01_in1.txt"   type="FlatFileReader"/>
        <Node id="reader1" guiX="50"   guiY="300" guiName="FlatFileReader" file="data-
in/others/example_01_in2.txt"   type="FlatFileReader"/>
        <Node id="writer0" guiX="1050" guiY="300" guiName="FlatFileWriter" file="data-
out/others/example_01_out1.txt" type="FlatFileWriter"/>
        <Node id="writer1" guiX="650"  guiY="100" guiName="FlatFileWriter" file="data-
out/others/example_01_out2.txt" type="FlatFileWriter"/>
        <Node id="concat"  guiX="250"  guiY="200" guiName="Concat" type="Concat"/>
        <Node id="copy"    guiX="450"  guiY="200" guiName="Copy" type="Copy"/>
        <Node id="sort"    guiX="650"  guiY="300" guiName="FileSortNode" sortKey="c(d)"
type="FileSortNode"/><!--"d" - descending-->

        <Node id="filter"  guiX="850" guiY="300" guiName="FilterNode" type="FilterNode">
            <Attr name="filterExpression">
                <![CDATA[//#Rust:closure
                |input| input.a.chars().count() < 15
                ]]>
            </Attr>
        </Node>
```

```
        <Edge id="edge0" fromNode="reader0:0" toNode="concat:0"/>
        <Edge id="edge1" fromNode="reader1:0" toNode="concat:1"/>
        <Edge id="edge2" fromNode="concat:0"  toNode="copy:0"/>
        <Edge id="edge6" fromNode="copy:0"    toNode="writer1:0" metadata="ObjectWithPos"/>
        <Edge id="edge5" fromNode="copy:1"    toNode="sort:0"/>
        <Edge id="edge3" fromNode="sort:0"    toNode="filter:0"/>
        <Edge id="edge4" fromNode="filter:0"  toNode="writer0:0" metadata="ObjectWithPos"/>
    </Phase>
</Graph>
```

4.3. Результат

На вход были поданы два файла:

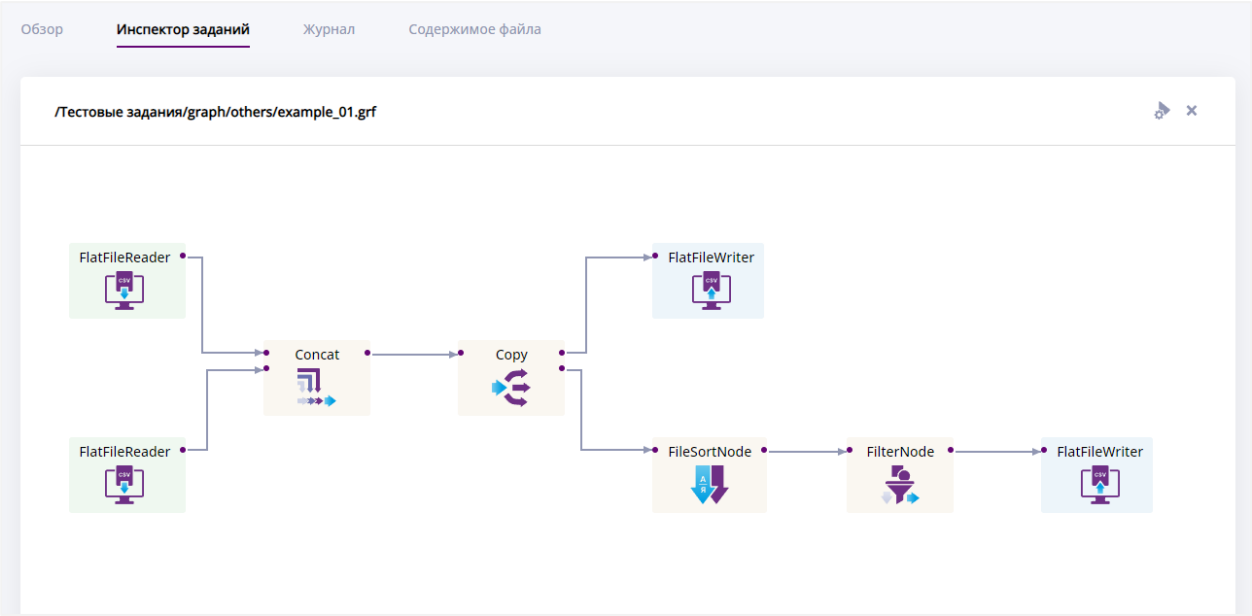
```
файл 1:
        werwr|234|25.45
        krlgkrgw|63|-65.8
        srgehdhdfxyjset___g_jgsn|72465|92745972
        fbsb!efb|-42536356|0
        glsdgh|453453|4524.545
файл 2:
        Лёша на горе рога нашёл|-123|-123
        Около Мити молоко|234|454.4
        Тина барабанит|12|12.12
```

На выходе получены файлы:

```
файл 1 (содержит результат работы задания целиком):
        glsdgh|453453|4524.545
        werwr|234|25.45
        Тина барабанит|12|12.12
        fbsb!efb|-42536356|0.0
        krlgkrgw|63|-65.8
файл 2 (содержит результат конкатенации двух потоков
данных на втором шаге задания):
        werwr|234|25.45
        krlgkrgw|63|-65.8
        srgehdhdfxyjset___g_jgsn|72465|92745970.0
        fbsb!efb|-42536356|0.0
        glsdgh|453453|4524.545
```

```
Лёша на горе рога нашёл|-123|-123.0
Около Мити молоко|234|454.4
Тина барабанит|12|12.12
```

На рисунке ниже представлено графическое представление этого задания в инспекторе задач:



4.4. Атрибуты используемых шагов

Атрибуты FlatFileReader:

Атрибут	Обязательный	Описание	Возможные значения
id	да	Название шага для указания в атрибутах ребра fromNode и toNode. Может быть произвольным.	id="reader"
guiX	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiX="50"
guiY	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiY="100"
guiName	нет	Имя шага, указываемое на графе. Равно type шага. Нельзя изменять.	guiName="FlatFileReader"

Атрибут	Обязательный	Описание	Возможные значения
file	нет	Путь к источнику данных для чтения	file="data-in/others/example_01_in1.txt"
type	да	Тип шага. Определяет функциональность данного шага.	type="FlatFileReader"

Атрибуты Concat:

Атрибут	Обязательный	Описание	Возможные значения
id	да	Название шага для указания в атрибутах ребра fromNode и toNode. Может быть произвольным.	id="concat"
guiX	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiX="50"
guiY	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiY="100"
guiName	нет	Имя шага, указываемое на графе. Равно type шага. Нельзя изменять.	guiName="Concat"
type	да	Тип шага. Определяет функциональность данного шага.	type="Concat"

Атрибуты Copy:

Атрибут	Обязательный	Описание	Возможные значения
id	да	Название шага для указания в атрибутах ребра fromNode и toNode. Может быть произвольным.	id="copy"
guiX	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiX="50"

Атрибут	Обязательный	Описание	Возможные значения
guiY	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiY="100"
guiName	нет	Имя шага, указываемое на графе. Равно type шага. Нельзя изменять.	guiName="Copy"
type	да	Тип шага. Определяет функциональность данного шага.	type="Copy"

Атрибуты FileSortNode:

Атрибут	Обязательный	Описание	Возможные значения
id	да	Название шага для указания в атрибутах ребра fromNode и toNode. Может быть произвольным.	id="sort"
guiX	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiX="50"
guiY	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiY="100"
guiName	нет	Имя шага, указываемое на графе. Равно type шага. Нельзя изменять.	guiName="FileSortNode"
sortKey	да	Ключ сортировки. «d» - «descending» - по убыванию «a» - «ascending» - по возрастанию	sortKey="component(d)"
type	да	Тип шага. Определяет функциональность данного шага.	type="FileSortNode"

Атрибуты FilterNode:

Атрибут	Обязательный	Описание	Возможные значения
id	нет	Название шага для использования в атрибутах ребра fromNode и toNode	id="filter"

Атрибут	Обязательный	Описание	Возможные значения
id	да	Название шага для указания в атрибутах ребра fromNode и toNode. Может быть произвольным.	id="sort"
guiX	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiX="50"
guiY	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiY="100"
guiName	нет	Имя шага, указываемое на графе. Равно type шага. Нельзя изменять.	guiName="FilterNode"
Filter expression	да	Выражение, по которому фильтруются записи. Возвращает логическое значение.	<Attr name="filterExpression"> <![CDATA[//#Rust:closure |input| input.a.chars().count() < 15]]></Attr>
type	да	Тип шага. Определяет функциональность данного шага.	type="FilterNode"

Атрибуты FlatFileWriter:

Атрибут	Обязательный	Описание	Возможные значения
id	нет	Название шага для использования в атрибутах ребра fromNode и toNode.	id="writer"
guiX	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiX="50"
guiY	нет	Координата верхнего левого угла шага для отрисовки графа в инспекторе заданий.	guiY="100"

Атрибут	Обязательный	Описание	Возможные значения
guiName	нет	Имя шага, указываемое на графе. Равно type шага. Нельзя изменять.	guiName="FlatFileWriter"
file URL	да	Путь к файлу, в который должен быть записан результирующий набор данных.	${WRITE_DIR}/out.txt
type	да	Тип шага. Определяет функциональность данного шага.	type="FlatFileWriter"