



FlexiLayout Studio 9.0

Обучение на примерах

Пример 4

Содержание

Краткое описание документа Invoice Multi-page.....	3
Шаг 1. Предварительные настройки.....	3
Шаг 2. Просмотр изображений и результатов предраспознавания.....	4
Шаг 3. Блоки.....	5
Шаг 4. Анализ изображений и определение последовательности поиска элементов и групп элементов.....	6
Шаг 5. Header документа, описание группы InvoiceHeader	7
Шаг 5.1 Поиск заголовка поля Invoice Number, элемент kwInvoiceNumber	7
Шаг 5.2 Поиск заголовка поля Delivery Address, элемент kwDeliveryAddress	9
Шаг 5.3 Поиск заголовка поля Invoice Date, элемент kwInvoiceDate	10
Шаг 5.4 Описание поля Invoice Number, элемент InvoiceNumber	12
Шаг 5.5 Описание поля Invoice Date, элементы grDate, InvoiceDate и InvoiceDateAsString....	14
Шаг 5.6 Описание поля Delivery Address, элементы grAddress, wgAddressAbove, DeliveryAddress.....	16
Шаг 6. Footer документа, описание группы InvoiceFooter.....	19
Шаг 6.1 Поиск заголовка столбца Product Total, элемент ExtraTag	20
Шаг 6.2 Поиск заголовка столбца Description, элемент AddTag	20
Шаг 6.3 Описание итоговой части документа, элемент LongFooter	20
Шаг 6.4 Описание итоговой части документа, элемент ShortFooter	21
Шаг 6.5 Описание итоговой части документа, элемент TextFragment	22
Шаг 7. Описание заголовков столбцов таблицы, группа TableHeader	22
Шаг 7.1 Поиск заголовка столбца Designation, элемент kwDesignation.....	23
Шаг 7.2 Поиск заголовка столбца ExtraQuantity, элемент ExtraQtyTag	23
Шаг 7.3 Поиск заголовка столбца Quantity, элемент kwQuantity	25
Шаг 7.4 Поиск заголовка столбца UnitPrice, элемент kwUnitPrice.....	25
Шаг 7.5 Поиск заголовка столбца Total, элемент kwTotal	26
Шаг 7.6 Поиск заголовка столбца Reference, элемент kwReference	27
Шаг 7.7 Поиск заголовка столбца Sales, элемент kwSales	28
Шаг 7.8 Поиск заголовка столбца Unit, элемент kwUnit	28
Шаг 8. Поиск табличного элемента, элемент InvoiceTable	29
Шаг 9. Поиск поля TotalAmount, описание составного элемента SumGroup.....	32
Шаг 9.1 Поиск заголовка поля Total Amount, элемент kwTotal	33
Шаг 9.2 Описание поля Total Amount, элемент TotalAmount	34
Шаг 10. Поиск поля Company, описание составного элемента CompanyGroup, описание элемента Company.	35
Шаг 11. Экспорт гибкого описания и подключение его к шаблону FlexiCapture.....	37

Краткое описание документа Invoice Multi-page

 **Внимание!** В этом примере рассматривается многостраничный документ.

Данный документ представляет собой типичный многостраничный счет. Его главной особенностью по сравнению с предыдущими примерами является то, что он состоит из нескольких страниц и содержит многостраничную таблицу.

 **Замечание.** Программа FlexiLayout Studio 9.0 позволяет создавать гибкие описания для многостраничных документов. В этом случае весь документ рассматривается как одно целое и поиск элементов может производиться независимо от страницы, на которой элементы находятся. Подробнее см. Справка раздел Многостраничное гибкое описание.

На примере документа **Invoices** мы проследим по шагам процесс создания гибкого описания, познакомимся с некоторыми нестандартными подходами к описанию местоположения объектов, а также научимся находить объекты изображения, соответствующие началу и концу многостраничного документа и многостраничные элементы (в данном примере многостраничную таблицу). Подробнее см. Справка раздел Элементы\Свойства элементов, задающие параметры искомого объекта изображения\Элементы Header и Footer.

Конечной целью создания гибкого описания данного документа в ABBYY FlexiLayout Studio является устойчивое нахождение во всех тестовых документах местоположения следующих полей:

- **Invoice Number**
- **Invoice Date**
- **Delivery Address**
- **Total Amount**
- **Invoice Table** (столбцы **Reference, Designation, Unit, Quantity, Unit Price, Total**)
- **Company**

Шаг 1. Предварительные настройки

1. Запустите программу ABBYY FlexiLayout Studio 9.0.
2. Создайте новый проект (меню **File>New Project...**).
3. В открывшемся диалоге задайте имя проекта **Invoices**.
4. Добавьте изображения в пакет (подробнее см. Пример 1, шаг 2).

 **Замечание.** Тестовые изображения четвертого примера хранятся в <имя диска>:\Documents and Settings\All Users\Application Data\ABBYY\FlexiCapture\9.0\Samples\FlexiLayout Studio\Invoice\InvoicesBatch.

5. Для каждого документа пакета, можно задать эталонную сборку, которая будет отражать желаемое распределение страниц по документам. Эталонная сборка может быть создана вручную или автоматически при добавлении изображений в пакет. Эталонная сборка дает возможность сравнивать результаты сборки страниц в документы при наложении гибкого описания и желаемое распределение страниц по документам. Подробнее об эталонной сборке см. Справка раздел Отладка гибкого описания\Эталонная сборка страниц в документы.

Для того чтобы правильно сгруппировать страницы в документы сначала необходимо их проанализировать и определить страницы, которые относятся к одному документу, а также их порядок. В окне **Batch** страницы одного пакета должны следовать строго одна за другой (первая страница>вторая и т.д.).

 **Замечание.** В случае, если страницы были добавлены в пакет не по порядку и их невозможно собрать в документы, их необходимо перенумеровать. Для этого перетащите мышкой страницу в окне Batch (программа автоматически перенумерует страницы).

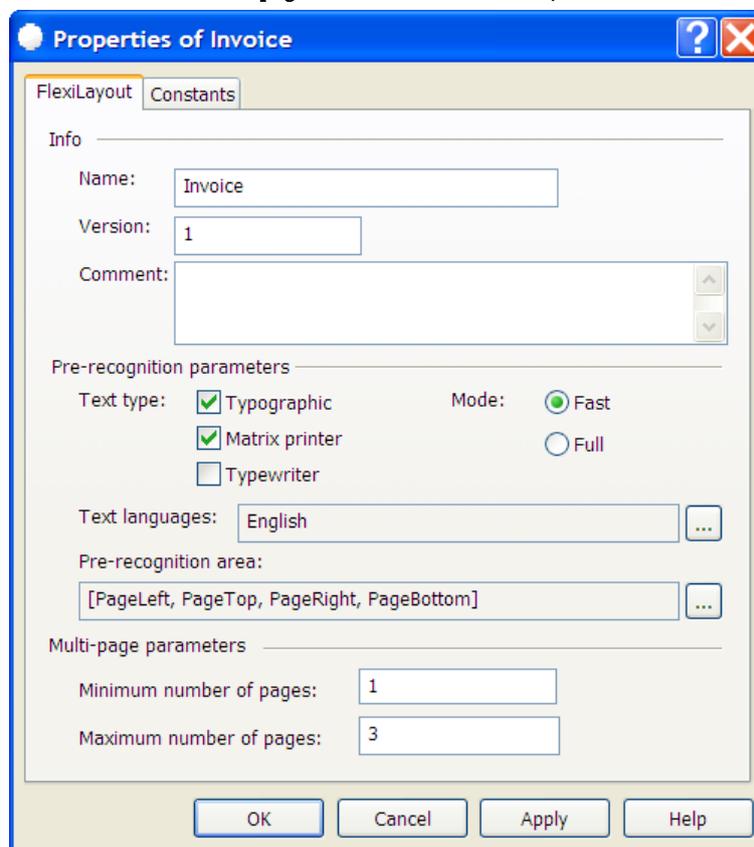
- В окне **Batch** выделите страницы, которые относятся к одному документу.
- Нажмите правую кнопку мыши и выберите в локальном меню пункт **Assemble to Reference Documents**.
- В диалоге **Assemble to Reference Document** выберите опцию **One Document**. Программа сгруппирует все выделенные страницы в один документ.

- Отметьте опцию **First page of document matches with header**. Этой опцией мы указываем программе, что элемент **Header** обязательно должен быть наложен на первую страницу документа.
- Отметьте опцию **Last page of document matches with footer**, так последнюю страницу документа мы будем описывать с помощью элемента **Footer**.
- Нажмите **ОК**. Эталонная разметка отобразится в режиме **Reference Layout**: начало и конец документа выделяются зелеными уголками.

 **Замечание.** В случае, если страницы были собраны неправильно, сборку можно отменить командой **Disassemble Reference Documents**.

6. Настройте свойства гибкого описания:

- **Name** – Invoice.
- **Text Type** - следует выбрать способ заполнения документа **Typographic** и **Matrix Printer**.
- **Text Language** - наш документ является англоязычным, поэтому в качестве языка предраспознавания в списке выберите **English**.
- **Mode** - режим предраспознавания выберите **Fast**.
- **Minimal number of pages** - 1, так как наш документ может состоять минимум из одной страницы.
- **Maximal number of pages** - 3, так как в одном документе может быть максимум 3 страницы.



 **Замечание.** Более подробно о критериях выбора параметров предраспознавания см. Пример 1, шаг 3.

Шаг 2. Просмотр изображений и результатов предраспознавания

Перед созданием элементов гибкого описания необходимо понять, на какие объекты изображений документов мы можем опереться для идентификации начала и конца документа, а также для поиска нужных полей. Опорными объектами могут служить только такие графические и/или текстовые фрагменты изображения, которые устойчиво находятся в процессе предраспознавания изображений документов

Замечание. Для определения начала и конца документа служат специальные элементы **Header** и **Footer**. Они могут состоять из одного или нескольких элементов, строго идентифицирующих первую и последнюю страницы документа соответственно.

Чтобы запустить процесс предраспознавания:

1. Выделите все изображения в пакете.
2. Выберите команду **Prerecognize** (меню **Batch** или локальное меню изображения).

Замечание. Более подробно о процессе запуска предраспознавания см. Пример 1, шаг 4.

Предварительный просмотр результатов предраспознавания всех тестовых страниц показывает, что, поскольку все текстовые объекты, соответствующие заголовкам искомым полям (для полей с заголовком), заголовки колонок таблицы и горизонтальные разделители обнаруживаются устойчиво, на них можно будет опираться при поиске других объектов изображения.

Замечание. Более подробно о просмотре страниц и результатах предраспознавания см. Пример 1, шаг 5, шаг 6.

Теперь можно приступить непосредственно к созданию и настройке опорных элементов и заданию критериев поиска блоков, соответствующих искомым полям. Для этого перейдите в окно **FlexiLayout** в главном окне программы ABBYY FlexiLayout Studio.

Шаг 3. БЛОКИ

Целью создания данного гибкого описания является нахождение местоположения следующих блоков (в случае их наличия на изображении):

- **Invoice Number**
- **Invoice Date**
- **Delivery Address**
- **Total Amount**
- **Invoice Table** (столбцы **Reference, Designation, Unit, Quantity, Unit Price, Total**)
- **Company**

Для удобства и наглядности при создании гибкого описания создайте список всех требуемых блоков:

1. Создайте блок **InvoiceNumber** (тип **Text**), соответствующий полю **Invoice Number**.
2. Создайте блок **InvoiceDate** (тип **Text**), соответствующий полю **Invoice Date**.
3. Создайте блок **DeliveryAddress** (тип **Text**), соответствующий полю **Delivery Address**.
4. Создайте блок **TotalAmount** (тип **Text**), соответствующий полю **Total Amount**.
5. Создайте блок **Company** (тип **Text**), соответствующий полю **Company**.
6. Создайте блок **InvoiceTable** (тип **Table**), соответствующий полю **Invoice Table**.
В диалоге свойств блока на закладке **Columns** последовательно добавьте названия столбцов в следующем порядке: **Reference, Designation, Unit, Quantity, Unit Price, Total**.

Замечание. Указанные названия столбцов и порядок их следования будут соответствовать данному блоку в шаблоне FlexiCapture.

Замечание. Более подробно о создании блока типа **Table** см. Пример 3, Шаг 3.

На этом процедура задания свойств гибкого описания и начальных приготовлений к работе закончена и можно приступить непосредственно к созданию элементов.

Шаг 4. Анализ изображений и определение последовательности поиска элементов и групп элементов

Необходимо выяснить:

- Какова закономерность расположения полей на изображении?
- На какие элементы мы можем опереться в поиске полей?
- Какую выбрать последовательность поиска (поскольку на каждом следующем шаге мы можем опираться только на описанные выше элементы)?

Так как мы создаем описание для многостраничного документа, то в первую очередь необходимо определить какие объекты изображения можно использовать для описания первой и последней страниц документа. Для этого в программе предусмотрены специальные составные элементы **Header** и **Footer**.

- Элемент **Header** должен накладываться по возможности только на первую страницу документа. При наличии шаблонов документов других типов в одном проекте программы FlexiCapture 9.0 Professional данный элемент также играет роль идентификационного (уникального признака данного типа документа).
- Элемент **Footer** должен накладываться только на последнюю страницу документа, он служит для определения конца документа. Рекомендуется создавать в этой группе обязательные подэлементы, которые не позволят наложить Footer на промежуточную страницу документа.

Анализ изображений документов показывает, что:

1. На первой странице документов располагается группа полей **Invoice Number, Invoice Date, Delivery Address**. При этом заголовок поля **Invoice Number** всегда присутствует в начале каждого документа, в то время как поля **Invoice Date** и **Delivery Address** могут и отсутствовать.
 - Поля **Invoice Number** и **Invoice Date** могут располагаться или справа от соответствующего заголовка или под ним.
 - **Delivery Address** тоже будем искать справа и под соответствующим заголовком, строго ограничив область поиска. Кроме этого нам понадобится дополнительный элемент для ограничения снизу.
 - Так как значения этих полей присутствует не на всех изображениях, то для ускорения процедуры наложения шаблона, можно задать дополнительное условие – не искать значение поля, если не найден его заголовок.
2. Мы можем использовать группу этих полей как идентификатор нашего документа. Поэтому эти поля будем описывать в составе predeterminedного в программе группового элемента **Header** (назовем его **InvoiceHeader**).
3. Последняя страница документа характеризуется словами **TOTAL AMOUNT MUST, Carried over, Total CHF, TOTAL**, которые находятся ниже таблицы. Но они не уникальны и в наших документах могут встречаться и в заголовке, и в теле таблицы. Поэтому для их поиска дополнительно потребуются опорные элементы, например заголовки некоторых столбцов, опираясь на которые, мы можем описать правильную область поиска наших слов.
4. Элементы, которыми мы будем описывать последнюю страницу, мы будем создавать в составе predeterminedного в программе группового элемента **Footer** (назовем его **InvoiceFooter**).
5. Для того чтобы **Footer** накладывался только на последнюю страницу документа, он должен содержать в себе обязательный элемент. Так как слова, которые идентифицируют последнюю страницу (см. п.3), находятся на каждой последней странице каждого документа, то элемент, описывающий их, сделаем обязательным.
6. Таблица (назовем ее **InvoiceTable**) в наших документах всегда начинается на первой странице и заканчивается на последней. Кроме этого, она всегда ограничена сверху заголовками ее столбцов на первой странице. А ограничение снизу (на последней странице) легче всего задать через вспомогательный элемент, например, мы можем использовать обязательный элемент из группы **InvoiceFooter**.

 **Замечание.** Множество всех страниц документа называется многостраничным полотном. Многостраничное полотно образуется при склеивании страниц документа сверху вниз, без стыков и зазоров, левые границы всех страниц лежат на одной прямой, проходящей через точку (0, 0). Порядок склейки страниц определяется порядком страниц в пакете. Поэтому мы можем задать только начало таблицы – ее **header** на первой странице и конец – **footer** на последней. Программа будет искать таблицу на протяжении всего документа, всего многостраничного полотна.

7. Название компании **Company** мы будем искать всегда на первой странице документа, причем только в первой ее трети.
8. Заголовок поля **Total Amount** всегда располагается на последней странице под таблицей. Значение поля будем искать или правее или ниже заголовка.

Шаг 5. Header документа, описание группы InvoiceHeader

В группе **InvoiceHeader** мы опишем элементы, которые помогут нам идентифицировать первую страницу документа.

- Для описания поиска ключевых слов (keywords) заголовков полей **Invoice Number**, **Delivery Address** и **Invoice Date** будем создавать следующие элементы типа **Static Text**:
 - Элемент **kwInvoiceNumber** - подробная инструкция в **Шаге 5.1**.
 - Элемент **kwDeliveryAddress** - подробная инструкция в **Шаге 5.2**.
 - Элемент **kwInvoiceDate** - подробная инструкция в **Шаге 5.3**.
- Поиск полей **Invoice Number** и **Invoice Date** будем осуществлять правее и под соответствующим заголовком с помощью элементов:
 - InvoiceNumber**, типа **Character String** - подробная инструкция в **Шаге 5.4**.
 - InvoiceDate**, типа **Date** (в случае хорошего качества изображения) и **InvoiceDateAsString**, типа **Character String**, (в том случае, если поиск элемента **InvoiceDate** оказался неуспешным), объединив их в составном элементе **grDate** - подробная инструкция в **Шаге 5.5**).

 **Замечание.** Более подробно про поиск даты в случае плохого качества изображений см. Справка раздел Советы и рекомендации.

- Для определения поля **Delivery Address**, которое является многострочным, будем использовать элемент **DeliveryAddress** типа **Text Fragment** (подробная инструкция в **Шаге 5.6**), задав область поиска самым строгим образом.
В качестве ограничителя области поиска снизу будем использовать элемент **wgAddressAbove** типа **White Gap**. (подробная инструкция в **Шаге 5.6**)
И объединим все эти элементы, служащие описанию местоположения поля **Delivery Address**, в отдельную группу - **grAddress**.

Шаг 5.1 Поиск заголовка поля Invoice Number, элемент kwInvoiceNumber

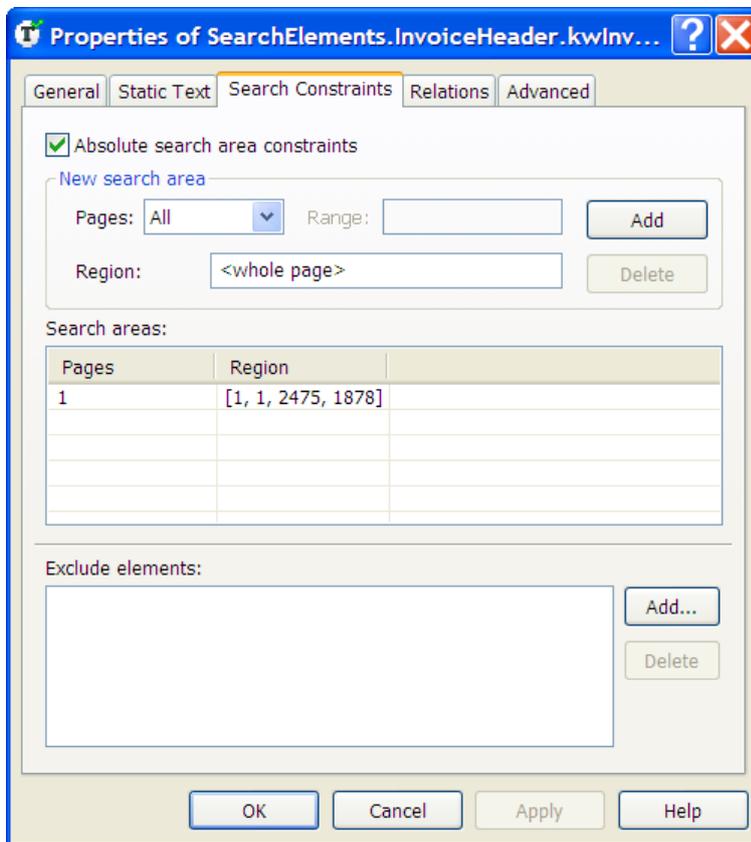
Анализ изображения показывает:

- Заголовок поля **Invoice Number** присутствует на всех изображениях и отличает их от других типов документов. Поэтому сделаем заголовок поля **Invoice Number** обязательным, то есть идентификационным элементом для данного типа документов. Подробнее см. Справка раздел Элементы\Обязательные и необязательные элементы.
- Заголовок на изображениях представлен фразами **Invoice no.**, **Credit note No**, **Invoice Number.**, **Invoice Number**. Поэтому составим нашу поисковую строку из всех возможных вариантов, при этом запишем каждую фразу слитно, чтобы ограничения на количество ошибок относились ко всей фразе целиком, а не к отдельным словам.
- Нас интересует поле только с первых страниц документов, причем только в первой части первой страницы (на первых страницах некоторых документов есть статический текст по содержанию похожий на наш заголовок, но располагается он ближе к концу страницы).

Чтобы создать элемент **kwInvoiceNumber**:

- В составе элемента **InvoiceHeader** создайте элемент **kwInvoiceNumber** (тип **Static Text**).
- Отмените заданное по умолчанию свойство **Optional** на закладке **General** (поскольку данный элемент мы решили сделать обязательным).
- Выберите закладку **Static Text**.
- В поле **Search Text** укажите значение искомого текста: **InvoiceNo.|CreditNoteNo|InvoiceNumber:|InvoiceNumber**
- Задайте значение свойства **Max error percentage** равным 20.
- Выберите закладку **Search Constraints** и задайте область поиска прямоугольником в абсолютных координатах на первой странице:

- отметьте опцию **Absolute search area constraints**,
- в поле **Pages**: из списка выберите **Range**,
- в поле **Range**: укажите 1,
- в поле **Search region**: укажите прямоугольник [1, 1, 2475, 1878]
- задав свойств области поиска нажмите кнопку **Add**. Область появится в списке **Search areas**.



7. Выберите закладку **Advanced**
8. В поле **Advanced pre-search relations** задайте дополнительные параметры:

```
WholeWordMode: true; // в качестве гипотез будут рассматриваться только целые текстовые объекты.
MaxGapInLine: 50dt; // максимально допустимое расстояние между соседними символами.
```

9. Задав дополнительную проверку в поле **Advanced post-search relations** мы можем повлиять на качество сформированных гипотез, например, в данном случае такую: *если элемент найден, вычислить разницу между эталонной шириной и шириной региона найденного элемента и проверить ее на вхождение в нечеткий интервал*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not IsNull then
{ FuzzyQuality: 600dt - width, {-70000,0,0,70000}*dt;
}
```

//такая проверка означает, что чем больше разница, тем выше штрафующий коэффициент

Замечание. Для описания положения объектов на странице и расстояний между ними используется двумерная система координат с горизонтальной и вертикальной осями. Точка отсчета (0, 0) координатных осей расположена в левом верхнем углу изображения. Положительное направление оси абсцисс - слева направо, оси ординат - сверху вниз.

10. Протестируйте изображения, временно исключив элемент **InvoiceFooter**.
 - Замечание.** Для того чтобы исключить элемент выделите его в дереве элементов и в меню FlexiLayout выберите команду **Disable**.

Шаг 5.2 Поиск заголовка поля *Delivery Address*, элемент *kwDeliveryAddress*

Анализ изображения показывает:

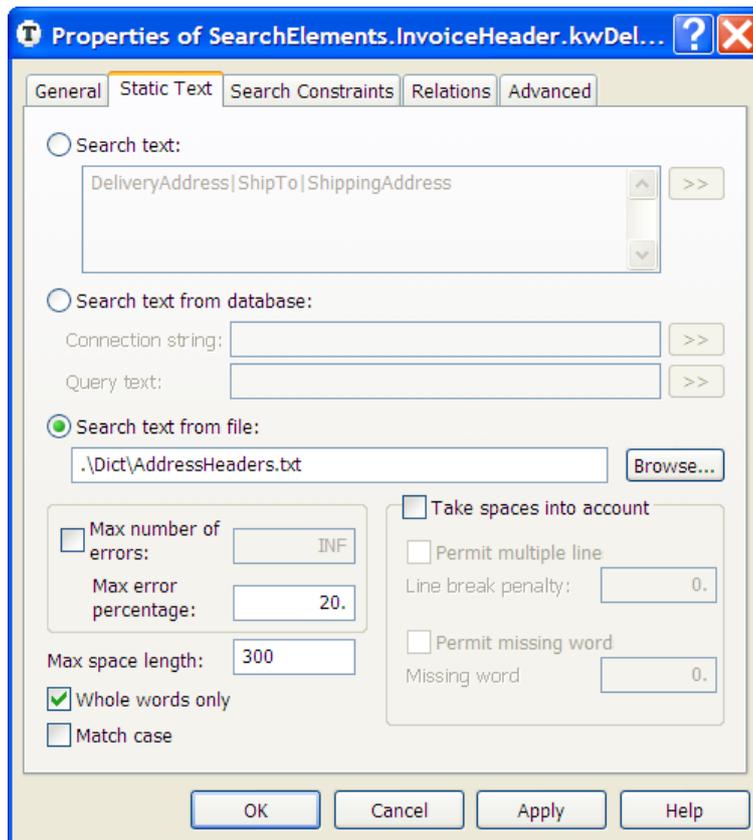
- Искомое поле находится всегда на первой странице документов, причем только в первой части первой страницы. Поэтому область поиска ограничим первой страницей, прямоугольником в абсолютных координатах.
- Поиск текста будем осуществлять на основе текста из файла **AddressHeaders.txt**.
Замечание. В интерфейсе программы и в языке гибких описаний поддерживается возможность обращения к базам данных и текстовым файлам, например, для задания набора строк для поиска на изображении. Текстовые файлы могут быть как в формате ANSI, так и в формате Unicode. Регистр написания не влияет на поиск и используется только для повышения читабельности. Каждый вариант текстовой строки для поиска должен быть определен на новой строке. Пустые строки игнорируются.
- Число ошибок в заголовке можно ограничить 20 процентами, что в нашем случае будет означать не более 3х ошибок во всей фразе.
Замечание. Подбор значения допустимого числа ошибок является творческим процессом и осуществляется путем подбора. Более подробно о параметрах элемента типа **Static Text** см. Справка, раздел Элементы\Свойства элементов, задающие параметры искомого объекта изображения\Static Text.

Чтобы создать элемент **kwDeliveryAddress**:

1. В составе элемента **InvoiceHeader** создайте элемент **kwDeliveryAddress** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. Отметьте опцию **Search Text from file:** и укажите путь к файлу **AddressHeaders.txt**.

Замечание. Файл **AddressHeaders.txt** хранится в папке
 <имя диска>:\Documents and Settings\All Users\Application Data\ABBYY\FlexiCapture\9.0\Samples\FlexiLayout Studio\Invoice\Dict.

Обращение к текстовому файлу производится по имени файла, которое может быть как абсолютным, так и задаваться относительно каталога проекта FlexiCapture (содержащего файл *.fproj) или каталога пакета проекта FlexiCapture Studio (содержащего изображения).



4. Задайте значение свойства **Max error percentage** равным 20.
5. Выберите закладку **Search Constraints** и задайте область поиска прямоугольником в абсолютных координатах на первой странице:
 - отметьте опцию **Absolute search area constraints**,
 - в поле **Pages**: из списка выберите **Range**,
 - в поле **Range**: укажите 1,
 - в поле **Search region**: укажите прямоугольник [1, 1, 2475, 1878] и нажмите кнопку **Add**. Область появится в списке **Search areas**.
6. На закладке **Advanced** в поле **Advanced pre-search relations** задайте WholeWordMode: true;
// в качестве гипотез будут рассматриваться только целые текстовые объекты.
7. Протестируйте изображения (временно исключите элемент **InvoiceFooter**).

Шаг 5.3 Поиск заголовка поля *Invoice Date*, элемент *kwInvoiceDate*

Анализ тестовых изображений показывает:

- Заголовок поля **Invoice Date** представлен коротким и очень распространенным словом **Date**. Следовательно, требуется максимально сузить область поиска.
- Заголовок поля **Date** (в случае наличия на изображении этого поля) располагается очень близко к заголовку поля **Invoice Number**.

Чтобы создать элемент **kwInvoiceDate**:

1. В составе элемента **InvoiceHeader** создайте элемент **kwInvoiceDate** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Date**.

4. Оставьте значение свойства **Max error percentage** равным 30 (что для нашего 4х буквенного слова означает ошибку не более чем в одной букве).
5. Выберите закладку **Relations**.
6. На закладке **Relations** укажите, что ключевые слова заголовка могут находиться не выше заголовка поля **Invoice Number**, чем на расстоянии 30 dot от его верхней границы, но и не ниже чем на расстоянии 700 dot от его нижней границы:

- **Below** верхней границы региона элемента **kwInvoiceNumber**, **Offset = -30**, **Element border = Top**.
- **Above** нижней границы региона элемента **kwInvoiceNumber**, **Offset = -700**, **Element border = Bottom**.

 **Замечание.** После создания нового отношения нажмите кнопку **Add** -оно появится в списке **Relations**.

 **Замечание.** При подборе значений для отступов (**offset**) бывает полезно обратить внимание на геометрические свойства (границы и размеры) опорных и искомых объектов изображения в режиме просмотра результатов предраспознавания. Для этого удобно пользоваться инструментом измерения параметров – **Measure Rectangle**. Значения для отступов (**offset**) подбираются опытным путем.

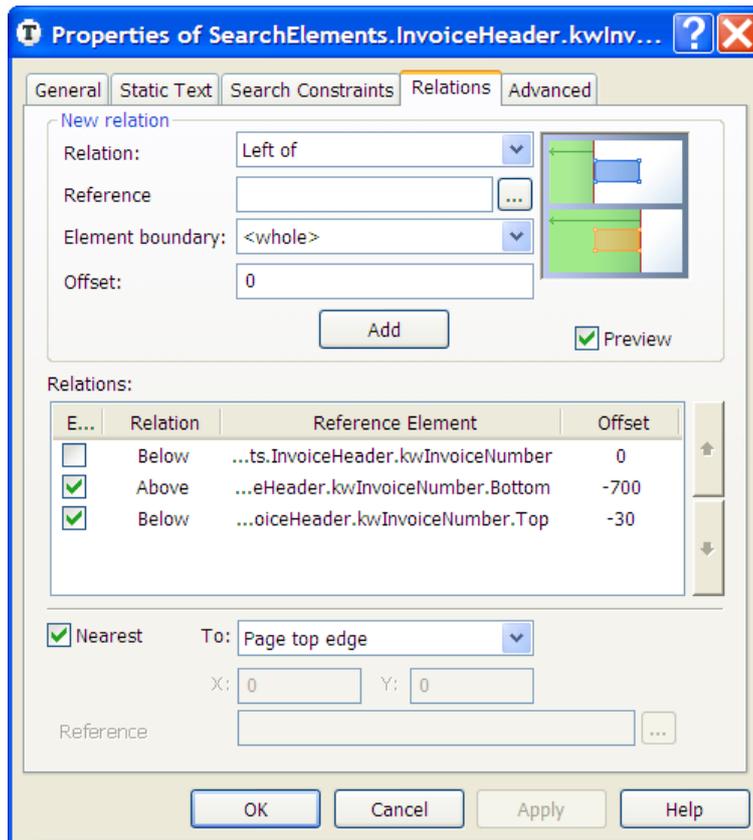
7. Протестируйте изображения (временно исключите элемент **InvoiceFooter**).

Наложение гибкого описания показывает, что

- на первой странице некоторых документов для **kwInvoiceDate** было построено несколько гипотез и выбрана не та, которая нам нужна. Чтобы исключить подобную ошибку надо скорректировать свойства элемента.
- повторно проанализировав наши изображения можно заметить, что интересующее нас поле находится ближе всего к верхней границе страницы.

Для того чтобы задать дополнительное условие поиска:

1. Откройте диалог **Properties** со свойствами элемента **kwInvoiceDate**.
2. Выберите закладку **Relations**.
3. Отметьте опцию **Nearest** и из выпадающего списка **To**: выберите **Page top edge**.



Повторное тестирование показывает, что заголовок поля **Invoice Date** обнаруживается устойчиво.

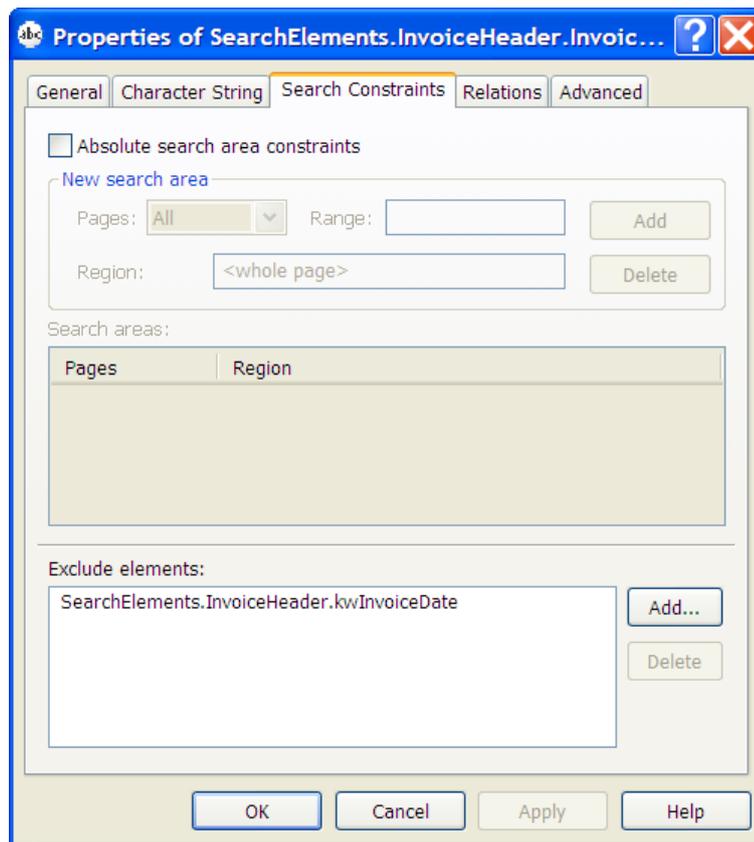
Шаг 5.4 Описание поля *Invoice Number*, элемент *InvoiceNumber*

Поле **Invoice Number** на некоторых документах располагается правее заголовка и на том же уровне, а на других - под заголовком (в первом и втором документах, значение поля указано непосредственно под заголовком).

Чтобы создать элемент **InvoiceNumber**:

1. В составе элемента **InvoiceHeader** создайте элемент **InvoiceNumber** (тип **Character String**).
Замечание. Совпадение имени элемента с именем блока, соответствующего полю **InvoiceNumber**, не обязательно, но удобно для дальнейшей работы с гибким описанием.
2. Выберите закладку **Character String**.
3. Задайте алфавит
 -./0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
4. Задайте значение параметра **Percentage of non-alphabet characters** равным 30%.
5. В поле **String Length** задайте нечеткий интервал {-1, 3, 20, INF}, оценивающий длину цепочки символов. Предполагаем, что все возможные значения будут находиться в интервале от 3 до 20 символов. Любая гипотеза, входящая за рамки данного интервала, будет штрафоваться.
6. На некоторых изображениях в область поиска поля **InvoiceNumber** попадает уже найденный элемент **kwInvoiceDate**. Для того чтобы не рассматривать значение этого элемента в качестве гипотезы поля **InvoiceNumber**, исключим этот элемент из области поиска:

- Нажмите кнопку **Add** рядом с полем **Exclude elements**;
- В списке элементов выберите **kwInvoiceDate**;
- Нажмите кнопку **Ok**.
- После этого в поле **Exclude elements**: вы увидите строку **SearchElements.InvoiceHeader.kwInvoiceDate**;



7. Перейдите на закладку **Advanced**.
8. В поле **Advanced pre-search relations** задайте дополнительные условия поиска: если заголовок **kwInvoiceNumber** не найден, то поле **InvoiceNumber** не искать, в противном случае задайте область поиска поля массивом прямоугольников: один справа от заголовка, другой - под заголовком с некоторым отступом. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
if kwInvoiceNumber.IsNull then Dontfind();
else
{ WholeWordMode: true;
  MaxGapInLine: 30dt;
```

```
  RectArray DataRegion;
```

```
  Let r1= Rect (kwInvoiceNumber.Rect.Right, kwInvoiceNumber.Rect.Top -20dt, kwInvoiceNumber.Rect.Right +
650dt, kwInvoiceNumber.Rect.Bottom + 50dt);
  // ограничим область поиска прямоугольником справа от заголовка поля InvoiceNumber
```

```
  Let r2 = Rect (kwInvoiceNumber.Rect.Left - 100dt, kwInvoiceNumber.Rect.Bottom, kwInvoiceNumber.Rect.Right
+ 100dt, kwInvoiceNumber.Rect.Bottom + 100dt);
  // ограничим область поиска прямоугольником снизу от заголовка поля InvoiceNumber
```

```
  DataRegion = RectArray (r1);
  DataRegion.Add (r2);
  RestrictSearchArea (DataRegion);
}
```

9. Область поиска значения поля обозначена двумя прямоугольниками – справа от заголовка и под ним. Причем, если значение поля располагается справа от его заголовка, то на большинстве тестовых изображений под заголовком присутствует какой-то текст (чаще всего заголовок и значение поля даты). Так как на большинстве изображений этот ненужный текст попадает в область поиска поля **InvoiceNumber**, программа создаст для него гипотезу. Чтобы повлиять на качество такой гипотезы, в поле **Advanced post-search relations** задайте дополнительные параметры проверки: чем значение поля ниже заголовка, тем сильнее штрафовать гипотезу. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
if not IsNull then
{ FuzzyQuality: Top - kwInvoiceNumber.Rect.Top, {-30000,0,0,10000}*dt;
}
// Это означает, что чем дальше верхняя граница поля от верхней границы заголовка, тем сильнее будет оштрафована гипотеза. В данном случае наибольшее качество будет у гипотезы того элемента, верхняя граница которого совпадает с верхней границей заголовка.
```

10. Протестируйте изображения (временно исключите элемент **InvoiceFooter**).
11. Задайте местоположение блока **InvoiceNumber** в виде прямоугольника региона элемента **InvoiceNumber**, увеличенного по высоте и ширине на 5 dot. Для этого выберите опцию **Region Expression** и задайте следующее выражение:

```
Rect outputRect;
outputRect = InvoiceHeader.InvoiceNumber.Rect;
IsNull = InvoiceHeader.InvoiceNumber.IsNull;
// Считать регион блока найденным, если найден элемент InvoiceNumber

OutputRegion = outputRect;
OutputRegion.Inflate (5dt, 5dt);
```

Шаг 5.5 Описание поля *Invoice Date*, элементы *grDate*, *InvoiceDate* и *InvoiceDateAsString*

Анализ изображения показывает:

- В некоторых документах поле **Invoice Date** располагается правее заголовка и на том же уровне, на других - под заголовком. Поэтому область поиска ограничим прямоугольниками справа и снизу заголовка.
- Для поиска даты воспользуемся элементом типа **Date**. Дополнительно зададим проверку – если заголовок не найден, то значение искать не нужно.
- На практике часто встречаются изображения, когда текст даты распознаётся не очень хорошо. Это может быть связано с различными дефектами сканирования или не предусмотренным форматом даты и т.д. Поэтому на тот случай, если поиск даты элементом типа **Date** не даст результатов, дополнительно опишем элемент типа **Character String**.

Для того чтобы задать настройки, общие для всех этих элементов, нам потребуется составной элемент.

Чтобы создать элемент **grDate**:

1. В составе элемента **InvoiceHeader** создайте элемент **grDate** (тип **Group**).
2. Перейдите на закладку **Advanced** и задайте дополнительные условия поиска: *ограничить область поиска поля массивом прямоугольников: один справа от заголовка, другой - под заголовком с некоторым отступом*. На языке описания FlexiLayout данное условие выглядит следующим образом:

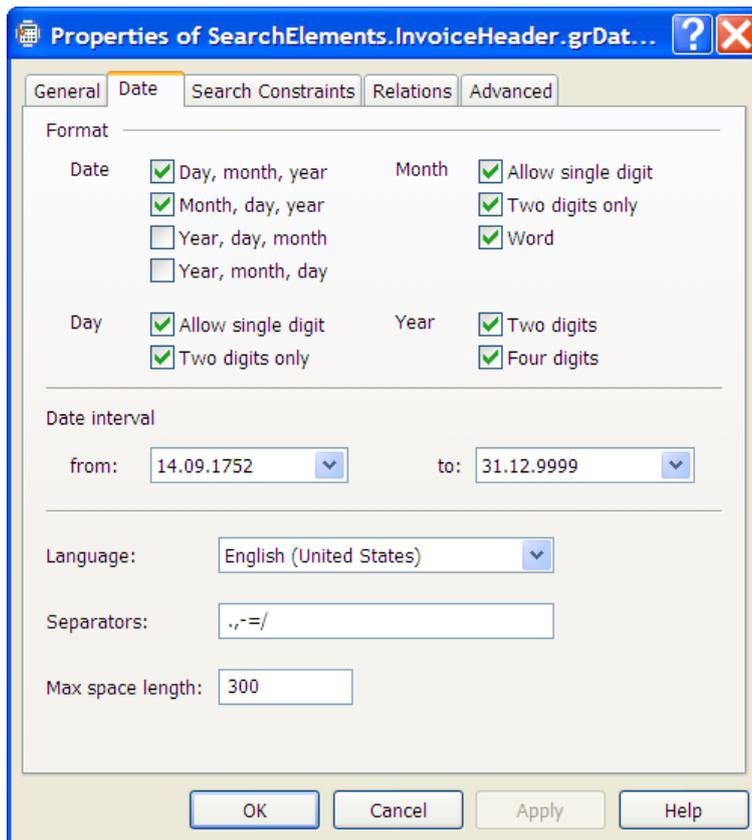
```
RectArray DataRegion;

Let r1= Rect (kwInvoiceDate.Rect.Right, kwInvoiceDate.Rect.Top -20dt, kwInvoiceDate.Rect.Right + 650dt,
kwInvoiceDate.Rect.Bottom + 50dt);
Let r2 = Rect (kwInvoiceDate.Rect.Left - 150dt, kwInvoiceDate.Rect.Bottom, kwInvoiceDate.Rect.Right + 100dt,
kwInvoiceDate.Rect.Bottom + 100dt);

DataRegion = RectArray (r1);
DataRegion.Add (r2);
RestrictSearchArea (DataRegion);
```

Чтобы создать элемент **InvoiceDate**:

1. В составе элемента **InvoiceHeader.grDate** создайте элемент **InvoiceDate** (тип **Date**).
2. Выберите закладку **Date**.
3. Для элемента **InvoiceDate** на закладке **Date** следует указать все допустимые форматы записи даты:



4. На некоторых изображениях в область поиска поля **InvoiceDate** попадают уже найденные элементы **kwInvoiceNumber** и **InvoiceNumber**. Для того чтобы не рассматривать значение этих элементов в качестве гипотез поля **InvoiceDate**, исключим эти элементы из области поиска:

- Нажмите кнопку **Add** рядом с полем **Exclude elements**,
- В списке элементов выберите **kwInvoiceNumber**,
- Нажмите кнопку **Ok**.
- После этого в поле **Exclude elements**: вы увидите строчку **SearchElements.InvoiceHeader.kwInvoiceNumber**;

Повторите эти действия для элемента **SearchElements.InvoiceHeader.InvoiceNumber**.

5. Выберите закладку **Advanced**.
6. Поле **Invoice Date** является необязательным элементом. Но в случае отсутствия даты в документе (поле **Invoice Date**) всегда присутствует и заголовок даты, описанный нами ранее с помощью элемента **kwInvoiceDate**. Таким образом, в поле **Advanced pre-search relations** можно задать дополнительные условия поиска: *Искать объект изображения, только если найден элемент kwInvoiceDate*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If InvoiceHeader.kwInvoiceDate.IsNull
Then DontFind();
```

7. Протестируйте изображения (временно исключите элемент **InvoiceFooter**).

*На случай изображений низкого качества, когда результаты распознавания не подходят ни под один из стандартных параметров для элемента **Date**, дополним гибкое описание альтернативным элементом, ищущим поле **Invoice Date**, с менее строгими условиями.*

Для того чтобы создать элемент **InvoiceDateAsString**:

1. В составе элемента **InvoiceHeader.grDate** создайте элемент **InvoiceDateAsString** типа **Character String**.

- Выберите закладку **Character String**.
- Задайте алфавит
`,-./0123456789ABCDEFGHIJKLMNORSTUVWXYZ abcdefghijklmnopqrstuvwxyz`
- Процент неалфавитных символов задайте равным 30%.
- В поле **String Length** задайте нечеткий интервал {-1, 8, 14, INF}, оценивающий длину цепочки символов. Предполагаем, что все возможные значения будут находиться в интервале от 8 до 14 символов. Любая гипотеза, выходящая за рамки данного интервала, будет штрафоваться.
- В поле **Max space length** задайте значение 20, ограничивающее допустимый размер просвета в текстовой строке в 20 dot;
- Не накладывайте других ограничений на параметры элемента, оставьте значения по умолчанию.
- Выберите закладку **Advanced**.
- Поскольку элемент **InvoiceDateAsString** мы хотим искать только в том случае, если элемент **InvoiceDate** не обнаружен на изображении, в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *Искать объект изображения, только если не найден элемент InvoiceDate*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If Not InvoiceDate.IsNull Then DontFind;
```

- Зададим аналогичные **InvoiceDate** дополнительные условия поиска: *Искать объект изображения, только если найден элемент kwInvoiceDate. Искать объект изображения, ближайший к элементу kwInvoiceDate*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If InvoiceHeader.kwInvoiceDate.IsNull Then DontFind;  
Nearest: InvoiceHeader.kwInvoiceDate;
```

- Задайте местоположение блока **InvoiceDate** в виде прямоугольника региона найденного элемента **InvoiceDate** или **InvoiceDateAsString**, увеличенного по высоте и ширине на 5 dot. Для этого выберите опцию **Region Expression** и задайте следующее выражение:

```
Rect outputRect;  
if not InvoiceHeader.grDate.InvoiceDate.IsNull then  
  outputRect = InvoiceHeader.grDate.InvoiceDate.Rect;  
else  
{ outputRect = InvoiceHeader.grDate.InvoiceDateAsString.Rect;  
  IsNull = InvoiceHeader.grDate.InvoiceDateAsString.IsNull;  
}  
OutputRegion = outputRect;  
OutputRegion.Inflate (5dt, 5dt);
```

Шаг 5.6 Описание поля *Delivery Address*, элементы *grAddress*, *wgAddressAbove*, *DeliveryAddress*.

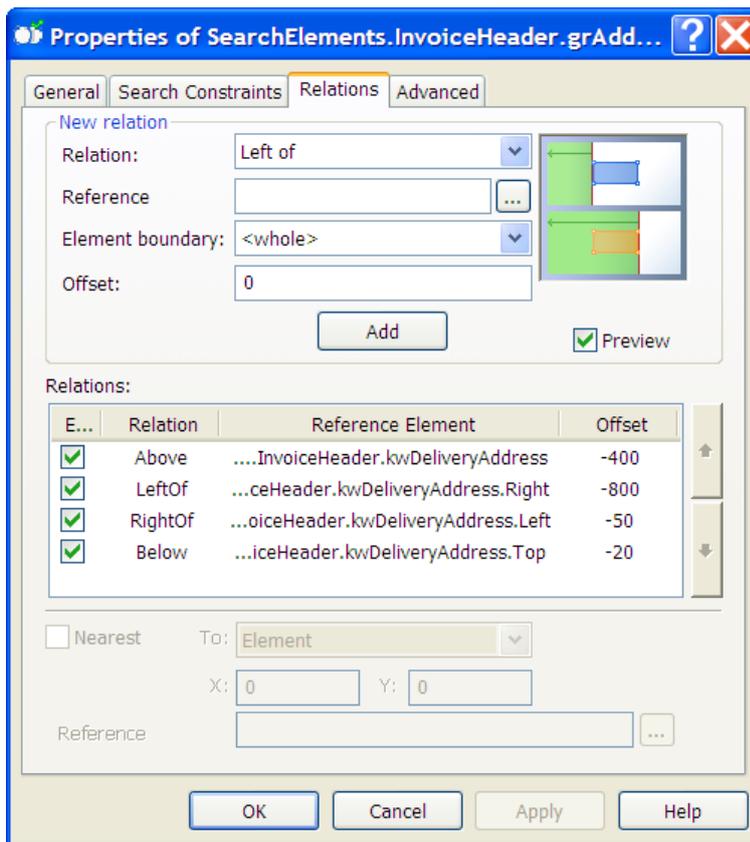
Местоположение поля *Delivery Address* мы решили искать:

- справа и снизу от заголовка поля.
- с помощью элемента **White Gap** – в качестве ограничителя снизу.
- с помощью элемента типа **Text Fragment**.

Чтобы задать настройки, общие для этих элементов, нам потребуется составной элемент. Для того чтобы создать элемент **grAddress**:

- В составе элемента **InvoiceHeader** создайте элемент **grAddress** (тип **Group**).
- Перейдите на закладку **Relations** и задайте следующие условия поиска элементов группы:
 - Above** элемента **kwDeliveryAddress**, **Offset** = -400
 - LeftOf** правой границы элемента **kwDeliveryAddress**, **Offset** = -800, **Element border** = **Right**.
 - RightOf** левой границы элемента **kwDeliveryAddress**, **Offset** = -50, **Element border** = **Left**.

- **Below** верхней границы элемента **kwDeliveryAddress**, **Offset = -20**, **Element border = Top**.



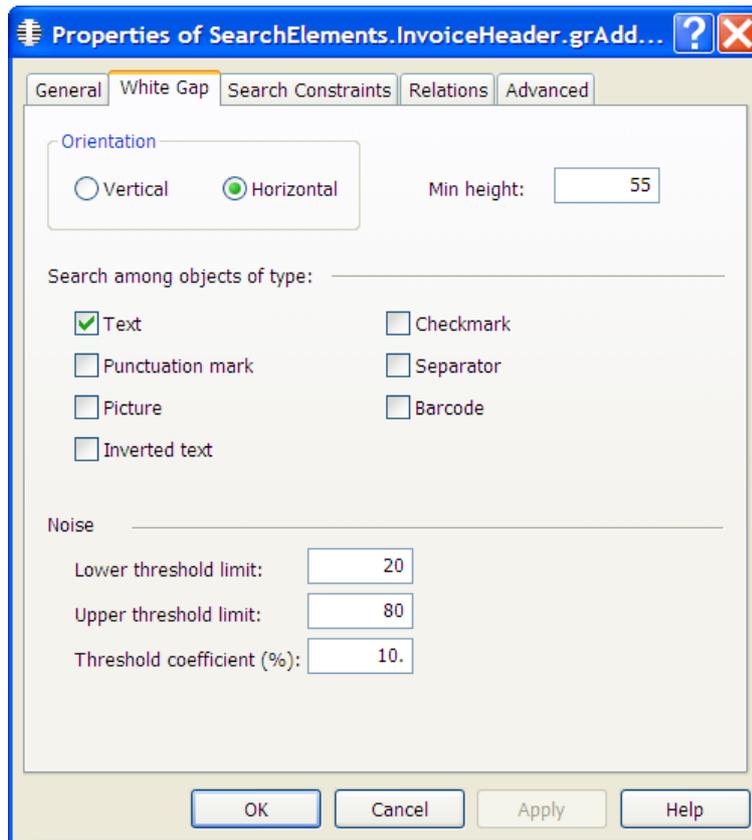
Для ограничения области поиска элемента **DeliveryAddress** снизу опишем горизонтальный просвет (**white gap**).

Чтобы создать элемент **wgAddressAbove**:

1. В составе элемента **InvoiceHeader.grAddress** создайте элемент **wgAddressAbove** (тип **White gap**).
2. Выберите закладку **White Gap**.
3. В группе **Orientation** выберите **Horizontal**. Минимальную высоту (**Min height**) поставьте равной 55.

Замечание. При подборе значений для высоты/ширины просветов бывает полезно обратить внимание на геометрические свойства (границы и размеры) близлежащих объектов изображения в режиме просмотра результатов предраспознавания.

4. Просвет мы будем искать среди текста, поэтому в группе **Search among objects of type**: отметьте только опцию **Text**.
5. В группе **Rubbish** установите следующие параметры для опций:
 - **Lower threshold limit** – 20;
 - **Upper threshold limit** – 80;
 - **Threshold coefficient** – 10.



Замечание. Значения свойств **Lower threshold limit**, **Upper threshold limit** и **Threshold coefficient** подбираются опытным путем. Подбор значений рекомендуется начинать со значений по умолчанию, если же поиск со значениями по умолчанию приводит к неудовлетворительным результатам, тогда следует подобрать такие значения свойств, которые позволят найти искомый просвет на всех тестовых изображениях.

6. Не накладывайте других ограничений на параметры элемента, оставьте значения по умолчанию.
7. Перейдите на закладку **Relations**.
8. Поскольку просвет может находиться только снизу от ключевых слов заголовка **kwDeliveryAddress**, задайте следующее условие поиска:

Below элемента **kwDeliveryAddress**, **Offset** = 20

Offset необходимо задать для того чтобы исключить нахождение просветов например между заголовком и текстом. Выберите закладку **Advanced**.

9. Поскольку поле **Delivery Address** может присутствовать на изображении, а может и нет, но при этом всегда присутствует вместе с заголовком, в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *Искать объект изображения, только если найден элемент **kwDeliveryAddress***. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If InvoiceHeader.kwDeliveryAddress.IsNull Then DontFind;
```

10. Протестируйте нахождение элемента **wgAddressAbove** на изображениях.
11. Поскольку в заданной области поиска на некоторых изображениях программа находит несколько просветов и выбирает не тот, который нас интересует, необходимо задать дополнительные ограничения. Анализ показывает, что интересующий нас просвет – самый верхний в заданной области поиска. Поэтому в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *Искать объект изображения ближайший по вертикали к верхней границе первой страницы*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
NearestY: Page(1).RectGlobal.Top;
```

12. Протестируйте изображения (временно исключите элемент **InvoiceFooter**).

Чтобы создать элемент **DeliveryAddress**:

1. В составе элемента **InvoiceHeader.grAddress** создайте элемент **DeliveryAddress** (тип **Text Fragment**).
2. Перейдите на закладку **Relations**.
3. Поскольку **DeliveryAddress** может находиться только выше просвета **wgAddressAbove** задайте следующее условие поиска:

Above элемента **wgAddressAbove**, **Offset** = 0.

5. Поскольку условие область поиска поля **DeliveryAddress** уже задано на уровне группы **InvoiceHeader.grAddress**, мы не будем его дублировать здесь.
6. Выберите закладку **Advanced**.
7. Поскольку заголовок поля **DeliveryAddress** также находится в области поиска, нам необходимо исключить его, чтобы получить только значение **DeliveryAddress**. Кроме этого в области поиска также присутствуют разделители и мусор, которые тоже желательно исключить. Поэтому нам не подходит просто исключение региона элемента **kwDeliveryAddress**. В данном случае будет правильно за основу взять область региона элемента **kwDeliveryAddress**, расширить его так, чтобы в него попал весь мусор и исключить уже этот прямоугольник. Для этого в поле **Advanced pre-search relations** задайте дополнительные условия поиска:

```
If InvoiceHeader.kwDeliveryAddress.IsNull Then DontFind;
Else

// опишем прямоугольник вокруг заголовка поля (задав некоторый отступ от границ)
{ Let Left = InvoiceHeader.kwDeliveryAddress.Rect.Left - 100dt;
  Let Top = InvoiceHeader.kwDeliveryAddress.Rect.Top - 100dt;
  Let Right = InvoiceHeader.kwDeliveryAddress.Rect.Right + 20dt;
  Let Bottom = InvoiceHeader.kwDeliveryAddress.Rect.Bottom;

// исключим из области поиска нашего поля описанный прямоугольник
  ExcludeRect (Left, Top, Right, Bottom);
}
```

8. Протестируйте изображения (временно исключите элемент **InvoiceFooter**).
9. Задайте местоположение блока **DeliveryAddress** в виде прямоугольника региона найденного элемента **DeliveryAddress**, упрощенного для улучшения визуального восприятия и увеличенного по высоте и ширине на 5 dot. Для этого выберите опцию **Region Expression** и задайте следующее выражение:

```
Rect outputRect;
OutputRegion = InvoiceHeader.grAddress.DeliveryAddress.SimplifiedRegion;
IsNull = InvoiceHeader.grAddress.DeliveryAddress.IsNull;
OutputRegion.Inflate (5dt, 5dt);
```

Шаг 6. Footer документа, описание группы InvoiceFooter

В группе **InvoiceFooter** мы опишем элементы, которые помогут нам идентифицировать последнюю страницу документа:

- на последней странице документа всегда присутствуют ключевые слова «**TOTALAMOUNTMUST**», «**Carriedover**», «**TotalCHF**», «**Total**».
 - Для поиска этого текста будем создавать два элемента: в первом укажем «**TOTALAMOUNTMUST**», «**Carriedover**», «**TotalCHF**» и создадим его в дереве элементов выше, чем второй, в котором укажем только «**Total**» (его будем искать только если не найден первый элемент). Это поможет нам сократить количество создаваемых гипотез для элемента и ускорит наложение гибкого описания.
 - эти слова всегда находятся ниже заголовка **Description** и **Product Total** столбцов таблицы на той же странице.
1. Для описания поиска ключевых слов (keywords) заголовков столбца **Product Total** и **Description** будем создавать элементы типа **Static Text**:

- Элемент **ExtraTag** - подробная инструкция в **Шаге 6.1**.
 - Элемент **AddTag** - подробная инструкция в **Шаге 6.2**.
2. Для описания поиска ключевых слов (keywords) «TOTALAMOUNTMUST», «Carriedover», «TotalCHF» и «Total» также воспользуемся элементами типа **Static Text**:
 - Элемент **LongFooter** - подробная инструкция в **Шаге 6.3**.
 - Элемент **ShortFooter** (для поиска ключевого слова «Total») - подробная инструкция в **Шаге 6.4**.
 3. Для того чтобы **Footer** накладывался только на последнюю страницу документа, он должен содержать в себе обязательный элемент. Так как наши ключевые слова, идентифицирующие последнюю страницу, находятся на каждой последней странице каждого документа, в группе InvoiceFooter создадим обязательный элемент **InvoiceFooter** типа **TextFragment**. Его область поиска зададим в виде прямоугольника региона найденного элемента либо **LongFooter**, либо **ShortFooter** немного расширив его по высоте и ширине - подробная инструкция в **Шаге 6.5**.

Шаг 6.1 Поиск заголовка столбца Product Total, элемент ExtraTag

Чтобы создать элемент **ExtraTag**:

1. В составе элемента **InvoiceFooter** создайте опциональный элемент **ExtraTag** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **PRODUCTTOTAL**.
4. Не накладывайте других ограничений на параметры элемента, оставьте значения по умолчанию.

Шаг 6.2 Поиск заголовка столбца Description, элемент AddTag

Чтобы создать элемент **AddTag**:

1. В составе элемента **InvoiceFooter** создайте опциональный элемент **AddTag** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Designation|Description**.
4. В поле **Advanced pre-search relations** задайте дополнительные параметры:

```
WholeWordMode: true; // в качестве гипотез будут рассматриваться только целые слова.
MaxGapInLine: 10dt; // максимально допустимое расстояние между соседними символами.
```

Шаг 6.3 Описание итоговой части документа, элемент LongFooter

Чтобы создать элемент **LongFooter**:

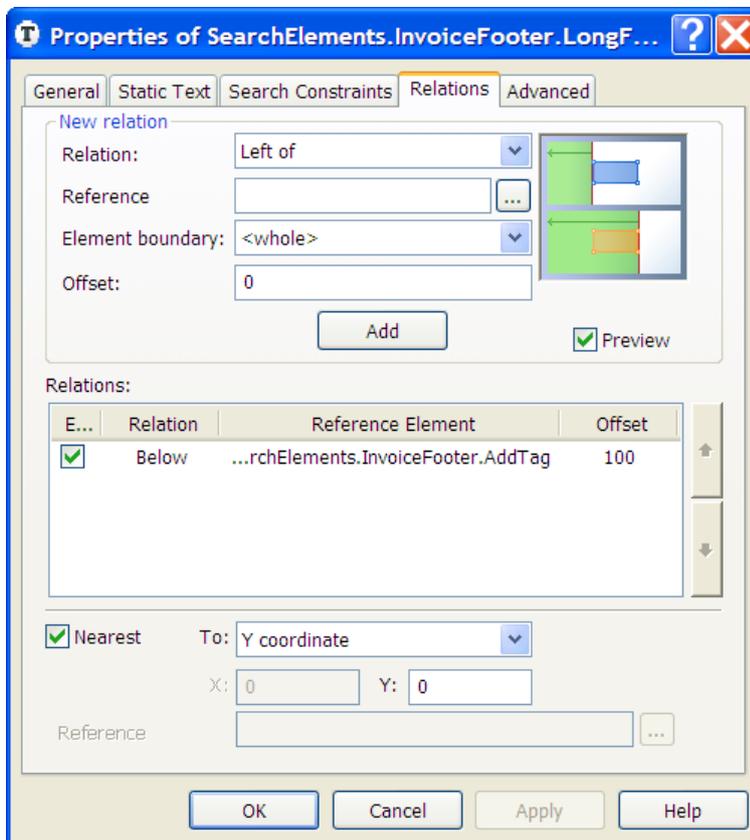
1. В составе элемента **InvoiceFooter** создайте опциональный элемент **LongFooter** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **TOTALAMOUNTMUST|Carriedover|TotalCHF**.
4. Поскольку искомый текст всегда находится ниже заголовка столбца **Description**, перейдите на закладку **Relations** и задайте отношение:

```
Below элемента AddTag, Offset = 100;
```

5. Поскольку в некоторых документах (например, документы номер 3 и 9) программа находит несколько вариантов заданного текста и выбирает не тот, который нас интересует, необходимо задать дополнительные ограничения. Анализ показывает, что интересующий нас текст – самый верхний в заданной области поиска (нам интересен текст,

расположенный наиболее близко к краю таблицы, так как в дальнейшем нам будет удобно использовать его как Footer таблицы). Поэтому на закладке **Relations** задайте дополнительные условия поиска: *Искать строку, ближайшую к верхнему краю документа*:

- Отметьте опцию **Nearest**
- В выпадающем списке поля **To**: выберите **Y coordinate**
- В поле **Y**: укажите 0.



Шаг 6.4 Описание итоговой части документа, элемент ShortFooter

Чтобы создать элемент **ShortFooter**:

1. В составе элемента **InvoiceFooter** создайте опциональный элемент **ShortFooter** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Total**.
4. Перейдите на закладку **Search Constraints** и исключите из области поиска **ShortFooter** уже найденный элемент **ExtraTag**, чтобы исключить нахождение слова «**Total**» в заголовке таблицы:
 - Нажмите кнопку **Add** рядом с полем **Exclude elements**,
 - В списке элементов выберите **ExtraTag**,
 - Нажмите кнопку **Ok**.
 - После этого в поле **Exclude elements**: вы увидите строчку **SearchElements.InvoiceFooter.ExtraTag**;
5. Поскольку искомый текст всегда находится ниже заголовка столбца **Description**, перейдите на закладку **Relations** и задайте отношение:

Below элемента **AddTag**, **Offset** = 100;

6. Мы договорились, что **ShortFooter** мы будем искать только в случае отсутствия **LongFooter**. Для того чтобы задать это условие в поле **Advanced pre-search relations** задайте дополнительные условия поиска – *не искать если найден longFooter*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not LongFooter.IsNull then Dontfind();
```

Шаг 6.5 Описание итоговой части документа, элемент TextFragment

Чтобы создать элемент **TextFragment**:

1. В составе элемента InvoiceFooter создайте неопциональный элемент **TextFragment** (тип **TextFragment**).
2. Перейдите на закладку **Advanced** и в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *не искать, если не найден ни LongFooter, ни ShortFooter; если найден LongFooter задайте область поиска поля прямоугольником региона этого элемента, расширив его на 10 dot по высоте и ширине; если найден ShortFooter задайте область поиска поля прямоугольником региона этого элемента, расширив его на 10 dot по высоте и ширине*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If LongFooter.IsNull and ShortFooter.IsNull then Dontfind();
// мы не будем искать элемент, если ни один из элементов не найден

else
{ if not LongFooter.IsNull then
// в случае если найден LongFooter, область поиска опишем на основе его региона
{ LeftOf: LongFooter.Rect.Right + 10dt;
RightOf: LongFooter.Rect.Left - 10dt;
Below: LongFooter.Rect.Top - 10dt;
Above: LongFooter.Rect.Bottom + 10dt;
}
else if not ShortFooter.IsNull then
// если же найден ShortFooter, то область поиска зададим регионом ShortFooter
{ LeftOf: ShortFooter.Rect.Right + 10dt;
RightOf: ShortFooter.Rect.Left - 10dt;
Below: ShortFooter.Rect.Top - 10dt;
Above: ShortFooter.Rect.Bottom + 10dt;
}
}
}
```

Протестируйте наложение гибкого описания и убедитесь, что страницы правильно собираются в документы в соответствии с описанными элементами **Header** и **Footer**. После наложения гибкого описания на все изображения сборка страниц в документы должна совпадать с эталонной сборкой, заданной нами в начале работы над данным гибким шаблоном.

 **Замечание.** Чтобы наложить гибкий шаблон на все тестовые изображения, удобно воспользоваться горячими клавишами:

- Поместите фокус в окно пакета изображений **Batch**.
- Для выбора всех страниц пакета нажмите клавиши **Ctrl + A**.
- Для запуска анализа выделенных страниц нажмите клавиши **Ctrl + Shift + E**.

 **Замечание.** В случае многостраничного документа важно накладывать гибкое описание не на часть страниц одного документа, а на все. Иначе, возможно, программа не сможет наложить гибкое описание на некоторые документы.

Шаг 7. Описание заголовков столбцов таблицы, группа TableHeader

Что касается поля **Invoice Table**, которое мы будем искать с помощью элемента типа **Table**, здесь нам тоже понадобятся предварительные приготовления – нам нужно определить местоположение заголовков столбцов таблицы. В данном примере, для поиска заголовков столбцов таблицы мы будем использовать дополнительные вспомогательные элементы.

Таким образом, прежде чем непосредственно приступить к поиску оставшихся полей, опишем все необходимые элементы:

Для описания заголовков создадим группу **TableHeader**, в которую включим описания всех заголовков таблицы. Здесь же ограничим область поиска первой страницей.

Чтобы создать элемент **TableHeader**:

1. создайте составной элемент **TableHeader** (тип **Group**) в дереве и.
2. перейдите на закладку **Search Constraints** и задайте область поиска элемента:
 - отметьте опцию **Absolute search area constraints**,
 - в поле **Pages**: из списка выберите **Range**,
 - в поле **Range**: укажите 1,
 - в поле **Search region**: оставьте значение по умолчанию **<whole page>**,
 - нажмите кнопку **Add**.

Поиск заголовков таблицы будем осуществлять с помощью элементов типа **Static Text**.

Для этого в составе группы **TableHeader** создадим элементы:

1. Элемент **kwDesignation** - подробная инструкция в **Шаге 7.1**,
2. Элемент **ExtraQtyTag** - подробная инструкция в **Шаге 7.2**,
3. Элемент **kwQuantity** - подробная инструкция в **Шаге 7.3**,
4. Элемент **kwUnitPrice** - подробная инструкция в **Шаге 7.4**,
5. Элемент **kwTotal** - подробная инструкция в **Шаге 7.5**,
6. Элемент **kwReference** - подробная инструкция в **Шаге 7.6**,
7. Элемент **kwSales** - подробная инструкция в **Шаге 7.7**,
8. Элемент **kwUnit** - подробная инструкция в **Шаге 7.8**.

Шаг 7.1 Поиск заголовка столбца Designation, элемент kwDesignation

Проанализировав результаты предраспознавания, видим, что в наших документах заголовок представлен только двумя вариантами текста «Designation» и «Description», и так как область поиска мы задали на уровне группы, в описании данного элемента мы не будем ее дублировать.

Чтобы создать элемент **kwDesignation**:

1. В составе элемента **TableHeader** создайте элемент **kwDesignation** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Designation|Description**.
4. Все остальные параметры не переопределяйте;
5. Протестируйте гибкое описание и убедитесь, что элемент находится на изображениях.

Шаг 7.2 Поиск заголовка столбца ExtraQuantity, элемент ExtraQtyTag

Далее нам необходимо найти заголовок **Quantity**, который в документах представлен следующими вариантами текста: «Qty», «Quantity», «ORDERED», «QtyNet». Но в нашей таблице есть еще один заголовок, в значении которого может встретиться слово «Quantity», кроме этого он содержит уникальный текст. Поэтому в первую очередь мы опишем заголовок **ExtraQtyTag**, а затем исключим его из области поиска заголовка **Quantity**.

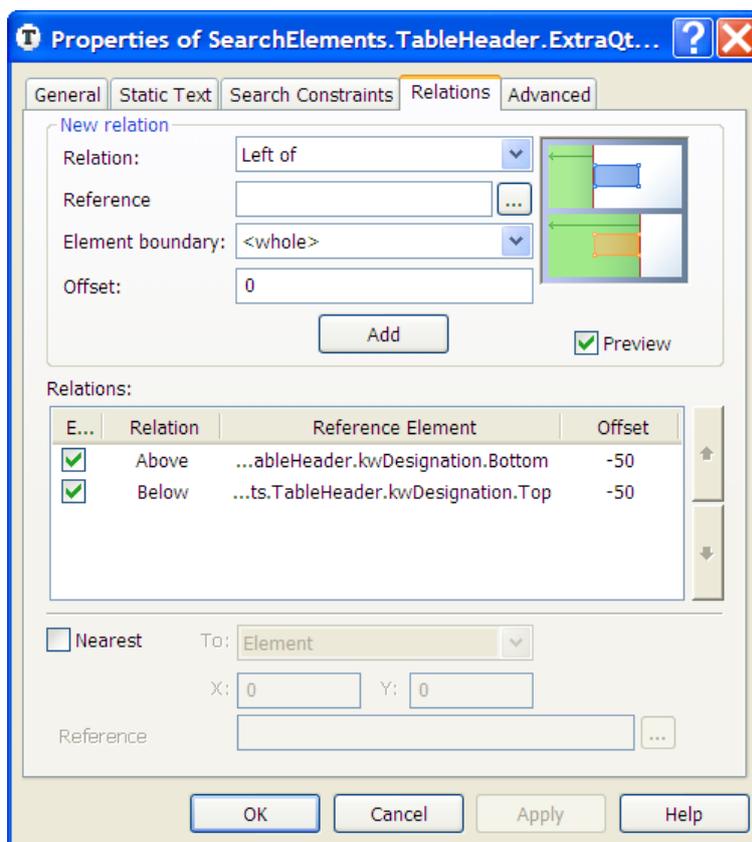
Во всех наших документах заголовок столбца **ExtraQtyTag** представлен только одним вариантом текста. Несмотря на то, что качество изображений достаточно высокое, зададим допустимый процент ошибок равным 35%, что для слова из 7 или 8 букв означает ошибку в двух буквах.

Кроме этого, поскольку все заголовки таблицы находятся на одном уровне, зададим дополнительное условие поиска относительно уже найденного заголовка столбца **Designation**.

Чтобы создать элемент **ExtraQtyTag**:

1. В составе элемента **TableHeader** создайте элемент **ExtraQtyTag** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **{Quantity *BK.ORD}**.
4. Задайте значение свойства **Max error percentage** равным 35.
5. Так как поле **ExtraQuantity** является многострочным отметьте опции **Allow for spaces** и **Permit multiple lines**.
6. Мы не будем штрафовать наши гипотезы за переход на другую строку, поэтому значение поля **Line break penalty** установите равным 1.
7. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска:

Above нижней границы элемента **kwDesignation**, **Offset** = -50, **Element border** = **Bottom**,
Below верхней границы элемента **kwDesignation**, **Offset** = -50, **Element border** = **Top**.



8. Для оптимизации поиска зададим дополнительное условие: не искать заголовки, если не найден заголовок **Designation**. Перейдите на закладку **Advanced** и в поле **Advanced pre-search relations** задайте это условие. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If kwDesignation.IsNull then Dontfind();
```

9. В поле **Advanced pre-search relations** также задайте дополнительные параметры:

```
WholeWordMode: true; // в качестве гипотез будут рассматриваться только целые слова.
```

Замечание. Данное условие также можно задать через интерфейс: отметьте опцию **Whole words only** на закладке **Static Text**.

10. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

Шаг 7.3 Поиск заголовка столбца *Quantity*, элемент *kwQuantity*

При описании этого заголовка мы будем опираться на уже найденные элементы *kwDesignation* и *ExtraQtyTag*.

Чтобы создать элемент **kwQuantity**:

1. В составе элемента **TableHeader** создайте элемент **kwQuantity** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Qty|Quantity|ORDERED|QtyNet**.
4. Задайте значение свойства **Max error percentage** равным 35.
5. Перейдите на закладку **Search Constraints** и исключите из области поиска уже найденный элемент **SearchElements.TableHeader.ExtraQtyTag**;
6. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска. Подробнее см. **Шаг 7.2**.
7. Для оптимизации поиска зададим дополнительное условие: не искать заголовков, если не найден заголовок **Designation**. Подробнее см. **Шаг 7.2**.
8. В поле **Advanced pre-search relations** также задайте дополнительные параметры:

```
WholeWordMode: true; // в качестве гипотез будут рассматриваться только целые текстовые объекты.
MaxGapInLine: 20dt; // максимально допустимое расстояние между соседними символами.
```

9. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

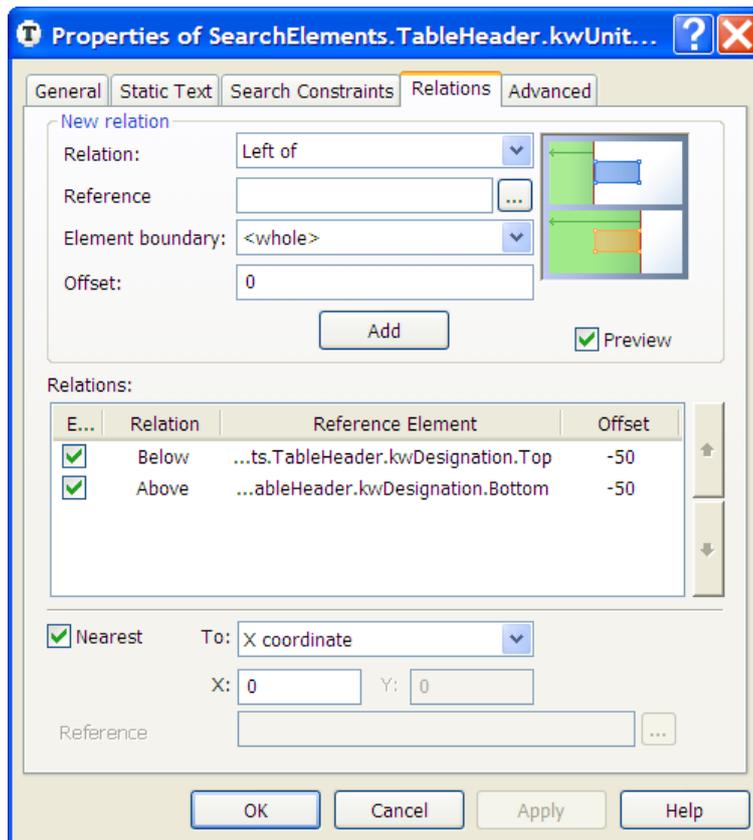
Шаг 7.4 Поиск заголовка столбца *UnitPrice*, элемент *kwUnitPrice*

Проанализировав изображения, видим, что заголовок **Unit Price** представлен следующими вариантами *search text*: «**UnitPrice**», «**Price**», «**RATE**», а заголовок столбца **Unit** – словом «**Unit**». Чтобы исключить нахождение слова «**Unit**» заголовка столбца **Unit** в заголовке столбца **Unit Price**, сначала опишем **Unit Price**, так как он содержит уникальный текст, а затем, задав дополнительное отношение, опишем заголовок столбца **Unit**.

При описании элемента для поиска ключевых слов заголовка столбца **Unit Price** примем во внимание, что столбец **Unit Price** всегда располагается правее столбца **Quantity**.

Чтобы создать элемент **kwUnitPrice**:

1. В составе элемента **TableHeader** создайте элемент **kwUnitPrice** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **UnitPrice|Price|RATE**.
4. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска. Подробнее см. **Шаг 7.2**.
5. Также на закладке **Relations** задайте условие искать наш элемент как *самый ближайший к X coordinate со значением = 0*.



6. Выберите закладку **Advanced**.
7. Поскольку заголовок столбца **Unit Price** располагается правее заголовка столбца **kwQuantity**, в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *Если элемент **kwQuantity** найден, то искать объект изображения правее элемента **kwQuantity***. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If Not (TableHeader.kwQuantity.IsNull) Then
  RightOf: TableHeader.kwQuantity, 0 * dot;
```

8. Для оптимизации поиска зададим дополнительное условие: не искать заголовков, если не найден заголовок **Designation**. Подробнее см. **Шаг 7.2**.
9. Задав дополнительную проверку в поле **Advanced post-search relations** мы можем повлиять на качество сформированных гипотез, например, в данном случае такую: *если элемент найден, вычислить разницу между эталонной шириной и шириной найденного элемента и проверить ее на вхождение в нечеткий интервал*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not IsNull then
  { FuzzyQuality: 400dt - width, {-50000,0,0, 50000}*dt;
  }
  //такая проверка означает, что чем больше разница, тем выше штрафующий коэффициент
```

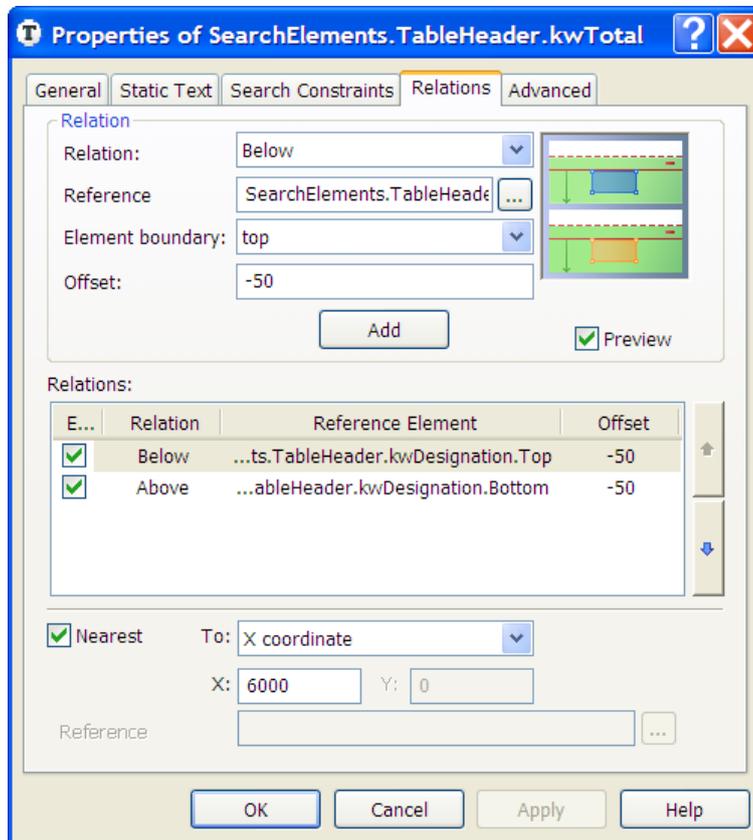
10. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

Шаг 7.5 Поиск заголовка столбца *Total*, элемент *kwTotal*

Проанализировав изображения, видим, что столбец **Total** на всех изображениях находится правее столбца **Unit Price**. Поэтому будем строить поиск заголовка столбца **Total** относительно заголовка столбца **Unit Price**.

Чтобы создать элемент **kwTotal**:

1. В составе элемента **TableHeader** создайте элемент **kwTotal** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **TotalChffTotal|Extension|AMOUNT|PRODUCTTOTAL**.
4. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска. Подробнее см. **Шаг 7.2**.
5. Также на закладке **Relations** задайте условие искать наш элемент как *самый ближайший к X coordinate со значением = 6000*.



6. Выберите закладку **Advanced**.
7. Поскольку заголовок столбца **Total** располагается правее заголовка столбца **kwUnitPrice** (если последний находится на изображении), в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *Если элемент **kwUnitPrice** найден, то искать **kwUnitPrice***. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If Not (TableHeader.kwUnitPrice.IsNull) Then
  RightOf: TableHeader.kwUnitPrice, 0 * dot;
```

8. Для оптимизации поиска зададим дополнительное условие: не искать заголовок, если не найден заголовок **Designation**. Подробнее см. **Шаг 7.2**.
9. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

Шаг 7.6 Поиск заголовка столбца *Reference*, элемент *kwReference*

Проанализировав изображения, видим, что в наших документах заголовок может быть представлен следующими вариантами текста: «Reference», «ID», «Item», «CODE».

Чтобы создать элемент **kwReference**:

1. В составе элемента **TableHeader** создайте элемент **kwReference** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Reference|ID|Item|CODE**.
4. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска. Подробнее см. **Шаг 7.2**.
5. Для оптимизации поиска зададим дополнительное условие: не искать заголовок, если не найден заголовок **Designation**. Подробнее см. **Шаг 7.2**.
6. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

Шаг 7.7 Поиск заголовка столбца *Sales*, элемент *kwSales*

Хотя сам столбец **Sales** нам не нужен, его заголовок необходим нам для ограничения таблицы справа.

Чтобы создать элемент **kwSales**:

1. В составе элемента **TableHeader** создайте элемент **kwSales** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Sales**
4. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска. Подробнее см. **Шаг 7.2**.
5. Выберите закладку **Advanced**.
6. Поскольку заголовок столбца **Sales** располагается правее заголовка столбца **kwTotal** (если последний находится на изображении), в поле **Advanced pre-search relations** задайте дополнительные условия поиска: *Если элемент **kwTotal** найден, то искать объект изображения правее элемента **kwTotal***. На языке описания FlexiLayout данное условие выглядит следующим образом:


```
If Not (TableHeader.kwTotal.IsNull) Then
  RightOf: TableHeader.kwTotal, 0 * dot;
```
7. Для оптимизации поиска зададим дополнительное условие: не искать заголовок, если не найден заголовок **Designation**. Подробнее см. **Шаг 7.2**.
8. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

Шаг 7.8 Поиск заголовка столбца *Unit*, элемент *kwUnit*

При описании элемента для поиска ключевых слов заголовка столбца **Unit** будем опираться на уже найденный столбец **UnitPrice** и зададим дополнительное отношение.- искать заголовок столбца **Unit** левее границы заголовка столбца **UnitPrice**.

Чтобы создать элемент **kwUnit**:

1. В составе элемента **TableHeader** создайте элемент **kwUnit** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **Unit**.
4. Для того чтобы описать, что наш заголовок находится на одном уровне с заголовком **Designation**, перейдите на закладку **Relations** и задайте дополнительные условия поиска. Подробнее см. **Шаг 7.2**.
5. Для того чтобы описать, что наш заголовок находится левее левой границы заголовка **UnitPrice**, на закладке **Relations** задайте дополнительные условия поиска:

```
LeftOf левой границы элемента kwUnitPrice, Offset = 20, Element border = Left;
```

6. Для оптимизации поиска зададим дополнительное условие: не искать заголовков, если не найден заголовок **Designation**. Подробнее см. **Шаг 7.2**.
7. Протестируйте гибкое описание на изображениях и убедитесь, что элемент корректно находится на первой странице документа.

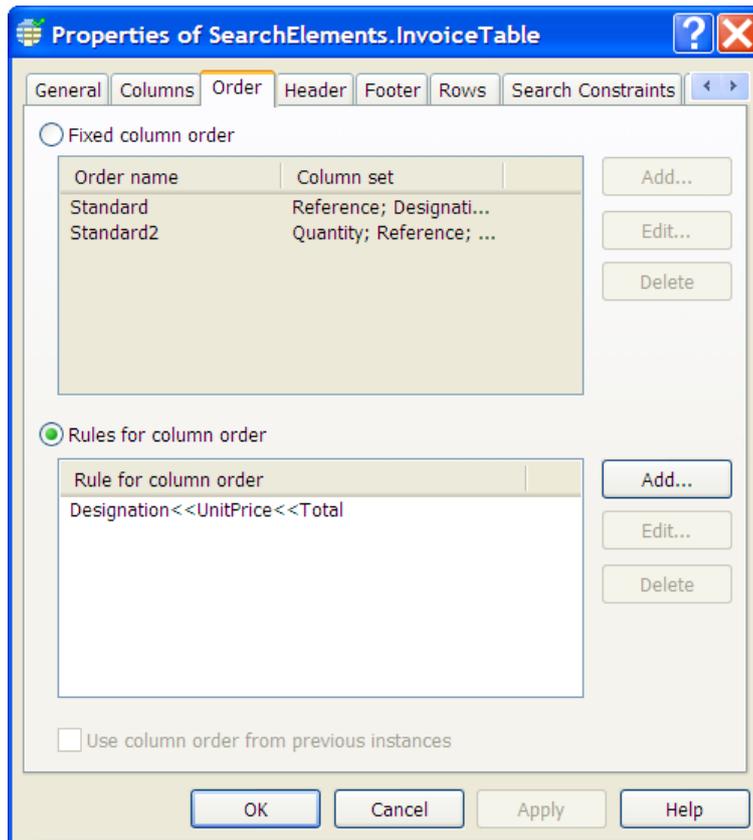
Шаг 8. Поиск табличного элемента, элемент InvoiceTable

Итак, все предварительные приготовления сделаны, и мы можем искать поле **Invoice Table**. Опишем элемент типа **Table**:

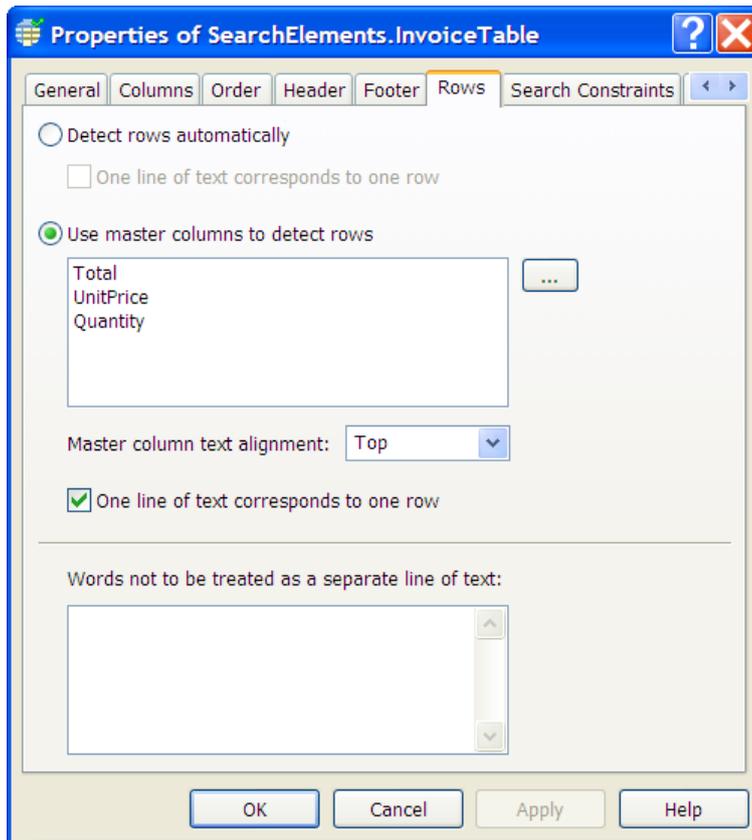
1. **Header** таблицы. Как уже было решено, заголовки столбцов мы зададим с помощью предварительно найденных вспомогательных элементов.
2. **Footer** таблицы. Для задания футера таблицы мы решили использовать уже найденный обязательный элемент **TextFragment** из группы **InvoiceFooter**, определяющий начало итоговой части. Этот элемент мы задавали как ближайший к нижнему краю таблицы
3. Область поиска таблицы. Ограничим область поиска справа относительно заголовка последнего, не нужного нам, столбца **Sales**.

Чтобы создать элемент **InvoiceTable**:

1. Создайте элемент **InvoiceTable** (тип **Table**) в дереве элементов **SearchElements**.
2. Выберите закладку **Columns**.
3. Выберите в качестве блока, для поиска которого служит данный элемент, ранее созданный (см. **Шаг 3**) блок **InvoiceTable**, нажав кнопку "...".
4. Задайте параметры поиска столбцов таблицы:
 - 4.1. Выберите в списке столбцов **Reference** и нажмите кнопку **Properties**.
Для заголовка данного столбца мы будем использовать ранее найденный элемент **TableHeader.kwReference**. Для этого отметьте опцию **Use found element as column title** и выберите элемент **TableHeader.kwReference**, нажав кнопку "...".
 - 4.2. Выберите в списке столбцов **Designation** и нажмите кнопку **Properties**.
Для заголовка данного столбца мы будем использовать ранее найденный элемент **TableHeader.kwDesignation**.
 - 4.3. Выберите в списке столбцов **Unit** и нажмите кнопку **Properties**.
Для заголовка данного столбца мы будем использовать ранее найденный элемент **TableHeader.kwUnit**.
 - 4.4. Выберите в списке столбцов **Quantity** и нажмите кнопку **Properties**.
Для заголовка данного столбца мы будем использовать ранее найденный элемент **TableHeader.kwQuantity**.
 - 4.5. Выберите в списке столбцов **UnitPrice** и нажмите кнопку **Properties**.
Для заголовка данного столбца мы будем использовать ранее найденный элемент **TableHeader.kwUnitPrice**.
 - 4.6. Выберите в списке столбцов **Total** и нажмите кнопку **Properties**.
Для заголовка данного столбца мы будем использовать ранее найденный элемент **TableHeader.kwTotal**.
5. Выберите закладку **Order**.
6. Поскольку искомые столбцы могут перемежаться другими (например, **ExtraQtyTag**), но всегда следуют в следующем порядке: сначала **Designation**, правее него **Unit Price**, и ещё правее **Total**, - то выберем вариант **Rules for column order** и, нажав кнопку "Add...", добавим столбцы в таком порядке: **Designation << Unit Price << Total**.



7. Выберите закладку **Header**. Заголовок таблицы нам нужен, но других ограничений мы накладывать не будем, поэтому оставьте значения по умолчанию.
8. Выберите закладку **Footer**. Для итоговой части таблицы мы будем использовать ранее найденный элемент **InvoiceFooter.TextFragment**. Для этого отметьте опцию **Use found element as footer** и выберите элемент **InvoiceFooter.TextFragment**, нажав кнопку "...". Выберите закладку **Rows**.
10. Поскольку в нашей таблице ячейки в некоторых столбцах могут содержать многострочные записи, необходимо задать ключевые столбцы для разделения на строки. Выберите опцию **Use master columns to detect rows**. В качестве разделителя на строки первым укажем столбец **Total**, так как в этом столбце всегда присутствуют данные, и каждой записи таблицы в этом столбце соответствует одна строка данных. На тот случай, если данный столбец не будет обнаружен на изображении, укажем следующими разделителями столбец **Unit Price** и столбец **Quantity**. Выберите данный набор столбцов с помощью кнопки "...".



11. Выберите закладку **Relations**. Поскольку интересующие нас столбцы таблицы располагаются левее начала заголовка столбца **Sales**, добавьте следующие условия поиска:

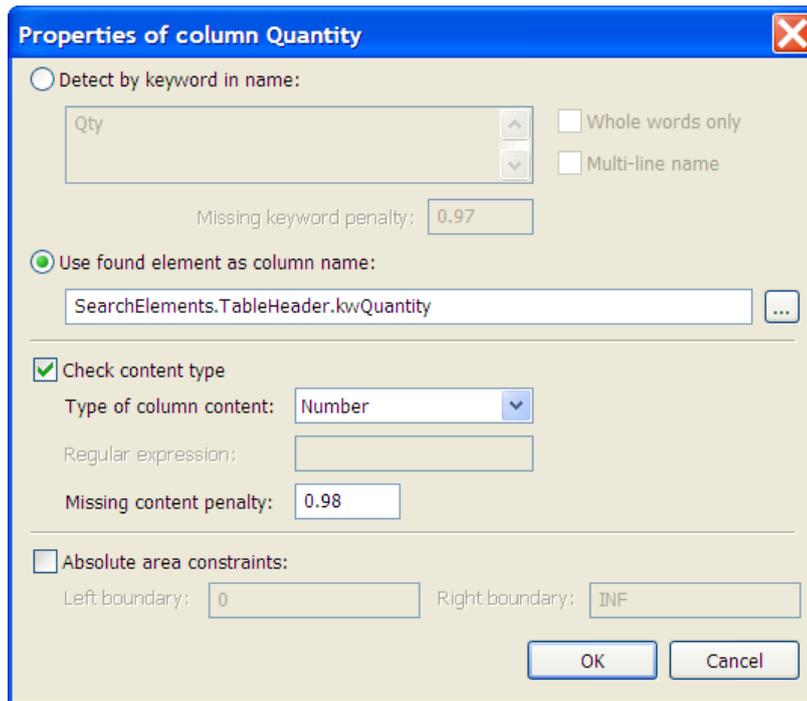
LeftOf элемента **TableHeader.kwSales**, **Offset = 0**;

Замечание. При подборе значений для отступов (**offset**) бывает полезно обратить внимание на геометрические свойства (границы и размеры) опорных и искомых объектов изображения в режиме просмотра результатов предраспознавания. В остальном подбор этих значений является творческим процессом и осуществляется путем подбора.

На случай изображений плохого качества (например, мусора между телом и footer таблицы или отсутствия header или footer таблицы) бывает полезным дополнительно задать ограничение таблицы сверху и снизу. В данном случае можно использовать элементы **kwDesignation** для ограничения сверху и **TextFragment** – снизу. На закладке **Relations** добавьте следующие условия поиска:

Below верхней границы элемента **kwDesignation**, **Offset = -50**, **Element border = Top**;
Above элемента **TextFragment**, **Offset = -50**; **Element border = Bottom**;

12. Наложим шаблон на тестовый набор изображений. Видим, что программа верно находит табличный элемент на всех изображениях, но на некоторых изображениях в столбец **Quantity** попадает текст из предыдущего, не нужного нам столбца. Чтобы отделить данные столбца **Quantity** от мусора, зададим тип контекста для данного столбца.
13. Откройте диалог **Properties** табличного элемента **InvoiceTable**. Перейдите на закладку **Columns**, откройте диалог **Properties** для столбца **Quantity**, отметьте опцию проверки содержимого столбца **Check content type**. В типе контекста **Type of column content**: выберите **Number**.



14. Если мы протестируем изображения теперь, то видим, что сейчас в столбец **Quantity** попадают только цифры, что и требовалось.
15. Задайте контекст также для столбцов **UnitPrice** равным **Currency** и для **Total** равным **Currency**.
16. Выберите в качестве описания местоположения блока **InvoiceTable** опцию **Source element**. В качестве source element укажите элемент **InvoiceTable**, нажав кнопку "...".

Шаг 9. Поиск поля TotalAmount, описание составного элемента SumGroup

Проанализировав тестовые документы, видим, что интересующее нас поле **Total Amount** присутствуют только вместе с заголовком на последней странице документа ниже таблицы. Кроме этого, само поле располагается либо на одном уровне, либо под заголовком. Так как поле однострочное, будем искать его с помощью элемента типа **Character String**.

Для того чтобы задать настройки, общие для поля и его заголовка, нам потребуется составной элемент.

Чтобы создать составной элемент **SumGroup**:

1. Создайте элемент **SumGroup** (тип **Group**).
2. Выберите закладку **Advanced** и в поле **Advanced pre-search relations** задайте дополнительные условия поиска для всех элементов группы: *искать ниже последней найденной таблицы*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not InvoiceTable.LastSubTable.Body.IsNull then
Below: InvoiceTable.LastSubTable.Body.Rect.Bottom;
```

Для описания поиска ключевых слов (keywords) заголовка поля **TotalAmount** воспользуйтесь элементом **SumGroup.kwTotal** типа **Static Text**. – Подробная инструкция в **Шаге 9.1**.

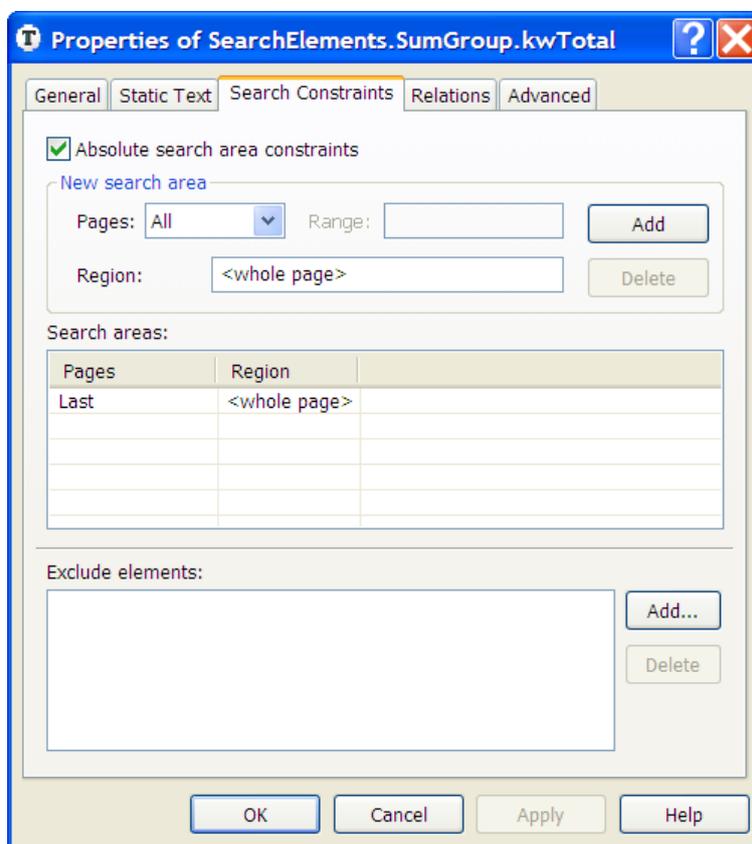
Для поиска однострочного поля **TotalAmount** воспользуйтесь элементом **SumGroup.TotalAmount** типа **Character String** - Подробная инструкция в **Шаге 9.2**.

Шаг 9.1 Поиск заголовка поля *Total Amount*, элемент *kwTotal*

Заголовок на изображениях представлен фразами «Total Chf», «Carried Over» и «Total». Кроме этого он располагается ниже таблицы (это условие мы уже задали для составного элемента *SumGroup*) и всегда на последней странице.

Чтобы создать элемент **kwTotal**:

1. В составе элемента **SumGroup** создайте элемент **kwTotal** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста: **TotalChf|CarriedOver|Total**.
4. Задайте значение свойства **Max error percentage** равным 20.
5. Выберите закладку **Search Constraints** и ограничьте область поиска последней страницей:
 - отметьте опцию **Absolute search area constraints**,
 - в поле **Pages**: из списка выберите **Last**,
 - остальные опции не отмечайте,
 - нажмите кнопку **Add**. Область появится в списке **Search areas**.



6. Выберите закладку **Advanced**
7. Среди множества всех найденных элементов с заданным текстом нас интересует тот, который располагается как можно ближе к нижней границе таблицы. Эта закономерность дает нам возможность повлиять на качество сформированных гипотез. В поле **Advanced post-search relations** задайте дополнительные параметры проверки гипотезы найденного элемента: *если была сформирована ненулевая гипотеза элемента, вычислить расстояние между верхней границей найденного элемента и нижней границей последней найденной в документе таблицы и проверить это расстояние на вхождение в нечеткий интервал*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not IsNull then
{ FuzzyQuality: Top - InvoiceTable.LastSubTable.Body.Rect.Bottom, {0,0,0, 50000}*dt;
}
//такая проверка означает, что чем больше расстояние между границами, тем выше штрафующий коэффициент
```

Шаг 9.2 Описание поля *Total Amount*, элемент *TotalAmount*

Поле **TotalAmount** будем одновременно искать как справа от заголовка, так и под заголовком.

Чтобы создать элемент **TotalAmount**:

1. В составе элемента **SumGroup** создайте элемент **TotalAmount** (тип **Character String**).
2. Выберите закладку **Character String**.
3. Задайте алфавит
'-.0123456789OSZosz
4. Задайте значение параметра **Percentage of non-alphabet characters** равным 30%.
5. В поле **String Length** задайте нечеткий интервал {-1, 1, 20, INF}, оценивающий длину цепочки символов. Предполагаем, что все возможные значения будут находиться в интервале от 1 до 20 символов. Любая гипотеза, выходящая за рамки данного интервала, будет штрафоваться.
6. Не накладывайте других ограничений на параметры элемента, оставьте значения по умолчанию.
7. Перейдите на закладку **Search Constraints** и исключите из области поиска элемента регион гипотезы элемента **kwTotal**.
8. Выберите закладку **Advanced**.
9. В поле **Advanced pre-search relations** задайте дополнительные условия поиска: *если найден заголовок kwTotal, то задайте область поиска поля массивом прямоугольников: один справа от заголовка, другой - под заголовком с некоторым отступом, иначе объект изображения не искать*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not kwTotal.IsNull Then // если найден заголовок kwTotal
{
  WholeWordMOde: true;
  MaxGapInLine: 30dt;

  RectArray DataRegion;

  Let r1= Rect (kwTotal.Rect.Right, kwTotal.Rect.Top -20dt, kwTotal.Rect.Right + 1800dt, kwTotal.Rect.Bottom + 50dt);
  // ограничим область поиска прямоугольником справа от заголовка поля
  Let r2 = Rect (kwTotal.Rect.Left - 100dt, kwTotal.Rect.Bottom, kwTotal.Rect.Right + 100dt, kwTotal.Rect.Bottom + 200dt);
  // ограничим область поиска прямоугольником снизу от заголовка поля

  DataRegion = RectArray (r1);
  DataRegion.Add (r2);

  RestrictSearchArea (DataRegion);
}
else Dontfind(); // иначе объект изображения не искать
```

 **Замечание.** При подборе значений для отступов (**offset**) бывает полезно обратить внимание на геометрические свойства (границы и размеры) опорных и искомых объектов изображения в режиме просмотра результатов предраспознавания. Значения для отступов (**offset**) подбираются опытным путем.

10. Область поиска значения поля обозначена двумя прямоугольниками – справа от заголовка и под ним. Причем, если значение поля располагается справа от его заголовка, то на большинстве тестовых изображений под заголовком присутствует какой-то текст. Так как на большинстве изображений этот ненужный текст попадает в область поиска поля **TotalAmount**, программа создаст для него гипотезу. Чтобы повлиять на качество такой гипотезы (занизить ее качество), в поле **Advanced post-search relations** задайте дополнительные параметры проверки: *чем значение поля ниже заголовка, тем сильнее штрафовать гипотезу*. На языке описания FlexiLayout данное условие выглядит следующим образом:

```
If not IsNull then
{FuzzyQuality: 400dt - width, {-50000,0,0, 50000}*dt;
```

```
//максимальное качество будет у гипотез элементов, ширина региона которых равна 400dot
if width < 50dt then Quality: 0;
// элементы ширины меньше 50 dot не будем рассматривать
if Top > kwTotal.Rect.Top then
{ FuzzyQuality: Top - kwTotal.Rect.Top, {0,0,0, 20000}*dt;
}
}
//такая проверка означает, что чем больше разница, тем выше штрафующий коэффициент
```

 **Замечание.** Для описания положения объектов на странице и расстояний между ними используется двумерная система координат с горизонтальной и вертикальной осями. Точка отсчета (0, 0) координатных осей расположена в левом верхнем углу изображения. Положительное направление оси абсцисс - слева направо, оси ординат - сверху вниз.

11. Протестируйте изображения и убедитесь, что программа верно находит элемент на всех изображениях, где он присутствует.
12. Задайте местоположение блока **TotalAmount** в виде прямоугольника региона элемента **TotalAmount**, увеличенного по высоте и ширине на 5 dot. Для этого выберите опцию **Region Expression** и задайте следующее выражение:

```
Rect outputRect;
outputRect = SumGroup.TotalAmount.Rect;
IsNull = SumGroup.TotalAmount.IsNull;
OutputRegion = outputRect;
OutputRegion.Inflate (5dt, 5dt);
```

Шаг 10. Поиск поля Company, описание составного элемента CompanyGroup, описание элемента Company

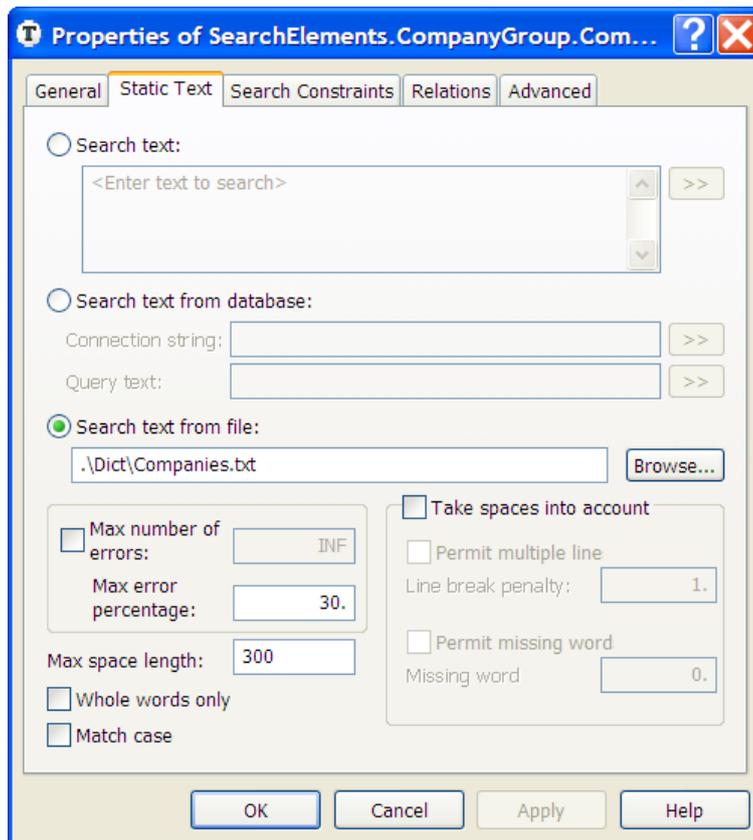
Проанализировав изображения, видим, что поле **Company** всегда располагается в первой трети первой страницы документов. Для поиска поля **Company** воспользуемся элементом типа **Static Text**, который опишем в составе элемента **CompanyGroup**, типа **Group**.

Чтобы создать элемент **CompanyGroup** создайте составной элемент **CompanyGroup** (тип **Group**). Не меняйте параметры, заданные по умолчанию.

Чтобы создать элемент **Company**:

1. В составе элемента **CompanyGroup** создайте элемент **Company** (тип **Static Text**).
2. Выберите закладку **Static Text**.
3. Отметьте опцию **Search Text from file**: и укажите путь к файлу **Companies.txt**.

 **Замечание:** Файл **Companies.txt** хранится в папке `<имя диска>:\Documents and Settings\All Users\Application Data\ABBYY\FlexiCapture\9.0\Samples\FlexiLayout Studio\Invoice\Dict`.



4. Выберите закладку **Advanced**.
5. В поле **Advanced pre-search relations** задайте дополнительное условие поиска: *искать объект изображения в первой трети первой страницы документа*. На языке описания FlexiLayout данное условие выглядит следующим образом:

Above: Page(1).RectGlobal.Top + Page(1).RectGlobal.Height/3;

6. Протестируйте изображения и убедитесь, что программа верно находит элемент на всех изображениях, где он присутствует.
7. Просмотр результатов наложения гибкого описания на все изображения показывает, что в некоторых документах не удалось найти название компании. Причиной этого явился тот факт, что в данном документе текст написан в две строки, а в свойствах элемента мы задали в качестве текста для поиска однострочный текст. Чтобы исключить подобную ситуацию, внесите следующие изменения:

- Откройте файл **Companies.txt** и укажите название компании **YOUR SERVICE** с пробелами. Это будет означать, что мы будем искать фразу, которая в общем случае может быть и не однострочной.
- Откройте диалог **Properties** элемента **Company** и перейдите на закладку **Static text**.
- Для задания многострочности искомой фразы выберите свойство **Permit multiple lines**.
- Задайте значение параметра **Line break penalty** равным 1. Это позволит сохранить качество сформированной гипотезы без изменения в случае многострочного текста. Задание меньшего значения приведет к штрафованию гипотезы.

8. Протестируйте изображения и убедитесь, что программа верно находит элемент в тестовых документах.
9. Задайте местоположение блока **Company** в виде прямоугольника региона элемента **Company**, увеличенного по высоте и ширине на 5 dot. Для этого выберите опцию **Region Expression** и задайте следующее выражение:

```
Rect outputRect;
outputRect = CompanyGroup.Company.Rect;
IsNull = CompanyGroup.Company.IsNull;
OutputRegion = outputRect;
OutputRegion.Inflate (5dt, 5dt);
```

Шаг 11. Экспорт гибкого описания и подключение его к шаблону FlexiCapture

Полученное гибкое описание можно экспортировать в формат ABBYY FlexiLayout (*.af) и подключить к шаблону в программе ABBYY FlexiCapture.

 **Замечание.** Подробнее об экспорте и подключении смотрите Пример1, Шаг 21 - Шаг 22.