



FlexiLayout Studio 8.0

Обучение на примерах

Пример 1

Содержание

Краткое описание формы Halloween Registration Form	3
Шаг 1. Создание нового проекта	3
Шаг 2. Добавление изображений в пакет	4
Шаг 3. Настройка свойств гибкого описания	6
Шаг 4. Предраспознавание	8
Шаг 5. Просмотр изображений и результатов предраспознавания.....	8
Шаг 6. Предварительный анализ результатов предраспознавания и определение опорных элементов	10
Шаг 7. Создание идентификационного элемента	10
Шаг 8. Проверка нахождения заголовка формы, соответствующего идентификационному элементу.....	12
Шаг 9. Корректировка свойств идентификационного элемента	13
Шаг 10. Описание поля YOUR PLANET NAME	14
Шаг 11. Описание заголовка поля YOUR PLANET NAME. Элемент PlanetNameHeader	15
Шаг 12. Описание поля YOUR PLANET NAME. Элемент PlanetName	16
Шаг 13. Проверка нахождения поля, содержащего имя планеты.....	20
Шаг 14. Описание поля YOUR PLANET NAME. Блок PlanetName	21
Шаг 15. Описание поля NAME.....	22
Шаг 16. Описание поля YOUR SPACESHIP NUMBER	23
Шаг 17. Описание поля DATE YOU ARRIVED AT THE EARTH	24
Шаг 18. Описание поля YOUR IDENTITY NUMBER ON THE PARTY	26
Шаг 19. Описание поля ПРОИЗВОЛЬНЫЙ ТЕКСТ.....	27
Шаг 20. Описание поля YOUR PHOTO IN FANCY DRESS	29
Шаг 21. Экспорт гибкого описания.....	32
Шаг 22. Подключение гибкого описания к шаблону в программе ABBYY FlexiCapture	32

Краткое описание формы Halloween Registration Form

Внимание! Для простоты в данном примере рассматривается одностраничный документ.

В данном руководстве ABBYY FlexiLayout Studio описывается процесс создания гибкого описания для простейшей регистрационной формы. Проект FlexiLayout Studio, содержащий тестовые изображения и отлаженное гибкое описание, находится в

<имя диска>:\Documents and Settings\All Users\Application Data\ABBYY\FlexiCapture\8.0\Samples\FlexiLayoutStudio\Halloween Form.

Этот документ является простейшим примером типичной регистрационной формы, на которой мы сможем по шагам проследить процесс создания гибкого описания, научимся находить объекты изображения, соответствующие элементам типа **Static Text**, **Character String**, **Barcode**, **Text Fragment**, **Date**, **Separator**, **White Gap** и **Object Collection**.

Конечной целью создания гибкого описания является устойчивое нахождение на всех тестовых страницах местоположения следующих полей:

Name
Your Planet Name
Your spaceship number
Date you arrived at the Earth
Your identity number on the party
Your photo in fancy dress

Кроме того, необходимо добиться обнаружения текстового поля произвольной формы, в котором каждый гость приводит любую информацию, которую ему удалось найти по теме Halloween. В отличие от всех вышеперечисленных полей, заголовок этого текстового поля произволен.

HALLOWEEN REGISTRATION FORM

NOTE: Please remember that this form should be completed by extraterrestrials only

YOUR PHOTO IN FANCY DRESS

NAME: BIG-TROUBLE-AND-EVEN-MORE-THAN-YOU-THINK

YOUR PLANET NAME: PLUTO

YOUR SPACESHIP NUMBER: 000 00 WD 01

DATE YOU ARRIVED AT THE EARTH: 10/ 10/ 2005

YOUR IDENTITY NUMBER ON THE PARTY IS: 

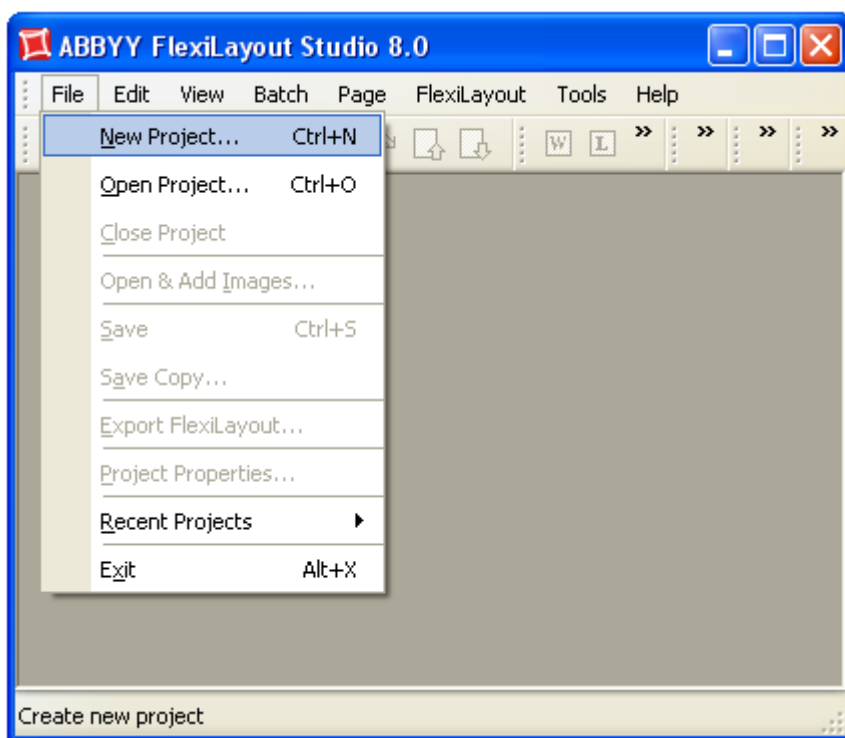
About some customs:

The Jack-o-lantern custom probably comes from Irish folklore. As the tale is told, a man named Jack, who was notorious as a drunkard and trickster, tricked Satan into climbing a tree. Jack then carved an image of a cross in the tree's trunk, trapping the devil up the tree. Jack made a deal with the devil that, if he would never tempt him again, he would promise to let him down the tree. According to the folk tale, after Jack died, he was denied entrance to Heaven because of his evil ways, but he was also denied access to Hell because he had tricked the devil. Instead, the devil gave him a single ember to light his way through the fright darkness. The ember was placed inside a hollowed-out turnip to keep it glowing longer. The Irish used turnips as their "Jack's lanterns" originally. But when the immigrants came to America, they found that pumpkins were far more plentiful than turnips. So the Jack-O-Lantern in America was a hollowed-out pumpkin, lit with an ember.

Шаг 1. Создание нового проекта

Замечание. Проект - это набор файлов, создаваемый программой ABBYY FlexiLayout™ Studio и содержащий всю информацию о гибком описании FlexiLayout.

1. Создайте новую папку **Sample1**.
2. Запустите программу ABBYY FlexiLayout Studio.
3. Создайте новый проект (выберите **New Project** в меню **File**).



4. В открывшемся диалоге задайте имя проекта **Sample1**.

После создания проекта в папке **Halloween Form** будут созданы:

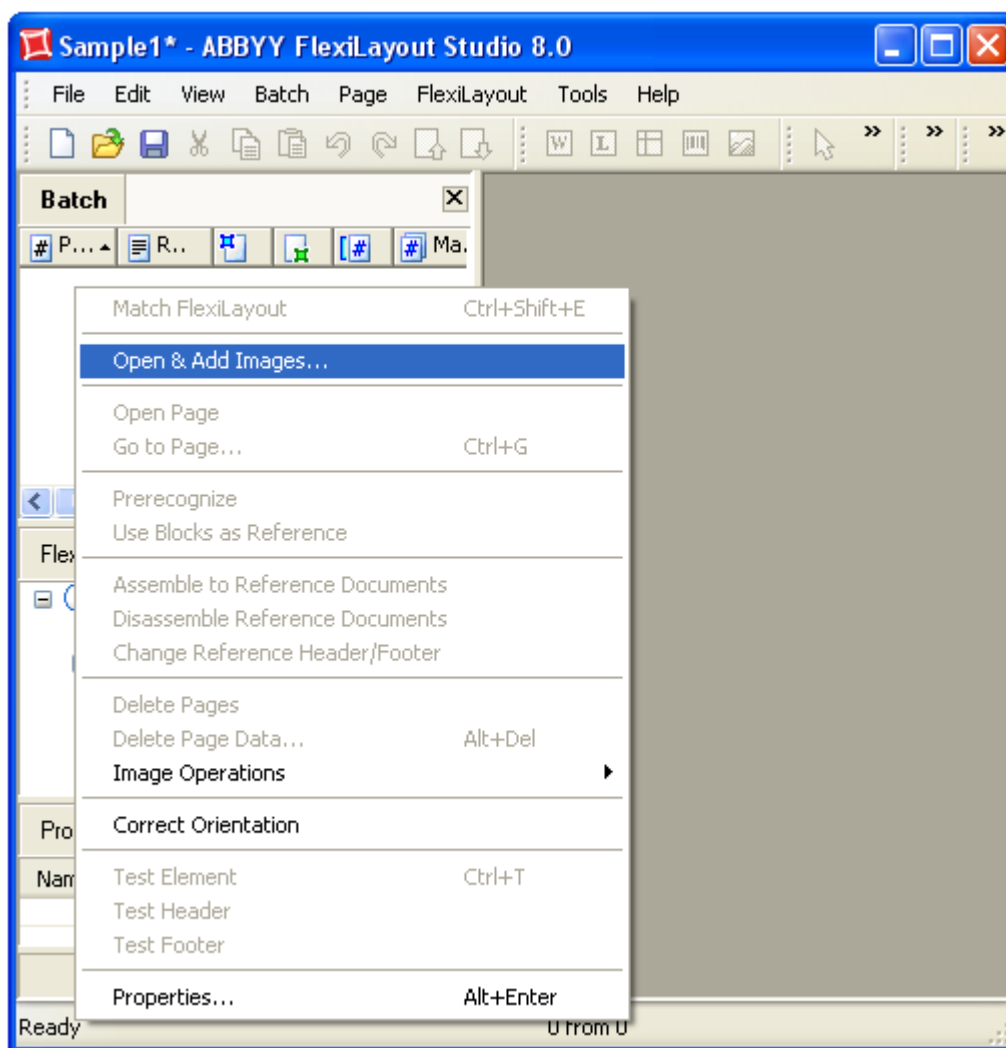
- o **Sample1.fsp** - файл проекта,
- o **Sample1Batch** – папка, в которую по умолчанию сохраняются изображения, добавляемые в пакет FlexiLayout,
- o **Sample1Templates** - папка, в которую по умолчанию сохраняется гибкое описание.

Шаг 2. Добавление изображений в пакет

 **Замечание.** Пакет - это набор тестовых изображений (страниц), с помощью которых будет создаваться и отлаживаться гибкое описание FlexiLayout.

После создания нового проекта необходимо добавить в пакет тестовые изображения, на которых будет осуществляться настройка и отладка гибкого описания. Для этого:

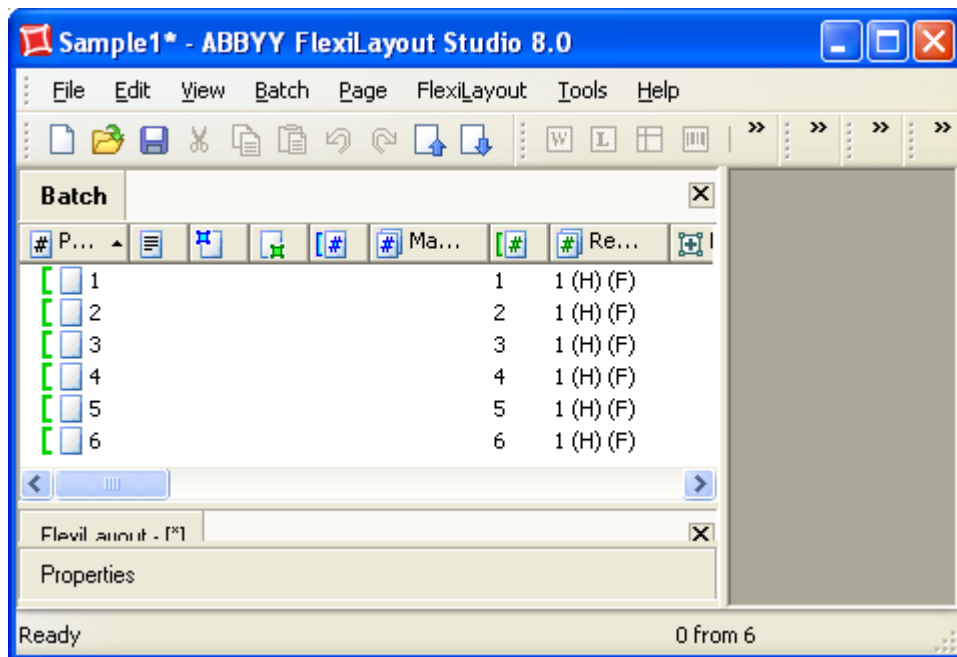
1. Перейдите в окно **Batch**.
2. Выберите команду **Open & Add Images** в контекстном меню, либо в меню **File** основного меню программы.



3. В открывшемся диалоге **Open Images**, отметьте опцию **Document per file** и укажите файлы тестовых изображений.

(тестовые изображения первого примера хранятся в
 <имя диска>:\Documents and Settings\All Users\Application Data\ABBYY\FlexiCapture\8.0\Samples\
 FlexiLayoutStudio\Halloween Form\Halloween FormBatch)

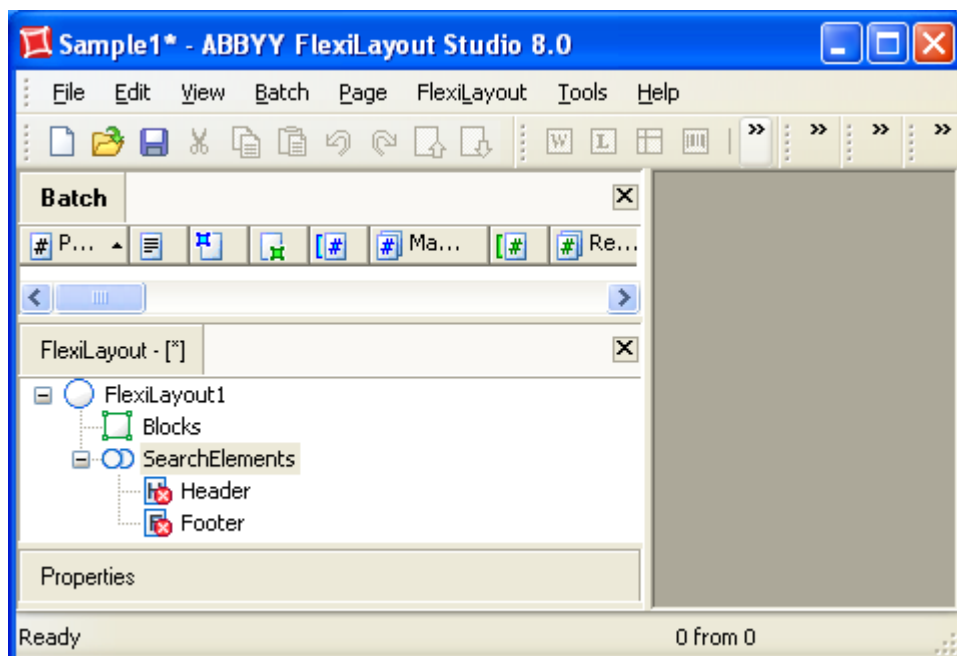
Добавленные в пакет тестовые изображения отобразятся в окне **Batch**.



Замечание. При добавлении изображений можно сразу определить принцип эталонной сборки изображений в документы. Выбрав опцию **Document per file**, мы указываем программе, что каждое изображение является отдельным документом. Подробнее об эталонной сборке см. Эталонная сборка страниц в документы

Шаг 3. Настройка свойств гибкого описания

После создания нового проекта гибкое описание по умолчанию называется **FlexiLayout1** (окно **FlexiLayout**).



Замечание. Значок ошибки рядом с **SearchElements** означает, что еще ни один элемент **FlexiLayout** не был создан.

Для удобства работы с гибким описанием рекомендуется его переименовать.

Для того чтобы задать свойства FlexiLayout (в том числе, для его переименования):

1. Дважды щелкните по имени гибкого описания (в данном случае **FlexiLayout1**) левой кнопкой мыши (либо выберите **Properties** в локальном меню, которое вызывается по нажатию правой кнопки мыши).

2. Задайте имя гибкого описания в поле **Name**, например, **RegistrationForm**.
3. Укажите параметры предраспознавания формы:

Text Language – наш документ является англоязычным, поэтому в качестве языка предраспознавания в списке выберите **English**.

В поле **Text Type** следует выбрать все необходимые способы заполнения документов, руководствуясь внешним видом тестовых изображений, или дополнительной информацией, получаемой от заказчика. В данном случае будем считать, что при заполнении документов не использовался ни матричный принтер, ни пишущая машинка. Поэтому оставляем **Typographic** (значение по умолчанию).

Режим предраспознавания **Mode**. При выборе режима следует руководствоваться в первую очередь качеством исходных изображений и размером шрифта. Изображения наших документов хорошего качества, на них нет мусора, на них практически отсутствуют текстовые участки, выполненные мелким шрифтом. Поэтому можно оставить режим **Fast**, предлагаемый по умолчанию. Это позволит ускорить процесс предраспознавания, как при отладке гибкого описания в программе FlexiLayout Studio, так и при наложении шаблона в программе ABBYY FlexiCapture. На этом процедура задания свойств гибкого описания закончена и можно приступать непосредственно к созданию элементов и блоков FlexiLayout.

Multi-page parameters – наш документ является одностраничным. Поэтому задайте значения параметров **Minimal number of pages** и **Maximal number of pages** равными 1.

Для одностраничных документов нет необходимости в описании начала и конца документа с помощью predefined составных элементов **Header** и **Footer**. Поэтому их можно удалить из списка элементов гибкого описания.

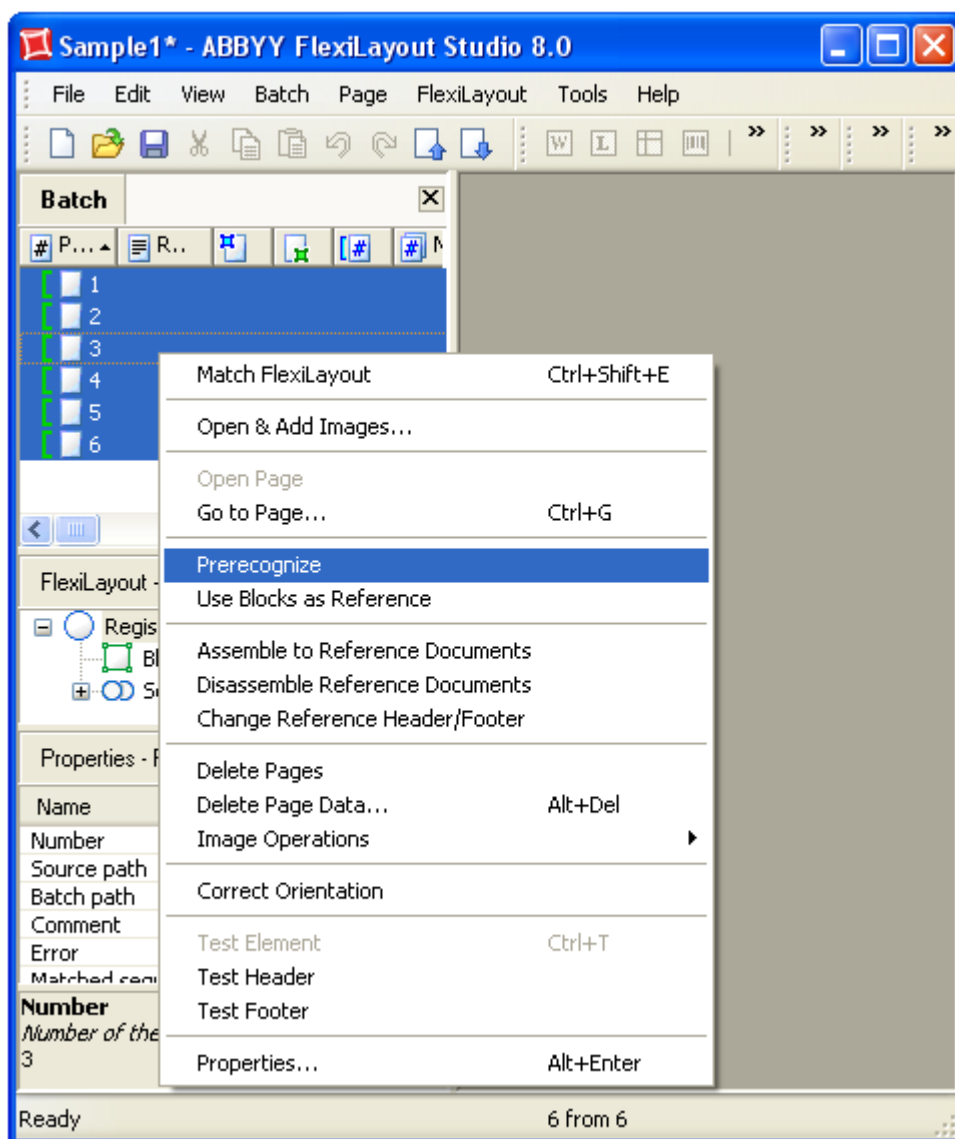
Замечание. Для определения начала и конца документа в гибкое описание могут быть добавлены специальные элементы: **Header** и **Footer**. Если в процессе наложения гибкого описания не найден ни элемент **Header** ни элемент **Footer**, то число страниц в документе считается равным наибольшему числу страниц, указанному в свойствах гибкого описания. Для данного документа в свойствах гибкого описания мы указали 1.

Шаг 4. Предраспознавание

Перед созданием элементов гибкого описания необходимо понять, на какие объекты изображений формы мы можем опереться для поиска ее полей данных формы. Такими объектами могут служить только такие графические и/или текстовые фрагменты изображения, которые устойчиво находятся в процессе предраспознавания изображений формы.

Для того чтобы запустить процесс предраспознавания:

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



Замечание. Если тестовые изображения в пакете меняются и/или добавляются, необходимо перезапустить процедуру предраспознавания. Если предраспознавание изображения не было выполнено предварительно, то оно будет сделано в процессе наложения гибкого описания.

Шаг 5. Просмотр изображений и результатов предраспознавания

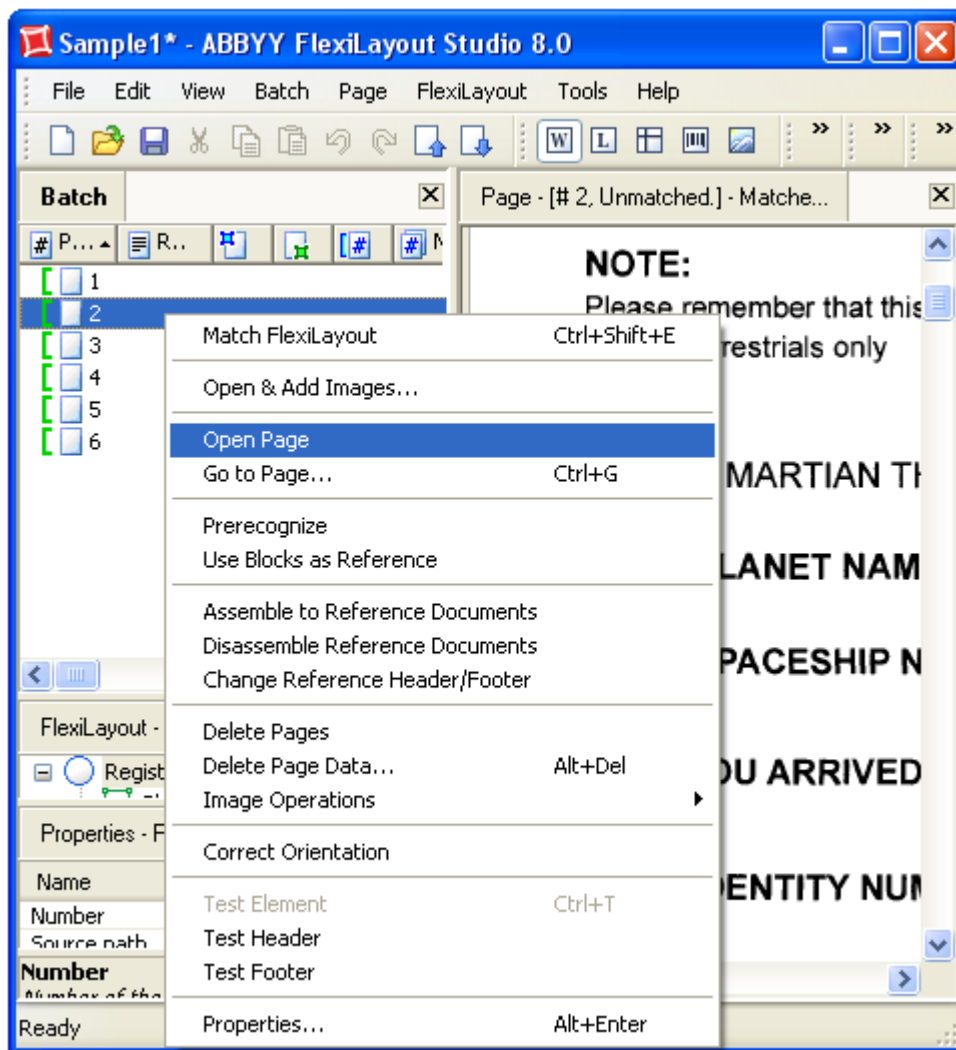
Откроем первую страницу пакета.

Для этого:

Дважды щелкните левой кнопкой мыши на соответствующей строке в окне **Batch**

или

Выберите команду **Open page** (меню **Batch** или локальное меню пакета).

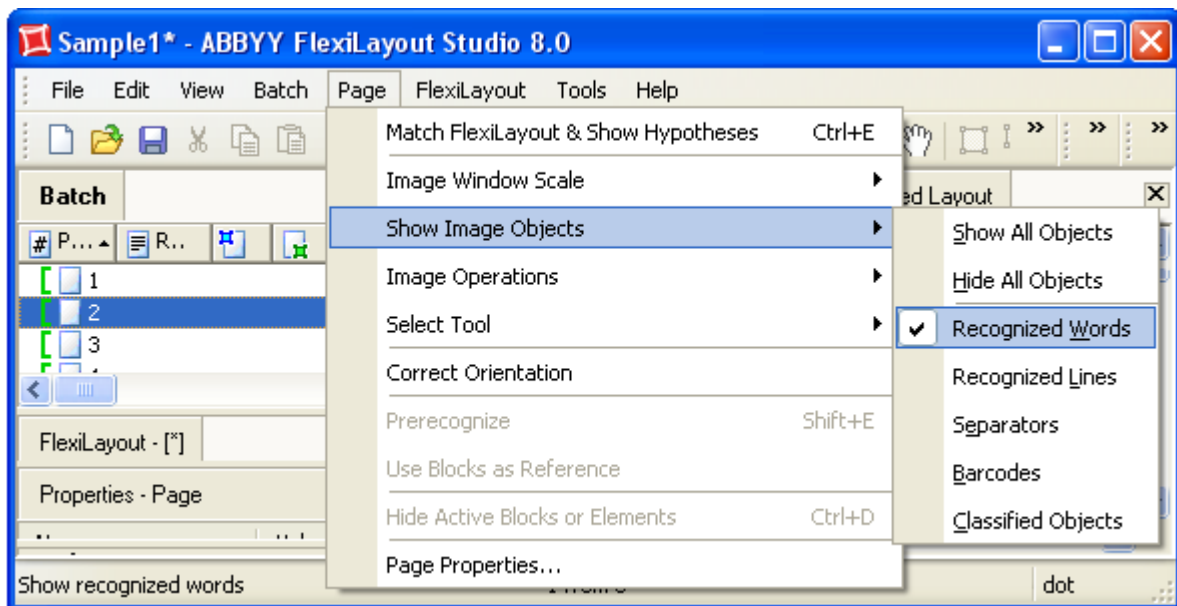


Открытое изображение отобразится в окне **Image** в режиме **Matched Layout**.

По умолчанию результаты предраспознавания не видны на изображении страницы сразу после окончания предраспознавания.

Для просмотра результатов предраспознавания:

Выберите команду **Show Image Objects** в меню **Page** основного меню программы и укажите, какой тип объектов изображения вы хотите посмотреть на изображении.



или

Выберите соответствующую кнопку на панели инструментов **Extracted Objects**.



Замечание. Также можно выбрать для просмотра все объекты **Show All Objects**.

Шаг 6. Предварительный анализ результатов предраспознавания и определение опорных элементов

Для анализа результатов предраспознавания бывает удобно просматривать объекты (изображения) каждого типа отдельно. Нас интересуют только объекты, которые, устойчиво обнаруживаются процедурой предраспознавания на каждом тестовом изображении, и которые впоследствии можно будет использовать в качестве опорных для поиска необходимых полей.

Предварительный просмотр результатов предраспознавания всех тестовых страниц показывает, что:

1. Все текстовые объекты, соответствующие заголовкам искомых полей, объекты типа **Barcode** и **Separator** вокруг фотографии обнаруживаются устойчиво. Поэтому на них можно будет опираться при поиске полей.
2. На изображениях 3, 5 и 6 обнаруживаются разделители, отсутствующие на других изображениях. На них нельзя опираться при поиске полей формы.
3. На изображениях 1 и 6 не все части фотографии распознаны как **Image Object** типа **Picture**. Поэтому для нахождения фотографии целиком поиска объекта типа **Picture** недостаточно, понадобятся дополнительные приемы. Их рассмотрим ниже при создании соответствующего элемента (Шаг 20. Описание поля Фотография).

Теперь можно приступить непосредственно к созданию и настройке опорных элементов и заданию критериев поиска блоков, соответствующих искомым полям.

Шаг 7. Создание идентификационного элемента

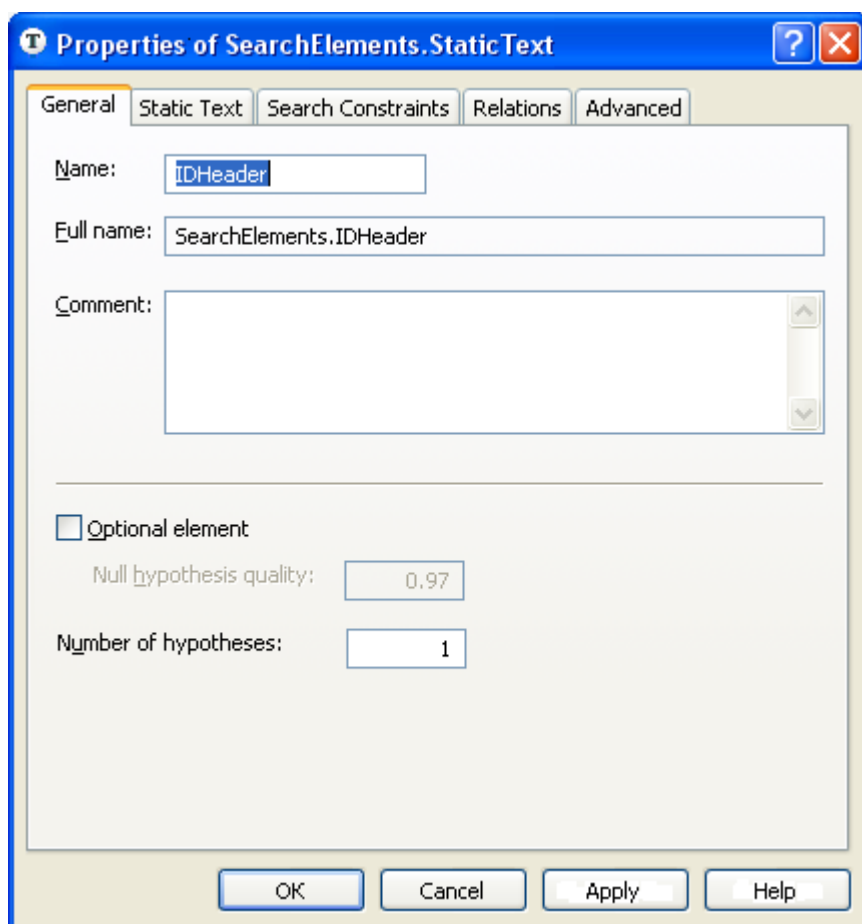
При обработке гибких форм в программе ABBYY FlexiCapture хотелось бы по возможности полностью исключить случайное распознавание форм, не относящихся к заданному типу. Одним из способов идентификации формы является задание, по крайней мере, одного обязательного элемента. Объект изображения, соответствующий этому элементу должен устойчиво обнаруживаться на всех документах данного типа (иначе гибкое описание не будет наложено на изображение). В нашем случае в качестве идентификатора будем использовать четкий, хорошо распознающийся текст заголовка формы **HALLOWEEN REGISTRATION FORM**.

 **Замечание.** Идентификационный элемент или идентификационный набор элементов можно описать в предопределенном составном элементе Header. В данном примере не рассматривается.

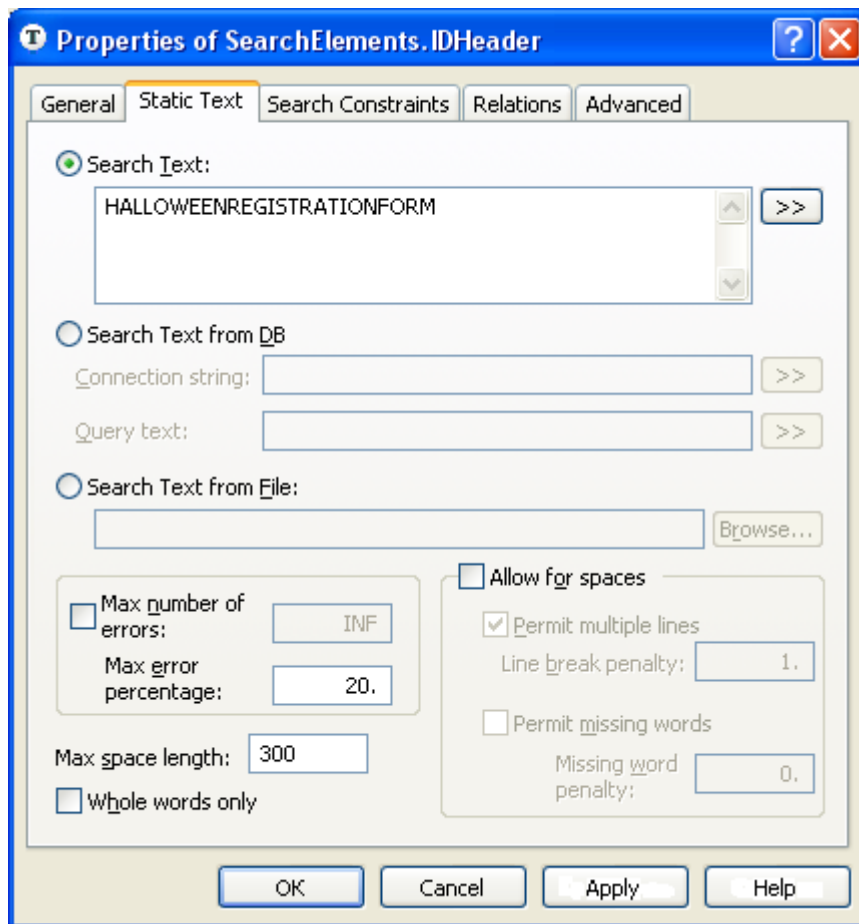
Заголовок формы нам понадобится исключительно для идентификации формы, а не для поиска полей, содержащих данные. Опишем его в шаблоне с помощью элемента типа **Static Text**.

Для того чтобы создать идентификационный элемент:

1. Перейдите в окно **FlexiLayout** в главном окне программы ABBYY FlexiLayout Studio.
2. В дереве FlexiLayout выделите **SearchElements**.
3. Выберите команду **Elements** (меню **FlexiLayout>Add element>Static Text** или локальное меню элемента).
4. Задайте имя элемента **IDHeader** в поле **Name**.
5. Отмените заданное по умолчанию свойство **Optional** на закладке **General** (поскольку данный элемент мы решили сделать обязательным).



6. Перейдите на закладку **Static Text**.



7. В поле **Search Text** укажите значение искомого текста **HALLOWEEN REGISTRATION FORM**. Ориентируясь на изображение первой страницы пакета, полагаем заголовок формы однострочным, поэтому запишем его без пробелов. Это позволит ускорить поиск однострочного статического текста.
8. Задайте максимально допустимое количество ошибок в искомом тексте в процентах или в явной форме (числом). Зададим в поле **Max error percentage** максимально допустимый процент ошибок 20. Таким образом, для нашего искомого заголовка в 25 символов мы допускаем 5 ошибок.

Замечание. Подбор значения допустимого числа ошибок является творческим процессом.

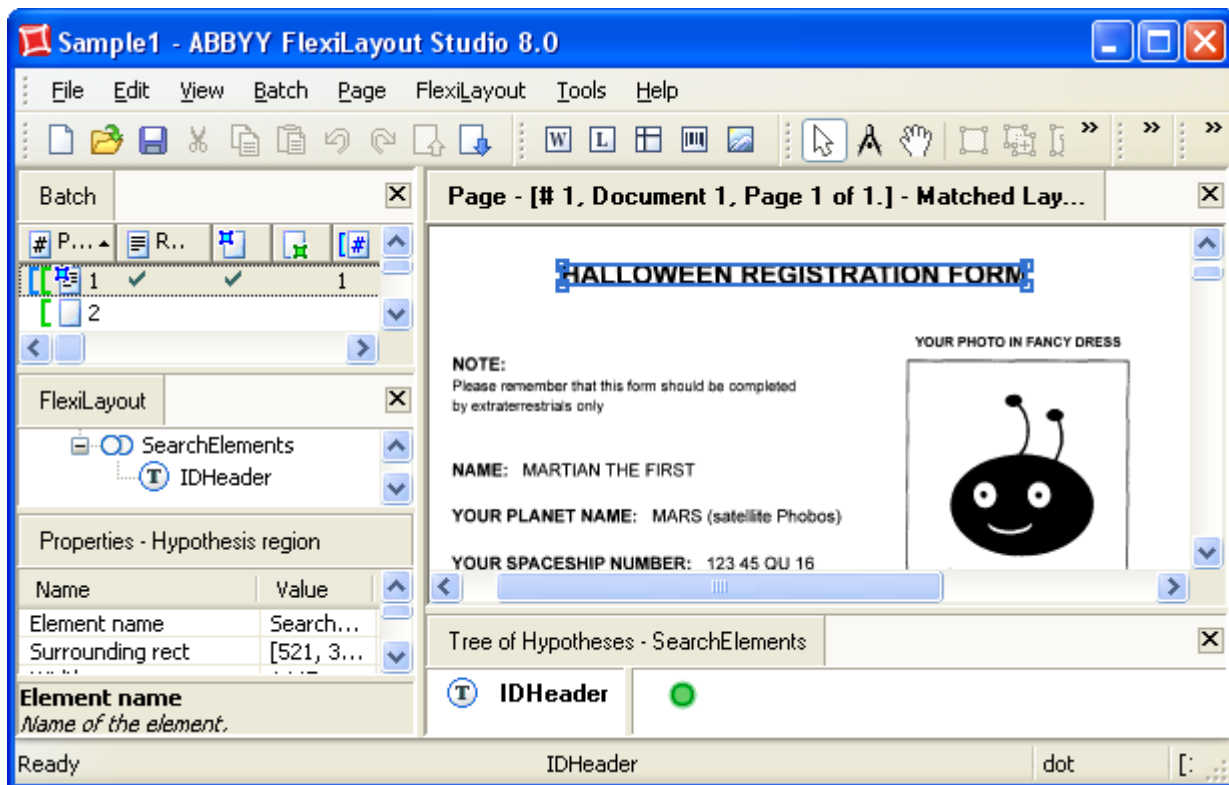
Шаг 8. Проверка нахождения заголовка формы, соответствующего идентификационному элементу

Для проверки обнаружения объекта, соответствующего идентификационному элементу **IDHeader**, необходимо наложить гибкое описание на каждое изображение пакета.

Для этого:

1. Откройте каждое изображение пакета.
2. Выберите команду **Match FlexiLayout & Show Hypotheses** (меню **Page** или локальное меню изображения).

При успешном наложении гибкого описания на изображении синим цветом выделяется регион объекта, соответствующего элементу **IDHeader**, а в окне **Tree of Hypotheses** сформированная гипотеза элемента **IDHeader** отображается значком . При нажатии на него курсором мыши в окне **Properties** отображаются свойства гипотезы.



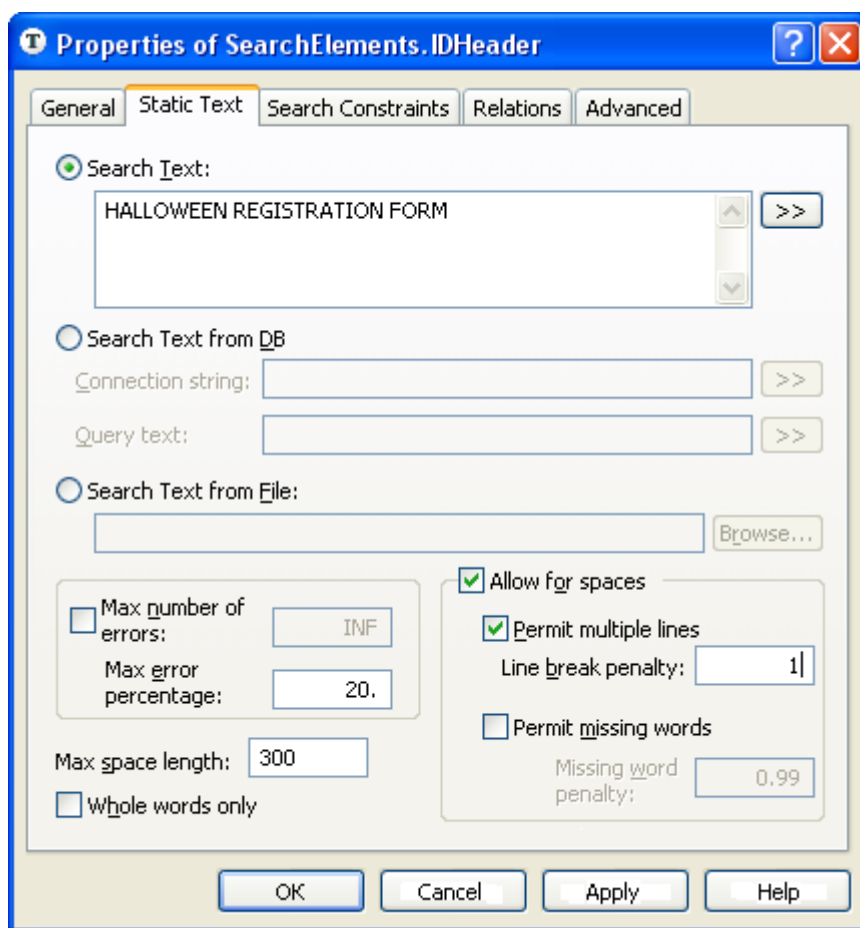
Просмотр результатов наложения гибкого описания на все изображения показывает, что на изображение 2 не удалось наложить гибкое описание. Причиной этого явился тот факт, что на данной странице заголовок написан в две строки, а в свойствах элемента **IDHeader** мы задали в качестве текста для поиска однострочный заголовок.

Шаг 9. Корректировка свойств идентификационного элемента

Для того чтобы объект изображения, соответствующий идентификационному элементу устойчиво обнаруживался на всех документах, необходимо скорректировать свойства идентификационного элемента.

Для этого:

1. Откройте диалог **Properties** со свойствами элемента.
2. Перейдите на закладку **Static Text**.



3. Укажите значение поля **Search text** с пробелами: **HALLOWEEN REGISTRATION FORM**. Это будет означать, что для идентификации формы мы будем искать фразу, которая в общем случае может быть и не однострочной.
4. Для задания многострочности искомой фразы отметьте опции **Allow for spaces** и **Permit multiple lines**.
5. Задайте значение параметра **Line break penalty** равным 1. Это позволит сохранить качество сформированной гипотезы без изменения при переходе ее на следующую строку. Задание меньшего значения приведет к штрафованию гипотезы.

Повторное наложение гибкого описания показывает, что объект изображения, соответствующий идентификационному элементу, устойчиво обнаруживается на всех тестовых изображениях.

Шаг 10. Описание поля YOUR PLANET NAME

Результаты предразpoznавания показывают, что на всех тестовых изображениях заголовки полей данных обнаруживаются устойчиво и являются уникальными в рамках данной формы. Кроме того, порядок следования полей на каждом изображении сохраняется, несмотря на различия в их абсолютном местоположении. Поэтому для настройки поиска всех полей имеет смысл:

1. Создать элементы типа **Static Text** для поиска заголовков.
2. Относительно них описать месторасположение элементов типа **Character String**, **Text Fragment**, **Barcode** и т.д., созданных для поиска полей.
3. Создать блоки (соответствующие полям), задав их месторасположение совпадающим с регионами соответствующих опорных элементов.

Для того чтобы описать месторасположение блока, соответствующего полю **Your Planet Name**, необходимо выполнить следующие шаги.

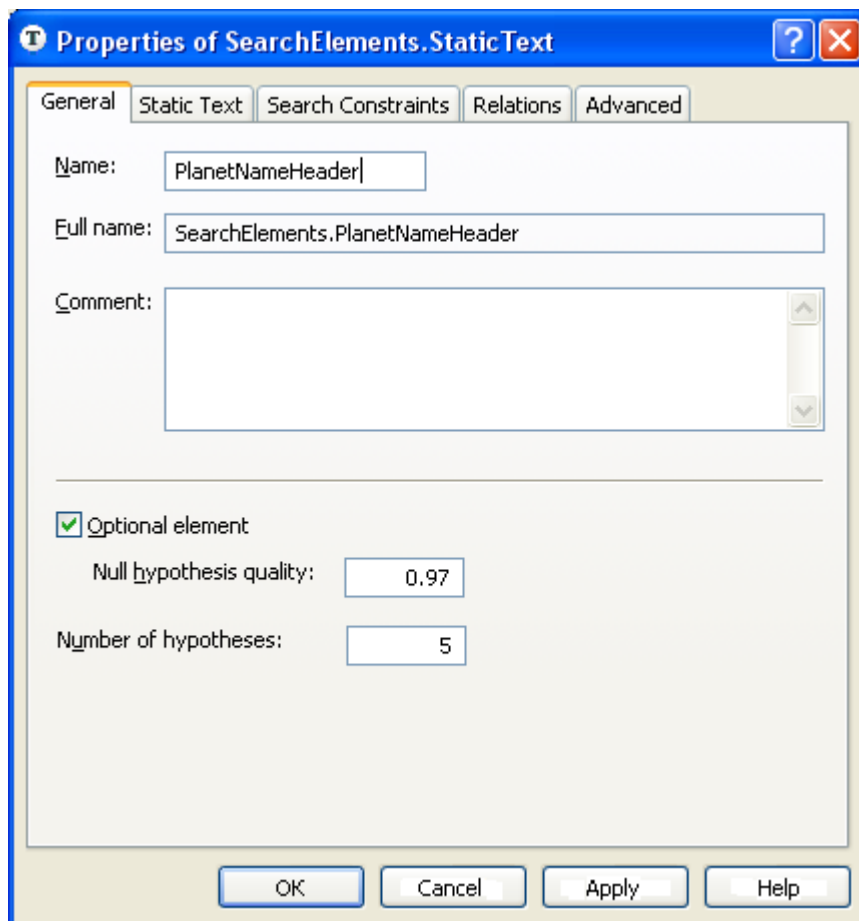
1. Создать элемент **PlanetNameHeader** (тип **Static Text**), соответствующий заголовку поля **YOUR PLANET NAME**.
2. Создать элемент **PlanetName** (тип **Character String**), соответствующий самому полю **YOUR PLANET NAME**.
3. Создать блок **PlanetName** (тип **Text**), соответствующий полю **YOUR PLANET NAME**.

Шаг 11. Описание заголовка поля YOUR PLANET NAME. Элемент PlanetNameHeader

Как можно заметить, полю, содержащему имя планеты, предшествует поле, содержащее имя гостя. Тем не менее, мы начнем именно с поиска поля **YOUR PLANET NAME**. Это необходимо для исключения неопределенности, которая может возникнуть, при поиске заголовков полей. Как можно видеть, слово **NAME**: входит в состав обоих заголовков. Поэтому при попытке найти слово **NAME**: без дополнительных ограничений будет сформировано несколько гипотез. Для избежания этой неоднозначности мы и будем искать сначала поле, содержащее имя планеты, а затем поле, содержащее имя гостя, наложив дополнительные ограничения на заголовок поля (Шаг 15).

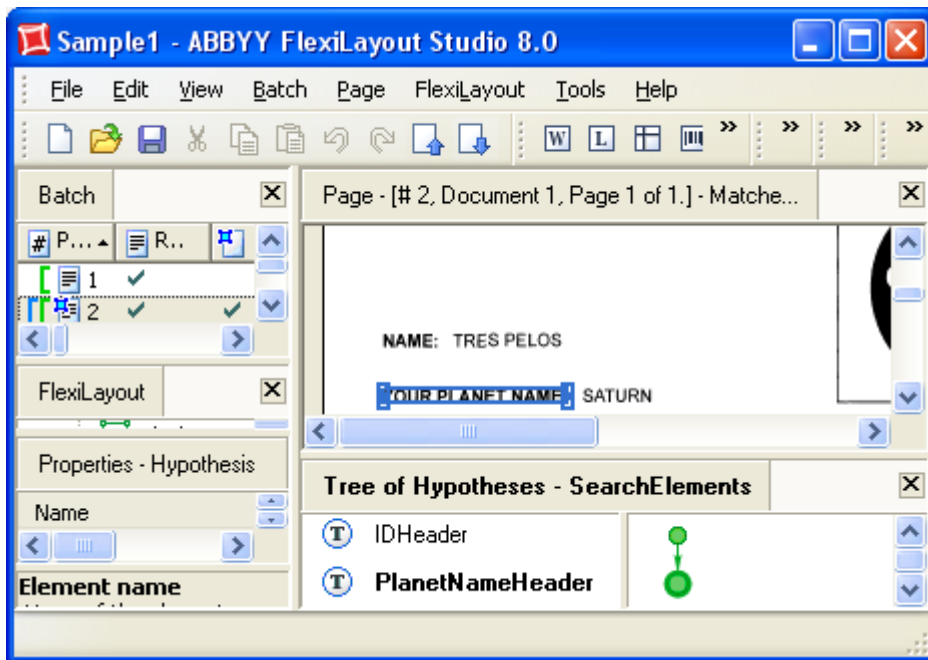
Для того чтобы создать элемент **PlanetNameHeader**:

1. Создайте элемент **Static text** с именем **PlanetNameHeader**. Оставим выбранным пункт **Optional element**. В отличие от элемента, соответствующего заголовку формы, который в целях идентификации был задан как обязательный, предполагается, что в общем случае все остальные элементы формы могут быть необязательными. Это позволяет формировать гипотезы (нулевые гипотезы), даже для тех элементов, для которых не будут найдены соответствующие объекты изображения.



2. Выберите закладку **Static Text**.
3. В поле **Search Text** укажите значение искомого текста. Как можно видеть, на всех тестовых документах заголовков является однострочным. Поэтому мы можем написать заголовок поля без пробелов **YOURPLANETNAME**.
4. Задайте значение свойства **Max error percentage** равным 20, поскольку заголовок поля состоит из нескольких слов и является достаточно длинным.

Наложение гибкого описания показывает, что заголовок поля, содержащего имя планеты, успешно найден на всех тестовых изображениях.



Шаг 12. Описание поля YOUR PLANET NAME. Элемент PlanetName

Будем полагать, что значение поля **YOUR PLANET NAME** всегда является однострочным. Поскольку значение этого поля не является фиксированным, в отличие от заголовка поля, для его нахождения следует создать элемент типа **Character String**.

Для того чтобы создать элемент **PlanetName**:

1. Создайте элемент типа **Character String** с именем **PlanetName**.

The screenshot shows the 'Properties of SearchElements.CharacterString' dialog box with the 'General' tab selected. The 'Name' field contains 'PlanetName', and the 'Full name' field contains 'SearchElements.PlanetName'. The 'Comment' field is empty. The 'Optional element' checkbox is checked. The 'Null hypothesis quality' is set to 0.97, and the 'Number of hypotheses' is set to 5. The 'OK', 'Cancel', 'Apply', and 'Help' buttons are at the bottom.

Properties of SearchElements.CharacterString

General Character String Search Constraints Relations Advanced

Name: PlanetName

Full name: SearchElements.PlanetName

Comment:

☒ Optional element

Null hypothesis quality: 0.97

Number of hypotheses: 5

OK Cancel Apply Help

2. Выберите закладку **Character String**.

The screenshot shows the 'Properties of SearchElements.CharacterString' dialog box with the 'Character String' tab selected. The 'Regular expression' radio button is unselected, and the 'Alphabets' radio button is selected. A table lists character sets, with the first entry being '()ABCDEFGHIJKLMNOPQRSTUVWXYZabcd...' with a 'Portio...' of 100. The 'Add...', 'Edit...', and 'Delete' buttons are to the right of the table. The 'Allow embedded hypotheses' checkbox is unselected, and the 'Percentage of non-alphabet characters' is set to 10. The 'Whole words only' checkbox is unselected. The 'Percentage of word ends' is set to 100, and the 'Max space length' is set to 300. The 'String length' is set to '{ 0, 3, 35, 100 }', and the 'Total length of spaces' is set to '{ -1, -1, INF, INF }'. The 'OK', 'Cancel', 'Apply', and 'Help' buttons are at the bottom.

Properties of SearchElements.CharacterString

General Character String Search Constraints Relations Advanced

☐ Regular expression:

☒ Alphabets:

Character Set	Portio...
()ABCDEFGHIJKLMNOPQRSTUVWXYZabcd...	100.

Add... Edit... Delete

☐ Allow embedded hypotheses

Percentage of non-alphabet characters: 10

☐ Whole words only

Percentage of word ends: 100. Max space length: 300

String length: { 0, 3, 35, 100 }

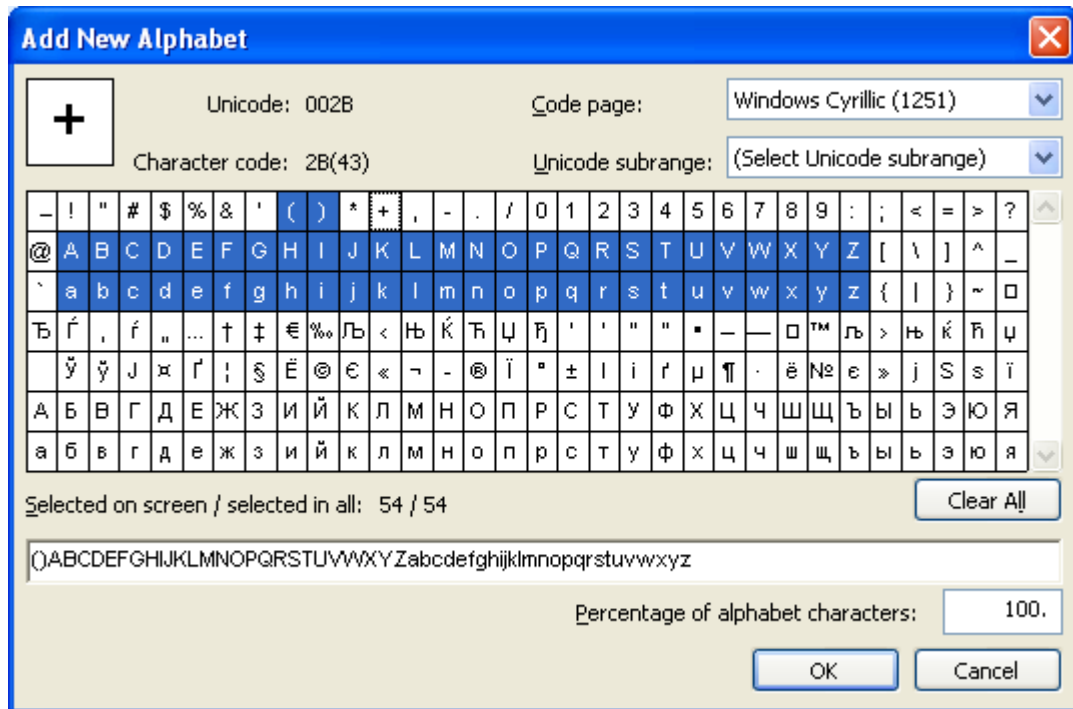
Total length of spaces: { -1, -1, INF, INF }

OK Cancel Apply Help

3. Задайте алфавит (все возможные символы, которые могут встретиться в названии планет и их спутников). Исходя из информации, полученной при просмотре тестовых изображений, предполагаем, что это – все символы латинского алфавита и круглые скобки. Для задания символов алфавита следует нажать кнопку **Add** и выделить соответствующие символы в открывшемся диалоге **Add New Alphabet**.

 **Замечание.** Выбранные символы отобразятся в столбце **Character Set**, поля **Alphabet**, закладки **Character String**.

4. Задайте значение параметра **Percentage of alphabet characters** равным 100. Это значит, что при поиске объекта, соответствующего элементу **PlanetName**, в расчет будут приниматься только символы данного алфавита.



 **Замечание.** В общем случае для элемента может быть создано несколько разных алфавитов. В поле **Percentage of alphabet characters** можно задать максимально допустимую долю участия символов каждого из них в данном элементе.

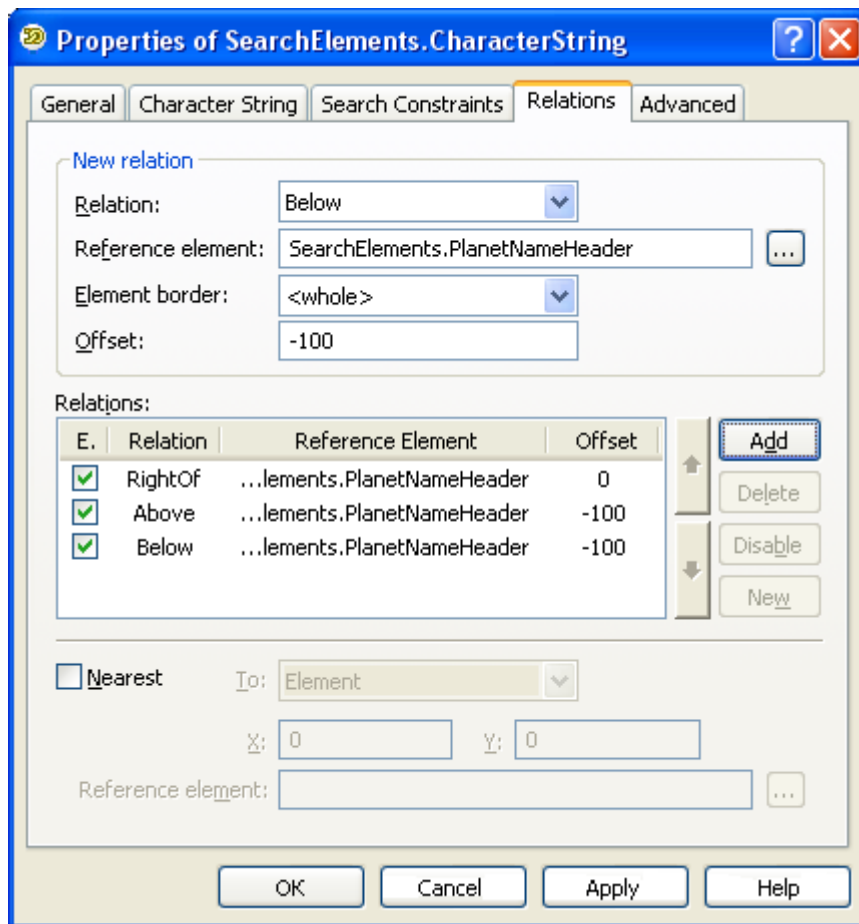
5. Снимите отметку с пункта **Allow embedded hypotheses**. Это позволит формировать гипотезы максимальной длины, удовлетворяющие всем условиям поиска. Иначе может быть сформировано несколько пересекающихся гипотез меньшей длины, каждая из которых состоит из кусочков, относящихся к одному из заданных алфавитов (с учетом параметра **Persantage of non-alphabet characters**, т.е. с добавлением допустимого числа неалфавитных символов), и удовлетворяет всем условиям поиска.
6. В поле **Persantage of non-alphabet characters** задайте допустимую долю символов, не входящих в алфавит данного элемента, равную 10

 **Замечание.** Значение данного свойства подбирается опытным путем и может быть изменено в процессе отладки гибкого описания.

7. В поле **String Length** задайте нечеткий интервал {0, 3, 35, 100}, оценивающий длину цепочки символов. Предполагаем, что все возможные значения имен планет, которые могут встретиться в данном поле, будут включать от 3 до 35 символов. Любая гипотеза, выходящая за рамки данного интервала, будет штрафоваться.

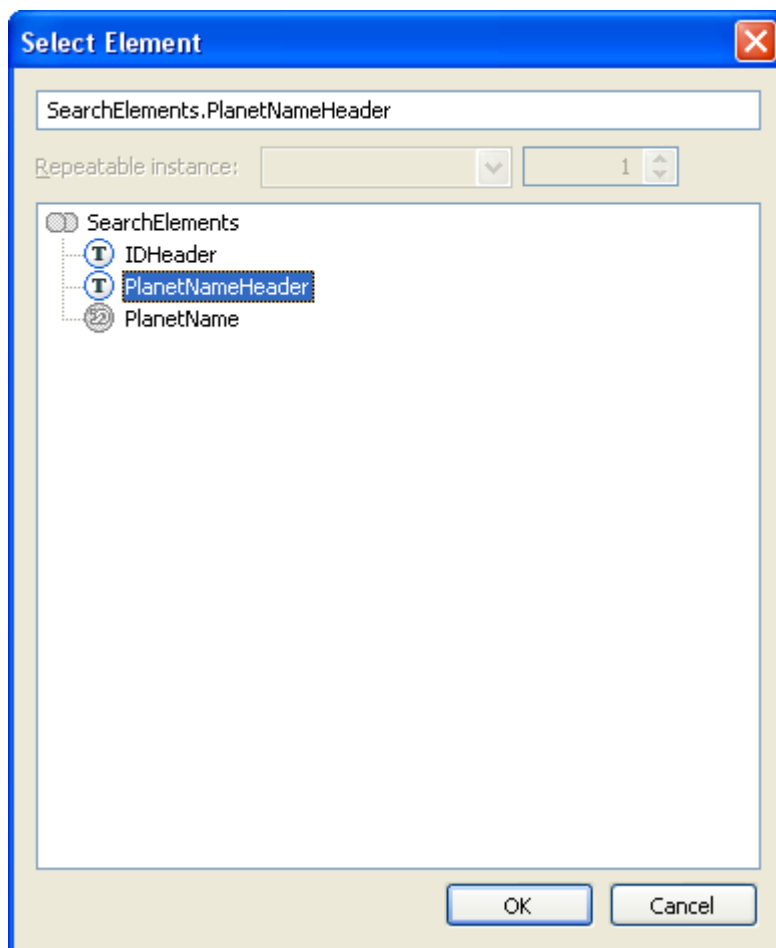
 **Замечание.** Значение данного свойства подбирается опытным путем и может быть изменено в процессе отладки гибкого описания.

8. В поле **Total length of spaces** задайте нечеткий интервал, оценивающий суммарную длину всех пробелов между символами в гипотезе (по умолчанию установлено значение dot = 1/300 дюйма). Оставляем значение по умолчанию { -1, -1, INF, INF }. Таким образом, мы не накладываем никаких ограничений на длину пробелов в гипотезе.
9. Перейдите на закладку **Relations**.



В качестве опорного элемента (элемента, относительно которого задается положение искомого элемента) для элемента **PlanetName** будем использовать элемент **PlanetNameHeader**. Имя планеты всегда находится справа от заголовка поля, а также на одном уровне с ним. Поэтому помимо отношения правее, необходимо задать положение имени планеты относительно верхней и нижней границы элемента заголовка. Иначе в качестве гипотезы может быть найден любой текст, соответствующий заданному алфавиту, и лежащий в правой плоскости относительно элемента заголовка, в том числе выше или ниже него. Итак, будем искать имя планеты, заданное элементом **PlanetName**, справа от заголовка, заданного элементом **PlanetNameHeader** и находящееся на одном уровне с заголовком.

10. Задайте положение элемента **PlanetName** справа от элемента **PlanetNameHeader**. Имя элемента **PlanetNameHeader** следует выделить курсором в окне **Select element**. В выпадающем списке **Relation** выберите значение **RightOf** и оставьте значение свойства **Offset** равным 0. Нажмите кнопку **Add**.



11. Задайте положение элемента **PlanetName** не ниже элемента **PlanetNameHeader**. Имя элемента **PlanetNameHeader** следует выделить курсором в окне **Select element**. В выпадающем списке **Relation** выберите значение **Above** и задайте значение свойства **Offset** равным **-100** (значение подбирается опытным путем), чтобы описать область поиска с небольшим запасом по высоте заголовка. Отрицательное значение смещения позволяет задавать положение элемента выше нижней границы заголовка. Нажмите кнопку **Add**.
12. Задайте положение элемента **PlanetName** не выше элемента **PlanetNameHeader**. Имя элемента **PlanetNameHeader** следует выделить курсором в окне **Select element**. В выпадающем списке **Relation** выберите значение **Below** и задайте значение свойства **Offset** равным **-100** (значение подбирается опытным путем), чтобы описать область поиска с небольшим запасом по высоте заголовка. Отрицательное значение смещения позволяет задавать положение элемента ниже верхней границы заголовка. Нажмите кнопку **Add**.

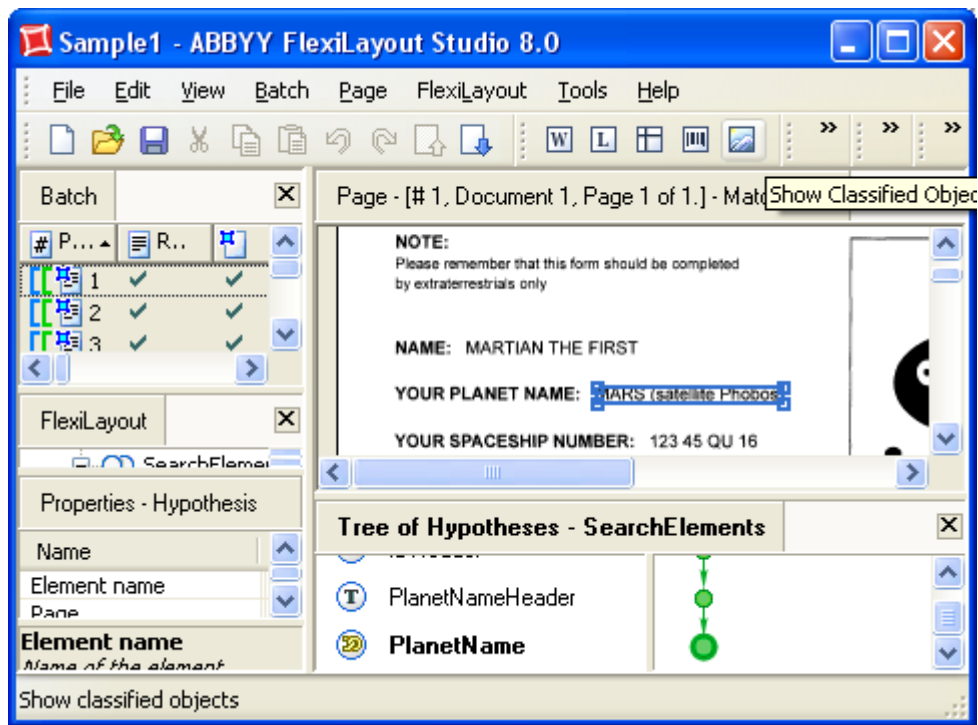
Замечание. Неотрицательные смещения при указании в поле **Element border** значения **<whole>** позволяют задавать только положение типа правее правой границы элемента, левее левой границы элемента, выше верхней границы элемента и ниже нижней границы элемента. Задание области поиска между верхней и нижней границами заголовка при указании в свойстве **Element border** значений **<top>** и **<bottom>** с неотрицательным значением свойства **Offset** ограничивает область поиска высотой заголовка, что исключает обнаружение гипотез, слегка выходящих по вертикали за горизонтальные границы заголовка.

Шаг 13. Проверка нахождения поля, содержащего имя планеты

Для проверки правильности нахождения поля, соответствующего элементу **PlanetName**, перейдите в окно **Batch**. И для каждого изображения запустите процедуру наложения гибкого описания. Для этого:

1. Откройте изображение.
2. Выберите команду **Match FlexiLayout & Show Hypotheses** (меню **Page** локальное меню изображения, окно **Image**).

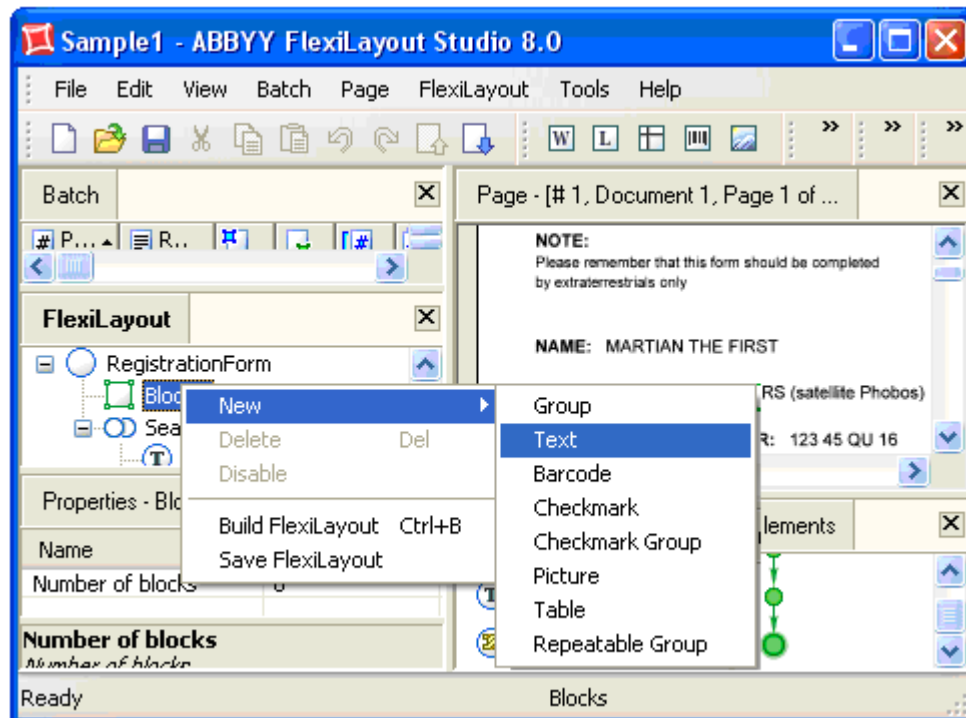
Проверка показала, что на всех изображениях было успешно найдено поле, содержащее имя планеты, соответствующее элементу **PlanetName**.



Шаг 14. Описание поля YOUR PLANET NAME. Блок PlanetName

Для того чтобы создать блок, соответствующий полю **YOUR PLANET NAME**:

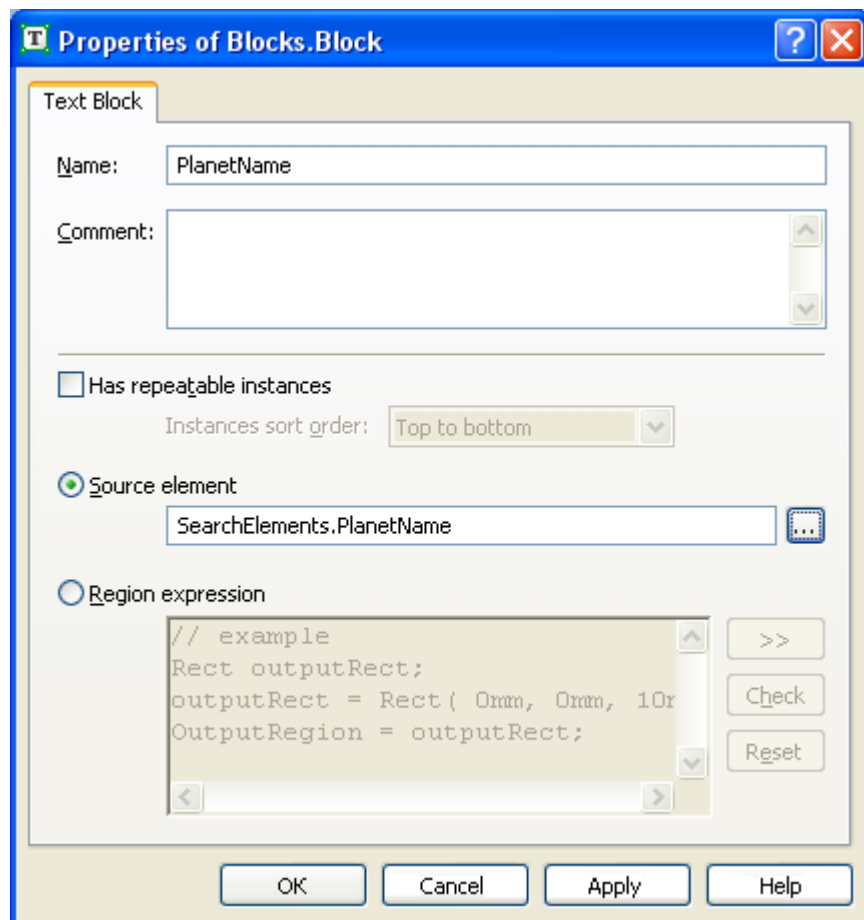
1. Перейдите в окно **FlexiLayout** в главном окне программы ABBYY FlexiLayout Studio.
2. В дереве **FlexiLayout** выделите объект **Blocks**.
3. Выберите команду **Add Block>Text** в меню **FlexiLayout** (или в локальном меню **New>Text**).



4. В открывшемся диалоге **Properties** задайте имя блока, например, **PlanetName** в поле **Name**.

 **Замечание.** Совпадение имени блока с именем элемента, соответствующего полю **Your Planet Name** не обязательно, но удобно для дальнейшей работы с гибким описанием.

5. Выберите в качестве описания местоположения блока опцию **Source element**. В качестве исходного элемента выберите элемент **PlanetName**, нажав кнопку "...".



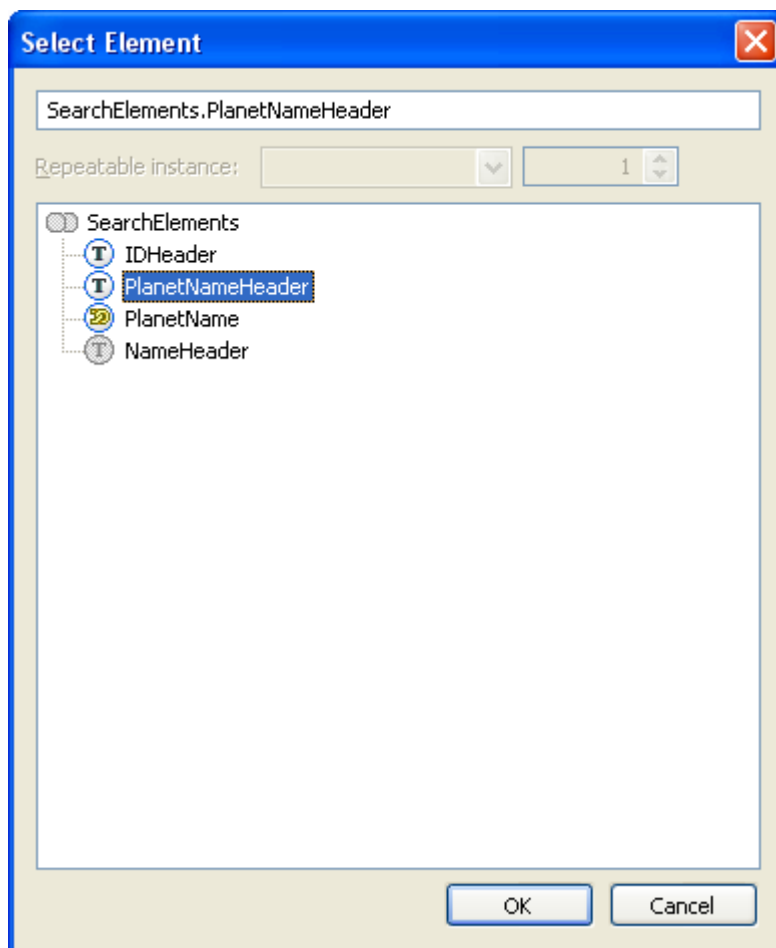
Шаг 15. Описание поля NAME

Для того чтобы описать месторасположение блока, соответствующего полю **NAME**, необходимо:

1. Создать элемент **NameHeader** (тип **Static Text**), соответствующий заголовку поля **NAME**.
2. Создать элемент **GuestName** (тип **Character String**), соответствующий самому полю **NAME**.
3. Создать блок **GuestName**, соответствующий полю **NAME**.

Процесс создания элементов и блока для данного поля аналогичен процессу создания элементов и блока для поля **YOUR PLANET NAME** (см. Шаг 10 – Шаг 14), поэтому мы остановимся более подробно лишь на отличительных особенностях:

При описании области поиска элемента **NameHeader** укажем, что он находится над элементом **PlanetNameHeader**. Тем самым мы исключаем возможность формирования ошибочной гипотезы, содержащей слово **Name** из заголовка **Your planet name**.



В отличие от элемента **PlanetName**, зададим для элемента **GuestName** алфавит, состоящий исключительно из букв латинского алфавита и символа "-". И в поле **String Length** зададим нечеткий интервал [0, 0, 50, INF]

При описании области поиска элемента **GuestName** укажем в качестве опорного элемента элемент **NameHeader**.

При создании блока **GuestName** укажем в качестве исходного элемента элемент **GuestName**.

Шаг 16. Описание поля YOUR SPACESHIP NUMBER

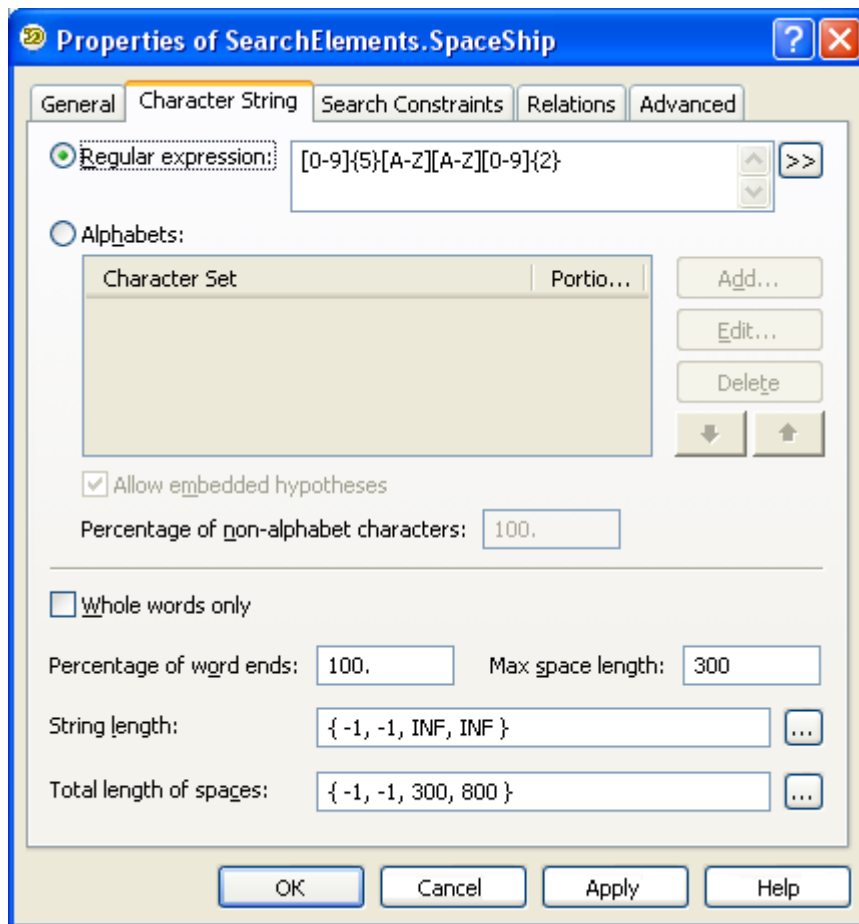
Для того чтобы описать месторасположение блока, соответствующего полю **YOUR SPACESHIP NUMBER**, необходимо:

1. Создать элемент **SpaceShipHeader** (тип **Static Text**), соответствующий заголовку поля **YOUR SPACESHIP NUMBER**.
2. Создать элемент **SpaceShip** (тип **Character String**), соответствующий самому полю **YOUR SPACESHIP NUMBER**.
3. Создать блок **SpaceShip**, соответствующий полю **YOUR SPACESHIP NUMBER**.

Процесс создания элементов и блока для данного поля аналогичен процессу создания элементов и блока для поля **YOUR PLANET NAME** (см. Шаг 10 – Шаг 14), поэтому мы остановимся более подробно лишь на отличительных особенностях:

В отличие от элемента **PlanetName**, зададим для элемента **SpaceShip** не алфавит, а регулярное выражение. Это обусловлено тем, что структура информации в поле **YOUR SPACESHIP NUMBER** одинакова на всех тестовых изображениях. Опишем номер космического корабля следующим регулярным выражением:

[0-9]{5}[A-Z][A-Z][0-9]{2} (подробнее о задании регулярного выражения см. в справке к программе ABBYY FlexiCapture).



При описании области поиска элемента **SpaceShip** укажем в качестве опорного элемента элемент **SpaceShipHeader**.
При создании блока **SpaceShip** укажем в качестве исходного элемента элемент **SpaceShip**.

Шаг 17. Описание поля DATE YOU ARRIVED AT THE EARTH

Для того чтобы описать месторасположение блока, соответствующего полю **DATE YOU ARRIVED AT THE EARTH**, необходимо:

1. Создать элемент **DateHeader** (тип **Static Text**), соответствующий заголовку поля **DATE YOU ARRIVED AT THE EARTH**.
2. Создать элемент **ArrivalDate** (тип **Date**), соответствующий самому полю **DATE YOU ARRIVED AT THE EARTH**.
3. Создать блок **ArrivalDate**, соответствующий полю **DATE YOU ARRIVED AT THE EARTH**.

Процесс создания элементов и блока для данного поля аналогичен процессу создания элементов и блока для поля **YOUR PLANET NAME** (см. Шаг 10 – Шаг 14), поэтому мы остановимся более подробно лишь на отличительных особенностях:

В отличие от элемента **PlanetNameHeader**, укажем для элемента **DateHeader** значение поля **Search Text** с пробелами: Date you arrived at the earth. Выберем опции **Allow for spaces** и **Permit multiple lines**, зададим значение свойства **Line break penalty** равным 1.

Properties of SearchElements.DateHeader

General Static Text Search Constraints Relations Advanced

☒ Search Text:

DATE YOU ARRIVED AT THE EARTH: >>

☐ Search Text from DB

Connection string: >>

Query text: >>

☐ Search Text from File:

Browse...

☐ Max number of errors: INF

Max error percentage: 20.

Max space length: 300

☐ Whole words only

☒ Allow for spaces

☒ Permit multiple lines

Line break penalty: 1.

☐ Permit missing words

Missing word penalty: 0.

OK Cancel Apply Help

Для поля **DATE YOU ARRIVED AT THE EARTH** (в отличие от поля **YOUR PLANET NAME**) создадим элемент **ArrivalDate** типа **Date**. Для элемента **ArrivalDate** на закладке **Date** следует указать все допустимые форматы записи даты. Предполагаем, что месяц будет всегда записан цифрами, а год всегда будет состоять из 4 цифр. Следовательно, отменим выбор пунктов **Word** (группа **Month**) и **Two digits** (группа **Year**). Остальные свойства оставим без изменения.

При описании области поиска элемента **ArrivalDate** укажем в качестве опорного элемента элемент **DateHeader**.
При создании блока **ArrivalDate** в качестве исходного элемента выберем элемент **ArrivalDate**.

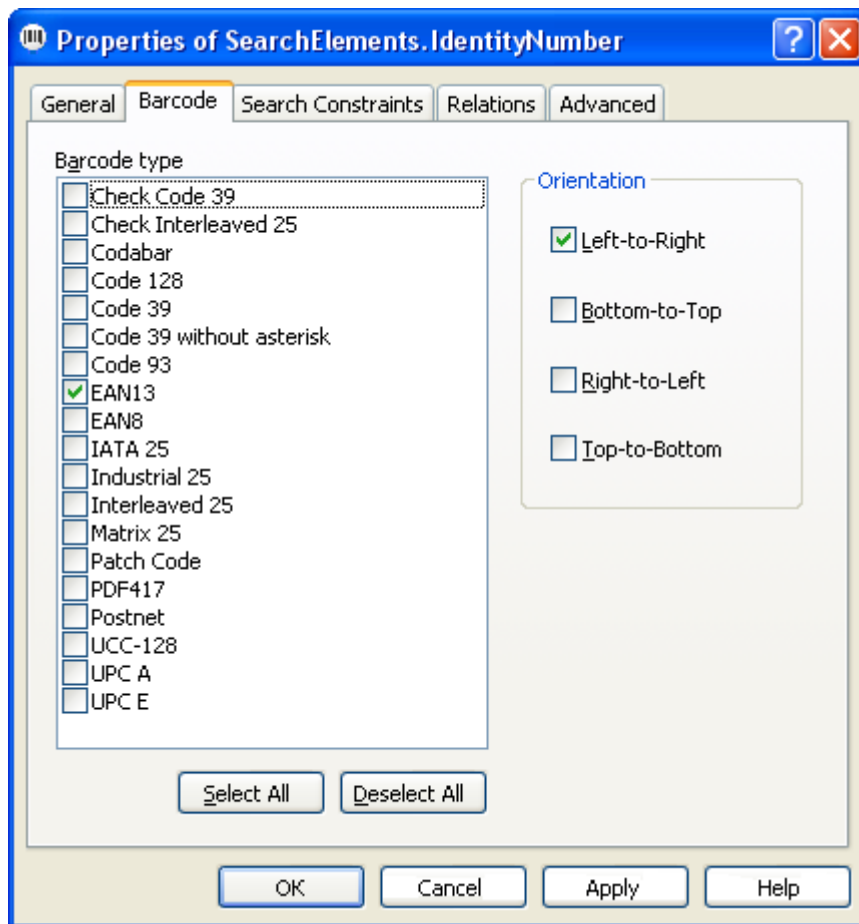
Шаг 18. Описание поля YOUR IDENTITY NUMBER ON THE PARTY

Для того чтобы описать месторасположение блока, соответствующего полю **YOUR IDENTITY NUMBER ON THE PARTY**, необходимо:

1. Создать элемент **BarcodeHeader** (тип **Static Text**), соответствующий заголовку поля **YOUR IDENTITY NUMBER ON THE PARTY**.
2. Создать элемент **IdentityNumber** (тип **Barcode**), соответствующий самому полю **YOUR IDENTITY NUMBER ON THE PARTY**.
3. Создать блок **IdentityNumber**, соответствующий полю **YOUR IDENTITY NUMBER ON THE PARTY**.

Процесс создания элементов и блока для данного поля аналогичен процессу создания элементов и блока для поля **YOUR PLANET NAME** (см. Шаг 10 – Шаг 14), поэтому мы остановимся более подробно лишь на отличительных особенностях:

Для поля **YOUR IDENTITY NUMBER ON THE PARTY** (в отличие от поля **YOUR PLANET NAME**) создадим элемент **IdentityNumber** типа **Barcode**. Для элемента **IdentityNumber** на закладке **Barcode** укажем его тип **EAN13** и ориентацию **Left-to-Right**.



При создании блока **IdentityNumber** в качестве исходного элемента выберем элемент **IdentityNumber**.

Шаг 19. Описание поля ПРОИЗВОЛЬНЫЙ ТЕКСТ

При просмотре всех тестовых изображений можно заметить, что в отличие от всех случаев, рассмотренных выше, у поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ**, располагающегося в нижней части формы, нет постоянного заголовка. Поэтому в качестве опорного объекта изображения нужно использовать не заголовок, а другие объекты. Для настройки нашего гибкого описания будем считать, что поле **ПРОИЗВОЛЬНЫЙ ТЕКСТ** содержит свой заголовок. В качестве опорного элемента для элемента, созданного для поиска поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ**, попробуем сначала использовать уже созданный элемент **IdentityNumber**, поскольку для всех тестовых изображений прослеживается закономерность размещения поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ** ниже поля **YOUR IDENTITY NUMBER ON THE PARTY**.

Замечание. Как уже отмечалось выше, случайные разделители, присутствующие на некоторых страницах, при поиске поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ** использовать не следует.

Для того чтобы найти поле **ПРОИЗВОЛЬНЫЙ ТЕКСТ**, используя в качестве опорного элемент **IdentityNumber** необходимо:

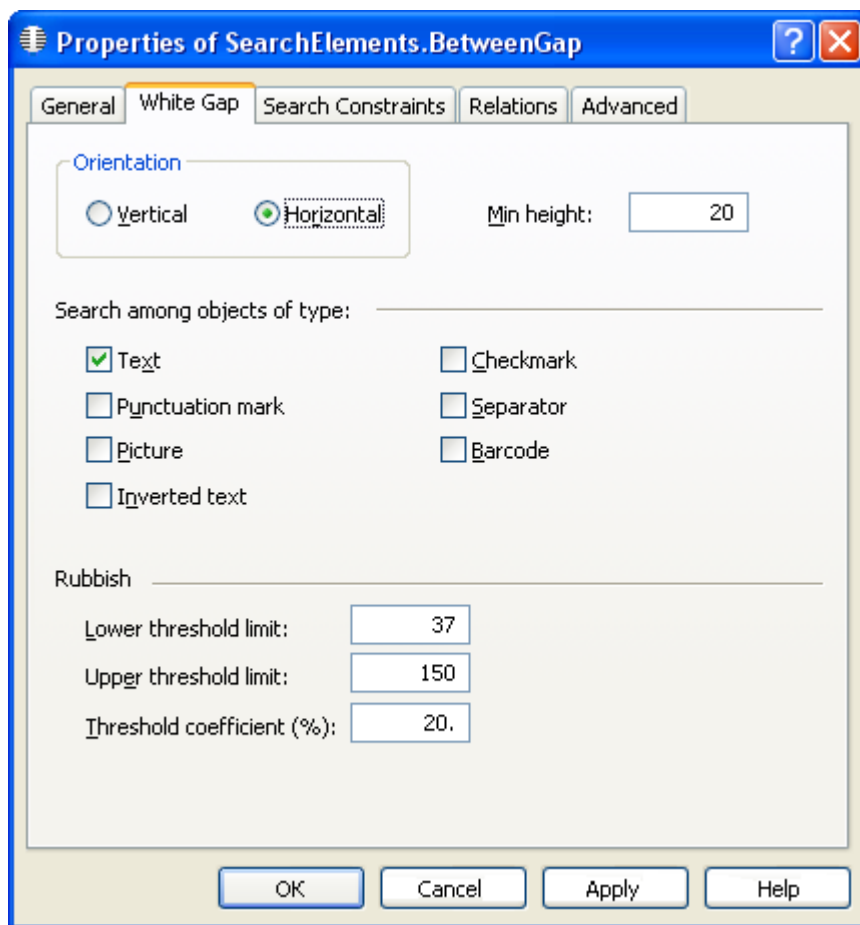
1. Создать элемент **TextField** типа **Text Fragment**, соответствующий полю **ПРОИЗВОЛЬНЫЙ ТЕКСТ**.
2. Задать положение элемента **TextField** на закладке **Relations** как **Below** элемента **IdentityNumber**.

При проверке нахождения поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ** с помощью элемента **IdentityNumber** заметим, что на некоторых тестовых изображениях в сформированную гипотезу элемента **TextField** попадут также и цифры штрих-кода. Поэтому надо найти другой, более точный, способ описания местоположения поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ**.

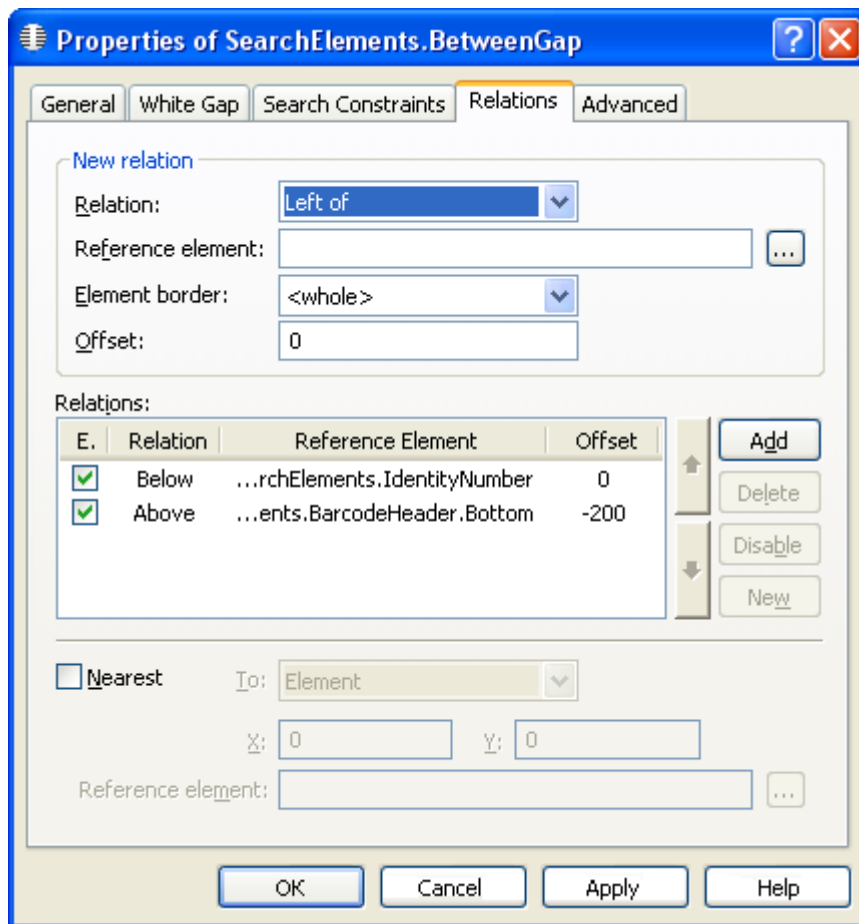
Создадим элемент типа **White Gap** с горизонтальной ориентацией, соответствующий просвету, находящемуся между штрих-кодом и произвольным текстом. Затем будем использовать его в качестве опорного для поиска поля **ПРОИЗВОЛЬНЫЙ ТЕКСТ**. Для этого в дереве элементов элемент **White Gap** обязательно нужно поместить выше элемента **TextField**!

1. Создайте элемент с именем **BetweenGap** типа **White gap**.

2. На закладке **White Gap** укажите ориентацию элемента **BetweenGap** – **Horizontal**.



3. На закладке **Relations** укажите область поиска элемента **BetweenGap** как находящуюся ниже элемента **IdentityNumber**, но на расстоянии не более 200 dot от нижней границы элемента **BarcodeHeader**.



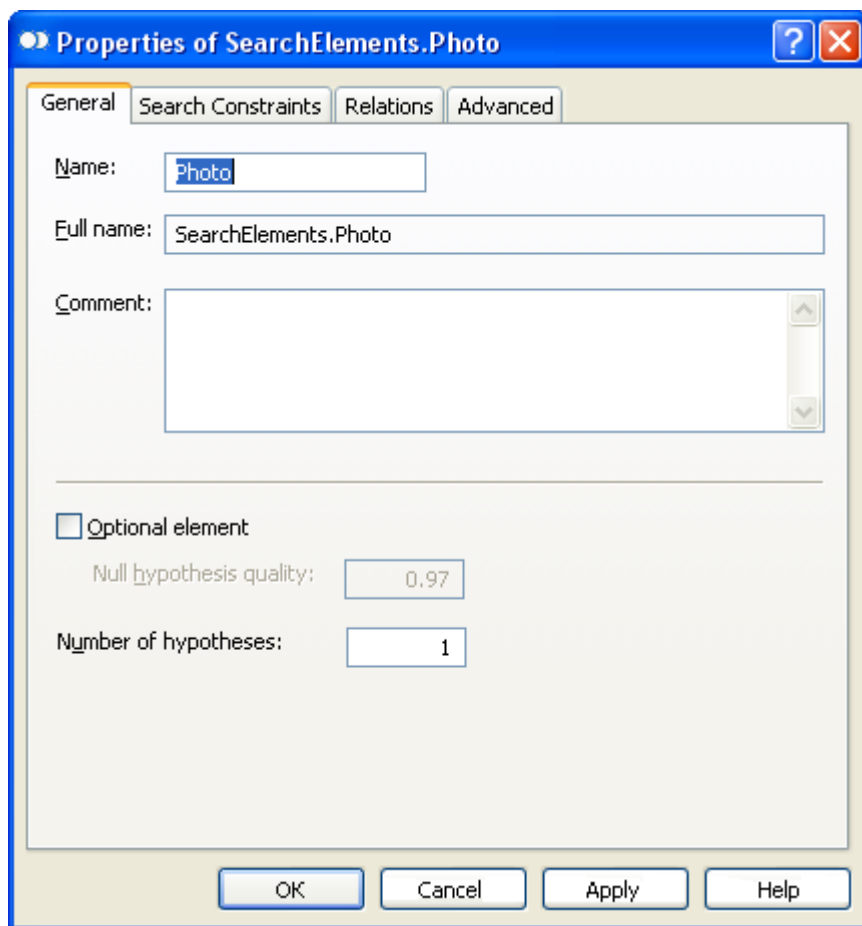
4. Создайте элемент с именем **TextField** типа **Text Fragment**.
5. Задайте положение элемента **TextField** на закладке **Relations** ниже (**Below**) элемента **BetweenGap**, Offset=0.

Наложив гибкое описание на все изображения пакета, заметим, что сформированная гипотеза элемента **TextField** не содержит лишних данных. Таким образом для элемента **TextField** будем использовать в качестве опорного элемент **BetweenGap**.

Для описания месторасположения блока, соответствующего полю **ПРОИЗВОЛЬНЫЙ ТЕКСТ**, создайте блок с именем **TextField** (тип **Text**), в качестве исходного элемента выберите элемент **TextField**.

Шаг 20. Описание поля YOUR PHOTO IN FANCY DRESS

Как уже отмечалось выше (Шаг 6), при просмотре результатов предраспознавания на изображениях 1 и 6 не все части фотографии распознаны как **Image Object** типа **Picture**. Для получения полной фотографии найдем все типы **Image Object**, расположенные в области фотографии. Для удобства описания всех частей фотографии и ограничения числа искомых гипотез объединим их в составной элемент и назовем его **Photo**.



Включим в элемент **Photo** (тип **Group**):

- o элемент **PhotoPicture** (тип **Object collection**), соответствующий фотографии,
- o элемент **PhotoHeader** (тип **Static text**), соответствующий заголовку **YOUR PHOTO IN FANCY DRESS**,
- o 4 разделителя вокруг фотографии, необходимые для ее поиска.

Поскольку текст заголовка (**YOUR PHOTO IN FANCY DRESS**) является уникальным в данном документе, соответствующий ему элемент мы добавили первым, а разделители будем искать уже относительно него.

Для создания элемента, соответствующего заголовку:

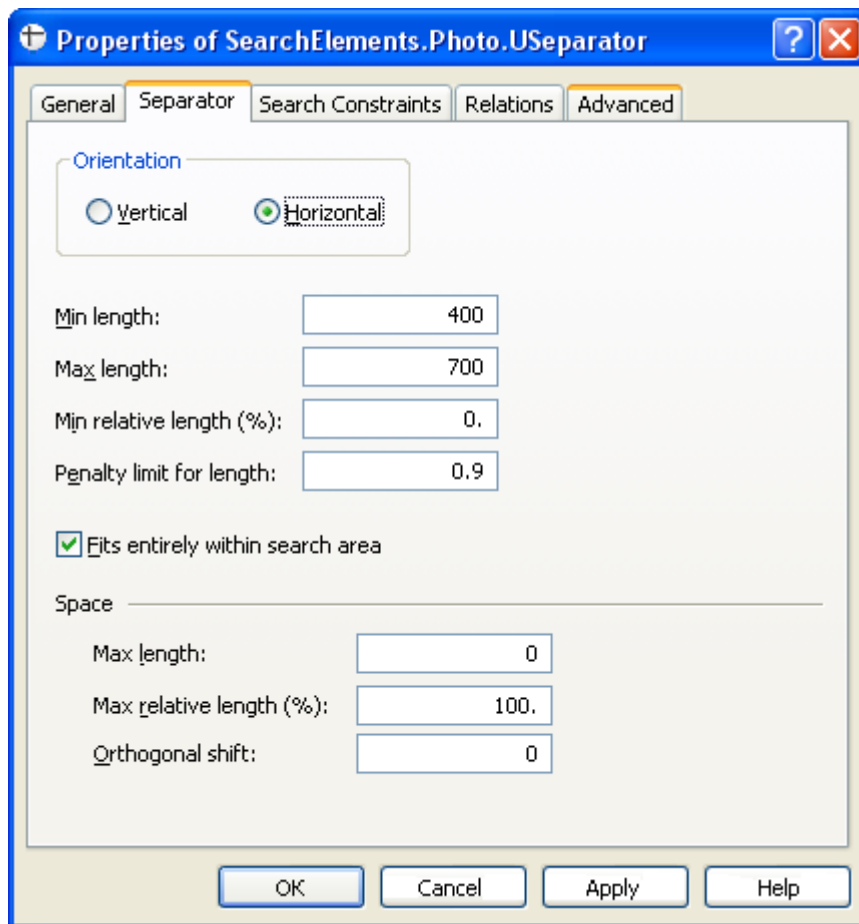
1. Создадим элемент с именем **PhotoHeader** типа **Static text**.
2. Свойства элемента задаются по аналогии с заголовками полей, рассмотренными ранее.

Проверка показывает устойчивое обнаружение заголовка фотографии на всех тестовых изображениях.

Создадим элементы типа **Separator** для четырех разделителей вокруг фотографии и назовем их:

- USeparator** – верхний разделитель,
- BSeparator** – нижний разделитель,
- LSeparator** – левый разделитель,
- RSeparator** – правый разделитель,

На закладке **Separator** (диалог **Properties**) для каждого из 4 разделителей необходимо выбрать соответствующую ему ориентацию (**Vertical** или **Horizontal**). Пока остальные свойства разделителей на данной закладке оставим без изменения.



Теперь, ориентируясь на тестовые изображения, зададим область поиска для каждого из разделителей.

Для верхнего разделителя (элемент **USeparator**):

ниже заголовка фото, но не дальше, чем на 100 dot от его верхней границы (**Below** элемента **PhotoHeader** и **Above** элемента **PhotoHeader**, **Offset** = -100).

внутри области, которая не превышает по ширине длину заголовка, более чем на 80 dot с каждой стороны (**LeftOf** правой границы региона элемента **PhotoHeader**, **Offset** = -80, **Element Border** = right и **RightOf** левой границы элемента **PhotoHeader**, **Offset** = -80, **Element Border** = left).

Для нижнего разделителя (элемент **BSeparator**):

ниже, чем заголовок поля с именем гостя (**Below** элемента **NameHeader**),

выше, чем нижняя граница штрих-кода (**Above** элемента **IdentityNumber**, **Element Border** = bottom),

внутри области, которая не превышает по ширине длину заголовка, более чем на 80 dot с каждой стороны (**LeftOf** правой границы элемента **PhotoHeader**, **Offset** = -80, **Element Border** = right и **RightOf** левой границы элемента **PhotoHeader**, **Offset** = -80, **Element Border** = left).

Для левого разделителя (элемент **LSeparator**):

внутри области, которая по высоте может быть больше, чем расстояние между верхним и нижним разделителями, но не более чем на 50 dot с каждой стороны (**Below** элемента **USeparator**, **Offset** = -50 и **Above** элемента **BSeparator**, **Offset** = -50),

разделитель находится в окрестности левой границы верхнего разделителя, которая не может быть более 40 dot (**RightOf** левой границы элемента **USeparator**, **Offset** = -20, **Element Border** = left и **LeftOf** левой границы элемента **USeparator**, **Offset** = -20, **Element Border** = left).

Для правого разделителя (элемент **RSeparator**):

внутри области, которая по высоте может быть больше, чем расстояние между верхним и нижним разделителями, но не более чем на 50 dot с каждой стороны (**Below** элемента **USeparator**, **Offset** = -50 и **Above** элемента **BSeparator**, **Offset** = -50),
 в окрестности правой границы верхнего разделителя, которая не может быть более 40 dot (**RightOf** правой границы элемента **USeparator**, **Offset** = -20, **Element Border** = right и **LeftOf** правой границы элемента **USeparator**, **Offset** = -20, **Element Border** = right).

Можно посмотреть, какими получились эти области поиска на данном изображении при наложении гибкого описания. Для этого в окне **Tree of Hypotheses** в локальном меню выделенной гипотезы выберите **Show search area**. Если условия были заданы с ошибкой, и область поиска получилась не там, где мы хотели, мы это сразу увидим. Или увидим, что даже для сформированной гипотезы область поиска стоит подкорректировать, например, она слишком мала и т.п.

Максимальный размер для всех разделителей определен опытным путем на основе изучения свойств объектов типа **Separator**, полученных в процессе предраспознавания.

Для того чтобы описать поле **YOUR PHOTO IN FANCY DRESS**:

1. Создадим элемент с именем **PhotoPicture** типа **Object collection**.
2. На закладке **Relations** зададим область поиска элемента **PhotoPicture**:
 ниже верхнего разделителя (**Below** элемента **USeparator**),
 выше нижнего разделителя (**Above** элемента **BSeparator**),
 правее левого разделителя (**RightOf** элемента **LSeparator**),
 левее правого разделителя (**LeftOf** элемента **RSeparator**).

Далее создадим блок с именем **PhotoPicture**, в качестве исходного элемента выберем элемент **PhotoPicture**.

Наложим гибкое описание на тестовые изображения. Результаты наложения показывают, что искомые поля обнаружены на всех изображениях.

Шаг 21. Экспорт гибкого описания

Перед тем, как подключить гибкое описание к шаблону в программе ABBYY FlexiCapture, его необходимо экспортировать в файл формата ABBYY FlexiLayout (*.afl).

Для того чтобы экспортировать гибкое описание:

1. Выберите команду **Export FlexiLayout...** в меню **File**.
2. В открывшемся диалоге **Export FlexiLayout** укажите имя файла.

Полученное таким образом гибкое описание FlexiLayout можно использовать в программе ABBYY FlexiCapture.

Шаг 22. Подключение гибкого описания к шаблону в программе ABBYY FlexiCapture

Экспортированное гибкое описание (файл формата ABBYY FlexiLayout) можно подключить к шаблону в программе ABBYY FlexiCapture. Для этого:

1. Откройте программу ABBYY FlexiCapture.
2. Создайте новый проект или откройте существующий.
3. Создайте новый шаблон документов или откройте существующий в Редакторе шаблонов.

 **Замечание.** Для того чтобы открыть существующий шаблон в Редакторе шаблонов откройте диалог **Document Templates** (меню **Project>Document Templates...**), в списке шаблонов выберите нужный, нажмите кнопку **Edit...**

4. Откройте диалог свойств раздела шаблона (Редактор шаблонов, меню **View>Properties...**).

5. Перейдите на закладку **Flexible Description**.
6. Нажмите на кнопку **Load...**
7. В открывшемся диалоге **Open** укажите файл гибкого описания (*.afl).

 **Замечание.** Блоки шаблона будут созданы автоматически, в соответствии с гибким описанием. Для того чтобы увидеть блоки на изображении, наложите шаблон (меню **Tools>Match Section**).

