

БИБЛИОТЕКИ ТЕГОВ (ТАГЛИБЫ)	3
МЕХАНИЗМ, АРХИТЕКТУРА РАБОТЫ ТЕГОВ НА ПРИМЕРЕ <RBL: TEXT/>	5
ОБЩЕПОТРЕБИТЕЛЬНЫЕ ТЕГИ	7
Label.....	7
Image	7
Submitlink.....	9
Button	11
Checkbox	12
Password.....	13
Text.....	14
Textarea	16
Drop-down.....	17
Radiogroup	19
Argument	20
Tabbar.....	21
orgStructureLabel (rbIDB).....	22
Masked_label	24
dd_label	25
DATEPICKER.....	26
filesizeLabel	28
webviewtabbar	29
htmleditor.....	30
uifieldfilter	30
uifieldintegerfilter	30
datagriduser.....	31
user	31
tree.....	31
Тег «дерево» TreeList	34
checkout.....	36
previewpopup.....	36
locale	37
form	37
divtag	37
popup.....	38
WEbviewheader	38
webviewfooter.....	38
formseparator	38
bookmark	39
facet	39
confirmdialog	40
htmlstart	40
htmlend	40
head	40
bodystart	40
bodyend	41
priorityimage.....	41

fileformatimage	41
helptooltip	42
calendar	42
fileupload	43
download	45
UI Фильтры (uifieldintegerfilter)	47
uifilterflobj	47
UifilterSuperUser	47
uigroupfilter	47
slideshow	47
АЯКС ТЕГИ	50
ajaxHelpTip	50
UserGroupSelection	52
Qeditor	54
orgStructureSelection	55
ГРИД ТЕГИ	57
<rbl:grid>	57
<rbl:gridcaption>	58
<rbl:griddata>	58
<rbl:gridheadrow>	59
<rbl:gridth>	59
<rbl:gridbodyrow>	60
<rbl:gridtd>	61
<rbl:gridcompletion>	61
<rbl:gridservice>	61
<rbl:gridpager>	62
5.5. ПРОВЕРОЧНЫЕ ТЕГИ	62
rbl:browsercheck	62
5.6. КОНЦЕПЦИЯ PPR	63
5.7. GRID В ВИДЕ ДЕРЕВА	63
5.7. СОЗДАНИЕ СОБСТВЕННОГО ТЕГА	64

БИБЛИОТЕКИ ТЕГОВ (ТАГЛИБЫ)

Тэг (Контроль) – элемент, несущий определенный функционал и используемый для построения web-форм. Каждый тег обладает набором атрибутов (могут быть обязательные и необязательные, например: name, size, value и другие) - входные параметры. За каждым атрибутом тега скрываются свойства, поведение, характеристики, имена методов формы, которые описаны в библиотеке тегов TLD, а конкретная функциональность этих элементов содержится в TagData. В проекте существует множество различных тегов (<rb1: text/>, <rb1: button/>, <rb1: submitlink/> и другие) (Рисунок 1):

ajaxHelpTipTag – отображение справочной информации.

Argument - Элемент данных, используемый для передачи нужного значения в метод вызываемого тега.

Button - элемент для выполнения какой-либо функции по нажатию.

Submitlink - внешнее отображение текста какого-либо элемента, с возможностью выполнять функцию по нажатию.

drop-down - тег отображающий данные в виде выпадающего списка.

Facet - Показывает закрывающуюся панель, с возможностью оформленную рамкой (fieldset'ом).

Checkout - предназначен для отображения той или иной картинки в зависимости от того кем выписан документ.

Fileformatimage – графическое изображение, отображающее формат файла, в зависимости от формата файла изображение изменяется.

Confirmdialog - информационное сообщение о подтверждении какого-либо действия («Вы уверены, что хотите удалить данный элемент»).

Fileupload - загрузка данных в систему.

htmlEditor - форма простого текстового редактора.

Popup - по нажатию правой кнопки мышки открывает информационное окно. В теге реализовано контекстное меню.

Textarea - область для ввода текста с заданным форматом.

Checkbox - флажок. Тег может представлять собой группу флажков с возможностью выбора одного или нескольких индикаторов.

Radiogroup - панель группы радио кнопок, содержит регулярно расположенные столбцами и строками радиокнопки.

Password - текстовое поле для ввода пароля пользователя.

Webviewheader и **webviewfooter** - используются для обрамления Webview.

Tab – неотъемлемая часть вебвью, отображает кнопки переключения между формами в вебвью.

Datericker – календарь

И другие.


Список rbl тегов	
Теги	Примеры
rbl:label	пример
rbl:image	
rbl:submitlink	пример
rbl:button	пример пример пример
rbl:checkbox	<input checked="" type="checkbox"/> <input type="checkbox"/>
rbl:password	<input type="password"/>
rbl:text	<input type="text"/>
rbl:textarea	<div style="border: 1px solid #ccc; padding: 5px; min-height: 40px;">пример</div>
rbl:drop-down	<div style="border: 1px solid #ccc; padding: 2px 5px; display: inline-block;">пример4 ▾</div>
rbl:radiobutton	<input type="radio"/> 1 <input type="radio"/> 2
rbl:argument	
rbl:tab	<div style="display: inline-block; border: 1px solid #ccc; padding: 2px 5px; margin-right: 5px;">активный таб</div> <div style="display: inline-block; border: 1px solid #ccc; padding: 2px 5px; margin-right: 5px;">неактивный таб</div>
rbl:ajaxcallWebTip	?
rbl:datericker	00:19 <input type="text"/>
rbl:ord-все-выходящие-в-него-теги	
rbl:group-rbl:label	пример
rbl:formseparator	
rbl:Content-Fileupload	File Upload
rbl:Content-Filedownload	123
rbl:nc:sidebshow	<div style="border: 1px solid #ccc; width: 80px; height: 60px; background-color: #f0f0f0; margin: 5px auto;"></div>

Рисунок 1. Пример внешнего вида тегов (тегов) проекта RBL.

МЕХАНИЗМ, АРХИТЕКТУРА РАБОТЫ ТЕГОВ НА ПРИМЕРЕ <RBL: TEXT/>

Например, тег <rbl: text/> (**Рисунок 13** – внешнее отображение; **Рисунок 12** – внутреннее представление в JSP): где **rbl:text** – имя тега, **name** – имя обязательного атрибута, **value** – имя атрибута.

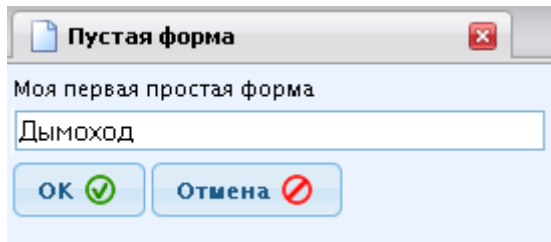


Рисунок 3

```
<rbl:text name="plusone" value="Значение"/>
```

Рисунок 2

В зависимости от состава атрибутов поведение и внешний вид тега может изменяться.

Для того чтобы пользователю отображался тот или иной тег он вносится разработчиком в JSP страницу какой-либо формы (текст в JSP представлен на **Рисунке 10**), для первичного наполнения формы используется массив TagData из которого, по заданному имени берется значение и отображается пользователю (см. **Рисунок 4**).

```
<rbl:form>
    <table border="0"><tr> <td>
        <rbl:label value="Моя первая простая форма"/>
    </td><td>
        <rbl:text name="plusone"/>
    </td><td>
        <rbl:text name="plustwo"/>
    </td><td>
        <rbl:text name="result" />
    </td></tr></table>
</rbl:form>
```

```
public class ExampleEmptyForm extends BaseForm{
    TextTagData s = (TextTagData)getControlData("plusone", TextTagData.class);
    TextTagData a = (TextTagData)getControlData("plustwo", TextTagData.class);
    TextTagData d = (TextTagData)getControlData("result", TextTagData.class);
    @Override
    s.setValue("Значение 1");
    a.setValue("Значение 2");
    d.setValue("Результат");
}
```

Массив всех значений контролов (TagData)

Рисунок 4. Механизм работы тега

Каждый тег несет свою функциональность. Например: Переход с формы login'a – после заполнения полей пользователь нажимает на кнопку (выполняется действие) и пользователю отображается другая форма.

Для перехода из одной формы в другую могут быть использованы теги `<tbl: button/>`, `<tbl: submitlink/>` и другие. При переходе из одной формы в другую пользователю необходимо, чтобы введенные ранее значения не терялись, здесь и используется механизм персистентности.

Персистентность (persistent) – изменения в web-форме не приводят к потере данных.

Сохранение состояния данных формы необходимо для того, чтобы у пользователя была возможность вернуться к ним в работе позже или же использовать их в другой форме. После ввода значений и нажатия на кнопку действия, пользователь не видит всей процедуры обработки этих значений и для него визуально страница не изменится, однако, введенные значения, отправляются на сервер для обработки и последующему представлению пользователю в заполненных полях.

Если бы механизм персистентности не использовался в web-программировании, то значения, указанные пользователям в полях тегов, при нажатии на кнопку действия не сохранялись, и пользователю было бы необходимо снова заполнять поля.

ОБЩЕУПОТРЕБИТЕЛЬНЫЕ ТЕГИ

LABEL

Тег label предназначен для вывода простого текста на страницу.

Label представлен в 2 tld файлах с префиксами rbl и rblDB.

Название	Обязательный	Описание
name	Нет	Используется для идентификации
value	Нет	Текст, выводимый на страницу
dateformat	Нет	Используется для вывода даты в определенном формате (или пустая строка, если используется формат по умолчанию)
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс
objectField	Нет	Имя поля объекта передаваемого в форму (присутствует только в rblDB)
field	Нет	Имя поля в гриде

В случае необходимости присваивания определенных CSS стилей текст на страницу будет выводиться в виде: `текст`. Формат даты может работать как в датагриде, так и при установке лебела, к примеру: `<rbl:label value="1236425448687" dateformat="" > </rbl:label>`.

В случае использования формата даты в гриде в value должен передаваться TimeStamp, в случае простого лебела - строковое значение long. При невозможности форматирования даты в system.out выдается ошибка RBLWEB.CONTROL.LABEL.DATA_FORMAT_ERROR.001 со значением value и форматом даты, в который пытались форматировать.

IMAGE

Тег предназначен для отображения изображений различных форматов, представлен в одном tld файле, имеющим префикс rbl.

Название	Обязательный	Описание
----------	--------------	----------

name	Нет	Используется для идентификации
src	Нет	Путь к картинке.Если usetheme=true указывается лишь название файла с расширением,поиск картинки будет осуществлен в папке img используемой темы
usetheme	Нет	Использовать тему или нет(true/false)по умолчанию true
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс(по умолчанию INPUT_SUBMITIMAGE_CLASS)
value	Нет	Отображаемый текст в случае неудачной загрузки картинки
valueField	Нет	Отображаемый текст в случае неудачной загрузки картинки(используется в гриде)
srcfield	Нет	Путь к картинке. Указывается лишь название файла с расширением. Поиск картинки будет осуществлен в папке img используемой темы(используется в гриде)
width	Нет	Ширина
height	Нет	высота
onclick	Нет	Метод вызываемый одиночным нажатием левой клавиши мыши
hotkey	Нет	Горячая клавиша, при нажатии на которую срабатывает метод Onclick
dialogId	Нет	Уникальный номер подтверждающего диалога (смотреть ConfirmDialogUiTag)

SUBMITLINK

Тег предназначен для отображения сабмит линков. Submitlink представлен в одном tld файле с префиксом rbl.

Название	Обязательный	Описание
name	Нет	Используется для идентификации
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс(по умолчанию INPUT_SUBMITLINK_CLASS)
field	Нет	Имя поля в гриде
onclick	Нет	Метод вызываемый одиночным нажатием левой клавиши мыши
value	Нет	Текст ссылки
clientScript	Нет	Скрипт запускаемый на клиенте. Не работает без onClient=true
onClient	Нет	На клиенте(true/false) по умолчанию false
span	Нет	Используется чтобы заблокировать возможность нажатия на ссылку (true/false) по умолчанию false
maxlen	Нет	Максимальная длина

Сабмит линк может использоваться в 3 режимах. В зависимости от режимов те или иные атрибуты тега становятся обязательными:

Серверный режим не в гриде: value и onclick;

Серверный режим в гриде: field и onclick ;

Клиентский режим: field, onClient, clientScript .

Пример КЛИЕНТСКОГО РЕЖИМА - скачивания файла - вне грида: Вот так работает сабмит линк на скачивание вне грида. В качестве дополнительного параметра можно передавать путь к форме, на которую надо вернуться, если будет ObjectNotFoundException; здесь это родительская форма - DownloadServlet.ERROR_STRING+"="+request.getQueryString() . Таким образом, если произойдет ошибка и указан путь, произойдет переход на форму - в данном случае она

просто обновится. Если не указан путь, то откроется пустое белое окно. Если же ошибок не будет, то файл будет передан браузеру.

```
<%  
String objectId2 = "WSP100000000000SB0000000000004";  
String fileName2 = "скачать существующий файл с обработкой ошибки,  
objectId="+objectId2;  
%>  
<rb:submitlink name="stest5" value="<%=fileName2%>" onClient="true"  
clientScript="<%=("view(this,false,""+fileName2+"",""+webApp  
+ "/download?" + DownloadServlet.OBJECT_ID + "=" + objectId2 + "&" + DownloadServlet.ERROR  
_STRING + "=" + request.getQueryString() + ",") %>" />
```

Пример одновременно СЕРВЕРНОГО РЕЖИМА - навигации, и КЛИЕНТСКОГО РЕЖИМА - скачивания файла - в гриде:

```
<rb:submitlink span = "false" name="sb2" field="<%= (TreeExplorer.FILENAME_FIELD) %>"  
onClient="true"  
clientScript="<%= ("view(this,$"+TreeExplorer.ISFOLDER_FIELD+"$, '$"+TreeExplorer.FILE  
NAME_FIELD+"$', ""+webApp  
+ "/download?" + DownloadServlet.OBJECT_ID + "=" + TreeExplorer.OBJECTID_FIELD + "$', ""+  
TreeExplorer.ONCLICKGOINTO_FUNCTION + ")") %>" >  
    <rb:argument name="arg3" argname="<%= (TreeExplorer.ID_PARAM_KEY) %>"  
field="<%= (TreeExplorer.NODEID_FIELD) %>" />  
    <rb:argument name="arg4" argname="<%= (TreeExplorer.NAME_PARAM_KEY) %>"  
field="<%= (TreeExplorer.FILENAME_FIELD) %>" />  
</rb:submitlink>
```

BUTTON

Тег предназначен для отображения кнопок. Button представлен в одном tld файле с префиксом rbl.

Название	Обязательный	Описание
name	Да	Используется для идентификации
value	Да	Надпись на кнопке
title	Нет	всплывающее описание
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
onclick	Нет	Метод вызываемый одиночным нажатием левой клавиши мыши
hotkey	Нет	горячая клавиша
type	Нет	Тип кнопки ok: ok button; cancel: cancel button; custom: button with rounded corners; specific: simple button; disabled: disabled button;
onClient	Нет	Вместо серверного метода будет вызываться js функция true/false по умолчанию false
outerStyleClass	Нет	Внешний стилевой класс

ЧЕКБВОХ

Тег предназначен для отображения чекбокса. Checkbox представлен в 2 tld файлах с префиксами rbl и rblDB

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс (по умолчанию INPUT_CHECKBOX_CLASS)
disabled	Нет	Блокировка(true/false)по умолчанию false
checked	Нет	Отмечена(true/false)по умолчанию false
objectField	Нет	Имя поля объекта передаваемого в форму (присутствует только в rblDB)
onclick	Нет	Метод вызываемый одиночным нажатием левой клавиши мыши

PASSWORD

Тег предназначен для отображения стандартного html тега password. Password представлен в одном tld файле с префиксом rbl

Атрибуты

Название	Обязательный	Описание
name	Нет	Используется для идентификации
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
value	Да	Введенный пароль
isrequiredvalue	Нет	Нужно ли пользователю ввести значение(true/false)по умолчанию false
visible	Нет	Видимость(true/false)по умолчанию true
enabled	Нет	Доступность для редактирования(true/false)по умолчанию true
size	Нет	Кол-во символов допустимое для ввода

ТЕХТ

Тег предназначен для вывода на экран текстового поля. Тег представлен в двух tld файлах с префиксами rbl и rblDB.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
value	Нет	Значение по умолчанию
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс
isRequiredvalue	Нет	Нужно ли пользователю ввести значение(true/false)по умолчанию false
visible	Нет	Видимость(true/false)по умолчанию true
enabled	Нет	Доступность для редактирования(true/false)по умолчанию true
size	Нет	Кол-во символов допустимое для ввода
disabled_render_as_label	Нет	Рендер как лейбл(true/false)по умолчанию false
id	Нет	Уникальный идентификатор
objectField	Нет	Имя поля объекта передаваемого в форму (присутствует только в rblDB)
fieldDomain	Нет	STRING (по умолчанию, если есть атрибут objectField); INTEGER; NUMBER; ID Используется при наличие атрибута objectField, что бы знать

		домен поля и отобразить данные в соответствующем виде (присутствует только в rblDB)
type	Нет	Вид: с круглыми краями(true);Обычный(false)

TEXTAREA

Поле <TEXTAREA> представляет собой элемент формы для создания области, в которую можно вводить несколько строк текста. В отличие от тега <button> в текстовом поле допустимо делать переносы строк, они сохраняются при отправке данных на сервер. Textarea представлен в одном tld файле с префиксом RBL.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
value	Нет	Текст по умолчанию
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс (по умолчанию TEXTAREA_CLASS)
cols	Нет	Кол-во столбцов
rows	Нет	Кол-во строк

DROP-DOWN

Тег <drop-down> позволяет создать элемент интерфейса в виде раскрывающегося списка, а также список с одним или множественным выбором. Ширина списка определяется самым широким текстом, указанным в теге <option>, а также может изменяться с помощью стилей. Можно создать группу пунктов с помощью тега <optgroup>, все пункты группы должны находится в контейнере <optgroup>. Каждый пункт создается с помощью тега <option>, который должен быть вложен в контейнер <drop-down>.

Атрибуты drop-down

Название	Обязательный	Описание
name	Да	Используется для идентификации самого OPTION, в случае, если в один выпадающий список будет произведена попытка записать несколько OPTION с одинаковым именем, предыдущие OPTION с таким же именем будут удалены
size	Нет	Кол-во видимых строк дропдаун листа (по умолчанию 1)
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс
disabled	Нет	Блокировка доступа и изменения элементов списка (по умолчанию false)
multiple	Нет	Возможность множественного выбора (по умолчанию false)
complex	Нет	Jquery menu (по умолчанию false)
onclick	Нет	Метод, который срабатывает при открытии drop-down
onClient	Нет	Values: all; onclick, onchange ... (если больше одного названия атрибута, то через пробел) По значению атрибута определяем, какие из методов выполняем на клиенте (т.е. будет вызываться не серверный метод с сабмитом, а какая-то js функция) 'all' - все на клиенте 'onclick' - только onclick 'onclick onchange' - onclick и onchange (в данном случае равносильно all, но вдруг новые методы добавятся)
onchange	Нет	Метод вызываемый при выборе одной из опций. Если multiple = true, то onchange игнорируется, что бы была возможность выбора нескольких элементов списка

Атрибуты optgroup

Название	Обязательный	Описание
----------	--------------	----------

name	Да	Используется для идентификации
label	Да	Заголовок группы опции
disabled	Нет	Блокировка доступа и изменения элементов группы (по умолчанию false)
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс

Атрибуты option

Название	Обязательный	Описание
name	Да	Используется для идентификации
label	Да	Заголовок опции
value	Нет	Используется в формах для передачи некоей потаенной информации
selected	Нет	Выделенная опция
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс
disabled	Нет	Блокирует использование опции.

RADIOGROUP

Тег является блоком для переключателей radiobutton. Используются для выбора одного из предложенных вариантов. Radiogroup представлен в одном tld файле с префиксом rbl.

Атрибуты radiogroup

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
position	Нет	Положение radiobuttons; Вертикально, если true. Горизонтально, если false. (по умолчанию false)
onclick	Нет	Метод вызываемый одиночным нажатием левой клавиши мыши

Атрибуты radiobutton

Название	Обязательный	Описание
value	Да	Значение radiobutton, передаваемое при сабмите формы
disabled	Нет	Блокировка доступа (по умолчанию false)
caption	Нет	Текст рядом с radiobutton. Если атрибут не указан текст будет value
checked	Нет	Отметка (true/false) по умолчанию false

ARGUMENT

Тег предназначен для вывода на страницу скрытого поля. На странице никак не отображается. Представлен в одном tld файле с префиксом gbl.

Название	Обязательный	Описание
name	Нет	Используется для идентификации
value	Нет	Значение, передаваемое при сабмите формы
argname	Нет	Имя параметра которое передается в хэш мэп параметров
field	Нет	Имя поля в гриде
renderHiddenControl	Нет	Отрисовка контрола (true/false) по умолчанию false

TABBAR

Тег предназначен для отрисовки табов, используется служебными функциями фреймворка, использование его разработчиками не предусматривается.

Каждый элемент меню добавляется при помощи тега <tab>, который должен находиться в контейнере <tabbar>. Тег представлен в одном tld файле с префиксом rbl.

Атрибуты Tab

Название	Обязательный	Описание
name	Да	Используются для идентификации
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс

Атрибуты tab

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс (по умолчанию TAB_ACTIVE_CLASS или TAB_INACTIVE_CLASS)
onTop	Нет	Активный таб (true/false) по умолчанию false
text	Да	Текст в табе

ORGSTRUCTURELABEL (RBLDB)

Тег служит для отображения локализованного бредкрамба по ID элемента оргструктуры(descriptorId) ...

Атрибуты userId, userDomain используются только когда необходимо вывести бредкрамб подразделения пользователя: в userId кидается id для пользователя, в descriptorId - его подразделения (пользователь должен сам их определить)

Тег может работать в трех вариантах:

В тег непосредственно кидаются необходимые ID.

Тег находится в гриде, и ID является каким-либо полем резултсета (название поля заключается в символы «\$»)

Тег используется в форме-карточке (когда в форму приходит ID объекта, следовательно по ID поднимается соответствующий IObject) – ID является каким-либо атрибутом объекта формы (название атрибута заключается в символы «#»)

Атрибуты orgStructureLabel

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
descriptorId	Нет	Используется совместно с атрибутом descriptorDomain. Один из ID, по которому можно определить элемент оргструктуры (ID группы или ID дескриптора), для которого отображается бредкрамб. Для использования в гриде поле типа нужно заключить в символ \$, например \$GRUOP_ID\$ Для извлечения из IObject формы (когда передается id), поле типа нужно заключать в символ #, например #RECIPIENT_ROLE_ID#
descriptorDomain	Нет	Тип ID элемента оргструктуры: либо сам ID дескриптора элемента, либо ID группы, связанной с элементом. Значения: ORGST_ELEMENT_ID, ORGST_GROUP_ID По-умолчанию: ORGST_GROUP_ID Пользуйтесь скриплетами с константами OrgStructureLabelTagData.GROUP_ID_DOMAIN,

		.ORGSTR_ID_DOMAIN
userId	Нет	<p>Требует атрибуты descriptorId, descriptorDomain. Используется совместно с атрибутом userDomain.</p> <p>Один из ID, по которому можно определить пользователя оргструктуры (ID пользователя, ID персональной группы или ID дескриптора), для которого отображается бредкрамб.</p> <p>Для использования в гриде атрибут типа нужно заключить в символ \$, например \$USER_ID\$</p> <p>Для извлечения из IObject формы (когда передается id), поле типа нужно заключать в символ #, например #RECIPIENT_ID#.</p>
userDomain	Нет	<p>Тип ID пользователя оргструктуры: либо сам ID пользователя, либо ID дескриптора пользователя, либо ID персональной группы</p> <p>Значения: ORGST_ELEMENT_ID, ORGST_GROUP_ID (и для ID пользователя). По-умолчанию: ORGST_GROUP_ID</p> <p>Пользуйтесь скриплетами с константами OrgStructureLabelTagData.GROUP_ID_DOMAIN, .ORGSTR_ID_DOMAIN, .USER_ID_DOMAIN</p>
size	Нет	<p>Длина выводимого бредкрамба (без учета пользователя): всё, что не влезло, отсекается и заменяется ... (полный бредкрамб виден в тултипе по наведению)</p>
visible	Нет	<p>Если false, то элемент не отображается. По-умолчанию true.</p>
onClick	Нет	<p>Серверная функция, срабатывающая при нажатии на отображаемое значение.</p>

Пример:

```
<rbldb:orgStructureLabel name="orgstrl_to"
descriptorId="<%= ("#+IDDAttrConstants.VFS_INCOMING_RECIPIENT_ROLE_ID+"#")
```

```
%>" descriptorDomain="<%= OrgStructureLabelTagData.GROUP_ID_DOMAIN %>"
userId="<%= ("#+IDDAAttrConstants.VFS_INCOMING_RECIPIENT_ID+"#") %>"
userDomain="<%= OrgStructureLabelTagData.GROUP_ID_DOMAIN %>" size="100"/>
```

Файлы:

[OrgStructureLabelTag.java](#) - класс-рендер

[OrgStructureLabelTagData.java](#) - модель

Серверная часть OrgStructureLabelTagData:

Стандартное получение контрола:

```
private OrgStructureLabelTagData orgstrl_to = (OrgStructureLabelTagData)
getControlData("orgstrl_to", OrgStructureLabelTagData.class);
```

Важные методы:

Одноименные атрибутам get/set... методы.

MASKED_LABEL

Выдает «маскированные значения». Например, нам нужно, чтобы отображалось вместо 1- Да, вместо 0 – нет. Тег представлен в одном tld файле с префиксом rbl.

Пример использования (показаны открывающие и закрывающие теги в гриде):

```
<rbl:gridTd>
```

```
<rbl:masked_label source_values="1;0" masked_values="MSG_ONLINE;MSG_OFFLINE"
field="<%=IDDAAttrConstants.ATTR_SCHEDULER_INFO_STATUS%>" />
```

```
</rbl:gridTd>
```

Вместо значений 1;0 будут показаны локализованные значения MSG_ONLINE;MSG_OFFLINE.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
source_values	Да	Значения в базе данных, вместо которых будут выводиться значения атрибута masked_values (разделяются “;”).
masked_values	Да	Значения, которые будут выводиться на экран вместо значений из базы данных указанных в атрибуте source_values (также

		разделяются “;”).
field	Нет	Имя поля в резултсете.
name	Нет	Используется для идентификации

DD_LABEL

Тег предназначен для отображения локализованного имени типа или атрибута. Для изменения локализованного названия типа необходимо зайти в справочник типов и внизу внести локализованные значения. Тег представлен в одном tld файле с префиксом gbl.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
value	Нет	Тип объекта
attr	Нет	Атрибут объекта
field	Нет	Имя поля в резултсете
Name	Нет	Используется для идентификации

DATEPICKER

Тег представлен в 2 tld файлах и может иметь префиксы: rbl и rblDB.

Тег служит для отображения, а так же выбора даты. При значении атрибута `showTime=true` добавляется возможность выбора времени.

Атрибуты `datepicker`

Название	Обязательный	Описание
<code>name</code>	Да	Используется для идентификации
<code>showTime</code>	Нет	Если указано <code>true</code> , то появится поле выборки времени (<code>timepicker</code>).
<code>readonly</code>	Нет	Тег становится неактивным - нельзя установить ни дату, ни время .
<code>onDateChange</code>	Нет	Серверный метод, срабатывающий при выборе даты.
<code>dateFormat</code>	Нет	Формат, в котором тег отображает дату. По-умолчанию отображается в соответствии с локалью пользователя в системе. * <code>d</code> - day of month (no leading zero) * <code>dd</code> - day of month(two digit) * <code>D</code> - day name short * <code>DD</code> - day name long * <code>m</code> - month of year (no leading zero) * <code>mm</code> - month of year (two digit) * <code>M</code> - month name short * <code>MM</code> - month name long * <code>y</code> - year (two digit) * <code>yy</code> - year (four digit) * <code>'...'</code> - literal text * <code>'</code> - single quote.
<code>locale</code>	Нет	Принудительное выставление локали (соответств. смена названий кнопок, формата даты и пр.). По-умолчанию совпадает с локалью пользователя в системе.
<code>animation</code>	Нет	Устанавливает эффект появления календарика <code>"fadeIn""slideDown""blind"</code> <code>"clip""drop""fold""puff""scale""slide"</code>
<code>size</code>	Нет	Размер текстового поля для даты
<code>objectField</code>	Нет	Название поля домена <code>DATE</code> в объекте <code>IOBJECT</code> формы. Значение данного поля будет отображено тегом

Пример:

```
<rbl:datepicker name="dp_repeated_start_date" dateFormat="D dd.mm.yy" size="15" />
```

Файлы:

[DatepickerTag.java](#) - класс-рендер

[DatepickerTagData.java](#) - модель

Серверная часть `DatepickerTagData`:

Стандартное получение контроля:

```
private DatePickerTagData dp_expired = (DatePickerTagData)  
getControlData("dp_expired", DatePickerTagData.class);
```

Важные методы:

```
dp_expired.getDate(); - получение даты из контроля.  
dp_expired.setDate(date); - установка даты.
```

FILESIZELABEL

Тег предназначен для вывода информации о размере файла.

Название	Обязательный	Описание
value	Нет	Значение, выводимое на страницу
field	Нет	Имя поля в гриде
objectField	Нет	Имя поля объекта передаваемого в форму
name	Нет	Используется для идентификации
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс

WEBVIEWTABBAR

Тег используется в веб вью для создания меню. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс

HTMLEDITOR

Тег предназначен для написания и редактирования текста. Представлен в одном tld файле с префиксом rbl.

Название	Обязательный	Описание
name	Да	Имя тэга
enabled	Нет	Доступность редактора для редактирования.
styleClass	Нет	Имя CSS-класса для окружающего дива.
style	Нет	Инлайн CSS-стили для окружающего дива.
value	Нет	Содержимое области редактирования.
toolbarSet	Нет	Название набора контролов для данного редактора. Определенные заранее названия: Default, Basic, default_rbl_toolbar_set(по-умолчанию). Если вы хотите определить свой набор контролов - это можно сделать в файле rbl_fck_editor_config.js.

UIFIELDFILTER

Тег представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Имя тэга
filter	Нет	Фильтр полей резалтсета

UIFIELDINTEGERFILTER

Тег должен применяться в гриде: сравнивается операцией operation целочисленное значение поля field резалтсета с указанным числом в атрибуте value. Если результат true, то боди отрисовывается, иначе нет. Тег представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Имя тэга
value	Нет	Значение, которое необходимо сравнить с полем резалтсета

field	Нет	Поле результата
operation	Нет	Логическая операция

DATAGRIDUSER

Тег предназначен для отображения фото и имени пользователя. Используется только в гриде. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Нет	Используется для идентификации
userId	Нет	Id пользователя или имя поля результата
onclick	Нет	Метод, вызываемый при клике на линк
styleClass	Нет	Стилевые CSS классы
style	Нет	Инлайн CSS стили

USER

Тег предназначен для отображения фото и имени пользователя. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Имя тэга
userId	Нет	Уникальный номер пользователя
onclick	Нет	Метод, вызываемый одиночным нажатием левой клавиши мыши
style	Нет	Инлайн CSS-стили
styleClass	Нет	Стилевой CSS-класс

TREE

Тег предназначен для отрисовки дерева на главной странице системы. Включает в себя тег <rbl:treeNode>. Тег представлен в одном tld файле с префиксом rbl.

Атрибуты <rbl:tree>

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
hierId	Нет	Идентификатор иерархии, которую отображает дерево
withFiles	Нет	Прятать файлы(true/false)по умолчанию false

Атрибуты <rbl:treeNode>

Название	Обязательный	Описание
name	Да	Используется для идентификации
attrName	Нет	use attrName with attrValue. если атрибут текущей ноды равен введенному, нода рендерится
attrValue	Нет	use attrName with attrValue. если атрибут текущей ноды равен введенному, нода рендерится
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
labelField	Нет	Имя атрибута ноды иерархии, которая будет отображаться в дереве в качестве имени файла или папки
onClickLabel	Нет	Метод, вызываемый при клике на имя ноды (только если labelField!=null) (может быть заменено submitLink)
withIcon	Нет	Отображать иконку файла или папки (true/false) по умолчанию false
onClickIcon	Нет	Метод, вызываемый при клике на картинку (только если withIcon=true) (может быть заменено submitLink)
withPlusMinus	Нет	Отображать +/- (true/false) по умолчанию false

onExpandCollapse	Нет	Метод, вызываемый при клике на +/- (только если withPlusMinus=true)
complexView	Нет	Показывать количество детей после ноды(true/false)по умолчанию false

ТЕГ «ДЕРЕВО» TREELIST

Общее описание:

Тег дерево служит для отображения элементов в виде дерева.

Тег имеет возможность динамической «подкачки» элементов через ajax при нажатии на (+) (-).

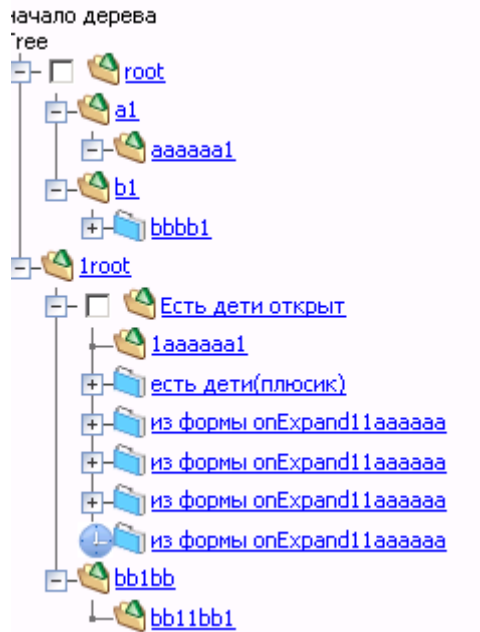


Рисунок 5 Внешний вид дерева

Пример использования:

```
<rbl:treelist name="myTree" onexpand="onExpandTree" ajax="true"
oncollapse="onCollapseTree" collapsedIcon='big_closed_folder.png'%>
expandedIcon='small_opened_folder.png'%>
leafIcon='small_opened_folder.png'%> onclick="onClick">
</rbl:treelist>
```

В этом примере будет создаваться дерево, при нажатии на элемент которого вызывается функция формы

```
public void onClick(Object controlData, HashMap<String, Object> params,
HttpServletRequest request)
```

при нажатии на иконку (+) будет вызываться функция

```
public void onExpandTree(Object controlData, HashMap<String, Object>
params, HttpServletRequest request)
```

при нажатии на иконку (-) будет вызываться функция

```
public void onCollapseTree(Object controlData, HashMap<String, Object>
params, HttpServletRequest request)
```

флаг `ajax="true"` показывает, что функции `onCollapseTree` и `onExpandTree` будут вызываться без перезагрузки формы с использованием технологии AJAX

Архитектура:

Каждый узел дерева состоит из класса `TreeNode`. Основными составляющими этого класса являются:

- Состояние узла (открыт, закрыт, имеет детей или нет), выраженное через `TREENODE_STATE`
{`HAS_CHILDREN_OPEN`,`HAS_CHILDREN_CLOSED`,`NO_CHILDREN`,`MAYBE_HAS_CHILDREN_CLOSED`}
- Иконка (необязательно)
- Наличие чекбокса и его состояние (помечен, нет)
- Лебел узла (отображаемый линк href, который показывается справа от иконки)
- Идентификатор узла

Пример создания элемента узла:

```
TreeNode f= new TreeNode("root", "root", TREENODE_STATE.HAS_CHILDREN_OPEN);
tree.addRootNode(f);
f.setHasCheckBox(true);
```

После того, как узел создан он добавляется в дерево либо путем добавления в корневые узлы, либо в дочерние элементы (TreeNode):

```
TreeListTagData tree = (TreeListTagData) getControlData("myTree",
TreeListTagData.class);
```

```
tree.addRootNode(f);
```

или

```
r1.addChildNode(new TreeNode(
"1aaaaaa1", "1aaaaaa1", TREENODE_STATE.NO_CHILDREN ) );
```

Ограничения и примечания:

1. В ajax режиме не поддерживается сохранение поставленной в чекбоксе галочки при экспанде
2. Для режима ajax в сервере приложений должен быть поставлен сервлет

```
<servlet>
    <servlet-name>TreeHelperServlet</servlet-name>
    <servlet-
class>org.corolla.rbl.web.dispatchers.servlets.ajax.controls.treelist.TreeLis
tHelperServlet</servlet-class>
    <load-on-startup>1</load-on-startup>
</servlet>

<servlet-mapping>
<servlet-name>TreeHelperServlet</servlet-name>
<url-pattern>/tree</url-pattern>
</servlet-mapping>
```

3. Пример использования находится в `form_descriptor=tree_demo`

CHECKOUT

Тег предназначен для отображения той или иной картинки в зависимости от того кем выписан документ. Тег должен получать либо status(вариант: через \$столбец\$ value) либо who (вариант: через \$столбец\$ whoField). Также может получать whenString (вариант: через \$столбец\$ whenField). Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
status	Нет	true, если выписано вами.false, если выписано не вами.null, если никем не выписано. По умолчанию null.
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс(по умолчанию INPUT_SUBMITIMAGE_CLASS)
value	Нет	Если в гриде, то это имя столбца. Значение должно иметь вид \$имя_столбца\$
whoField	Нет	Если в гриде, то это имя столбца с ID выписавшего пользователя. Значение должно иметь вид \$имя_столбца\$
who	Нет	Если не в гриде: ID выписавшего пользователя whenString Нет Если не в гриде: когда было выписано
whenField	Нет	Если в гриде, то это имя столбца, где указано, когда было выписано. Значение должно иметь вид \$имя_столбца\$

PREVIEWPOPUP

Тег предназначен для предпросмотра картинок. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Нет	Используется для идентификации

icon	Нет	Заполнить
thumbnail	Нет	Заполнить
thumbnailField	Нет	Заполнить
content	Нет	Заполнить
contentField	Нет	Заполнить

LOCALE

Тег предназначен для вывода флага текущей локали. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс

FORM

Тег – контейнер. Все теги системы должны находится в форме. Аналог html тега <form>. Тег представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс

DIVTAG

Тег контейнер. Аналог html тега <div>. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
onclick	Да	Метод, вызываемый при клике на элемент
style	Нет	Инлайн css-стили

styleClass	Нет	Стилевой CSS-класс
------------	-----	--------------------

POPUP

Тег предназначен для отображения попапа. Представлен в одном tld файде с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Нет	Используется для идентификации
src	Нет	URL Servlet, jsp или html, к-ый будет загружен в попап если ajax=true, То сервлет contextMenu , параметры: objectId =fileId, onContext=onContextFunction , можно добавить menuType=files пример использования см. commonExplorer
ajax	Нет	JS-запрос(true/false)по умолчанию false
width	Нет	Ширина
height	Нет	Высота
styleClass	Нет	Стилевой CSS-класс(по умолчанию eg_context_menu)

WEBVIEWHEADER

Тег – контейнер. Используется в веб вью для отрисовки элементов, заключенных в этот контейнер, перед формами, включенными в веб вью. Представлен в одном tld файле с префиксом rbl. Атрибутов нет.

WEBVIEWFOOTER

Тег – контейнер. Используется в веб вью для отрисовки элементов, заключенных в этот контейнер, после форм, включенных в веб вью. Представлен в одном tld файле с префиксом rbl. Атрибутов нет.

FORMSEPARATOR

Тег предназначен для отделения одной формы от другой. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
label	Нет	Заголовок

visible	Нет	Видимость(true/false)по умолчанию true
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс(по умолчанию GRID_CAPTION_CLASS)

BOOKMARK

Тег предназначен для создания закладок в браузере. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Нет	Используется для идентификации
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
bookmark	Нет	Строка, заносимая в закладку браузера
value	Нет	То, что будет отображаться в тексте пользователя
url	Нет	Адрес, который будет добавлен в закладку
addcontext	Нет	Необходимо ли добавлять путь контекста приложения (http://server...) к ссылке(true/false)по умолчанию true

FACET

Тег предназначен для скрытия или открытия элементов внутри данного тега. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
visible	Нет	Видимость(true/false)по умолчанию true

label	Нет	Текст ссылки скрывающей контент
minimizedlabel	Нет	Текст ссылки со скрытым контентом
isrenderFieldSet	Нет	Вывод fieldset(true/false)по умолчанию true
minimized	Нет	Свернуто(true/false)по умолчанию false

CONFIRMDIALOG

Тег предназначен для отображения диалога подтверждения с вариантами ответа. Представлен в одном tld файле с префиксом gbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Имя тэга
messageText	Да	Сообщение
yesText	Да	Ответ на кнопке да
noText	Да	Ответ на кнопке нет

HTMLSTART

Тег – контейнер, заключает в себе все содержимое веб-страницы. Аналог html тега <html>, но также включает в себя определение типа документа (Document Type Definition, DTD). Представлен в одном tld файле с префиксом gbl. Атрибутов нет. Открывающий тег.

HTMLEND

Тег – контейнер, заключает в себе все содержимое веб-страницы. Аналог html тега <html>. Представлен в одном tld файле с префиксом gbl. Атрибутов нет. Закрывающий тег.

HEAD

Тег – контейнер. Предназначен для подключения файлов с css стилями. Представлен в одном tld файле с префиксом gbl.

Атрибуты

Название	Обязательный	Описание
Title	Нет	Определяет заголовок вкладки

BODYSTART

Тег – контейнер. Аналог html тега <body>. Здесь подключаются js скрипты системы. Представлен в одном tld файле с префиксом gbl. Открывающий тег.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс

BODYEND

Тег – контейнер. Аналог html тега <body>. Представлен в одном tld файле с префиксом rbl. Открывающий тег.

Атрибуты

Название	Обязательный	Описание
bottomFade	Нет	Отрисовка полупрозрачного градиента внизу страницы (true/false) по умолчанию false

PRIORITYIMAGE

Тег предназначен для отображения степени важности, отрисовывается в виде флажков разного цвета. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс
value	Нет	Если содеожимое заключено в \$, то это имя поля грида
type	Нет	Расширение файлов, содержащих изображение: ico, png, gif ...
width	Нет	Ширина
height	Нет	Высота

FILEFORMATIMAGE

Тег показывает тип ноды (разные типы файла или вообще папка). Если есть onclick, то вызывает функцию в форме. Иначе будет вызывать карточку, то есть command

id="properties" из конфига контекстного меню. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css-стили
styleClass	Нет	Стилевой CSS-класс(по умолчанию INPUT_SUBMITIMAGE_CLASS)
type	Нет	Расширение файла картинки(по умолчанию png)
value	Нет	Лучше заполнять value для contentType, иначе будет браться объект по объектИд, и это будет медленно в гриде
width	Нет	Ширина
height	Нет	Высота
onclick	Нет	Метод, вызываемый одиночным нажатием левой клавиши мыши
objectId	Нет	Указывает Id документа, для прыжка в контекстное меню
dialogId	Нет	Уникальный номер подтверждающего диалога(смотреть ConfirmDialogUiTag)

HELPTOOLTIP

Нигде не используется. Представлен в одном tld файле с префиксом rbl.

CALENDAR

Тег предназначен для отображения календаря. Представлен в одном tld файле с префиксом rbl.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
displayType	Нет	Варианты: day, week, month. (по умолчанию day)

FILEUPLOAD

Тег предназначен для асинхронной загрузки файлов на сервер. Представлен в одном tld файле с префиксом rblContent.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
onComplete	Да	Имя серверной функции, которая будет вызвана по удачной загрузке файла на контент-сервер.
targetURI	Нет	URI, на который будет загружаться файл.
defaultType	Нет	Тип по умолчанию для аплоада.
failureCallback	Нет	js-функция, которая будет вызвана, если от сервера пришел ответ о том, что загрузка прошла неуспешно.
progressRefreshDelay	Нет	Период обновления информации о прогрессе загрузки файла.
style	Нет	Стиль для ссылки
styleClass	Нет	Стилевой класс для ссылки
cssBundle	Нет	Объект стилей для всплывающей формы в виде: {mainContr:'eg_upl_mainContr',bgContr:'eg_upl_bgContr',mainPane:'eg_upl_mainPane',formPane:'eg_upl_formPane',ctrlContr:'eg_upl_ctrlContr',ctrlBgContr:'eg_upl_ctrlBgContr',ctrlBtnsContr:'eg_upl_ctrlBtnsContr',form:'eg_upl_form',table:'eg_upl_table',pBar:'eg_upl_pb'}
value	Нет	Будет содержимым тэга A, заменяя любую определенное внутри тэга разметку.

fadingTimeFactor	Нет	Определяет, с какой скоростью будет производиться фейдинг формы при открытии и закрытии.
successHandler	Нет	Выполняется перед вызовом successCallback над самим объектом аплоада. С помощью этого метода можно определить поведение формы аплоада после загрузки (к примеру, заставив ее скраться). Пример: <pre>function(thisUploader){ thisUploader.uploader.progressBar.setProgress(0); thisUploader.uploader.fInput.value = ''; thisUploader.uploader.fNameInput.value = ''; thisUploader.setStatus(''); thisUploader.hide(); }</pre>
failureHandler	Нет	Выполняется перед вызовом failureCallback над самим объектом аплоада. С помощью этого метода можно определить поведение формы аплоада после загрузки (к примеру, заставив ее скраться).
pbResizeProperties	Нет	Настройки плавности хода прогресс-бара в формате [Число шагов при изменении прогресса, задержка между шагами]. Пример: [20,20].
backendPollDelay	Нет	Период опроса сервера о статусе загрузки файла в бэкэнд.

DOWNLOAD

Тег предназначен для выбора формата файла при даунлоаде. Возможные типы конверсий файла и его имя получаются ajax - запросом сервлета. Представлен в одном tld файле с префиксом rblContent.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
targetURI	Нет	URI, на который будет загружаться файл.
objectId	Нет	Идентификатор объекта для скачивания.
startHideDelay	Нет	Время через которое форма автоматически будет скрыта после нажатия на кнопку на старт.
styleClass	Нет	Имя стилевого css класса для тэга А.
style	Нет	Инлайн css стили для тэга А.
cssBundle	Нет	Объект стилей для всплывающей формы в виде: { mainContr:'eg_dwld_bf_mainContr',bgContr:'eg_dwld_bf_bgContr',mainPane:'eg_dwld_bf_mainPane',formPane:'eg_dwld_bf_formPane',ctrlContr:'eg_dwld_bf_ctrlContr',ctrlBgContr:'eg_dwld_bf_ctrlBgContr',ctrlBtnsContr:'eg_dwld_bf_ctrlBtnsContr',table:'eg_dwld_table',loader:'eg_dwld_loader'}
id	Нет	HTML id для элемента А.
value	Нет	Содержимое тэга А, заменяет любую

		определенную внутри тэга разметку.
fadeSpeed	Нет	Определяет, с какой скоростью будет производиться фейдинг формы при открытии и закрытии.

UI ФИЛЬТРЫ (UIFIELDINTEGERFILTER)

Тег предназначен для фильтрации показа содержащихся в нем элементов (например, если значение поля 1, то не показывать то, что находится внутри тега

```
<rb:uifieldintegerfilter name="ddd" field=""></rb:uifieldintegerfilter>
```

operation – операция (< или > или <>)

value – значение, с которым будут сравнивать

UIFILTERFLOBJ

Тег предназначен для фильтрации показа содержимого тега, в случае отсутствия прав доступа на FObject.

visibleIfNotrights – не показывает содержимое внутри тега, если нет прав на FObject.

UIFILTERSUPERUSER

Тег предназначен для фильтрации показа содержимого тега, в случае если пользователь супервизор.

name – имя тега.

UIGROUPFILTER

Тег предназначен для фильтрации отображения содержимого тега в зависимости от пользовательской группы

groupName – группа, к которой должен принадлежать пользователь, чтобы отрисовалось содержимое.

SLIDESHOW

Тег предназначен для отображения попапа-слайдшоу. Представлен в лдном tld файле с префиксом rbImg.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
targetURI	Нет	URI, к которому будут производиться обращения за изображениями.
slideshowId	Нет	Идентификатор слайдшоу для бэкэнда.
thumbId	Нет	Идентификатор маленького изображения для бэкэнда. Если не определено - будет использован первый слайд.

controlBarOpacity	Нет	Прозрачность бара с кнопками.
pollDelay	Нет	Период опроса, готово ли очередное изображение.
slideCount	Нет	Количество слайдов в этом слайдшоу. Если параметр не указан, тэг произведет запрос к бэкэнду для определения количества слайдов.
slideNum	Нет	Слайд, скоторого надо начинать показ.
hideTimeout	Нет	Задержка при закрытии слайдшоу.
ctrlBarFading_Steps	Нет	Количество шагов прозрачности, за которое производится показ кнопок управления.
ctrlBarFading_TimeForStep	Нет	Время между изменениями прозрачности для кнопок управления.
imgFading_Steps	Нет	Количество шагов прозрачности, за которое производится показ изображения.
imgFading_TimeForStep	Нет	Время между изменениями прозрачности для изображения.
resizeSteps	Нет	Количество шагов, за которое происходит трансформация слайда при различных размерах изображений.
initialDimention	Нет	Начальный размер прямоугольника при первом открытии слайдшоу. В формате "длина, высота".
slideshowCssClases	Нет	Классы для объекта слайдшоу. В формате "'eg_ss_mainContr','eg_ss_bgContr','eg_ss_mainPane','eg_ss_imgPane','eg_ss

		<code>_mainCtrlContr\ ,\ 'eg_ss_ctrlContr\ ,\ 'eg_s s_ctrlBgContr\ ,\ 'eg_ss_ctrlBtnsContr\ ,\ 'eg _ss_loaderSmall\ ,\ 'eg_ss_loaderBig\ ''</code> .
<code>imgMaxDimention</code>	Нет	Максимальный размер, который может принять слайд. Слайд всегда помещается в размер экрана, даже если максимальный размер изображения не достигнут. В формате "длина, высота".
<code>imgOffsets</code>	Нет	Отступы от изображения до краев главного прямоугольника. В формате "длина, высота".
<code>styleClass</code>	Нет	Стилевой css класс для тамбнейла.
<code>style</code>	Нет	Инлайн css стиль для тамбнейла.

АЯКС ТЕГИ

АJAXHELPTIP

Тег `ajaxHelpTip` представляет собой, так называемый тег помощник. На веб-странице размещается элемент, по наведению или нажатию на который всплывает веб-страница, которая подтягивается при помощи ajax технологии (смотри в разделе javascript). Тег представлен в одном tld файле с префиксом `gblajax`.

Атрибуты

Название	Обязательный	Описание
<code>name</code>	Да	Используется для идентификации
<code>helpContextId</code>	Нет	Страница, которая будет показана (например, "1.html") К примеру, если <code>helpContextId</code> = "123.html" и локаль соответствует <code>ru_RU</code> , то подсказка будет загружена из файла <code>/help/ru_RU/123.html</code> . <code>helpContextId</code> можно брать из модели формы или определять на <code>jsp</code> .
<code>id</code>	Нет	Html id
<code>trigger</code>	Нет	DOM-событие, открывающее подсказку. По умолчанию используется "hover". Возможно также использовать "click".
<code>value</code>	Нет	Заголовок помощи
<code>padding</code>	Нет	
<code>width</code>	Нет	
<code>spikeGirth</code>	Нет	
<code>fill</code>	Нет	
<code>strokeWidth</code>	Нет	
<code>strokeStyle</code>	Нет	
<code>timeout</code>	Нет	
<code>interval</code>	Нет	
<code>cornerRadius</code>	Нет	
<code>cssStyles</code>	Нет	

spikeLength	Her	
-------------	-----	--

USERGROUPSELECTION

Тег предназначен для отображения диалога выбора пользователей и (или) групп пользователей. UserGroupSelection представлен в одном tld файле с префиксом rblajax.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
onOkDialog	Да	Серверный метод, срабатывающий при нажатии на кнопку Ok диалогового окна
onCancelDialog	Да	Серверный метод, срабатывающий при нажатии на кнопку Cancel диалогового окна
onAddUserOrGroup	Нет	Серверный метод добавления пользователя/группы в список выбаранных (должен принимать один параметр String) Заменяет метод контроля public void addDefaultUserOrGroup(String id)
onGetUsersAndGroups	Нет	Серверный метод, который возвращает список пользователей и групп (должен принимать один параметр String) Заменяет метод контроля public ArrayList==String==getDefaultUsersAndGroups(String param)
onFilterUsersAndGroups	Нет	Серверный метод, который возвращает список пользователей и групп при фильтрации (должен принимать один параметр String) Заменяет метод контроля public ArrayList==String==getDefaultFilteredUsersAndGroups(String filterValue)
mode	Нет	Режим отображения - только пользователи (users_only), только группы (groups_only), по-умолчанию

		все (all)
dialogTitle	Нет	Заголовок диалогового окна
controlTitle	Нет	Заголовок контрола, на который навешано диалоговое окно
width	Нет	Ширина диалогового окна
height	Нет	Высота диалогового окна
size	Нет	Количество отображаемых элементов в списке
multiselect	Нет	Возможность выбора сразу нескольких объектов true/false (по умолчанию false)
visible	Нет	Видимость true/false (по умолчанию true)

QEDITOR

Тег предназначен для быстрого редактирования. Qeditor представлен в одном tld файле с префиксом gblajax.

Атрибуты

Название	Обязательный	Описание
name	Нет	Используется для идентификации
value	Нет	При добавлении этого атрибута выдает ошибку
title	Нет	Заголовок диалога
ongetcontent	Нет	Серверный метод, который срабатывает при нажатии на кнопку сохранить (в tld так)
onsavecontent	Нет	Серверный метод, который срабатывает при нажатии на кнопку сохранить
enabled	Нет	Возможность редактирования true/false (по умолчанию true)
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс

`Popuptreeselector` неизвестно нужен ли он вообще. Используется только в `popuptreedemo`

`Tormenu` кроме `treeexplorer` нигде не используется, да и там закоменчен

ORGSTRUCTURESELECTION

Тег представляет собой всплывающее диалоговое окно выбора элемента(-ов) из оргструктуры(оргструктур) системы с кнопками подтверждения и отмены. В теге для предоставления данных используется технология ajax, поэтому сабмит формы происходит только при нажатии диалоговых кнопок подтверждения или отмены.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
dialogTitle	Нет	Заголовок диалогового окна
controlTitle	Нет	Заголовок ссылки, по нажатию на к-ую вызывается диалоговое окно
label	Нет	Заголовок к дереву оргструктуры
singleselect	Нет	Если true, то можно выбрать только один элемент. По-умолчанию false.
enabled	Нет	Если false, то ссылка на диалоговое окно будет не видна. По-умолчанию true.
onOkDialog	Нет	Серверная функция, к-ая будет срабатывать по нажатию кнопки подтверждения.
selectableTypes	Нет	Допустимые типы оргструктуры для выбора, указывать через запятую: возможны ORGST_UNIT_DESCR, ORGST_ROLE_DESCR, ORGST_USER_DESCR. По-умолчанию можно выбрать всех.

Пример:

```
<rbajax:orgStructureSelection name="orgss_to"
```

```
selectableTypes="<%= IDDDTypeConstants.ORGST_USER %>" singleselect="true"
onOkDialog="onClickSelectToPerson"></rblajax:orgStructureSelection>
```

Файлы:

OrgStructureSelectionTag.java - класс-рендер

OrgStructureSelectionTagData.java - модель

/org/corolla/taghelpers/orgstructuretreedialog/orgstr_selection_dialog_init.j

sp - jsp, инициализирует непосредственно сам диалог выбора через ajax механизм - отрисовка необходимых html элементов, javascript функций и пр. для начала работы тега.

/org/corolla/taghelpers/orgstructuretreedialog/orgstr_selection_helper.jsp - jsp, передает начальные данные и обрабатывает действие пользователя в диалоговом окне через ajax механизм - инициализация левой и правой частей диалога, сворачивание/разворачивание узла дерева, удаление, фильтрация)

Серверная часть OrgStructureSelectionTagData:

Стандартное получение контроля:

```
private OrgStructureSelectionTagData orgss_to =
(OrgStructureSelectionTagData) getControlData("orgss_to",
OrgStructureSelectionTagData.class);
```

Важные методы:

orgss_to.getSelectedOrgStrElements() - возвращает список выбранных элементов (бинов) List<OrgStructElement> (может быть пуст, но не null). Через методы бина уже можно получить конкретную информацию по элементу - id элемента, id группы, связанной с элементов, имя, тип и прочее...

Пример использования функции, вызываемой по возвращению из тега (имя функции пишется в параметрах тега)

```
public void onSelectUser(Object controlData, HashMap<String, Object> params,
HttpServletRequest request) {
    OrgStructureSelectionTagData os1 =
(OrgStructureSelectionTagData) controlData;
    List<OrgStrElementBean> res = os1.getSelectedOrgStrElements();
    if (res == null || res.isEmpty()) {
        return;
    }
    OrgStrElementBean elem = res.get(0);
    actorId = elem.getGroupId();
    actorOrgstLabel.setDescriptorId(actorId);
}
```


ГРИД ТЕГИ

Таблица (грид) является формой представления данных. Описание всех тегов грида находится в одном tld файле с префиксом rbl. Общая структура грида имеет следующий вид:

```
<rbl:grid>
  <rbl:gridCaption></rbl:gridCaption>

  <rbl:gridData>
    <rbl:gridHeadRow>
      <rbl:gridTh></rbl:gridTh>
    </rbl:gridHeadRow>
    <rbl:gridBodyRow>
      <rbl:gridTd></rbl:gridTd>
    </rbl:gridBodyRow>
  </rbl:gridData>
  <rbl:gridCompletion>
    <rbl:gridService/>
    <rbl:gridPager/>
  </rbl:gridCompletion>
</rbl:grid>
```

<RBL:GRID>

Тег - контейнер. Все теги, относящиеся к гриду, должны находиться внутри него. Грид отрисовывается только при наличии данных.

Атрибуты

Название	Обязательный	Описание
name	Да	Используется для идентификации
type	Нет	true - без обрезания содержимого ячеек, false - с обрезанием содержимого ячеек, ширина колонок должна быть определена (по умолчанию)

style	Нет	Инлайн css стили
itemsperpage	Нет	Количество строк на одной странице
pager	Нет	Непонятно
ajaxpaging	Нет	По умолчанию false непонятно
styleClass	Нет	Стилевые css классы
popupURL	Нет	УРЛ для поппа в гриде
query	Нет	Значением данного атрибута должен являться запросом к FL серверу, результат данного запроса будет помещён в грид

<RBL:GRIDCAPTION>

Тег представляет собой заголовок грида. В атрибуте label указывается текст, который будет выведен в заголовке.

Атрибуты

Название	Обязательный	Описание
label	Нет	Текст, отображаемый в заголовке грида
styleClass	Нет	Стилевые css классы(по умолчанию GRID_CAPTION_CLASS)
style	Нет	Инлайн css стили

<RBL:GRIDDATA>

Тег служит контейнером для элементов, определяющих содержимое грида. Внутри <rbl:gridData> допустимо использовать следующие теги: <rbl:gridHeadrow>, <rbl:gridTh>, <rbl:gridBodyRow>, <rbl:gridTd>. Аналог html тега <table>.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
evenStyle	Нет	Инлайн css стили для четной

		строки грида
oddStyle	Нет	Инлайн css стили для нечетной строки грида
styleClass	Нет	Стилевой css класс (по умолчанию GRID_DATA_CLASS)
tableStyle	Нет	Инлайн css стили для таблицы в гриде
tableStyleClass	Нет	Стилевой css класс для таблицы в гриде(по умолчанию GRID_DATA_TABLE_CLASS)

<RBL:GRIDHEADROW>

Тег предназначен для хранения одной или нескольких строк, которые представлены вверху таблицы. Допустимо использовать не более одного элемента <rbl:gridHeadRow> в пределах одного грида и он должен идти в исходном коде сразу после тега <rbl:grid>. Аналог html тега <thead>

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс

<RBL:GRIDTH>

Тег <rbl:gridTh> предназначен для создания одной ячейки грида, которая обозначается как заголовочная. Тег должен размещаться внутри контейнера <rbl:gridHeadRow>. Аналог html тега <th>.

Атрибуты

Название	Обязательный	Описание
sort	Нет	Задумывалось, как первоначальная сортировка резалтсета. Но сейчас не функционально - тк резалтсет эффективнее сортировать соответствующим методом

		мануального резалтсета, или предикатом обычного запроса
filter	Нет	Если в поле будет фильтрация (указаны className и field), то можно указать здесь дефальтное значение фильтрации. В этом случае при нажатии на кнопку фильтрации в поле ввода будет это значение. Фильтрация пока работает только для строковых полей - String
field	Нет	Имя поля резалтсета для сортировки и фильтрации
className	Нет	Необходим для включения фильтрации по строкам типа String
width	Нет	ширина столбца
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
name	Нет	Используется для идентификации

<RBL:GRIDBODYROW>

Тег предназначен для хранения одной или нескольких строк грида. Допустимо использовать не более одного элемента <rbl:gridBodyRow> в пределах одного грида. Аналог html тега <tbody>.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс

<RBL:GRIDTD>

Тег <rbl:gridtd> Предназначен для создания одной ячейки грида. Тег должен размещаться внутри контейнера “rbl:gridBodyRow”. Аналог html тега <td>.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс
width	Нет	ширина ячейки
align	Нет	Выравнивание в ячейке(по умолчанию left)
popupURL	Нет	УРЛ для поппа

<RBL:GRIDCOMPLETION>

Тег – контейнер. Внутри <rbl:gridCompletion> допустимо использовать следующие теги: <rbl:GridService> и <rbl:gridPager>.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс (по умолчанию GRID_COMPLETION_CLASS)

<RBL:GRIDSERVICE>

Тег предназначен для добавления элементов поиска и обновления грида. (Насколько я понял, уже нигде не используется).

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс (по умолчанию GRID_SERVICE_CLASS)

<RBL:GRIDPAGER>

Тег предназначен для осуществления навигации в гриде. Состоит из нескольких элементов:

- кнопка перехода на первую страницу грида
- кнопка перехода на предыдущую страницу грида
- текстовое поле, отображающее текущий номер страницы грида
- лейбл, отображающий общее количество страниц грида
- кнопка перехода на следующую страницу грида
- кнопка перехода на последнюю страницу грида
- количество элементов отображаемых на одной странице грида

Тег <rbl:gridPager> отображается только в случае наличия нескольких страниц грида.

Атрибуты

Название	Обязательный	Описание
style	Нет	Инлайн css стили
styleClass	Нет	Стилевой css класс (по умолчанию GRID_PAGER_CLASS)

5.5. ПРОВЕРОЧНЫЕ ТЕГИ

RBL:BROWSERCHECK

Тег предназначен для проверки соответствия браузера пользователя требованиям системы. При несоответствии требованиям выдается сообщение о текущем типе и версии браузера пользователя, и указываются рекомендации к выбору браузера.

Атрибутов нет.

5.6. КОНЦЕПЦИЯ PPR

Partial page rendering – частичная прорисовка страницы.

Часто возникает необходимость в частичной (а не полной) перерисовки страницы, для этого предусмотрен специальный механизм PPR.

Основа его использования составляет в разделении jsp страницы на разделяемые элементы, отмечаемые специальным тегом `ajax_not_render`.

Все, находящееся внутри тега `ajax_not_render` не будет прорисовываться при запросе `ajax`.

Такое поведение элемента осуществляется за счет включения в область видимости страницы специальной переменной:

```
if  
(getPageContext().getAttribute(IWebConstants.REQUEST_AJAX_NOT_RENDER_FLAG,  
PageContext.REQUEST_SCOPE) != null) return SKIP_BODY;
```

5.7. GRID В ВИДЕ ДЕРЕВА

Суть "деревянного" грида в том, что некоторые элементы (связанные документы, маршрут согласования, резолюции) удобнее показывать именно в дереве с таблицей.

Архитектурно грид выглядит в виде основного грида на форме и дополнительной формы, содержащей грид, из которой берутся "подуровни через аякс запрос".

По событию раскрытия узла дерева вызывается ф-ция формы, в которую параметром передается значение аргумента (поля грида).

Есть возможность указать автоматическое раскрытие нескольких уровней дерева при первой отрисовке формы.

При загрузке показывается значок "часики", показывающий, что идет загрузка элемента дерева.

Ограничения: на форме может быть только один такой грид (и он не может быть на вью). ВСЕ ЭЛЕМЕНТЫ (ТЕГИ) ВНУТРИ ГРИДА ДОЛЖНЫ ИМЕТЬ ИМЯ.

Демонстрационная форма `tree_grid_demo`.

Tree Structure	ID	Value
[-] Первая строка	1234	5678
★ [-] ._Arg1Первая строка	1234__Arg1	5678
★ ★ [-] ._Arg1Arg1Первая строка	1234__Arg1Arg1	5678
★ ★ [-] ._Arg1Arg1Вторая строка	2345__Arg1Arg1	67890
★ ★ [-] ._Arg1Arg1Третья строка	3456__Arg1Arg1	78901
★ [-] ._Arg1Вторая строка	2345__Arg1	67890
★ ★ [-] ._Arg1Arg2Первая строка	1234__Arg1Arg2	5678
★ ★ [-] ._Arg1Arg2Вторая строка	2345__Arg1Arg2	67890
★ ★ [-] ._Arg1Arg2Третья строка	3456__Arg1Arg2	78901
★ [-] ._Arg1Третья строка	3456__Arg1	78901
[-] Вторая строка	2345	67890
[-] ._Arg2Первая строка	1234__Arg2	5678
★ ★ [-] ._Arg2Arg1Первая строка	1234__Arg2Arg1	5678
★ ★ [-] ._Arg2Arg1Вторая строка	2345__Arg2Arg1	67890
★ ★ [-] ._Arg2Arg1Третья строка	3456__Arg2Arg1	78901
★ [-] ._Arg2Вторая строка	2345__Arg2	67890
★ ★ [-] ._Arg2Arg2Первая строка	1234__Arg2Arg2	5678
★ ★ [-] ._Arg2Arg2Вторая строка	2345__Arg2Arg2	67890
★ ★ [-] ._Arg2Arg2Третья строка	3456__Arg2Arg2	78901
★ [-] ._Arg2Третья строка	3456__Arg2	78901
[-] Третья строка	3456	78901

5.7. СОЗДАНИЕ СОБСТВЕННОГО ТЕГА

Собственные теги позволяют отделить бизнес-логику от логики отображения. Для создания простого тега необходимо создать 2 файла с расширениями tld и java. В tld описываем тег, даем ему название, перечисляем атрибуты (обязательные и необязательные). Рассмотрим код файла tld простейшего тега:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<taglib xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee/web-
jsptaglibrary 2.0.xsd" version="2.0">
```

```
<tlib-version>1.0</tlib-version>
```

```
<short-name>rbl</short-name>
```

```
<uri>\WEB-INF\tlds\rbl_1_0.tld</uri>
```

```
<tag>
```

```
<name>label</name>
```

```
<tag-class>
```

```
org.corolla.rbl.web.taglib.tags.html.body.basic.label.LabelTag
```



```

</tag-class>

<body-content>jsp</body-content>

<attribute>

    <name>name</name>

    <required>>true</required>

    <type>java.lang.String</type>

    <description>HTML имя тега</description>

</attribute>

<attribute>

    <name>style</name>

    <required>>false</required>

    <type>java.lang.String</type>

    <description>инлайн css стили</description>

</attribute>

</tag>

```

Прежде всего в tld файле указываем версию xml и кодировку.

```
<?xml version="1.0" encoding="UTF-8"??>
```

Затем следует идентификатор того, что тег создается для спецификации JSP (и TLD) определенной версии.

```

<taglib xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee/web-
jsptaglibrary\_2\_0.xsd" version="2.0">

```

Далее указываем версию библиотеки тегов, короткое название и URI.

```

<tlib-version>1.0</tlib-version>

<short-name>rbl</short-name>

<uri>\WEB-INF\tlds\rbl_1_0.tld</uri>

```

Теперь рассмотрим непосредственно сам тег

```

<tag> начало тега

    <name>label<name> название тега

    <tag-class> класс - обработчик

    org.corolla.rbl.web.taglib.tags.html.body.basic.label.LabelTag

</tag-class>

```

```

<body-content>jsp</body-content>
<attribute> начало атрибута
    <name>name</name> имя атрибута
    <required>true</required> обязательность употребления
    <type>java.lang.String</type> тип данных вводимых значений
    <description>HTML имя тега</description> описание атрибута
</attribute> конец атрибута
<attribute>
    <name>style</name>
    <required>false</required>
    <type>java.lang.String</type>
    <description>инлайн css стили</description>
</attribute>
</tag> конец тега

```

Далее рассмотрим код класса обработчика. Здесь описывается логика функционирования нашего тега.

```

public class LabelTag extends BaseControlTag {
    public String value = "";
    public String getValue() {
        return value;
    }
    public void setValue(String value) {
        this.value = value;
    }
    public String name = "";
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}

```

```

    }

    public int doStartTag() throws JspException {

        super.doStartTag();

        return EVAL_BODY_INCLUDE;

    }

    public int doEndTag() throws JspException {

    try {

        if (value==null) value="";

        if (!value.trim().equals("") && data.getVisible()) {

if (getStyleString()==null || getStyleString().equalsIgnoreCase("")) {
getOut().write(getL10nMessage(value));

        }else {

getOut().write("<span name="+name+" + getStyleString() + " >");
getOut().write(getL10nMessage(value));
getOut().write("</span>");

        }

        }

        return EVAL_PAGE;

    } catch (IOException e) {

        e.printStackTrace();

        throw new JspTagException(e);

    }

}

}

```

Все классы обработки тегов должны расширять базовый класс `BaseControlTag` и иметь как 2 обязательных метода, а именно: `doStartTag()` и `doEndTag()`. Так же необходимо сделать методы `get` и `set` для всех атрибутов, описанных в `tld` файле.

