Инструкция по реализации контекстных отчетов JasperReport  
на платформе SBF

В платформе SBF реализован механизм создания контекстных отчетов JasperReport. JasperReports - механизм отчетности с открытым исходным кодом. Он полностью написан на Java и способен использовать данные, поступающие из любого источника данных, и создавать документы с точностью до пикселей, которые можно просматривать, печатать или экспортировать в различных форматах документов, включая HTML, PDF, Excel, OpenOffice и Word.

Развертывание, настройка и подключение приложения

Чтобы начать использовать JasperReports в приложении на базе SBF, необходимо скачать два приложения: JaspersoftStudio и JasperServer.

JasperServer <https://community.jaspersoft.com/project/jasperreports-server/releases> — это серверное приложение, куда из JaspersoftStudio деплоятся и хранятся файлы шаблонов. К ним можно получить доступ посредством REST API. В JasperServer есть UI, с его помощью можно просматривать отчеты, создавать учетные записи для разных пользователей и давать им соответствующие доступы.

С помощью сетапа устанавливаем приложение на сервере, доступном для сервера приложения.

Web консоль приложения доступна по адресу http://<IP сервера>:<порт установки (8080)> /jasperserver/. Логин и пароль по умолчанию jasperadmin/ jasperadmin.

На сервере необходимо установить коннект к базе данных приложения. Для этого слева в дереве выбираем DataSources, по правой кнопке мыши выбираем AddResource→ DataSource, в предложенном окне заполняем параметры jdbc-коннекта к БД, жмем тест, сохраняем коннект с именем DataSource.

JaspersoftStudio <https://community.jaspersoft.com/project/jaspersoft-studio/releases> — среда разработки на основе Eclipse со встроенной Java-библиотекой JasperReports, где разрабатываются шаблоны отчетов и публикуются на сервере JasperReport.

С помощью сетапа устанавливаем JaspersoftStudio на локальной машине разработчика отчетов. После установки настраиваем подключение к JasperServer и к БД приложения. Выбираем вкладку Repository Explorer, на элементе Servers жмем правою кнопку Create JasperServer connection, заполняем параметры подключения к JasperServer, нажимаем тест, сохраняем. Выбираем элемент Data Adapters, по правой кнопке мыши выбираем Create Data Adapter, выбираем Database JDBC Connection, заполняем параметры подключения к БД приложения, нажимаем тест, сохраняем.

Настройка приложения

На сервере приложения необходимо настроить параметры подключения к JasperServer:

<system-properties>

<property name="ru.sbsoft.jasperserver.url" value="http://<ip server>:<port>/jasperserver/"/>

<property name="ru.sbsoft.jasperserver.user" value="<login>"/>

<property name="ru.sbsoft.jasperserver.password" value="<password>"/>

</system-properties>

Контекстный отчет привязывается к гриду приложения. Доступные отчеты показываются на инструментальной панели браузера и в меню.   
Возможны варианты, когда при различных условиях выбора пользователем, необходимо отображать различный набор отчетов. Для этого реализован механизм задания набора фильтров и задания значений фильтров.

Для задания набора фильтров для каждого грида в интерфейсе IMetaTemplateInfo опубликован метод default List<CustomReportFilterInfo> getReportFilters() – по умолчанию возвращает null.

Для задания дополнительных фильтров отчетов в темплейте грида необходимо переопределить данный метод, который возвращает набор CustomReportFilterInfo. CustomReportFilterInfo содержит следующую информацию:

* String code – уникальный код фильтра в пределах набора,
* ILocalizedString description – описание назначения фильтра, отображается в настройках отчета в приложении (описано ниже в разделе создания отчета).

На клиентской части в реализации браузера грида необходимо задавать значение фильтров методом Browser.setReportFilters(List<StringParamInfo> reportFilters).

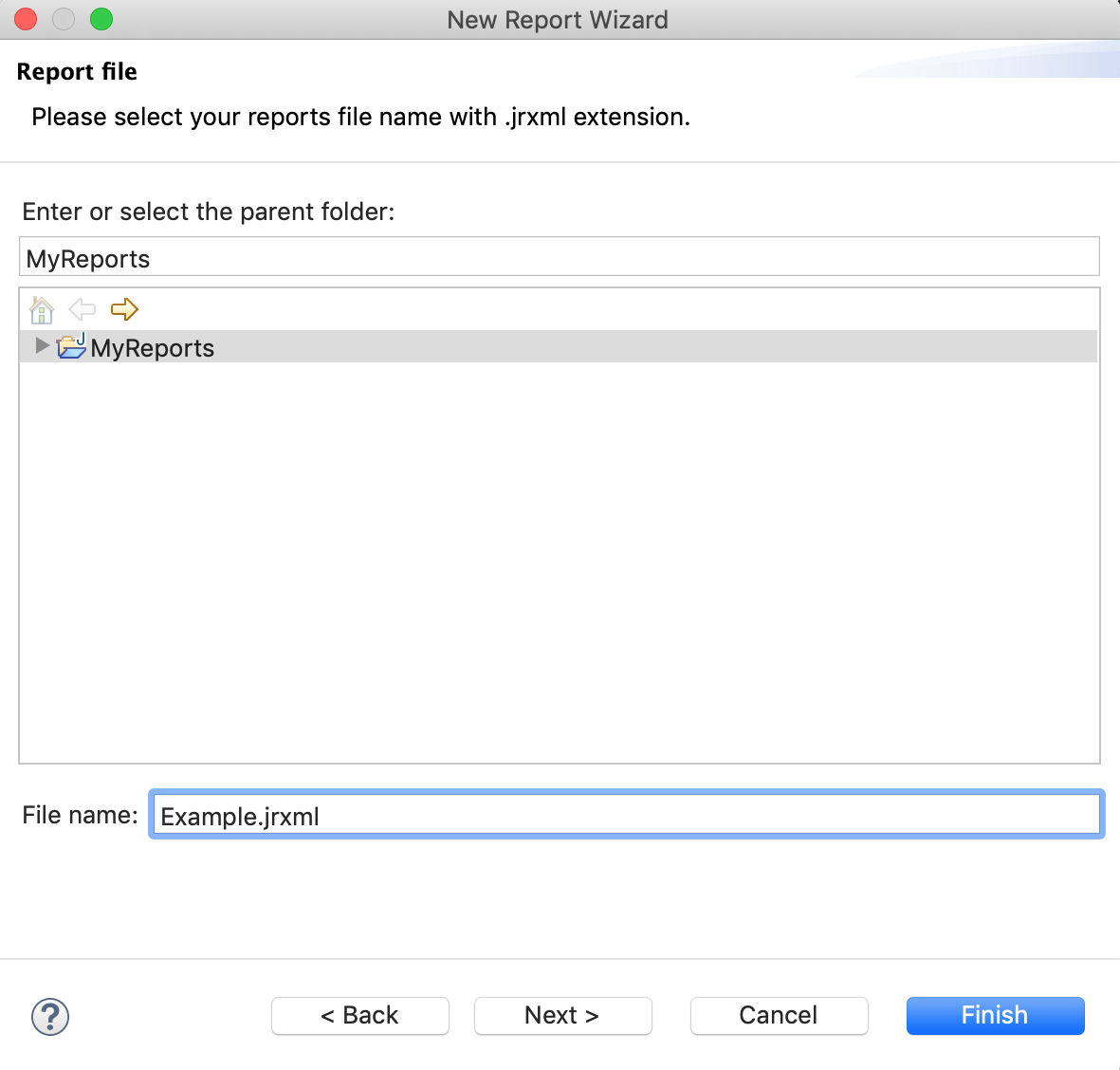
Создание контекстного отчета

Создание отчета производиться в три этапа:

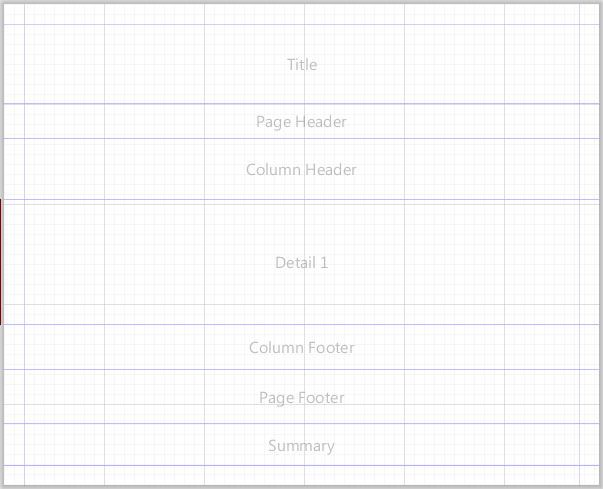
1. Создание шаблона отчета в JasperReport Studio.
2. Публикация шаблона на JasperReport Server.
3. Регистрация и настройка отчета в приложении.

Создание шаблона отчета

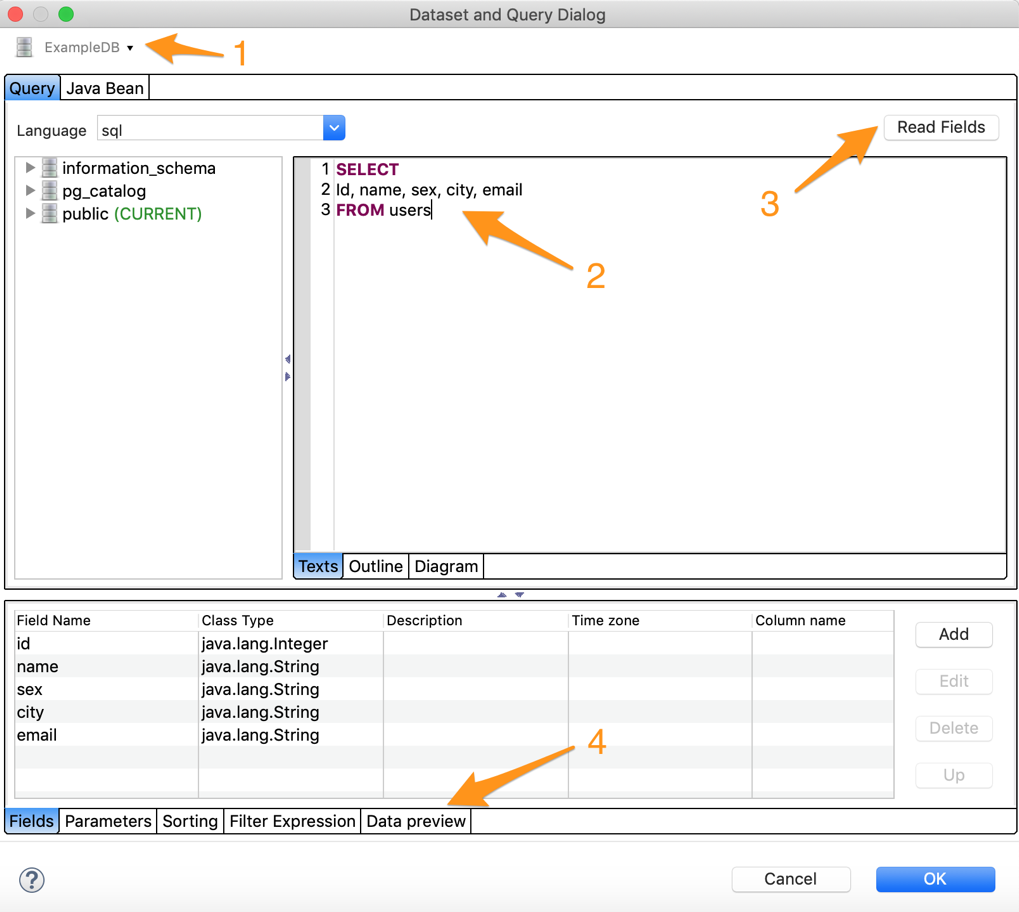
В JasperReport Studio создаем новый шаблон отчета. Нажимаем File → New → Jasper Report → Blank A4 → указываем имя JRXML-файла → выбираем Data Adapter.



После создания отчета видим следующую картинку.



Семь разных блоков — для каждого блока прописано свое поведение, отличающееся от других. Более подробно об этом можно почитать в документации <https://community.jaspersoft.com/documentation/tibco-jaspersoft-studio-user-guide/v630/understanding-bands> . Нажав на Source можно увидеть структуру этих блоков в XML виде. Запрос в БД можно написать нажать на кнопку DataSet and Query editor dialog. После этого откроется новое окно, где необходимо выбрать источник данных (1), написать запрос (2) и объявить переменные поля. Кнопка Read Fields (3) прочитает все поля автоматически при валидном запросе. Чтобы просмотреть данные, нужно нажать на Data preview (4).



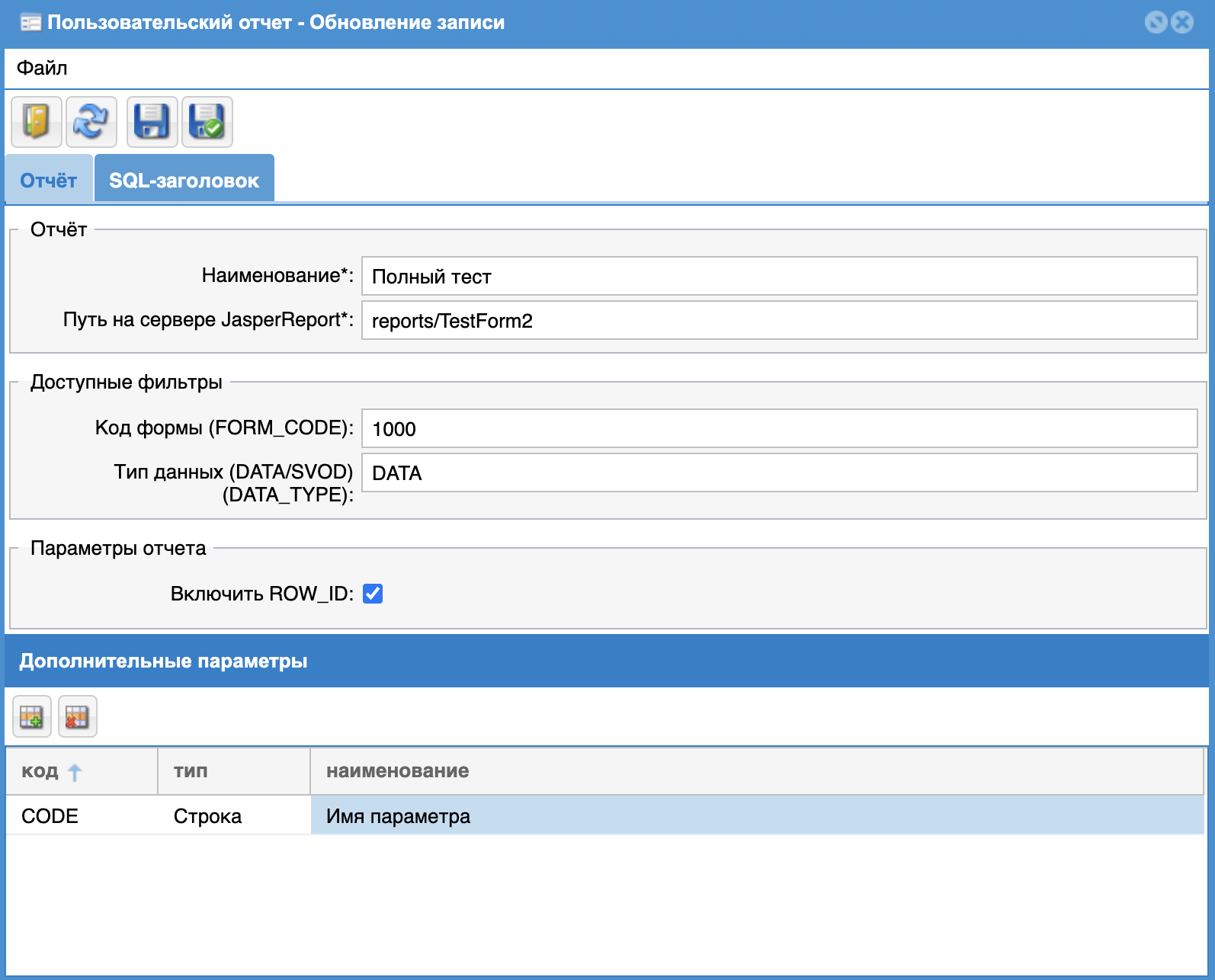
Параметры отчета задаются на вкладке Parameters. Их также можно перетащить в запрос.

Публикация отчета на сервере

Для публикации отчета на сервере нужно нажать кнопку Publish report to JasperReport Server, выбрать сервер и папку Reports.

Регистрация и настройка отчета в приложении

Для регистрации отчета в приложении нужно зайти в режим Администрирование → Пользовательские отчеты. Сверху выбрать требуемый грид, в нижней части будет показан список зарегистрированных отчетов. Создаем новый отчет:



Указываем следующие свойства отчета:

* Наименование – наименование отчета. Отображается в гриде в списке отчетов.
* Путь на сервере JasperReport: локальный путь на сервере JasperReport, можно посмотреть на сервере в подсказке, подержав мышь под требуемом отчете.
* Доступные фильтры: отображаются дополнительные фильтры, определяемые в темплейте грида getReportFilters(). Фильтры работают от разрешения: если значение фильтра установлено – фильтр применяется, в противном случае не применяется.
* Включить ROW\_ID: если данная опция включена, то в отчет передается параметр ROW\_ID с id текущей выбранной строки в гриде. При включенной опции отчет доступен только при выборе какой-либо записи в гриде.
* Дополнительные параметры: задаются дополнительные параметры, которые надо будет установить в окне запуска операции формирования отчета. Параметры передаются в отчет с установленным кодом.
* SQL-заголовок: так как в шаблоне отчета можно привязать только один SQL-запрос, формирующий набор данных отчета, возникает необходимость передачи дополнительных данных для заголовка. Например, для формы наименование структуры формы и дата. Для этого можно сформировать SQL-запрос, возвращающий одну запись результата. В запросе можно использовать параметры отчета с префиксом «:», например :ROW\_ID. Результат запроса передается в отчет в виде набора параметров с кодом имени поля запроса в верхнем регистре.

Решение проблемы экспорта отчета в PDF

При использовании текстового элемента внутри Jaspersoft Studio пользователи могут выбрать используемый для него шрифт. Хотя это кажется простым, при использовании шрифтов возникает много проблем. Основная проблема заключается в том, что доступные шрифты предоставляются операционной системой, и по этой причине у вас могут быть:

* Шрифт, который доступен в одной операционной системе, недоступен в другой. В этом случае в качестве запасного решения для элемента используется шрифт по умолчанию.
* Шрифт может быть доступен в разных операционных системах, но может немного отличаться от одной к другой.

Во многих случаях это просто недопустимо. использование шрифта, отличного от запланированного, вероятно, потребует больше или меньше места для текста, и это может повлиять не только на графический вид, но и на макет самого отчета. Чтобы решить эту проблему, пользователь может использовать внешние файлы шрифтов вместо шрифтов, предоставляемых системой. Таким образом, отчет не будет зависеть от шрифтов операционной системы и везде будет выглядеть одинаково. Это можно сделать с помощью расширения шрифта.

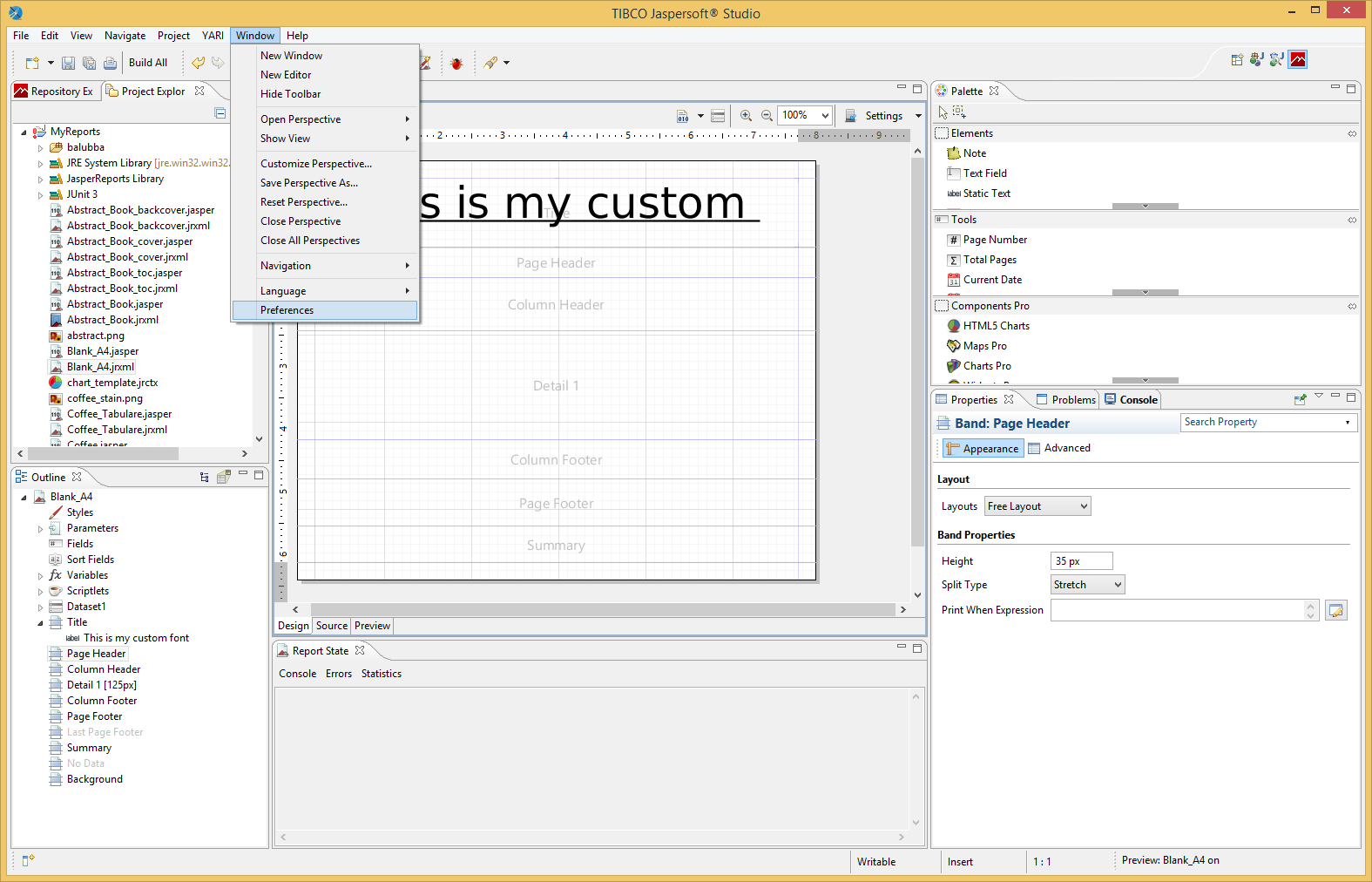
PDF-экспортер

Проблема со шрифтом может быть еще более проблематичной при использовании экспортеров, в частности, экспортера PDF. Формат PDF имеет собственный набор шрифтов. И, если в документе необходимо использовать другие шрифты, они должны быть встроены в PDF. В первых версиях JasperReports эта проблема была решена путем введения некоторых свойств в текстовые элементы, позволяющих указать имя и кодировку шрифта PDF, а также, будет ли шрифт встроен в окончательный PDF-файл или нет.

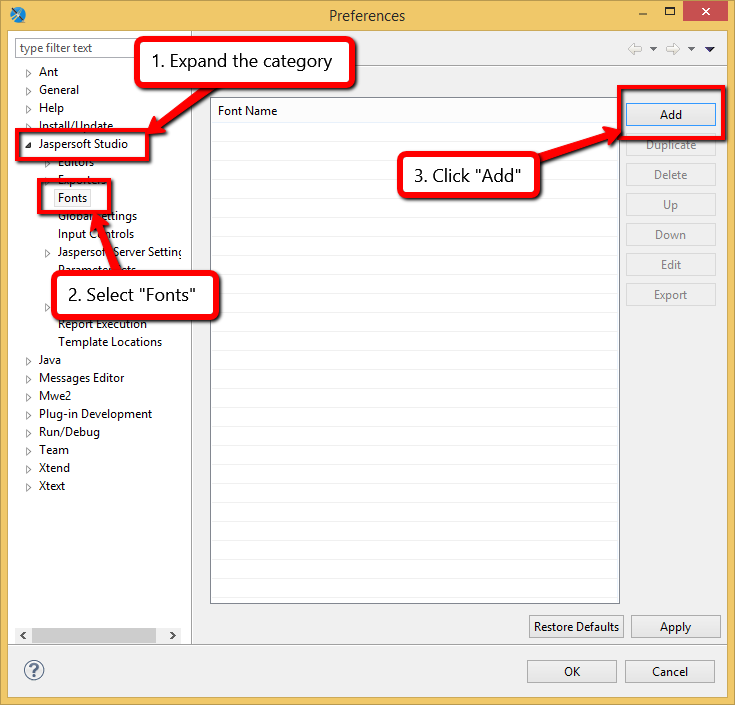
Этот подход все еще можно использовать для обратной совместимости, но сейчас он устарел и не рекомендуется. Фактически, в последней версии Jaspersoft Studio текстовый элемент, использующий это свойство, показывает значок предупреждения и сообщение. Это потому, что Font Extension лучше справляется с этой проблемой. Итак, если пользователь хочет использовать собственный шрифт при экспорте PDF, то это следует сделать с помощью расширения шрифта.

Создать расширение шрифта

Откройте Jaspersoft Studio, перейдите в  ***Window-> Preferences.***

[](https://community.jaspersoft.com/sites/default/files/images/1(105).png)

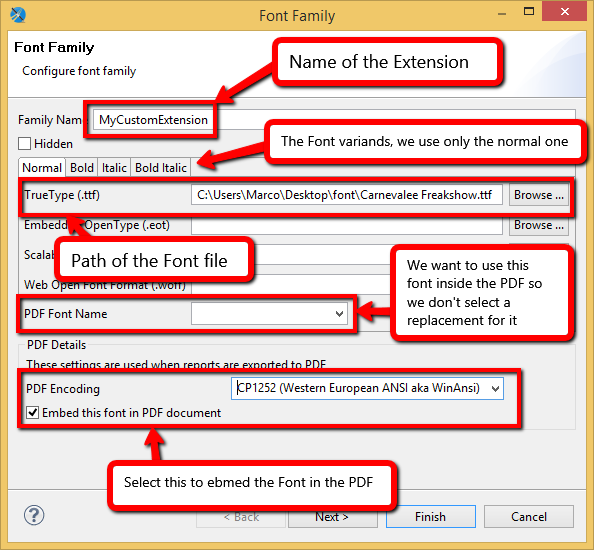
В новом диалоговом окне разверните категорию ***Jaspersoft Studio*** слева, выберите « ***Шрифт»*** и нажмите « ***Добавить».***



На этом этапе вы должны предоставить уникальное имя для расширения шрифта и путь к его расширению.

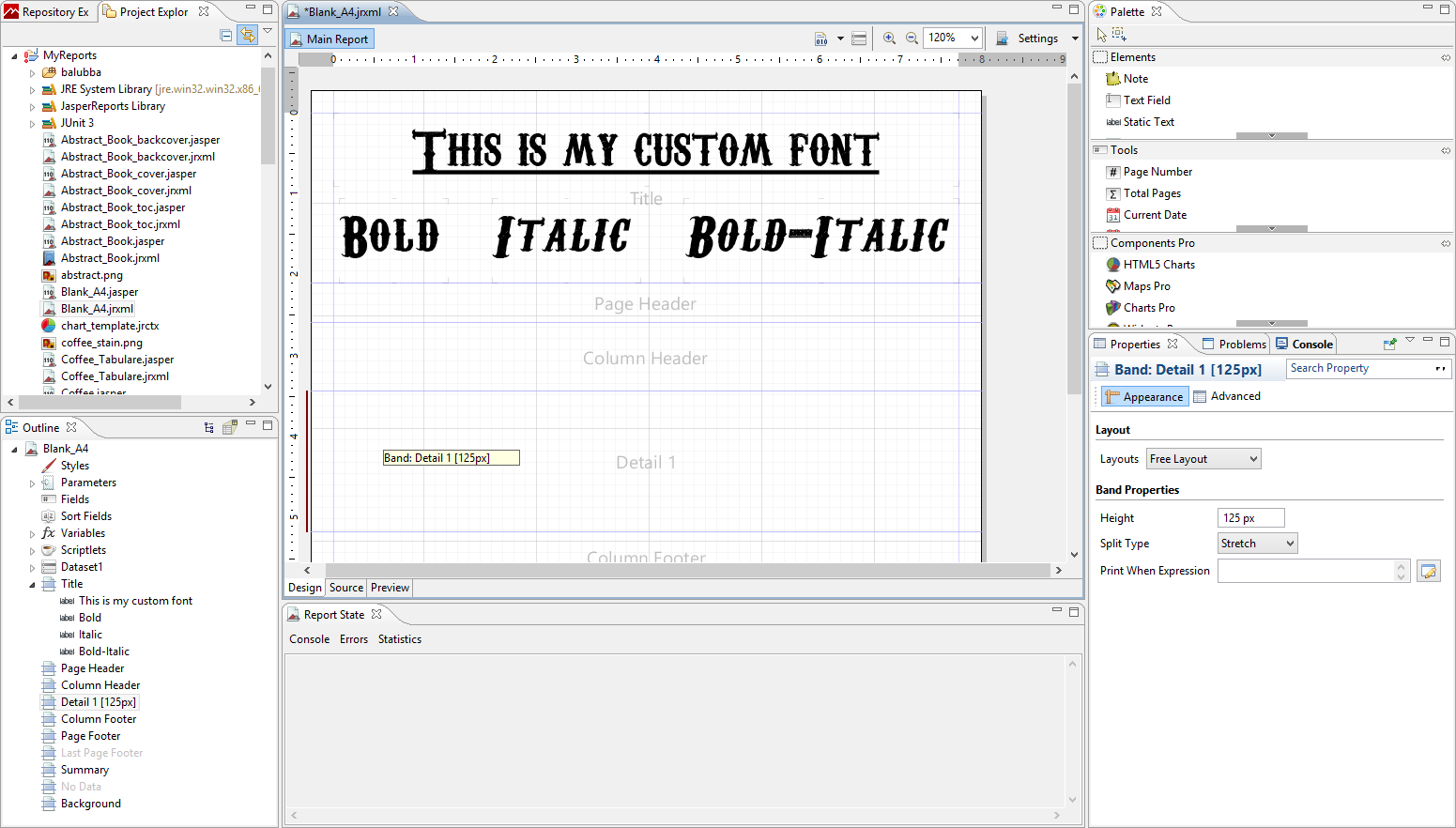
Вы также можете указать вариант шрифта для текста, выделенного жирным, курсивом и полужирным курсивом. Если вы не предоставите это, вы все равно сможете использовать шрифт с одним из этих стилей. Обычный шрифт будет изменен, чтобы получить стилизованный вариант. Но вы также можете указать разные шрифты для каждого из вариантов. Это твой выбор. В этом уроке мы будем использовать только один шрифт Normal.

Вы также можете указать, заменяется ли этот шрифт при экспорте в PDF. Но мы оставим это поле пустым, так как мы хотим использовать этот шрифт также в экспорте PDF. По этой причине мы выбираем « ***Вставить этот шрифт в документ PDF*** » и совместимую кодировку, например  ***CP1251*** . Обратите внимание, что встраивание шрифта в экспорт PDF увеличит размер итогового файла PDF.



Теперь нажмите " ***Далее"*** . В следующих шагах вы можете предоставить дополнительную информацию для других экспортеров и ограничить использование этого шрифта только подмножеством локалей. Нам это не нужно, поэтому просто нажмите « ***Готово*** », чтобы закрыть диалоговое окно, и « ***ОК*** », чтобы закрыть диалоговое окно настроек.

Теперь вы можете открыть отчет, и среди доступных шрифтов вы найдете один с именем, похожим на ваше расширение шрифта, « ***MyCustomExtension*** » (если вы его не видите, попробуйте закрыть и снова открыть отчет). Выберите его, чтобы увидеть, как он используется в элементе. Вы можете установить для стиля полужирный, курсив и полужирный курсив, даже если вы не указали конкретный шрифт при создании расширения шрифта.

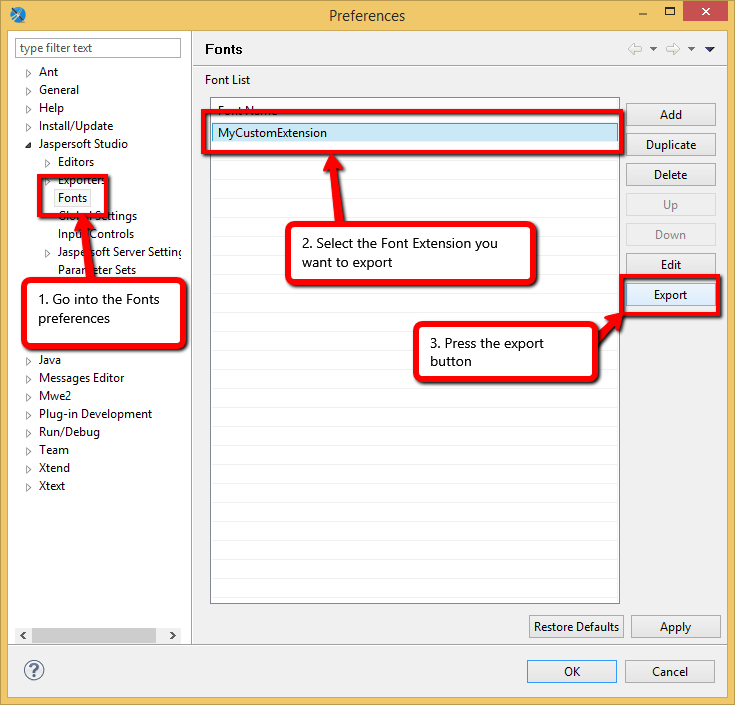
[](https://community.jaspersoft.com/sites/default/files/images/4(41).png)

Вы также можете предварительно просмотреть и экспортировать отчет в формате PDF, чтобы убедиться, что ваш шрифт используется правильно. Расширение, определенное таким образом, будет иметь глобальную область действия и будет отображаться в каждом проекте в вашем рабочем пространстве.

Экспорт расширения шрифта

Одна из проблем, которую решает Font Extensions, состоит в том, чтобы использовать один и тот же шрифт во всех операционных системах, чтобы внешний вид текста был одинаковым. Но, используя то, что объясняется в этом руководстве, мы должны скопировать файл TTF в каждую систему и в каждой системе воссоздать расширение шрифта с точно таким же именем, иначе отчет, сделанный в одной, не найдет шрифт в других. Если бы мы использовали больше файлов TTF для стилей текста, мы должны также скопировать их и более глубоко настроить расширение. Это несложно, но это может занять некоторое время, и повторение этого много раз может привести к ошибкам, например, опечатке в имени расширения. Чтобы упростить это, есть возможность экспортировать наше расширение шрифта как JAR и использовать его в любом проекте.

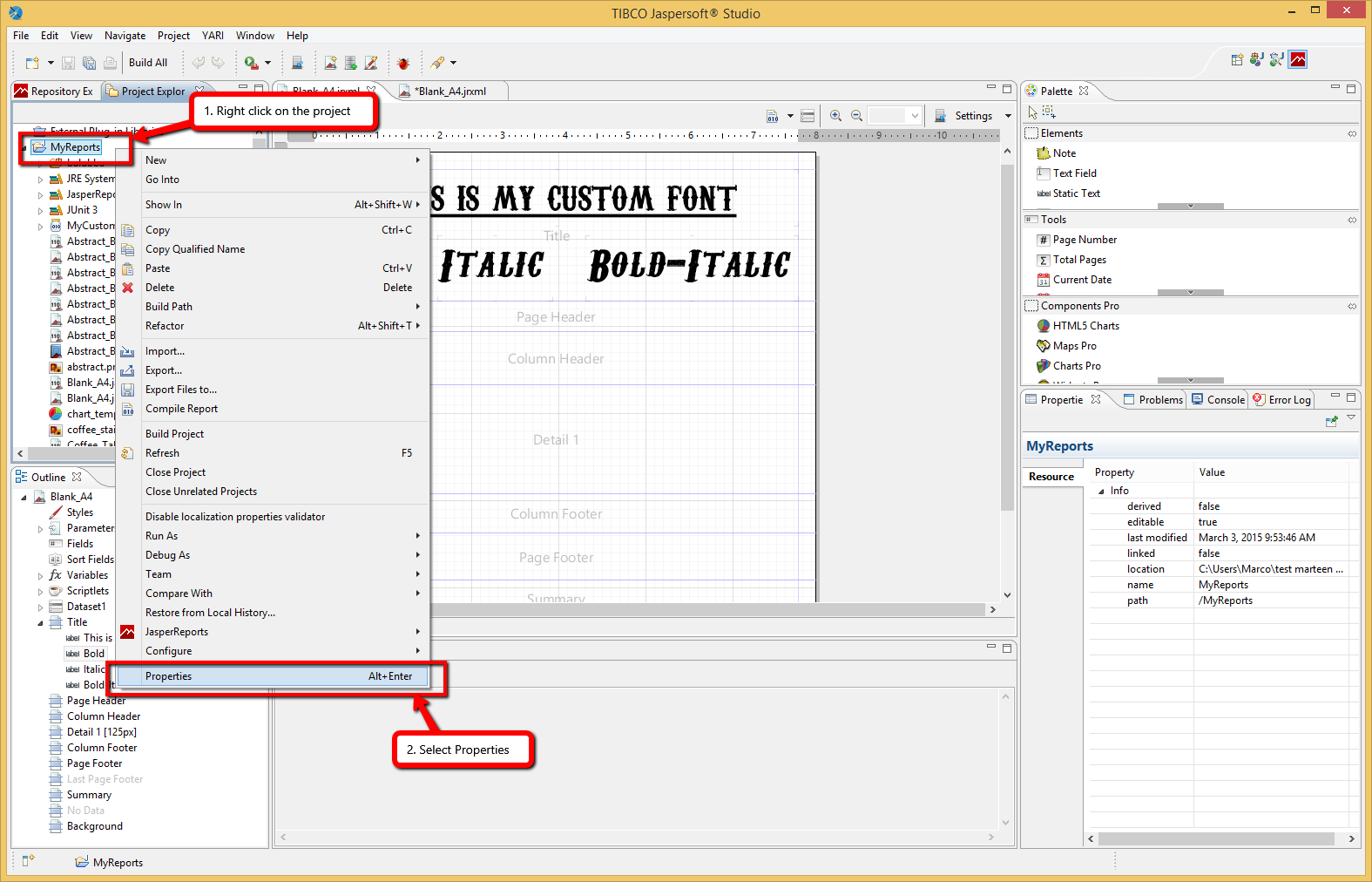
Сначала откройте диалоговое окно « ***Настройки*** » и перейдите в ***раздел*** « ***Шрифты»,*** чтобы увидеть все ваши расширения шрифтов. Выберите ранее созданное расширение « ***MyCustomExtension*** » и нажмите « ***Экспорт*** ».



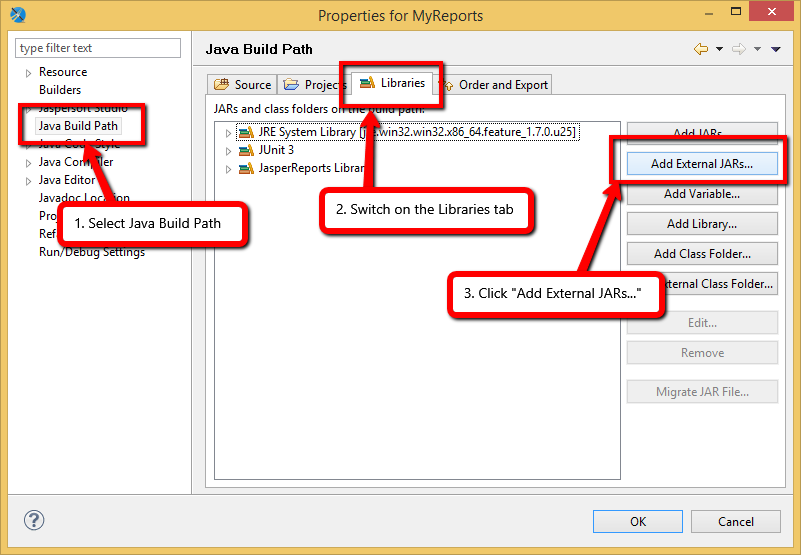
На этом этапе появится диалоговое окно сохранения, в котором определите имя целевого файла и сохраните его. В этом JAR есть все, что нужно для использования расширения в другой системе, вам нужно только поделиться этим единственным файлом, чтобы перенести расширение шрифта на другой компьютер. Вы можете найти созданный таким образом файл в приложении к этому руководству.

Импортируйте расширение шрифта

Теперь у вас есть расширение шрифта. Чтобы использовать его, вы должны добавить его в путь к классам проекта. Например, чтобы добавить расширение шрифта в проект по умолчанию ***MyReports*** , щелкните его правой кнопкой мыши в проводнике проекта и выберите « ***Свойства*** ».

[](https://community.jaspersoft.com/sites/default/files/images/6(26).png)

В появившемся диалоговом окне выберите « ***Путь сборки Java*** » слева, затем включите вкладку « ***Библиотеки*** » справа и нажмите « ***Добавить внешние JAR-файлы*** ...»



На этом этапе выберите JAR-файл расширения шрифта и нажмите « ***ОК»*** . Теперь для каждого отчета, содержащегося в проекте MyReports, вы увидите шрифт « ***MyCustomExtension*** ». В некоторых версиях Jaspersoft Studio новый шрифт не будет отображаться в отчетах, которые в данный момент открыты в редакторе. В этом случае закройте отчет и снова откройте его. Помните также, что расширение шрифта, определенное в настройках Jaspersoft Studio, является глобальным. Это видно по всем отчетам в рабочей области. Один, определенный с помощью JAR в пути к классам проекта, будет виден только в отчетах внутри проекта.

Опубликовать шрифт на сервере

После того, как у вас есть файл font.jar, поместите его в путь к классам вашего приложения (jasperserver-pro / WEB-INF / lib) и перезапустите сервер отчетов jasper.