

Diese Website verwendet Cookies von Google, um ihre Dienste bereitzustellen, Anzeigen zu personalisieren und Zugriffe zu analysieren. Informationen darüber, wie Sie die Website verwenden, werden an Google weitergegeben. Durch die Nutzung dieser Website erklären Sie sich damit einverstanden, dass sie Cookies verwendet.

WEITERE INFORMATIONEN OK

Постоянные
читатели

Архив блога


▼ 2009 (1)


▼ февраля (1)

[Почему gwt date picker
медленно работает
\(тормозит...\)](#)

► 2008 (3)

Авторы

 [Alexey Zaprudnev](#)

 [Viktor Zaprudnev](#)

Подпишитесь на

 Сообщения 

 Все комментарии 

С Р Е Д А , 1 1 Ф Е В Р А Л Я 2

Почему gwt date picker медленно работает (тормозит)

Недавно я обнаружил, что GWT DatePicker (из инкубатора) заметно притормаживает под Firefox. Заметно - я имею ввиду - на любой клик приходится ждать порядка 5 сек.

Самое удивительное, что та же самая аппликация работала на других машинках как заводная. При этом, заметил, что торможение бывает только под Firefox, но без разницы, на какой версии.

Оказывается, что причиной этому был FireBug плагин для firefox.

Когда я его отключил, аппликация заработала как положено.

Надеюсь, если вы столкнетесь с этим, отключение FireBag вам поможет!

Автор: [Viktor Zaprudnev](#) на 0:08 [Комментариев нет:](#) [Ссылки на это сообщение](#)
Ярлыки: [gwt](#)

С Р Е Д А , 1 2 Н О Я Б Р Я 2 0

GWT (GWT-EXT) + JasperReports быстро и просто.

Часть первая, как все началось

Вы энтузиаст новых инновационных технологий, разрабатываете корпоративное решение на Google Web Toolkit (GWT), но вот однажды в третьем часу ночи из бани звонит вам клиент, и сообщает, что зайдет днем пообщаться насчет работы с отчетами.

Какие еще отчеты, вон какие симпатичные панельки, окошки, кнопки думаете вы. Но вспомнив, что клиенто-ориентированность это хорошо, решаем ввести клиента в восторг.

Начнем, как водится, с постановки задачи.

Постановка задачи

Задача у нас не очень сложная, но вместе с тем не совсем тривиальная – создать прототип запуска, просмотра отчетов, а также уметь экспортировать в различные форматы.

Вам понадобится

Чтобы собрать приложение своими руками, вам понадобятся следующие инструменты:

☼ Eclipse IDE - <http://www.eclipse.org/downloads/>

☼ Google Web Toolkit - <http://code.google.com/webtoolkit/gettingstarted.html>

☼ JasperReports - <http://www.jaspersoft.com/>

☼ iReport - <http://ireport.sourceforge.net/>

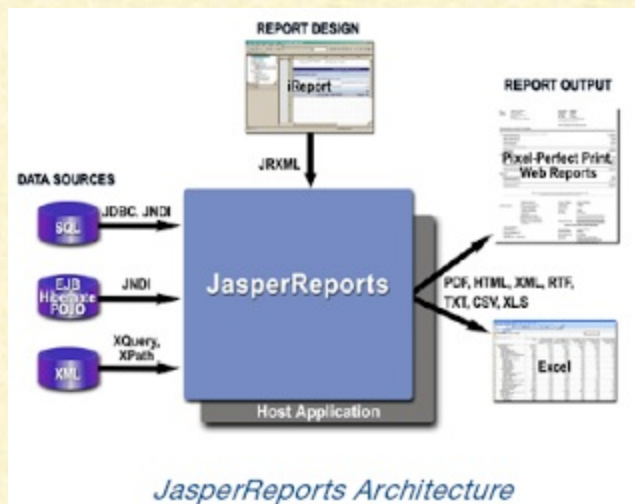
☼ Tomcat 5.x/6 - <http://tomcat.apache.org/>

☼ Ant - <http://ant.apache.org/>

☼ Gwt (Gwt-Ext) приложение :)

Немного о JasperReports

JasperReports один из популярных Open Source библиотек для создания отчетов. Очень просто интегрируется в любые Java приложения.



Отчеты описываются в формате XML, что в последующем возможно создание отчетов на базе шаблонов, используя Velocity (<http://velocity.apache.org/>), Jelly (<http://commons.apache.org/jelly/>) или другие инструментарии. Файлы имеют расширение *.jrxml.

Далее файл отчета компилируется, и на выходе получаем файлы *.jasper.

```
JasperCompileManager.compileReportToFile("WebappReport.jrxml");
```

В некоторых случаях это удобно, так как дает возможность в эксплуатации поставлять клиентам готовый отчет, а не исходники его.

После этого отчет запускается с помощью `JasperFillManager.fillReport()` в результате получаем `JasperPrint` объект, также возможно сохранить в файл.

В конце используя один из классов `JexcelApiExporter`, `JexcelApiExporterNature`, `JexcelApiExporterParameter`, `JRCsvExporter`, `JRGraphics2DExporter`, `JRHtmlExporter`, `JRPdfExporter`, `JRPrintServiceExporter`, `JRRtfExporter`, `JRTextExporter`, `JRXlsExporter`, `JRXmlExporter` можем экспортировать отчет в любой из данных форматов.

Прототип приложения

Для примера взят протип приложения GWT-Ext Showcase2, переделанный в рамках требований клиента.



Когда мы кликаем “Logged users”, информация разворачивается в новом табе, показывая детали. Пока все просто и ясно? Теперь все по порядку.

Первый шаг

Итак, перво-наперво поработаем с Jasper и создадим сам отчет

⊗ Имеем jasperreports-3.1.2-project.zip, разархивируем его в C:\demo\jasper. Находим в директории “demo\samples\webapp” (отличный пример для запуска отчетов под web) отчет WebappReport.jrxml

⊗ Далее используя iReport подправляем отчет под наши нужды, заменяем заголовок отчета, а также настраиваем JDBC DataSource и Report Query.



⊗ Слегка меняю исходники в директории “demo\samples\webapp\WEB-INF\classes\servlets” – делаем возможность запуска отчета используя параметр reportfile: String rp = req.getParameter(“reportfile”); String reportFileName = context.getRealPath(“/reports/”+rp+“.jasper”); “demo\samples\webapp” для тестирования – в HTML все ссылки прописываем название отчета “?reportfile=WebappReport”;

⊗ В корневом каталоге “demo\samples\webapp” запускаем ANT.

```

D:\...ports-3.1.2\demo\samples\webapp>ant
Buildfile: build.xml

prepare:

javac:
[javac] Compiling 6 source files to D:\Temp\Downloads\jasper\jasperreports-3
.1.2-project\jasperreports-3.1.2\demo\samples\webapp\WEB-INF\classes
[javac] Note: Some input files use unchecked or unsafe operations.
[javac] Note: Recompile with -Xlint:unchecked for details.

BUILD SUCCESSFUL
Total time: 10 seconds

```

Итак, после настройки Tomcat, копируем содержимое
 “demo\samples\webapp” в Tomcat “webapps\jasper”, и запускаем его.

Открываем браузер вводим туда ссылку ниже (Внимание у вас
 может отличаться порт!).

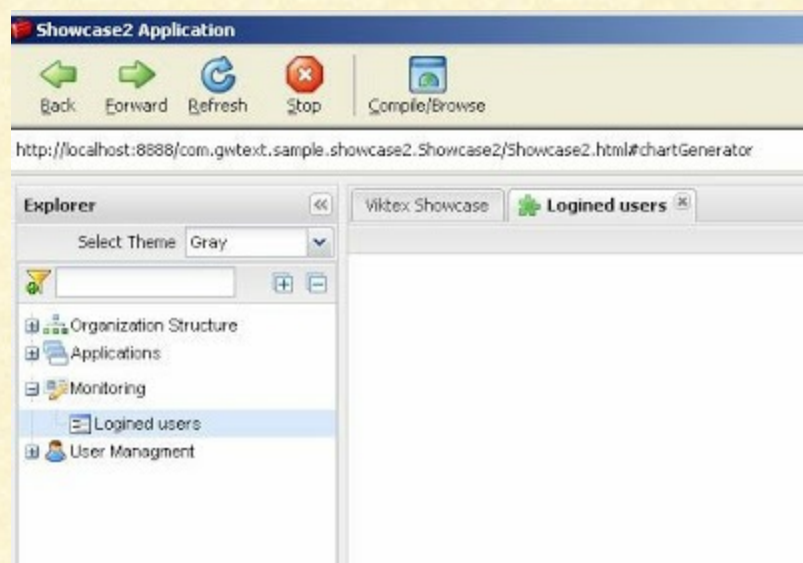
<http://localhost:8887/jasper/>



Здесь же и проверяем отчет.

Второй шаг

Итак, отчет у нас работает, теперь приступаем к GWT. При нажатии
 на “Logged users”, открывается пустой таб:



Осталось интегрировать Jasper.

Создаем простейший ReportManager class со статическим методом:


```

public class ReportManager {
public static String getExportReportURL(String report, String exptype){
return "http://localhost:8887/jasper/servlets/" + exptype + "?reportfile=" +
report;
};
}

```

Далее создаем панель:

```

Panel centerPanel = new Panel();
centerPanel.setLayout(new BorderLayout());
centerPanel.setTitle("Sessions");
...

```

```

String rep_url =
ReportManager.getExportReportURL("WebappReport", "html");
final Frame frame = new Frame(rep_url);
centerPanel.add(frame, new BorderLayoutData(RegionPosition.CENTER));

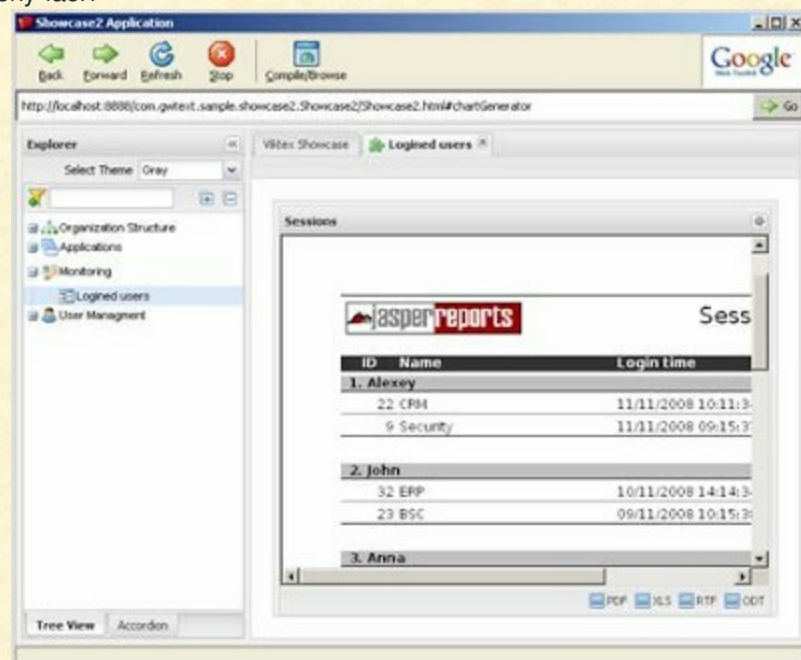
```

```

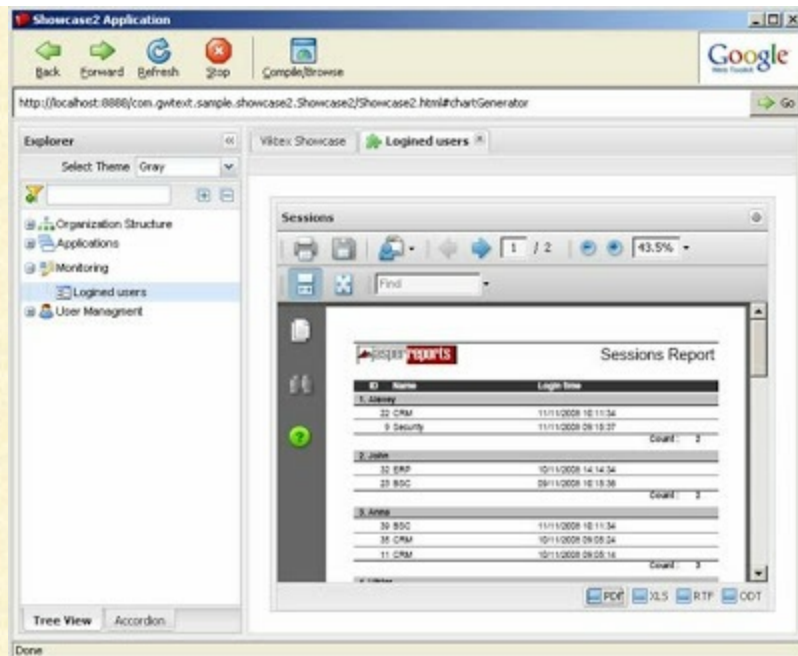
Toolbar toolbar = new Toolbar();
toolbar.addFill();
ToolbarButton toolbarButton = new ToolbarButton("PDF", new
ButtonListenerAdapter() {
public void onClick(Button button, EventObject e) {
frame.setUrl(ReportManager.getExportReportURL("WebappReport", "pdf"));
}
});
...

```

Запускаем проект в Host Mode, нажимаем на "Logged users" и
получаем



При нажатии PDF – мы увидим отчет уже в Pdf формате:



Теперь осталось дожидаться клиента и за чашечкой кофе рассказать о будущих перспективах :)

Автор: [Alexey Zaprudnev](#) на 20:45 Комментариев нет: [Ссылки на это сообщение](#)
Ярлыки: [ant](#) , [eclipse](#) , [gwt](#) , [ireport](#) , [jasper](#) , [reports](#) , [tomcat](#) , [viktex](#)

Ч Е Т В Е Р Г , 6 Н О Я Б Р Я 2

Как сделать GWT StackPanel похожим на Outlook

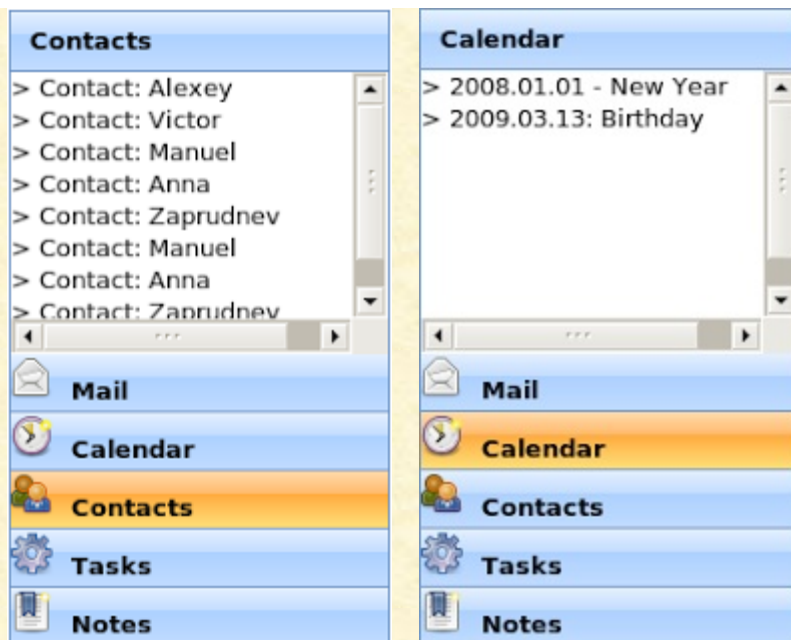
На этот раз хочется поделиться новым компонентом на GWT - Outlook Stack Panel.

Почему я решил написать его?

Проблема в том, что стандартный StackPanel от GWT, работает так, что при выборе закладки, она сдвигает другие закладки и показывается ее содержание. При этом получается так, что содержание закладки все время скочет вверх и вниз, что на мой взгляд не очень user friendly.

Пример stack панели, как бы я ее хотел видеть - Outlook stack panel: содержимое всегда находится в одном месте, кнопки не скочат и имеют один и тот же порядок.

Вот что у меня получилось.



Как видите, при смене закладки сами закладки остаются на том же месте, меняется заголовок и содержание.
Реализуется это используя DeckPanel компонент.

```
/**
 * Creates an empty stack panel.
 */
public OutlookStackPanel() {
    content.setAnimationEnabled(true);
    content.setWidth(DEFAULT_WIDTH);

    container.setStyleName(DEFAULT_STYLE);
    header.setSize(DEFAULT_WIDTH, DEFAULT_HEADER_HEIGHT);
    footer.setSize(DEFAULT_WIDTH, "");

    ScrollPanel scroller = new ScrollPanel();
    scroller.setAlwaysShowScrollBars(false);
    scroller.add(content);

    container.setVerticalAlignment(HasVerticalAlignment.ALIGN_TOP);
    container.add(header);
    container.add(scroller);

    container.setVerticalAlignment(HasVerticalAlignment.ALIGN_BOTTOM);
    container.add(footer);

    content.setAnimationEnabled(true);

    header.setStyleName(DEFAULT_HEADER_STYLE);

    initWidget(container);

    sinkEvents(Event.ONCLICK);
    sinkEvents(Event.ONMOUSEOVER);
    sinkEvents(Event.ONMOUSEOUT);
}
```

Затем не хитрый код, для отображения нужного содержания

```

@Override
public void onBrowserEvent(Event event) {

    Element target = event.getTarget();
    if (DOM.eventGetType(event) == Event.ONCLICK) {
        for (Widget widget : buttons) {
            if (widget.getElement() == target) {
                buttons.get(header.getVisibleWidget()).removeStyleDependentName("selected");
                widget.addStyleDependentName("selected");

                int index = buttons.indexOf(widget);
                if (index >= 0) {
                    header.showWidget(index);
                    content.showWidget(index);
                }
            }
        }
    } else if (DOM.eventGetType(event) == Event.ONMOUSEOVER) {
        ...
    } else if (DOM.eventGetType(event) == Event.ONMOUSEOUT) {
        ...
    }
}

```

Вы можете скачать исходники [здесь](#) ;) Удачи!
 В скором времени, я надеюсь выложить GWT application starter [here](#) (gwtfuse):

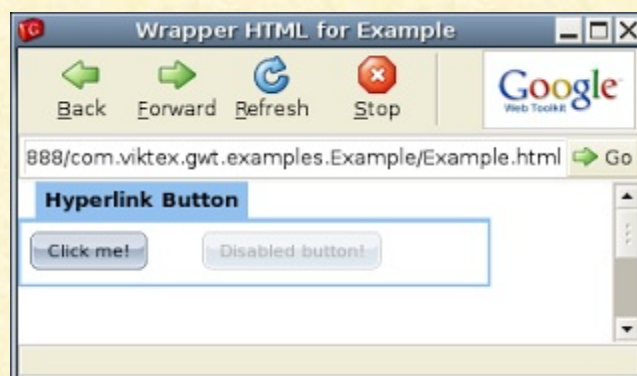
Если кто либо сделал это более красиво и элегантно, то прошу поделиться идеями ;)

Автор: [Viktor Zaprudnev](#) на 21:15 [Комментариев нет:](#) [Ссылки на это сообщение](#)
 Ярлыки: [button](#) , [gwt](#) , [outlook](#)

С У Б Б О Т А , 1 Н О Я Б Р Я 2

Как сделать красивую кнопку в GWT

GWT прекрасный инструмент для разработки веб приложений. На мой взгляд, единственный недостаток его на сегодняшний день - недостаток красивых компонентов. Конечно, это легко решить с помощью Gwt-ext или Ext-gwt, но, хотите ли вы добавлять на ваш сайт всю библиотеку этих скриптов и становиться зависимым от них? А что если вам нужно всего навсего несколько компонентов? Недавно, работая в одном GWT проекте нам понадобилось создать кнопки определенного вида и я решил, почему бы не поделиться этим с другими любителями GWT.



Так может выглядеть кнопка. Используя CSS и GWT Hyperlink, можно создать кнопку любого вида.

Я не нашел, как можно поместить рабочий пример на сайт, и поместил линк, по которому можно скачать скомпилированный пример (если вам лень настраивать проект в Eclipse):

Скомпилированная версия, а также вы можете скачать исходники по этой ссылке: Скачать исходники

Если у вас есть лучше идеи, как сделать эти кнопки, пишите, буду рад!

Автор: Viktor Zaprudnev на 20:08 Комментариев нет: Ссылки на это сообщение
Ярлыки: button , gwt

[Главная страница](#)

Подписаться на: [Сообщения \(Atom\)](#)

