

# GWT - Introduction

Dr. Lofi Dewanto

<https://lofidewanto.blogspot.de>

# Agenda

- Webapps - State of the Art
- Components of GWT
  - Java to JavaScript Transpiler
  - Emulated Java Runtime Environment
  - Interoperability Layer to JavaScript
  - Tools
  - User Interface Components

# Webapps - State of the Art

- See: Webapps - [State of the Art](#)
- State of the Art:
  - SOFEA (Service Oriented Front End Architecture)
  - ROCA (Resource Oriented Client Architecture)

# Components of GWT

- Java to JavaScript Transpiler
- Emulated Java Runtime Environment
- Interoperability Layer to JavaScript
- Tools
- User Interface Components

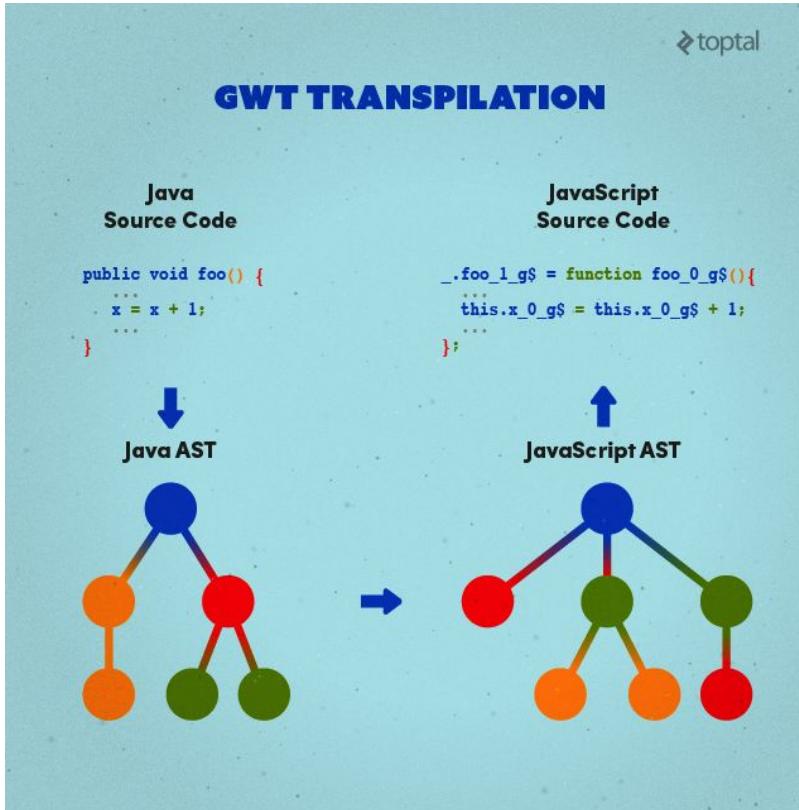
# Java to JavaScript Transpiler

# Java to JavaScript (JS) Transpiler

- Main component: Java (Source) to JavaScript (Source) transpiler
- Optimize the JS output at compile time
  - Naming the shortest possible name (attribute, method, class)
  - Pruning unused classes, methods and attributes

```
gwtmaterial.onScriptDownloaded(["var $wnd = $wnd || window.parent;var __gwtModuleFunction = $wnd.gwtmaterial;var $sendStats = __gwtModuleFunction.__sendStats;"..."
```

# Java to JavaScript (JS) Transpiler

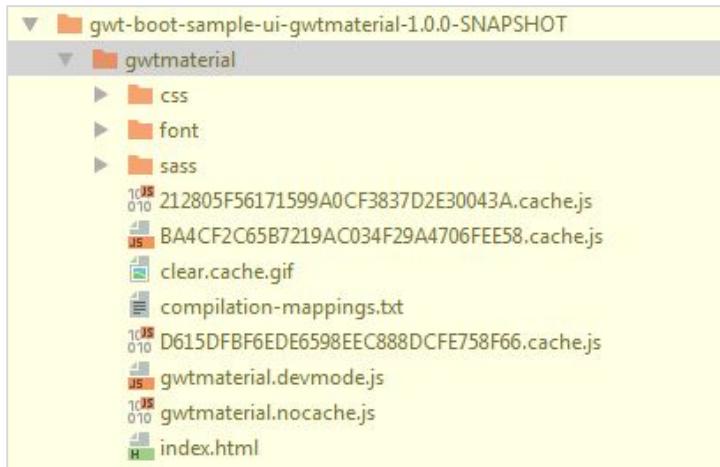


# Java to JavaScript (JS) Transpiler

- Inside GWT transpiler: <http://bit.ly/2InB0AW>

# Java to JavaScript (JS) Transpiler

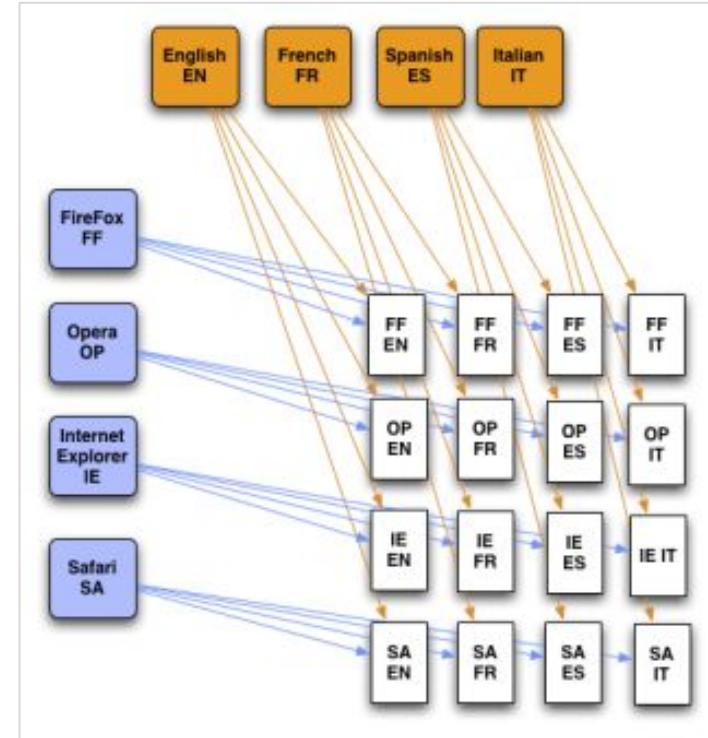
- Perfect caching
  - \*.nocache.js or \*.nocache.\*
  - <hashname>.cache.js or <hashname>.cache.\*



compilation-mappings.txt	
1	212805F56171599A0CF3837D2E30043A.cache.js
2	user.agent gecko1_8
3	
4	BA4CF2C65B7219AC034F29A4706FEE58.cache.js
5	user.agent ie10
6	
7	D615DFBF6EDE6598EEC888DCFE758F66.cache.js
8	user.agent safari
9	
10	Devmode:devmode.js

# Java to JavaScript (JS) Transpiler

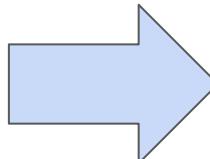
- Separation of
  - User agent (IE, Firefox, Safari)
  - Locale (EN, DE, ...)
- Reducing the download size of JS



# Java to JavaScript (JS) Transpiler

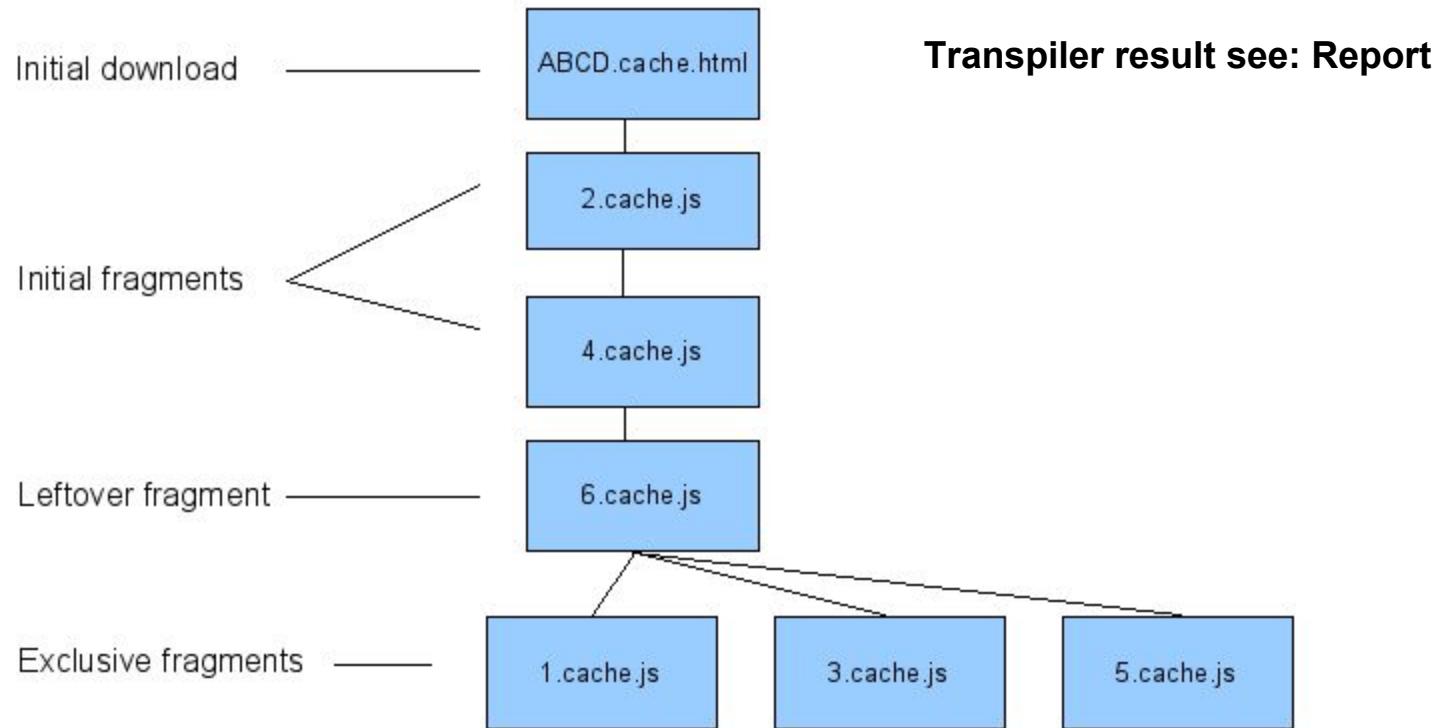
- Dead-for-now (DFN) code-Splitting “on demand”

```
public class Hello implements EntryPoint {  
    public void onModuleLoad() {  
        Button b = new Button("Click me", new ClickHandler() {  
            public void onClick(ClickEvent event) {  
                Window.alert("Hello, AJAX");  
            }  
        });  
  
        RootPanel.get().add(b);  
    }  
}
```



```
public class Hello implements EntryPoint {  
    public void onModuleLoad() {  
        Button b = new Button("Click me", new ClickHandler() {  
            public void onClick(ClickEvent event) {  
                GWT.runAsync(new RunAsyncCallback() {  
                    public void onFailure(Throwable caught) {  
                        Window.alert("Code download failed");  
                    }  
  
                    public void onSuccess() {  
                        Window.alert("Hello, AJAX");  
                    }  
                });  
            }  
        });  
  
        RootPanel.get().add(b);  
    }  
}
```

# Java to JavaScript (JS) Transpiler



# Java to JavaScript (JS) Transpiler

- Compiler CLI
  - [com.google.gwt.dev.Compiler](#)
- Compiler options (2.8.x)

```
> java -cp gwt-dev.jar com.google.gwt.dev.Compiler
Missing required argument 'module[s]'

Google Web Toolkit 2.8.0
Compiler [-logLevel (ERROR|WARN|INFO|TRACE|DEBUG|SPAM|ALL)] [-workDir dir] [-X[no]closureFormattedOutput] [-[no]compileReport] [-X[no]checkCasts] [-X[no]classMetadata] [-[no]draftCompile] [-[no]checkAssertions] [-XfragmentCount] [-XfragmentMerge] [-gen] [-[no]generateJsInteropExports] [-XmethodNameDisplayMode] [-Xnamespace] [-optimize] [-[no]saveSource] [-setProperty] [-style] [-[no]failOnError] [-[no]validateOnly] [-sourceLevel] [-localWorkers] [-[no]incremental] [-war] [-deploy] [-extra] [-saveSourceOutput] and module[s]
```

The level of logging detail: ERROR, WARN, INFO, TRACE, DEBUG, SPAM or ALL (defaults to INFO)  
The compiler's working directory for internal use (must be writeable; defaults to a system temp d  
EXPERIMENTAL: Enables Javascript output suitable for post-compilation by Closure Compiler (default  
Compile a report that tells the "Story of Your Compile". (defaults to OFF)  
EXPERIMENTAL: DEPRECATED: use jre.checks.checkLevel instead. (defaults to OFF)  
EXPERIMENTAL: Include metadata for some java.lang.Class methods (e.g. getName()). (defaults to ON  
Compile quickly with minimal optimizations. (defaults to OFF)  
Include assert statements in compiled output. (defaults to OFF)  
EXPERIMENTAL: Limits of number of fragments using a code splitter that merges split points.  
DEPRECATED (use -XfragmentCount instead): Enables Fragment merging code splitter.  
Debugging: causes normally-transient generated types to be saved in the specified directory  
Generate exports for JsInterop purposes (defaults to OFF)  
EXPERIMENTAL: Specifies method display name mode for chrome devtools: NONE, ONLY\_METHOD\_NAME, ABB  
Puts most JavaScript globals into namespaces. Default: PACKAGE for -draftCompile, otherwise NONE  
Sets the optimization level used by the compiler. 0=none 9=maximum.  
Enables saving source code needed by debuggers. Also see -debugDir. (defaults to OFF)  
Set the values of a property in the form of propertyName=value1[,value2...].  
Script output style: DETAILED, OBFUSCATED or PRETTY (defaults to OBFUSCATED)  
Fail compilation if any input file contains an error. (defaults to OFF)  
Validate all source code, but do not compile. (defaults to OFF)  
Specifies Java source level (defaults to 1.8)  
The number of local workers to use when compiling permutations  
Compiles faster by reusing data from the previous compile. (defaults to OFF)  
The directory into which deployable output files will be written (defaults to 'war')  
The directory into which deployable but not servable output files will be written (defaults to 'W  
The directory into which extra files, not intended for deployment, will be written  
Overrides where source files useful to debuggers will be written. Default: saved with extras.

Specifies the name(s) of the module(s) to compile

# Story of your Compile Report (SoyC)

**Compile report**

**Google web toolkit**

**Compile report**

**Overview of permutations**

- Permutation 0 ('locale': 'default', 'user.agent': 'ie6')
- Permutation 1 ('locale': 'en', 'user.agent': 'ie6')
- Permutation 2 ('locale': 'default', 'user.agent': 'ie8')
- Permutation 3 ('locale': 'en', 'user.agent': 'ie8')
- Permutation 4 ('locale': 'default', 'user.agent': 'safari')
- Permutation 5 ('locale': 'en', 'user.agent': 'safari')
- Permutation 6 ('locale': 'default', 'user.agent': 'gecko1\_8')
- Permutation 7 ('locale': 'en', 'user.agent': 'gecko1\_8')
- Permutation 8 ('locale': 'default', 'user.agent': 'gecko')
- Permutation 9 ('locale': 'en', 'user.agent': 'gecko')
- Permutation 10 ('locale': 'default', 'user.agent': 'opera')
- Permutation 11 ('locale': 'en', 'user.agent': 'opera')

**Compile report: Permutation 2**

Full code size: **432956 Bytes**   Initial download size: **197340 Bytes**   Left over code: **41555 Bytes**

#	Location	Size
1	@com.google.gwt.sample.showcase.client.content.i18n.CwConstantsExample.asyncOnInitialize	3952 Bytes (0.91%)
2	@com.google.gwt.sample.showcase.client.content.i18n.CwConstantsWithLookupExample.asyncOnInitialize	2509 Bytes (0.58%)
3	@com.google.gwt.sample.showcase.client.content.i18n.CwDateTimeFormat.asyncOnInitialize	4367 Bytes (1.01%)
4	@com.google.gwt.sample.showcase.client.content.i18n.CwDictionaryExample.asyncOnInitialize	2574 Bytes (0.59%)
5	@com.google.gwt.sample.showcase.client.content.i18n.CwMessagesExample.asyncOnInitialize	2352 Bytes (0.54%)
6	@com.google.gwt.sample.showcase.client.content.i18n.CwNumberFormat.asyncOnInitialize	7439 Bytes (1.72%)
7	@com.google.gwt.sample.showcase.client.content.i18n.CwPluralFormsExample.asyncOnInitialize	1785 Bytes (0.41%)
8	@com.google.gwt.sample.showcase.client.content.lists.CwListBox.asyncOnInitialize	2439 Bytes (0.56%)
9	@com.google.gwt.sample.showcase.client.content.lists.CwMenuBar.asyncOnInitialize	3785 Bytes (0.87%)
10	@com.google.gwt.sample.showcase.client.content.lists.CwStackPanel.asyncOnInitialize	16814 Bytes (3.88%)
11	@com.google.gwt.sample.showcase.client.content.lists.CwSuggestBox.asyncOnInitialize	13837 Bytes (3.2%)
12	@com.google.gwt.sample.showcase.client.content.lists.CwTree.asyncOnInitialize	5230 Bytes (1.21%)
13	@com.google.gwt.sample.showcase.client.content.other.CwAnimation.asyncOnInitialize	2603 Bytes (0.6%)
14	@com.google.gwt.sample.showcase.client.content.other.CwCookies.asyncOnInitialize	3646 Bytes (0.84%)
	www.corp.google.com/~kprobst/.../sp14-2-overallBreakdown.html#asyncOnInitialize	493 Bytes (0.11%)

# Bootstrap JS with GWT

- Host HTML file with JS to bootstrap the JS

```
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <link type="text/css" rel="stylesheet" href="Hello.css">
    <title></title>
  </head>
  <body>

<script type="text/javascript" language='javascript' src='hello/hello.nocache.js'></script>
    <!-- Along with page title and table headers defined -->
  </body>
</html>
```

# Bootstrap JS with GWT

- Example

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
<html>
  <body onload='alert("w00t!")'>
    <img src='bigImageZero.jpg'></img>
    <script source='externalScriptZero.js'></script>
    <img src='bigImageOne.jpg'></img>
    <img src='reallyBigImageTwo.jpg'></img>
    <script src='myApp/myApp.nocache.js'></script>
    <script src='externalScriptOne.js'></script>
  </body>
</html>
```

# Bootstrap JS with GWT - Principles

- `<script>` tags always block evaluation of the page until the script is fetched and evaluated.
- `<img>` tags do not block page evaluation.
- Most browsers will allow a maximum of two simultaneous connections for fetching resources.
- The `body.onload()` event will only fire once all external resources are fetched, including images and frames.

# Bootstrap JS with GWT - Principles

- The GWT selection script (i.e. `myApp/myApp.nocache.js`) will be fetched and evaluated like a normal script tag, but the compiled script will be fetched asynchronously.
- Once the GWT selection script has started, its `onModuleLoad()` can be called at any point after the outer document has been parsed.

# Bootstrap JS with GWT - Example Sequence

- The HTML document is fetched and parsing begins.
- Begin fetching `bigImageZero.jpg`.
- Begin fetching `externalScriptZero.js`.
- `bigImageZero.jpg` completes (let's assume). Parsing is blocked until `externalScriptZero.js` is done fetching and evaluating.
- `externalScriptZero.js` completes.
- Begin fetching `bigImageOne.jpg` and `reallyBigImageTwo.jpg` Simultaneously.
- `bigImageOne.jpg` completes (let's assume again). `myApp/myApp.nocache.js` begins fetching and evaluating.

# Bootstrap JS with GWT - Example Sequence

- `myApp/myApp.nocache.js` completes, and the compiled script (`<hashname>.cache.js`) begins fetching in a hidden IFRAME (this is non-blocking).
- `<hashname>.cache.js` completes. `onModuleLoad()` is not called yet, as we're still waiting on `externalScriptOne.js` to complete before the document is considered 'ready'.
- `externalScriptOne.js` completes. The document is ready, so `onModuleLoad()` fires.
- `reallyBigImageTwo.jpg` completes.
- `body.onload()` fires, in this case showing an `alert()` box.

# Bootstrap JS with GWT - Remember...

- You want to put the GWT selection script as early as possible within the body, so that it begins fetching the compiled script before other scripts (because it won't block any other script requests).
- If you are going to be fetching external images and scripts, you want to manage your two connections carefully.
- `<img>` tags are not guaranteed to be done loading when `onModuleLoad()` is called.
- `<script>` tags are guaranteed to be done loading when `onModuleLoad()` is called.

# Emulated Java Runtime Environment

# Emulated Java Runtime Environment (JRE)

- The Emulated JRE is by no means a full re-implementation of the Java JRE, but is rather a sort of selection of classes and methods that can be useful (and usable) client-side.
- The functionalities that are in the Java JRE but which you will not find inside the Emulated JRE fall into **three categories**.

# Emulated Java Runtime Environment (JRE)

1. **Things that cannot be ported client-side.** For instance, `java.lang.Thread` or `java.io.File` cannot be implemented in a browser with the same semantics of Java. The browser page is single-threaded and has no direct access to the filesystem.
2. **Things that could be implemented** but that would “cost too much” in terms of code size, performance, or dependencies, and which the community thus prefers not to have inside GWT. Included in this category, for instance, is Java reflection (`java.lang.reflect`) which would require the transpiler to keep class information for each type, and that would cause the size of the compiled JavaScript to balloon.
3. **Things nobody had interest** in and therefore have not been implemented.

# Emulated Java Runtime Environment (JRE)

- Write your own implementation using <super-source>
- List of all implemented JRE classes
  - <http://www.gwtproject.org/doc/latest/DevGuideCodingBasicsCompatibility.html>
  - <http://www.gwtproject.org/doc/latest/RefJreEmulation.html>

# Interoperability Layer to JavaScript

# Interoperability Layer to JS

- JSNI (Java Script Native Interface)
  - **Deprecated**
  - Similar to JNI (Java Native Interface)
- JsInterop (Java Script Interoperability)
  - From GWT 2.7.x
  - Annotation-based

# JSNI

- Example

```
public static native void alert(String msg) /*-{  
    $wnd.alert(msg);  
}-*/;
```

# JsInterop

- Example

```
@JsType(namespace = JsPackage.GLOBAL, name = "window", isNative = true)
public class Window {
    public static native void alert(String message);
}
```

# Tools

# Tools

- Web browser
  - Chrome with source maps support for debugging
  - FF with source maps support for debugging
- IDE
  - Eclipse / STS
    - With SDBG (Super Dev Mode Debugger) plugin: <https://sdbg.github.io>
  - IntelliJ
- Maven with GWT Maven Plugin
  - <https://troyer.github.io/gwt-maven-plugin>
- GWT Boot
  - One-stop-shop for all the GWT and related technology that you need without having to hunt through sample code and copy paste loads of dependency descriptors.

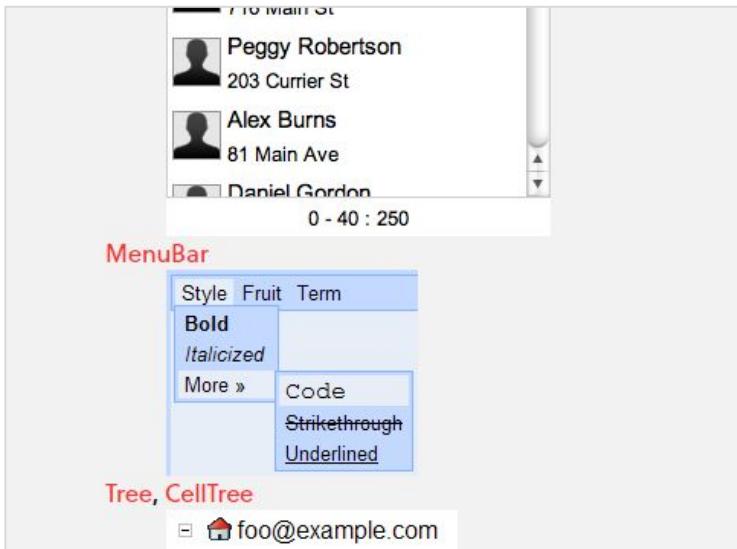
# User Interface Components

# GWT UI Components

- Lot of choices
  - Standard GWT UI included: Panels and Widgets (Button, Grid, Tree, TextField, ...)
    - Based on GWT standard UI:
    - GWTBootstrap3 (Open Source)
    - GWTMaterial (Open Source)
    - GXT (commercial)
    - ...
  - Based on new Browser Elemental (“to the metal”HTML5: DOM, WebGL, WebAudio, Shadow DOM, File API, ...)
    - DominoUI (Open Source)
    - VueGWT (Open Source)
    - GWTReact (Open Source)
    - Errai (Open Source)
    - ...

# GWT Standard UI Components

- Integrated in GWT libs
- [Widgets Dokumentation](#)



## Widget Gallery

The following are **widgets** and **panels** available in the GWT user-interface [Showcase](#) sample application.

### Widgets:

Button PushButton RadioButton CheckBox DatePicker ToggleButton TextBox PasswordTextBox  
RichTextArea FlexTable Grid CellTable CellBrowser TabBar DialogBox

### Panels:

PopupPanel StackPanel StackLayoutPanel HorizontalPanel VerticalPanel FlowPanel VerticalS  
DockLayoutPanel TabPanel TabLayoutPanel DisclosurePanel

## Widgets

### Button



### PushButton



### RadioButton

Choice 1  Choice 2 (Disabled)

### CheckBox

Normal Check  Disabled Check

### DatePicker



# GWTBootstrap3

- Widgets Dokumentation

## Tables

Cell Table

Field 1	Field 2	Field 3	Buttons
Test 0	Test 0	Test 0	<button>Click Me</button>
Test 1	Test 1	Test 1	<button>Click Me</button>
Test 2	Test 2	Test 2	<button>Click Me</button>
Test 3	Test 3	Test 3	<button>Click Me</button>
Test 4	Test 4	Test 4	<button>Click Me</button>
Test 5	Test 5	Test 5	<button>Click Me</button>
Test 6	Test 6	Test 6	<button>Click Me</button>
Test 7	Test 7	Test 7	<button>Click Me</button>
Test 8	Test 8	Test 8	<button>Click Me</button>
Test 9	Test 9	Test 9	<button>Click Me</button>

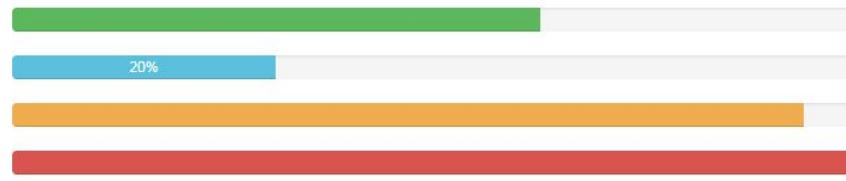
« 1 2 3 »

## Progress Bars

### Cross-browser compatibility

Progress bars use CSS3 transitions and animations to achieve some of their effects. These and below or older versions of Firefox. Opera 12 does not support animations.

### Basic



```
<b:Progress>
  <b:ProgressBar type="SUCCESS" percent="40"/>
</b:Progress>
<b:Progress>
  <b:ProgressBar type="INFO" percent="20" text="20%"/>
</b:Progress>
<b:Progress>
  <b:ProgressBar type="WARNING" percent="60"/>
</b:Progress>
<b:Progress>
  <b:ProgressBar type="DANGER" percent="80"/>
</b:Progress>
```

# DominoUI

- [Widgets Dokumentation](#)

## Domino UI demo

### Loaders

Use loaders to mask an element until some action is completed.

**BOUNCE**  
bounce loader effect.

Quis pharetra a pharetra fames blandit. Risus faucibus velit. Risus imperdiet mattis neque volutpat, etiam lacinia netus dictum magnis per facilisi sociosqu. Volutpat. Ridiculus nostra.

[CLICK ME](#)

**FACEBOOK**  
facebook loader effect

Quis pharetra a pharetra fames blandit. Risus faucibus velit. Risus imperdiet mattis neque volutpat, etiam lacinia netus dictum magnis per facilisi sociosqu. Volutpat. Ridiculus nostra.

Loading ...

[CLICK ME](#)

## Domino UI demo

### DATE PICKERS

#### INLINED

Header visible

Monday

May

14

2018

< 2018 >

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

**TODAY**

# References

- <https://dzone.com/articles/understanding-gwt-compiler>
- <http://www.gwtproject.org/doc/latest/DevGuideCompilingAndDebugging.html#DevGuideJavaToJavaScriptCompiler>
- <http://www.gwtproject.org/doc/latest/DevGuideOrganizingProjects.html#DevGuideBootstrap>
- <http://www.gwtproject.org/doc/latest/DevGuideCodeSplitting.html>
- <https://www.toptal.com/front-end/javascript-front-ends-in-java-with-gwt>
- <http://bit.ly/2InB0AW>
- <http://www.gwtproject.org/doc/latest/RefJreEmulation.html>
- <http://www.gwtproject.org/doc/latest/DevGuideCodingBasicsCompatibility.html>
- <http://www.gwtproject.org/doc/latest/DevGuideCompileReport.html>
- [http://www.gwtproject.org/articles/fragment\\_merging.html](http://www.gwtproject.org/articles/fragment_merging.html)
- <http://www.gwtproject.org/doc/latest/DevGuideCodingBasicsJSNI.html>
- <http://www.gwtproject.org/doc/latest/DevGuideCodingBasicsJsInterop.html>
- <http://www.g-widgets.com/2017/06/29/quick-tip-debugging-a-gwt-application-using-chrome-dev-tools>
- <https://www.sencha.com/products/gxt/#overview>