# The Unilingual Web

## Interactive Web Apps and Server Programming using only Java and the Google Web Toolkit
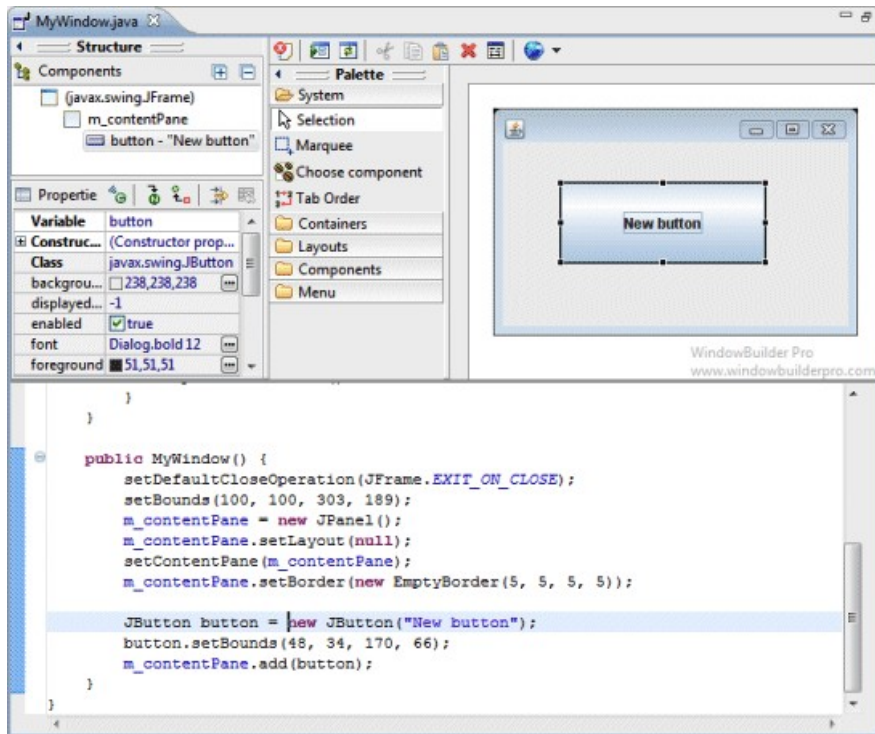
Brian Van Klaveren

# Why GWT?

"GWT's mission is to radically improve the web experience for users by enabling developers to use existing Java tools to build no-compromise AJAX for any modern browser."
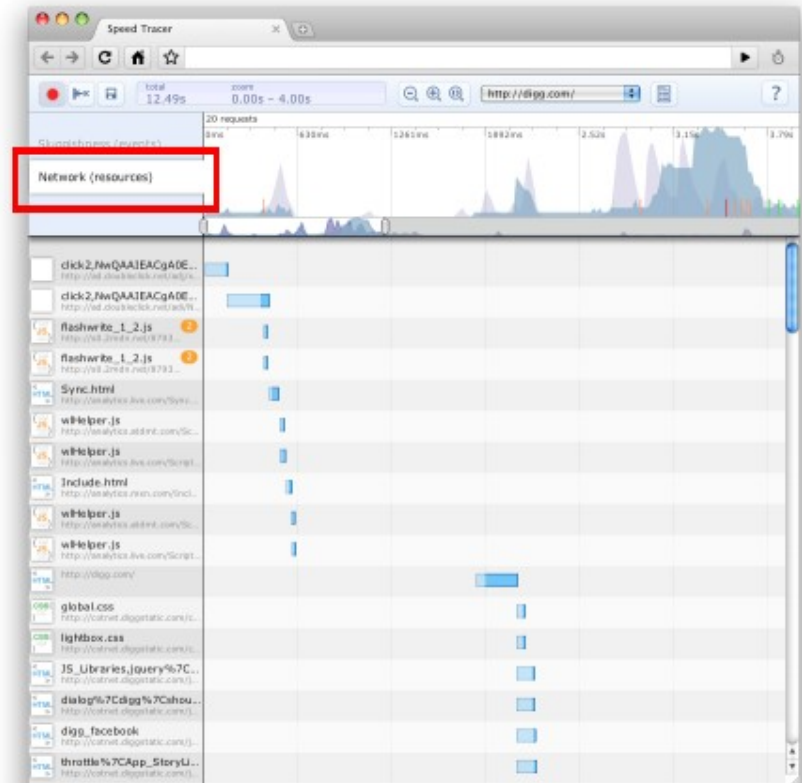
- GWT is natively Java – On both Server and Client
  - For the client, Java is compiled into JavaScript
  - You get to use standard Java dev tools, including a debugger
    - And plugins for Eclipse, NB, Maven, a UI designer, and profiler
- Server side coding, especially AJAX/RPC calls/interfaces is **simple**.
- Tons of widgets and third party libraries available
  - gwt-ext and smartgwt, for example
- Your platform target is the *modern browser*
- Apps behave more like desktop apps than web pages

SLAC
NATIONAL ACCELERATOR LABORATORY

# Additional Tools

## GWT Designer

## Speedtracer

SLAC

NATIONAL ACCELERATOR LABORATORY

# EXO DAQ GUI

- ## Application is distributed
  - ### Everyone has access
  - ### State maintained on server, dynamic updates in real time
- ## Mixed JSP/GWT
- ## Compatible with all browsers

# Fermi Datacatalog



Declarative layout using CSS, XML Better resizing, Use of space

UI modules can be loaded and unloaded on demand, without refreshing page

DnD functionality
Drag a dataset, show a plot

Dynamically loading data as needed to speed up page load time.

NATIONAL ACCELERATOR LABORATORY

# Wrapping it up

## Client Side:

- Embedding in a page means adding a script tag since code is compiled down to JavaScript.
- Basic set of widgets and elements makes development easy
- No Plugins!
- A modern browser is recommended:
  - Above FF 3.5, IE 7, and Safari 3

## Server Side:

- Generally, you will write servlets implementing your remote services to enable an interactive page
- But you don't you don't have to if you don't need AJAX
- A lot of code can be shared
  - Restricted to a subset of Java

SLAC
NATIONAL ACCELERATOR LABORATORY

# Alternatives for a unilingual web

- Python instead of Java:
  - Pyjamas (pyjs.org) is a python based implementation of a subset of GWT
  - Server side is a bit more flexible
    - Django, wsgi, mod_python, etc...
- Pure Javascript with node.js:
  - Server side javascript programming
  - Client side is whatever you want
- Both tend to require hand coding JSON/XML for serialization, however.