



After Work GWT

Mardi 25 Novembre 2008

Anthony Dahanne

anthony.dahanne@valtech.fr

<http://blog.valtech.fr>

delivering business agility

A decorative graphic at the bottom of the slide featuring overlapping, curved shapes in shades of orange and yellow, resembling a stylized leaf or a modern architectural element.

- **Présentation de GWT**
- **Mise en place de l'environnement de dév. (It0)**
- **Création d'une interface graphique (It1)**
- **Appels distants via RPC (It2)**
- **Conclusion**



- **Problème : Difficulté d'écriture du code JavaScript**

- Outillage (IDE) encore immature, débog compliqué (alert !)
- interprétation différente selon le navigateur
- manque de compétence JS



- **Solution : Générer du code JavaScript à partir de code Java**

- IDE (ex:Eclipse) nombreux et riches, débog pas à pas possible
- 1 code source pour tous les navigateurs !
- *Optimisation (code JS "compacté", génération de ce qui est utile uniquement)*
- Compétences nombreuses (!)

Présentation (2/5) : Historique

- **Projet racheté à 1 start up**
- **Créé par Bruce Johnson**
- **Proposé en Open source fin 2006**
- **Dès le début, une forte communauté, nombreuses bibliothèques dont EXT GWT, EXT, etc...**

- **Optimisation (temps de chargement réduits) au coeur de GWT :**
 - Fichiers échangés compressés avec gzip
 - Allégement de la taille des fichiers grâce aux noms raccourcis de variables et fonctions JS
 - Chargement des scripts dans une iframe cachée en tâche de fond
 - Utilisation de la mise en cache des navigateurs
 - Le navigateur ne charge que les fonctions dont il a besoin
 - Lors de la compilation, le code JS pour chacun des navigateurs est généré : le navigateur gardera en cache le code approprié

Présentation (4/5) : Des concurrents...

- **Microsoft Volta (C# , VB.NET) très en retard (moins performant)**
- **Morfik (<http://www.morfik.com>) approche orientée BDD et XML (!)**
- **GWT est aujourd'hui le framework de génération d'application web riches le plus avancé.**

Présentation (5/5) GWT ?

- **GWT est :**
 - Un outil, une librairie Java
 - Un traducteur de code Java vers Javascript
 - Un outil permettant de générer une application cliente riche (RIA)
- **GWT n'est pas :**
 - Une librairie Javascript (comme Scriptaculous ou YUI ou JQuery)
 - Un framework de développement Web (comme Struts ou JSF ou Rails)
 - Un outil pour générer un site web avec du contenu (wiki, blog, CMS, etc...)

- **Nous utiliserons :**

- Eclipse 3.3
- Une jdk Sun moderne (≥ 5.0)
- Windows ou Linux ou Mac OS
- GWT 1.5.3 (sorti le 17/10/2008)
- Tomcat (embarqué dans le projet GWT)
- Navigateur Web (Firefox et ses plug-ins comme Firebug sera bienvenue !)

Notre premier projet GWT : le Projet RH

- **Nous avons été sélectionné pour une refonte du module de recherche d'emplois**
- **Le client a entendu beaucoup de bien de GWT et a choisi cette technologie pour cette refonte**
- **Nous disposons de 3 itérations :**
 - Itération 0 : prise de connaissance avec l'outil
 - Itération 1 : création de la partie cliente
 - Itération 2 : lien avec la partie serveur

Appli Rh

ProjetRh : Iteration0

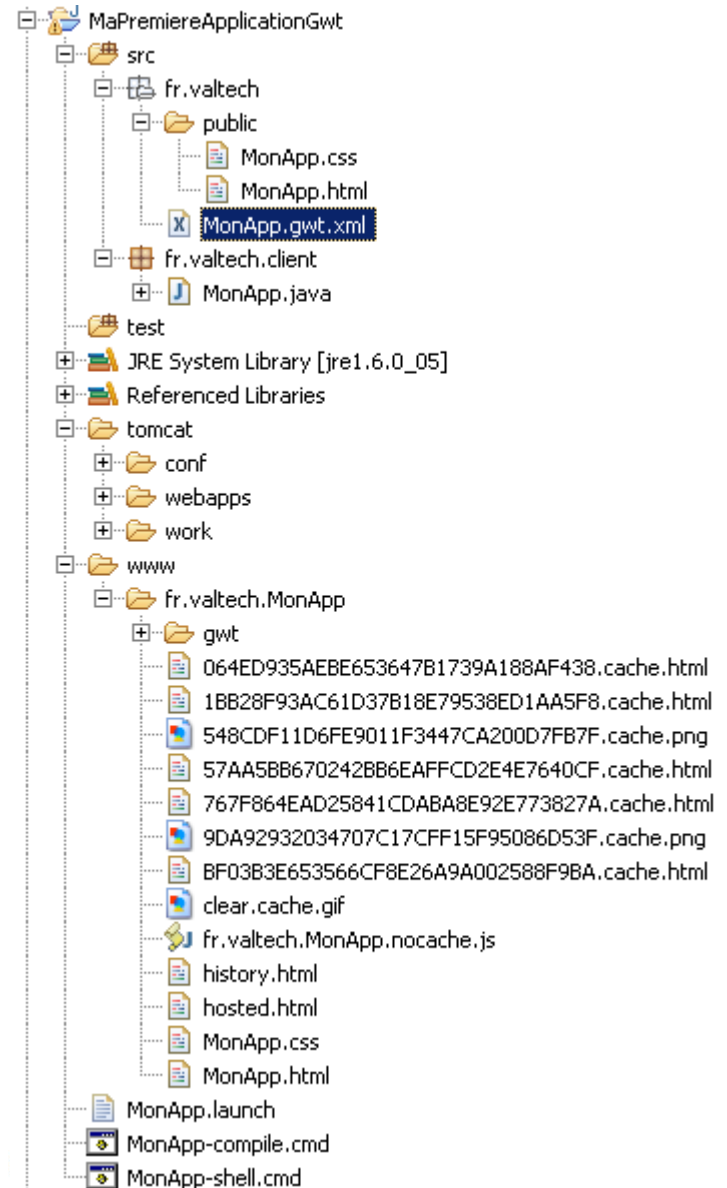
- **ProjetRh : Une application permettant aux visiteurs de consulter les offres d'emploi selon des critères**
- **Sprint Backlog de la première Itération :**
 - Création du projet avec les outils GWT
 - Intégration du projet dans Eclipse
 - Tour du propriétaire

Itération 0 : création du projet

- **Un projet GWT avec (ou sans) Eclipse en quelques commandes**
- **Création d'un répertoire contenant le projet :**
 - `>mkdir MonProjetRh puis >cd MonProjetRh`
- **Création du projet Eclipse :**
 - `>GWT_HOME\projectCreator.cmd -eclipse MonProjetRh`
- **Création de l'application :**
 - `>GWT_HOME\applicationCreator.cmd -eclipse MonProjetRh
fr.valtech.client.AppliRh`
- **Il n'y a plus qu'à lancer Eclipse et importer le projet créée !**

Itération 0 : Tour du propriétaire

- **Structure du projet :**
 - Pour lancer le projet, il suffit d'utiliser le MonProjetRh.launch et de le lancer en cliquant droit dessus "Run as Java Application"
 - La console GWT ainsi que le navigateur GWT se lancent !



- **2 modes de lancement :**
 - Hosted (ou géré) : application exécutée en tant que bytecode Java : permet de faciliter le code-compile-test-**debug** (console de debug, permet d'exploiter les points d'arrêt dans Eclipse, etc...)
 - Web : Le navigateur lit simplement le code généré par le compilateur GWT et l'interprète naturellement

- **Le module principal est représenté par un fichier XML ApplicationRh.gwt.xml , il est composé de :**
 - Un ou plusieurs modules hérités
 - Un point d'entrée (le “main”)
 - Un chemin des sources, un autre des ressources
 - Scripts JS et CSS
 - Config vers les servlets RPC
 - Etc...

Iteration 1 : Création du GUI

- **Dans cette itération, nous allons :**
 - Utiliser des widgets pour créer l'interface graphique
 - Utiliser un ImageBundle pour rassembler les images
 - Créer un tableau pour accueillir les résultats
 - Remplir dans un premier temps ce tableau (pour préparer le service GWT RPC)

Iteration 1 : Présentation des Widgets

- **Pleins de Widget : boutons, labels, cases à cocher, listes, arbres, etc...**
- **UIObject est la super classe , qui gère coordonnées, titre, visibilité et taille de tout objet graphique**
- **Les widgets héritent de la super classe Widget qui proposent en plus des opérations hiérarchiques attachement au parent, etc...)**
- **Les widgets proposent des événements basés sur des listeners Java**
 - FocusListener, MouseListener, KeyBoardListener, etc...

Iteration 1 : Présentation des Widgets

- **TextBox** : zone de saisie
- **Button** : bouton HTML
- **Label** : créer une zone de texte
- **Hyperlink** : lien “interne”
- **ListBox** : liste de valeurs
- **HTMLTable, FlexTable, Grid**
- **MenuBar** : création de menu pop-up
- **Tree** : décomposition d’une liste sous un arbre
- **SuggestBox** : interroge un oracle pour prédiction

- **HTML** : permet de placer du code HTML
- **Image** : insère une portion d'image ou une image entière
- **ImageBundle** :
 - permet d'atténuer les latences réseaux en évitant les aller et retour
 - Regroupe plusieurs images en une : en JS + CSS, les images seront chargées sous la forme de portion d'une unique grosse image
 - Nous rappelle le Keep Alive HTTP !

- **A comparer à Swing ou Windows.form**
- **En GWT, on a plutôt fusionné la notion de Layout et de Panel**
- **4 types de Panel :**
 - Simple
 - Complexe
 - Table
 - Split

Iteration 1 : Présentation des Panels

- **FocusPanel** : les éléments fils gèrent le focus
- **FormPanel** : simule soumission d'un form. HTML ?
- **PopUpPanel** et **DialogBox**
- **AbsolutePanel**
- Les enfants de **CellPanel** : **DockPanel**, **HorizontalPanel**, **VerticalPanel**
- **FlowPanel**, **StackPanel**
- **HTMLPanel** : intégration de GWT dans sites existants
- **DisclosurePanel** : une encoche démasque les fils

- **Dériver de la classe Widget ou d'une de ses sous classes**
 - Bien observer la classe DOM
 - Bien réfléchir aux événements gérés, notamment en utilisant sinkEvent, gestion unifiée des événements entre navigateurs
 - UIObject ferait l'affaire, mais ne gère pas les événements
- **Ou dériver de la classe Composite, pour contenir d'autres Widgets**

- **GWT plutôt sobre par rapport à Flex et SilverLight**
- **Les extensions rajoutent de l'AJAX (DnD, transitions, etc...)**
- **2 familles d'extensions :**
 - Les wrapper, widgets qui encapsulent du code JavaScript
 - Les natives, directement écrites en GWT, n'importent pas de code JavaScript supplémentaire (permettent l'optimisation du code, d'être paramétrées, etc...)
- **Des problèmes de licence apparaissent !**
- **GWT s'appuie sur GWT-WL, basé sur, entre autres, Scriptaculous**

Présentation d'extension GWT

- **GWT-DnD : le célèbre Glisser/Déplacer**
- **GWT-Ext, wrapper ExtJS**
- **GXT ou ExtGWT, librairie native GWT, en retard par rapport à GWT-Ext**
- **GWT Window Manager**

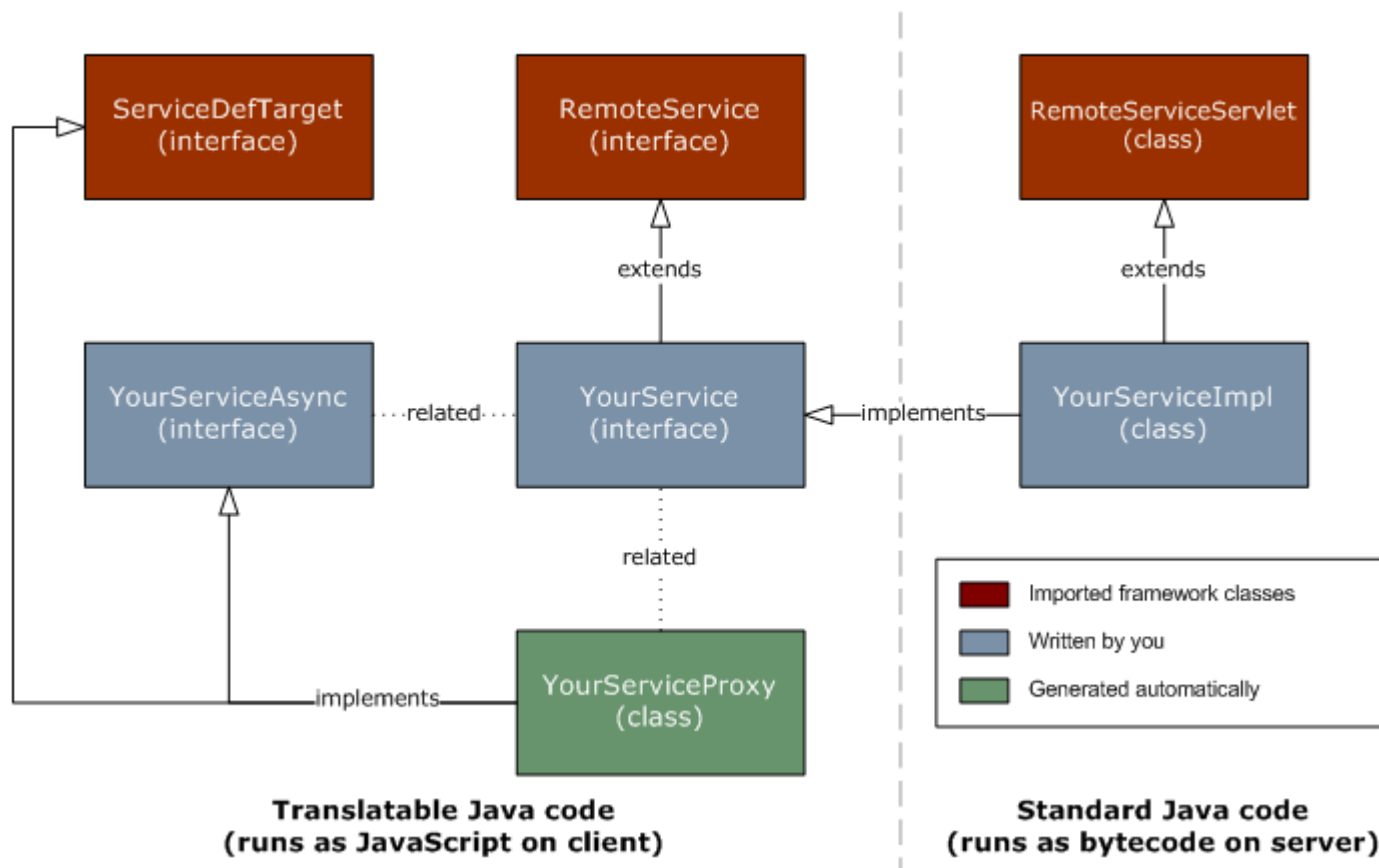
Itération 3 : appel de services distants

- **GWT apporte d'origine une fonctionnalité d'appels distants de méthode, RPC (Remote Procedure Call)**
- **Nous allons utiliser GWT RPC pour lancer nos recherches sur le serveur Tomcat embarqué dans la console GWT**

Itération 3 : Appels distants via GWT RPC

- **Appels asynchrones !**
- **Rappelle Corba !**
 - Une interface de service (signature de méthodes distantes)
 - Implémentée côté serveur
 - Une interface plus technique, représente l'interface de service mais en plus technique, faisant apparaître un AsyncCallback qui contiendra les réponses de l'appel

Diagramme de classes GWT RPC



- **Attention aux types de retour des services ! Ils doivent être sérialisables !**
- **JavaScript n'est pas aussi riche que Java !**
- **Ne pas oublier le fichier des modules XML dans lequel on explicite le service utilisé et son implémentation**

Conclusion : Des limitations...

- **GWT ne supporte pas toute l'API Java**
- **L'interpréteur JS est mono thread**
- **GWT ne supporte pas le chargement dynamique de classes**
 - `Class c = Class.forName("");`
 - `Object o = c.getInstance();`
- **Une application GWT (ou Ajax) n'est pas référencée par les moteurs de recherche**

- **I18N**

- **Deferred Binding**

- Alimentation de variables à l'exécution (runtime)
- `Imp I = GWT.create(ClassName);`
- Le compilateur a la visibilité du code lors de la compilation, il peut l'optimiser
- GWT génère le JS pour chaque navigateur, pour chaque d'implémentation de DOM.java

Mais encore , GWT c'est ...

- **Des tests unitaires, à travers GWTTestCase que l'on hérite**
- **On implémente `getModuleName()` pour récupérer son module**
- **On le teste (assertions)**

FIN

- **Mais peut être ne s'agit il que du début !**

- **Cours GWT par Sami Jaber (Valtech Training)**
 - <http://www.dng-consulting.com>
- **GWT DnD :**
 - http://code.google.com/p/gwt-dnd/wiki/GettingStartedWith2_x
- **GXT (ext GWT)**
 - <http://extjs.com/products/gxt/>
- **Ongwt.com**
- **Une appli GWT riche**
 - <http://www.queweb.com>
- **La démo de GWT**
 - <http://gwt.google.com/samples/KitchenSink/KitchenSink.html>

Licence du document

- Ce document est protégé par le contrat :
- Creative Commons Paternité-Pas d'Utilisation Commerciale-Partage des Conditions Initiales à l'Identique 2.0 France License.

