



zenika  
ARCHITECTURE INFORMATIQUE

www.zenika.com

**Durée**  
4 jours (28h)

**Répartition**  
50% théorie  
50% pratique

**Pré-requis**  
Connaissance de Java

**Public**  
Architecte  
Développeur  
Chef de projet

**Tarif (déjeuners inclus)**  
1 750 € (HT)

**Sessions 2008 - Paris**  
7 au 10 janvier  
4 au 7 février  
3 au 6 mars  
7 au 10 avril  
13 au 16 mai  
2 au 5 juin  
7 au 10 juillet  
4 au 7 août  
1 au 4 septembre  
6 au 9 octobre  
3 au 6 novembre  
1 au 4 décembre

**Intra-entreprise sur demande**

**Renseignements et inscription**  
+33(0)1.45.26.19.15  
training@zenika.com  
www.zenika.com

# GWT (Google Web Toolkit)

## Développer une application Web 2.0 avec GWT

### Objectifs

- Comprendre le fonctionnement de GWT
- Connaître les façons d'intégrer GWT aux architectures JEE
- Réaliser une application web mettant en oeuvre GWT et JEE
- Savoir appliquer les best practices

### Contenu

#### Présentation

- Au cœur du contexte Web 2.0 et des technologies AJAX
- L'originalité du Google Web Toolkit
  - Principe de la compilation Java vers JavaScript
  - Indépendance vis-à-vis des navigateurs
  - Limites du domaine: réflexion, annotations, generics
- Maturité du toolkit et pérennité
- Positionnement par rapport aux autres frameworks  
AJAX: YahooUI, Dojo, DWR, Echo2...

#### Démarrer un projet GWT sous Eclipse

- Structure d'un projet
- Notion de modules, découpage en répertoires: public, client, server, bibliothèque gwt-user.jar
- Principe et utilisation du GWTShell pour l'exécution et le debugging d'un projet Eclipse
- Paramétrage des launch configurations: spécification des répertoires de génération
- Étude de la bibliothèque gwt-user-dev.jar
- Compilation d'un projet: principe et mise en oeuvre, pièges à éviter

#### Utiliser la bibliothèque de composants graphiques

- Utilisation des widgets basiques
  - Boutons, liens, images
  - Champs de formulaires, file upload
- Mise en forme: les Panels
- Widgets complexes
  - Menus
  - Composites
  - Popups
- Principe et utilisation du modèle événementiel
- Mise en place de l'internationalisation

#### Techniques plus avancées

- Créer son propre composant graphique
- Utilisation de JSNI pour générer son propre JavaScript
- Framework de commandes: utilisation de la DeferredCommand
- Gestion de la navigation web: management de l'historique

#### Accéder à la couche services

- Architecture de type AJAX et étude de la communication RPC
- Implémentation d'un service asynchrone en GWT
- Gestion de la sérialisation: principes, héritage et polymorphisme
- Traitements des Exceptions
- Découverte du protocole JSON et utilisation avec des serveurs non-Java
- Traitement de fragments XML côté client avec les classes DOM

#### Implémenter une politique de tests

- Utilisation du GWTTestCase pour JUnit
- Fonctionnement du mode de test et limitations
- Principes et implémentation de tests asynchrones

#### Présentation de « GWT Designer »

- Installation et configuration
- Principe et fonctionnement
- Édition WYSIWYG et génération du code Java
- Fonctionnalités fournies

#### Google Web Toolkit avancé: Architecture et intégrations

#### Cadre de développement & Intégration continue

- Intégration de GWT dans un projet Eclipse WTP
- Compilation et intégration continue des composants GWT
- Intégration de Maven
- Découplage du GWTShell de son serveur d'applications Tomcat embedded
- Couplage avec le serveur d'applications choisi, fourni par WTP
- Explication du nouveau processus de debug obtenu

#### Mise en place d'une architecture J2EE complète intégrant GWT

- Mise en oeuvre d'une couche de services SOA
- Encapsulation des services GWT par Spring
- Intégration entre MVC classique provenant d'un framework (SpringMVC, Struts...) et GWT selon une approche de type portail
- Choix du développement de la couche présentation
  - Approche entièrement GWT (type client riche)
  - Approche mixte: concept d'enrichissement de la vue produite par un couple classique Contrôleur/JSP
- Mise en oeuvre du databinding
  - Illustration des best practices en termes de modèle événementiel et de value objects
  - Intégration avec le FormPanel