



Hornet

## Guide de déploiement du framework



Version 1.1 du 04/03/2013

Etat : Validé

## SUIVI DES MODIFICATIONS

Version	Rédaction	Description	Vérification	Date
1.0	A. Frigout F. Bernier- Malcoiffe	Initialisation	C. Rocheteau	04/04/12
1.1	F. Bernier- Malcoiffe	Ajout §pré-requis (Repository)		04/03/13
		<b>Document validé dans sa version 1.1</b>		

## Liste de diffusion

Organisation	Nom	Info	Commentaire	Validation
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## SOMMAIRE

<b>SUIVI DES MODIFICATIONS .....</b>	<b>2</b>
<b>LISTE DE DIFFUSION .....</b>	<b>2</b>
<b>SOMMAIRE .....</b>	<b>3</b>
<b>TABLEAUX .....</b>	<b>3</b>
<b>FIGURES .....</b>	<b>3</b>
<b>1 INTRODUCTION .....</b>	<b>5</b>
<b>2 CONSTRUCTION DES LIVRABLES .....</b>	<b>6</b>
2.1 Pré-requis.....	6
2.1.1 Logiciels.....	6
2.1.2 Repository Ivy.....	6
2.2 Hornet server.....	6
2.2.1 Description.....	6
2.2.2 Configuration Ivy.....	6
2.2.3 Construction des livrables .....	7
2.3 Hornet client.....	8
2.3.1 Description.....	8
2.3.2 Configuration Ivy.....	8
2.3.3 Construction des livrables .....	8
2.4 Hornet template.....	9
2.4.1 Description.....	9
2.4.2 Construction des livrables .....	9
<b>3 DEPLOIEMENT DES LIVRABLES .....</b>	<b>10</b>
3.1 Intégration de hornetsserver dans un projet Hornet.....	10
3.2 Arborescence de la partie cliente.....	10
3.2.1 Schéma de l'arborescence.....	10
3.2.2 Création de l'arborescence .....	10
3.2.3 Configuration du serveur.....	12
<b>4 ANNEXES.....</b>	<b>13</b>
4.1 Utilisation de umlGraph .....	13

## TABLEAUX

Aucune entrée de table d'illustration n'a été trouvée.

## FIGURES

Aucune entrée de table d'illustration n'a été trouvée.

## DOCUMENTS DE REFERENCE



Version	Titre

# 1 INTRODUCTION

Ce document précise les étapes pour la construction et le déploiement des livrables.

## 2 CONSTRUCTION DES LIVRABLES

### 2.1 PRE-REQUIS

#### 2.1.1 LOGICIELS

- Eclipse Galiléo SR1 (v 3.5.1) et supérieur
- Ant 1.7.1 et supérieur

#### 2.1.2 REPOSITORY IVY

Le repository Hornet le plus récent, utilisé par Ivy, doit être installé en local.

Créer un répertoire local qui sert de référence dans la suite du document pour la configuration des propriétés Ivy. Le chemin vers ce répertoire est représenté par la variable <REPertoire REPOSITORY>.

Récupérer l'archive contenant le repository (de la forme « Repository\_AAAAMMDD.zip »). Cette archive contient un répertoire « Repository\_AAAAMMDD », qui contient lui-même les sous-répertoires « technique », « cache », « snapshot » et « metier ».

Extraire l'archive et copier le contenu du répertoire « Repository\_AAAAMMDD » dans <REPertoire REPOSITORY>.

L'arborescence créée est la suivante :

- <REPertoire REPOSITORY>/technique
- <REPertoire REPOSITORY>/cache
- <REPertoire REPOSITORY>/snapshot
- <REPertoire REPOSITORY>/metier

### 2.2 HORNETSERVER

#### 2.2.1 DESCRIPTION

Ce livrable constitue la partie serveur de Hornet.

Il est constitué de hornetserver.

#### 2.2.2 CONFIGURATION IVY

Un environnement Ivy est inclus dans le projet Hornet. Vous devez utiliser cet environnement.

Le fichier « ivysettings.xml » permet de configurer :

- Localisation des repositories
  - fichier « ivysettings.properties »
- Accès aux repositories
- Cache
- Statuts

### 2.2.3 CONSTRUCTION DES LIVRABLES

- Pré-requis :
  - La construction des livrables se fait avec une version récente d'Eclipse,
  - Ivy doit être configuré (cf. 2.2.2 Configuration Ivy),
  - L'installation des outils de génération de graphiques pour la Javadoc (voir annexes : 4.1 Utilisation de umlGraph).
- Récupérer le projet « hornetserver »

#### A noter :

*Dans certaines configurations d'Eclipse, le build peut se lancer automatiquement et échouer. Cet échec ne remet pas en cause la reconstitution des livrables.*

- Dans ivysettings.properties, modifier les trois lignes suivantes pour indiquer le répertoire de votre workspace Eclipse

```
repository.technique.url=<REPERTOIRE WORKSPACE ECLIPSE>/Repository/technique
...
repository.cache.url=<REPERTOIRE WORKSPACE ECLIPSE>/Repository/cache
...
repository.snapshot.url=<REPERTOIRE WORKSPACE ECLIPSE>/Repository/snapshot
```

- Lancer la tâche livraison du build.xml.
  - Les fichiers sont créés dans le dossier **livrable**.
  - Des fichiers sont créés dans le repository « **technique** » du référentiel de composants d'Ivy en respectant l'arborescence utilisée :
    - Le dossier « fr.gouv.diplomatie.hornet » contient les différents composants générés.
      - Pour chaque composant, on retrouve les dossiers contenant le jar, la Javadoc dans un fichier zip, ainsi que les sources archivées.

#### A noter :

```
Project 'hornetserver' is missing required library: 'lib/compile/catalina-5.5.23.jar'   hornetserver
Build path      Build Path Problem
Project 'hornetserver' is missing required library: 'lib/compile/naming-factory-5.5.23.jar'
hornetserver      Build path      Build Path Problem
```

Si des problèmes sont remontés par Eclipse (croix rouge sur le projet) :

- Lancer la tâche generation-classpath du build.xml.

## 2.3 HORNETCLIENT

### 2.3.1 DESCRIPTION

Ce livrable constitue la partie client riche de Hornet.

Il est constitué de hornetclient.

### 2.3.2 CONFIGURATION IVY

Le fichier « ivy.xml » permet de définir les points suivants :

- Information du projet
- Plan de configuration
- Artefacts à publier
- Dépendances

Le dossier de configuration d'Ivy est « ivysettings ».

Le fichier « ivy.xml » s'appuie sur le fichier « common.xml » présent dans le dossier de configuration.

« common.xml » est utilisé par le build (tâche "getYUIAndTools")

« common.xml » est un script de gestion des dépendances du projet qui s'appuie sur les fichiers « ivysettings.xml » et « ivysettings.properties » présents dans le dossier de configuration.

Le fichier « ivysettings.xml » permet de configurer :

- Localisation des repositories
  - fichier « ivysettings.properties »
- Accès aux repositories
- Cache
- Statuts

### 2.3.3 CONSTRUCTION DES LIVRABLES

- Pré-requis :
  - La construction des livrables se fait avec une version récente d'Eclipse,
  - Ivy doit être configuré (cf.2.3.2 Configuration Ivy),
  - Dans ivysettings.properties (dans le dossier de configuration d'Ivy), modifier les lignes suivantes pour indiquer le répertoire de votre workspace Eclipse

```
repository.technique.url=<REPertoire WORKSPACE ECLIPSE>/Repository/technique
...
repository.cache.url=<REPertoire WORKSPACE ECLIPSE>/Repository/cache
...
```

- Récupérer le projet « hornetclient »
- Lancer la tâche "getYUIAndTools" du build.xml à la racine du projet.



- Les répertoires « 2in3 », « builder », « gallery » et « yui3 » sont créés. Le répertoire « zip-yui » contient les zips de ces répertoires.

#### 2.3.3.1.1 Composants

- Lancer la tâche all du build.xml du répertoire « fwk » :
  - Une fois l'exécution terminée, rafraîchir.
  - Pour chaque composant, un répertoire est créé dans « hornet/build ». Il contient les fichiers JavaScript en version normale (« hornet-NOM\_COMPOSANT.js »), minimum (« hornet-NOM\_COMPOSANT-min.js ») et debug (« hornet-NOM\_COMPOSANT-debug.js »)

NB : Le composant « config » sert à générer le fichier de configuration « hornetconfig.js »

- Lancer la tâche livraison-fwk du build.xml à la racine du projet.
  - Une fois l'exécution terminée, rafraîchir.
  - Une archive **livrable-fwk-VERSION-DATE.zip** est créée dans le répertoire « livrable ».

#### 2.3.3.1.2 Thèmes

- Lancer la tâche all du build.xml du répertoire « themes » :

Pour chaque thème, un répertoire est créé dans « themes/build ». Il contient les fichiers css en version normal (« hornet-skin-NOM\_THEME.css »), minimum (« hornet-skin-NOM\_THEME-min.css ») et le css principal (« theme.css »)

- Lancer la tâche livraison-themes du build.xml à la racine du projet.
  - Une fois l'exécution terminée, rafraîchir.
  - Une archive **livrable-themes-VERSION-DATE.zip** est créée dans le répertoire « livrable ».

## 2.4 HORNET TEMPLATE

### 2.4.1 DESCRIPTION

Ce livrable constitue la partie génération de projet de Hornet.

Il est constitué des éléments suivants :

- hornettemplate

### 2.4.2 CONSTRUCTION DES LIVRABLES

- Récupérer le projet « hornettemplate »
- Effectuer un export du projet « hornettemplate » vers un répertoire.
  - Clic droit sur le projet « Export / General / Archive File »

## 3 DEPLOIEMENT DES LIVRABLES

### 3.1 INTEGRATION DE HORNETSERVER DANS UN PROJET HORNET

Le fichier de configuration du framework Ivy, pour une application, permet de définir les dépendances vers le framework hornetserver.

Ce fichier de configuration ivy.xml implémente des dépendances vers le framework hornetserver avec les profils définis dans la configuration.

```
<dependencies>
  <dependency org="fr.gouv.diplomatie.hornet"
    name="hornetserver-all"
    rev="1.0.0"
    conf="compile->core,libDependances;runtime->runtime;test->test"
    transitive="true" />
</dependencies>
```

### 3.2 ARBORESCENCE DE LA PARTIE CLIENTE

#### 3.2.1 SCHEMA DE L'ARBORESCENCE

L'arborescence finale attendue est la suivante :

(Le répertoire parent sera appelé répertoire [PARENT])

- [PARENT]

- hornetclient
  - version
    - fmk
    - themes
      - francediplo
      - diplomnet
      - default
- yui
  - yui
    - version
  - 2in3
    - version
  - gallery
    - version

#### 3.2.2 CREATION DE L'ARBORESCENCE

Rappel : les zips de hornetclient précédemment créés sont :

- **livrable-fwk-VERSION-DATE.zip**

- **livrable-themes-VERSION-DATE.zip**
- **2in3-VERSION.zip**
- **gallery-VERSION.zip**
- **yui-VERSION.zip**

### 3.2.2.1 PRE-REQUIS

Avoir l'arborescence suivante :

- [PARENT]
  - hornetclient
  - yui
    - yui
    - 2in3
    - gallery

### 3.2.2.2 HORNETCLIENT

Dans le répertoire [PARENT]/hornetclient, créer le répertoire qui correspond au numéro de version [VERSION] (ex : « 1.0.0 »).

#### 3.2.2.2.1 Arborescence des composants

- Dans le répertoire [PARENT]/hornetclient/[VERSION], décompresser le contenu de l'archive **livrable-fwk-VERSION-DATE.zip**

NB : Le composant « hornetconfig », à intégrer dans chaque page web, contient la configuration avec des chemins d'accès relatifs vers les répertoires des thèmes et de YUI.

#### 3.2.2.2.2 Arborescence des thèmes

- Dans le répertoire [PARENT]/hornetclient/[VERSION], décompresser le contenu de l'archive **livrable-themes-VERSION-DATE.zip**

### 3.2.2.3 YUI (LIBRAIRIES)

#### 3.2.2.3.1.1 YUI

- Dans le répertoire [PARENT]/yui/yui, le répertoire qui correspond au numéro de version [VERSION] (ex : « 3.4.1 ») doit être présent. Sinon, le créer.
  - Dans ce répertoire, décompresser le contenu de l'archive **yui-VERSION.zip**.

#### 3.2.2.3.1.2 gallery

- Dans le répertoire [PARENT]/yui/gallery, le répertoire qui correspond au numéro de version [VERSION] (ex : « 2011.06.29-23-18 ») doit être présent. Sinon, le créer.
  - Dans ce répertoire, décompresser le contenu de l'archive **gallery-VERSION.zip**.

#### 3.2.2.3.1.3 2in3

- Dans le répertoire [PARENT]/yui/2in3, le répertoire qui correspond au numéro de version [VERSION] (ex : « 2.9.0 ») doit être présent. Sinon, le créer.
  - Dans ce répertoire, décompresser le contenu de l'archive **2in3-VERSION.zip**.

### 3.2.3 CONFIGURATION DU SERVEUR

La configuration du serveur Apache qui suit permet d'utiliser le cache des navigateurs pour les images et les fichiers YUI.

1. Activer le module expires (cache)

Ajouter ou décommenter la ligne suivante :

```
LoadModule expires_module modules/mod_expires.so
```

2. Créer un fichier hornet-cache.conf (dans le répertoire conf/).

Le fichier contient:

```
<Location ~ "/PARENT/" >  
ExpiresActive On  
ExpiresDefault A15000000  
</Location>
```

Où PARENT est le répertoire parent qui contient les fichiers. .

3. Indiquer à Apache de charger la configuration créée en ajoutant la ligne suivante à la fin du httpd.conf :

```
Include conf/hornet-cache.conf
```

## 4 ANNEXES

### 4.1 UTILISATION DE UMLGRAPH

Objectifs	Installer GraphVIZ
Entrées	Installateur de graphVIZ ( <a href="#">graphviz-2.24.msi</a> )
Sorties	Exécutable « dot » dans le path

Procédure (testée avec la version 2.24):

- Ouvrir l'exécutable graphviz-2.24.msi
- Suivre l'assistant en cliquant 3 fois sur « next », puis close.
- Lancer une invite de commande et taper :

```
dot
```

- Puis valider, la commande doit être connue.

**FIN DU DOCUMENT**