

Table des matières

Table des matières	1
I Manuel utilisateur	3
1 Introduction	5
2 Mise en place du proxy	7
2.1 Composition du répertoire Proxy	7
2.1.1 Compilation des sources	7
2.2 Configuration du proxy	7
2.2.1 Mise en place d'une plate-forme	8
3 Utilisation	11
3.1 Utilisation	11
3.1.1 Préparation du lancement	11
3.1.2 Lancement	11
3.1.3 Commandes disponibles lors de l'utilisation	11
4 Débuguage	13
4.1 Utilisation du traceur	13
II Annexes	15
A Exemple de fichier de configuration	17



Première partie
Manuel utilisateur

Chapitre 1

Introduction

Le proxy assure l'interconnection des plate-formes robotique de l'INRIA avec le monde extérieur à savoir les différents clients. Il assure la diffusion des données des serveur vers les clients. il est générique et peut supporter plusieurs plateformes simultanément. Pour plus d'informations veuillez consulter son document de conception.



Chapitre 2

Mise en place du proxy

2.1 Composition du répertoire Proxy

Le proxy doit contenir 4 sous-répertoires qui sont

- package/ qui contient le fichier manifest pour la génération du JAR
- lib/ qui contient les librairies nécessaires au proxy.
- doc/ qui la documentation doxygen.
- src/ qui contient les fichiers sources.

il contient aussi un fichier proxy.sh et proxy.bat ; ces deux fichiers sont des scripts pour simplifier la compilation.

2.1.1 Compilation des sources

Pour compiler sous linux il faut taper ./proxy.sh qui utilise ANT.

Pour compiler sous windows il faut lancer proxy.bat.

L'utilitaire ANT crée alors un répertoire install qui contient les classes dans /classes, les librairies dans /lib et proxy.jar un jar executable via la commande java -jar proxy.jar "chemin complet du fichier de configuration". Il existe aussi d'autres possibilités avec le script pour les découvrir faites /proxy.sh usage.

2.2 Configuration du proxy

La configuration du proxy se fait grâce au fichier ConfProxy.xml qui se trouve par défaut dans le répertoire /install/conf/. Ce fichier contient les informations nécessaires à la prise en charge des plates-formes sur le proxy.

Le fichier se compose d'une partie racine :

```
<JavaXML:Proxy-config xmlns:JavaXML="http://www.oreilly.com/catalog/javaxml/">
```



```
...  
</JavaXML:Proxy-config>
```

qui contient une zone de définition des plates-formes :

```
<JavaXML:plateformesConfig>  
...  
</JavaXML:plateformesConfig>
```

2.2.1 Mise en place d'une plate-forme

Pour mettre en place une plate-forme sur le proxy, voici les informations nécessaires :

- le nom de la plate-forme.
- l'IP du serveur de la plate-forme.
- le port du serveur de la plate-forme.
- le port de connexion des clients sur le proxy.(libre de choix)
- la liste des identifiants des messages générés par la plate-forme.(libre de choix)

Paramètres principaux

Voici la syntaxe à respecter pour renseigner le fichier.

Pour définir une plate-forme on utilise les balises suivantes

```
<JavaXML:plateformeConfig>  
...  
</JavaXML:plateformeConfig>
```

Pour renseigner les différentes informations :

```
<JavaXML:identifiant>BIP</JavaXML:identifiant>  
<JavaXML:portClient>7032</JavaXML:portClient>  
<JavaXML:IPPlateforme>194.199.21.187</JavaXML:IPPlateforme>  
<JavaXML:portPlateforme>7031</JavaXML:portPlateforme>
```

Listes des messages générés par la plate-forme

Le proxy se base sur un système d'abonnements à des messages générés par la plate-forme physique. il faut donc définir dans la configuration de la plate-forme la liste des identifiants des messages. Tous les messages envoyés par la plate-forme sur le proxy doivent commencer par un identifiant connu du proxy. Le proxy l'utilise pour rediriger le message aux clients abonnés. Pour s'abonner les clients doivent aussi avoir connaissance de ces identifiants.

Définition des identifiants de messages :



```
<JavaXML:requestlist>  
  <JavaXML:request_data>TM</JavaXML:request_data>  
<JavaXML:request_data>DREAM</JavaXML:request_data>  
<JavaXML:request_ctrl>CMD</JavaXML:request_ctrl>  
</JavaXML:requestlist>
```

Vous remarquerez qu'il existe deux types d'identifiants le *requestdata* et le *requestctrl*

Les *request data* sont des messages unidirectionnels du serveur vers le client. Il peut y avoir autant de messages que l'on veut et le proxy ne connaît pas leurs contenus ni leurs formes. Il incombe au programmeur des parties clients et serveur de définir une structure commune.

Le *request ctrl* est un message particulier qui s'adresse s'il existe au client de contrôle d'exécution. Le proxy doit connaître cet identifiant pour des traitements internes gérant le contrôle d'accès des clients à cette fonctionnalité.



Chapitre 3

Utilisation

3.1 Utilisation

3.1.1 Préparation du lancement

Avant de lancer le proxy il faut s'assurer que les plates-formes sont lancées, que le proxy est compilé.

3.1.2 Lancement

Ensuite pour lancer le proxy il faut taper la commande

```
java -jar proxy.jar -classpath ../install/lib
```

ou encore utiliser le script avec la commande `./proxy.sh run`.

Par défaut, le proxy cherche le fichier de configuration dans `/install/conf/Confproxy.xml` pour modifier le chemin vous pouvez utiliser :

- `java -jar proxy.jar -classpath ../install/lib [lechemin]`
- `./proxy.sh run -Dconf=lechemin`

attention le chemin ne doit pas être relatif.

3.1.3 Commandes disponibles lors de l'utilisation

Pendant l'utilisation, vous pouvez :

- démarrer le proxy avec la commande `start`
- arrêter le proxy avec la commande `stop`
- obtenir des infos sur l'exécution avec la commande `info`
- tout arrêter avec la commande `quit`



Chapitre 4

Débugage

4.1 Utilisation du traceur

Pour pouvoir faire du déboguage, il est possible d'activer un tracage en modifiant le fichier `proxy/traceur/Traceur.java` dans les sources. Pour obtenir le tracage décommentez la fonction `debug`. Cette action affiche différents messages qui aident au debug parmi-eux :

- Les messages d'erreurs
- Le suivit des abonement desabonement
- Tracage des fonctions utilisées



Deuxième partie

Annexes

Annexe A

Fichier exemple de configuration

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<JavaXML:Proxy-config xmlns:JavaXML="http://www.oreilly.com/catalog/javaxml/">

  <!-- Données générales sur le proxy -->

  <JavaXML:version>1.0</JavaXML:version>
  <JavaXML:hostname>AKILA</JavaXML:hostname>
  <!-- Données de configuration des plateformes -->
  <JavaXML:plateformesConfig>
    <!-- Données de configuration de la plateforme bip -->

    <JavaXML:plateformeConfig>
      <JavaXML:identifiant>BIP</JavaXML:identifiant>
      <JavaXML:portClient>7032</JavaXML:portClient>
      <JavaXML:IPPlateforme>194.199.21.187</JavaXML:IPPlateforme>
      <JavaXML:portPlateforme>7031</JavaXML:portPlateforme>
      <JavaXML:timeout>10</JavaXML:timeout>
      <JavaXML:requestlist>
        <JavaXML:request_data>TM</JavaXML:request_data>
        <JavaXML:request_data>DREAM</JavaXML:request_data>
        <JavaXML:request_ctrl>CMD</JavaXML:request_ctrl>

      </JavaXML:requestlist>
    </JavaXML:plateformeConfig>
    <!-- Données de configuration de la plateforme RX90 -->
    <JavaXML:plateformeConfig>
      <JavaXML:identifiant>RX90</JavaXML:identifiant>
      <JavaXML:portClient>5006</JavaXML:portClient>
      <JavaXML:IPPlateforme>194.199.21.187</JavaXML:IPPlateforme>
      <JavaXML:portPlateforme>5000</JavaXML:portPlateforme>
      <JavaXML:timeout>10</JavaXML:timeout>
      <JavaXML:requestlist>
        <JavaXML:request_data>TM</JavaXML:request_data>
        <JavaXML:request_data>DREAM</JavaXML:request_data>
        <JavaXML:request_ctrl>CMD</JavaXML:request_ctrl>

      </JavaXML:requestlist>
    </JavaXML:plateformeConfig>
  </JavaXML:plateformesConfig>
</JavaXML:Proxy-config>
```

