VERSION 2.6
DATE Mai 2011

Table des matières

Chapitre 1

Deploying a DIET platform

Deployment is the process of launching a DIET platform including agents and servers. For DIET, this process includes writing configuration files for each element and launching the elements in the correct hierarchical order. There are three primary ways to deploy DIET.

Launching **by hand** is a reasonable way to deploy DIET for small-scale testing and verification. This chapter explains the necessary services, how to write DIET configuration files, and in what order DIET elements should be launched. See Section **??** for details.

GoDIET is a Java-based tool for automatic DIET deployment that manages configuration file creation, staging of files, launch of elements, monitoring and reporting on launch success, and process cleanup when the DIET deployment is no longer needed. See Section 1.1 for details.

Writing your own scripts is a surprisingly popular approach. This approach often looks easy initially, but can sometimes take much, much longer than you predict as there are many complexities to manage. Learn GoDIET– it will save you time !

1.1 GoDIET

GoDIET is an cross-platform tool that helps you automate ad-hoc deployment and management procedures for DIET infrastructure. It manages configuration file creation, staging of files, launch of elements, monitoring and reporting. GoDIET is extremely useful for large deployments on a complex physical infrastructure. The mains features are :

- Complete command line interface.
- Distributed command execution via SSH.
- Real time monitoring applications state.
- Complex physical infrastructure management with firewall and multiple network lan.

1.1.1 Installing GoDIET

The following operating systems are known to support GoDiet :

- Linux : Most recent distributions are likely to work
- Mac OS X 10.4 or later

You need to have the Sun Java 6 or OpenJDK6 installed. Download GoDIET on the project website [1].

1. http ://graal.ens-lyon.fr/DIET/godiet.html

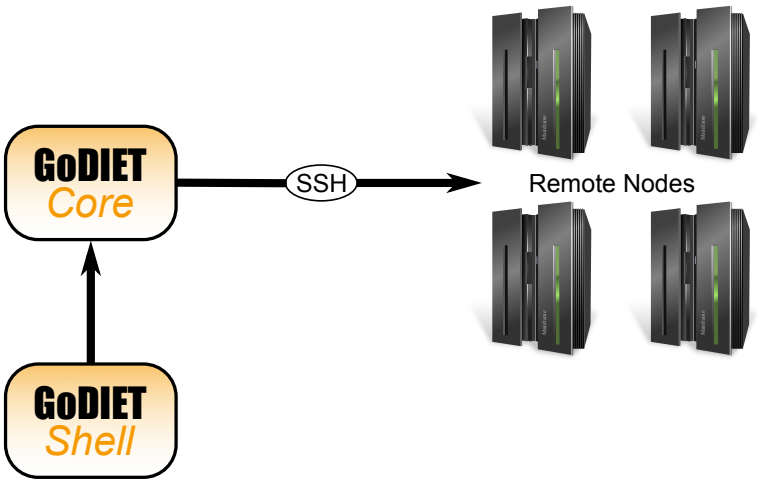

FIGURE 1.1 – Design principle of GODIET.

Extract the archive just and launch run.bat or run.sh script. You need to load your physical platform on which will GODIET will running.

1.1.2 GODIET setup

Avant de pouvoir utiliser GODIET, vous devez créer un fichier de configuration, un fichier d'infrastructure, et un fichier de déploiement de votre plateforme Diet. Ces fichiers sont au format XML et doivent repecter une grammaire, respectivemment Configuration.xsd, Infrastructure.xsd et Diet.xsd. Des examples de fichiers se trouvent dans le répertoire examples du projet.

Configuration

Dans ce fichier doit être obligatoirement setter depuis quel noeud de l'infrastructure est lancé GODIET. En effet, il est **obligatoire** de lancer GODIET depuis une ressource décrite dans le fichier d'infrastructure?? 1.1.3

Par défaut GODIET cherche le fichier dans le répertoire `${HOME}/.godiet/configuration.xml`. Si ce fichier est absent il charge un fichier par défaut qui ne spécifie pas de clef ssh et dont l'hôte source à pour nom `local` (et votre description d'infrastructure devra déclarer une resource du nom `local`).

General configuration description layout (some parts are omitted) :

```
<configuration schema="Configuration.xsd">
<goDietConfiguration localNode="local">
<localscratch dir="/tmp/scratch_godiet" />
  <user>
<ssh>
<key path="/home/.ssh/id_dsa"/>
<key path="/home/.ssh/admin_cluster2"/>
</ssh>
   </user>
</goDietConfiguration>
```

Infrastructure

GoDiet needs to have the description of infrastructure on which Diet will be running. Cette description doit être fait dans un fichier au format XML qui listera tous les éléments physiques

disk scratch space available at each site for storage of configuration files, and which machines share the same disk to avoid unecessary copies.

General infrastructure description layout :

```
<infrastructure schema="Infrastructure.xsd">
 <domain id="idDomain1">
 ..
    <storage id="idStorage">..</storage>
    <node id="idNode">..</node>
    <gateway id="idGateway1">..</gateway>

 </domain id="idDomain2">
 ..
   <gateway id="idGateway2"/>
 ..
 </domain>
 <link from="idGateway1" to="idGateway2"/>
</infrastructure>
```

Un domaine définit un ensemble de machines qui peuvent communiquer directement (au sens IP). Dans le cas où des machines sont séparer par un dispositif de filtrage (mur en feu ou autres), il faut les décrire dans deux domaines différents et GoDiet s'occupera d'instancier automatiquement les forwarders indispensables aux communications Diet.

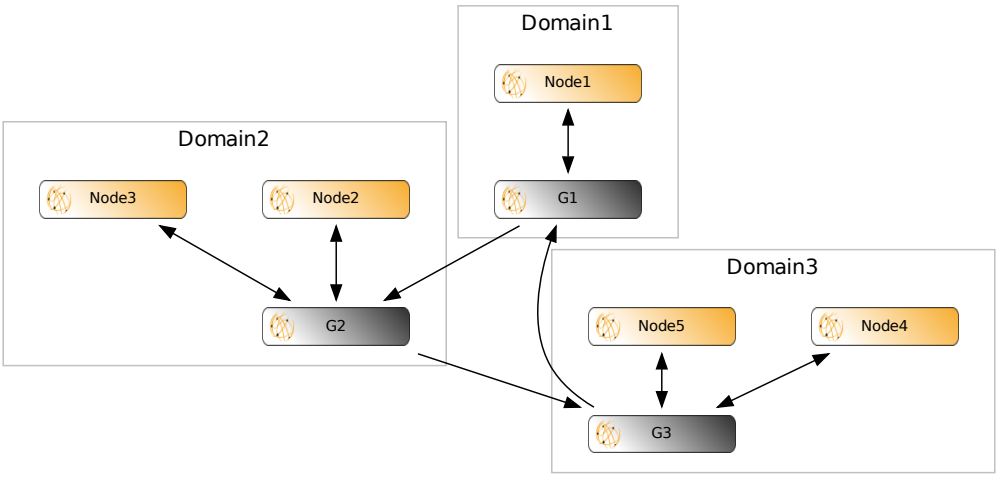

Figure 1.2 – Three domains infrastructure representation example (godiet-

Diet platform

The user of GoDiet could describes the desired deployment in an XML file including all needed external services (*e.g.*, omniNames and LogService) ; the desired hierarchical organization of agents and servers is expressed directly using the hierarchical organization of XML.

1.1.3 Godiet shell

Completion historique coloration syntaxique le fichier de configuration est chargé automatiquement depuis le répertoire home/.godiet.

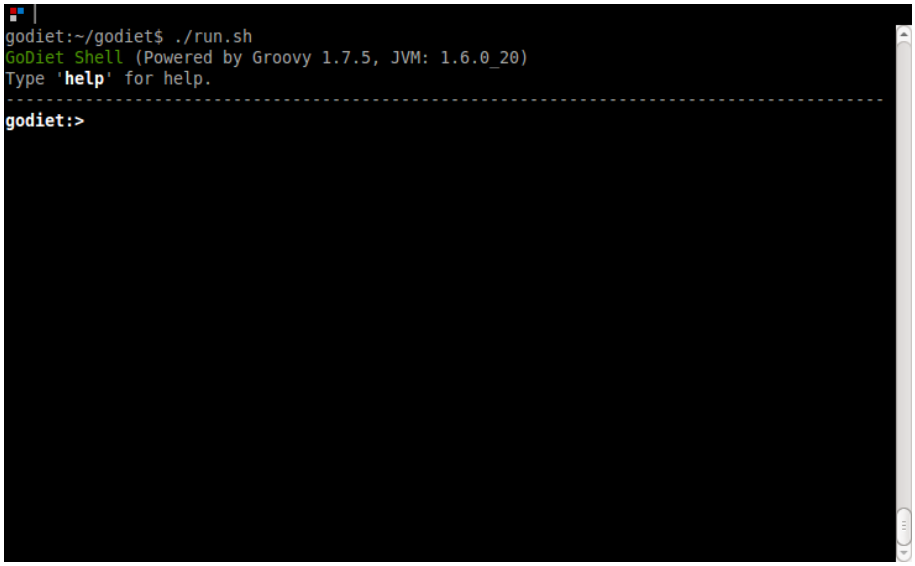

Figure 1.3 – GoDiet shell startup.

Help command

SSH command

Load commands

LoadInfrastructure permet de charger un fichier de description d'infrastructure
calcule et création automatique des Forwarders

Start & Stop commands

Status command

Status command options :
- **ma** : Display the masters agents status
- **la** : Display the locals agents status
- **seds** : Display the servers daemons status
- **all** : Display all diet softwares status

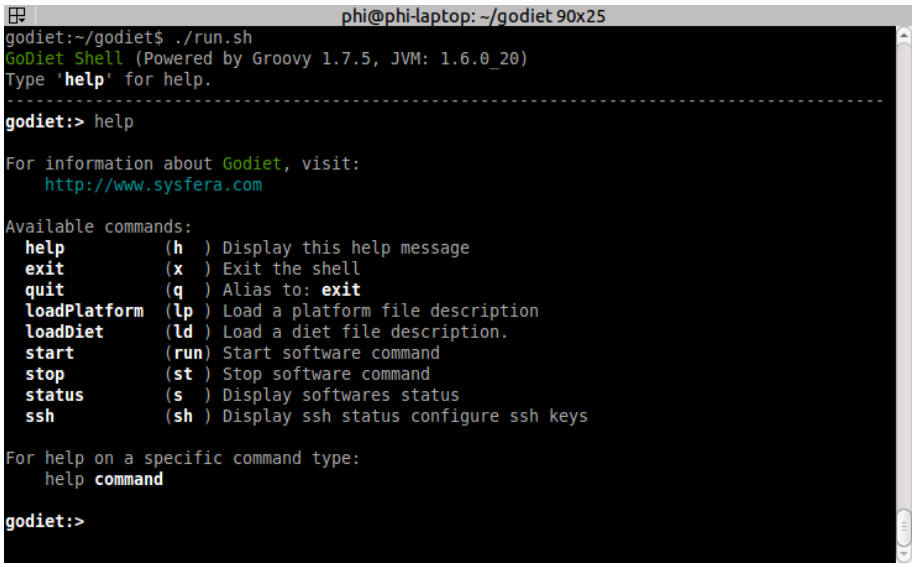

FIGURE 1.4 – The help command.

La commande status affiche sous forme de tableau l'état des éléments gérés pas GoDiet. Un exemple d'execution est affiché sur la figure 1.5. De gauche à droite les informations affichés sont :

- Le nom du logiciel comme décrit dans le fichier de description d'infrastructure.
- Son état et depuis quand il s'y trouve.
- La machine sur laquelle le logiciel va être executé ou est actuellement executé (selon son état).
- Le message d'information dans le cas où la ressource est dans l'état erreur.

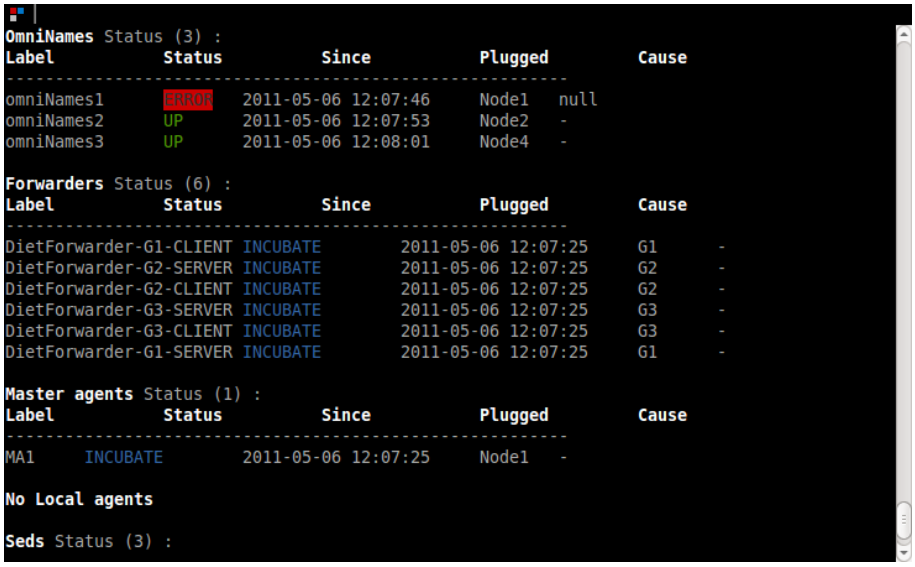

FIGURE 1.5 – The 'status all' command.

Draw command

Draw command options :
- **infrastructure** : Draw the infrastructure
- **platform** : Draw the platform infrastructure

Génère un fichier au format Dot. La figure1.2 est le resultat de généré avec cette commande