

VERSION 2.6
DATE Mai 2011
Copyright SysFera

Table des matières

1	Deploying a DIET platform	5
1.1	GoDIET	5
1.1.1	Installing GoDIET	5
1.1.2	GoDIET setup	6
1.1.3	Godiet shell	7

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 website¹. Extract the archive just and launch run.bat or run.sh script. You need to load your

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

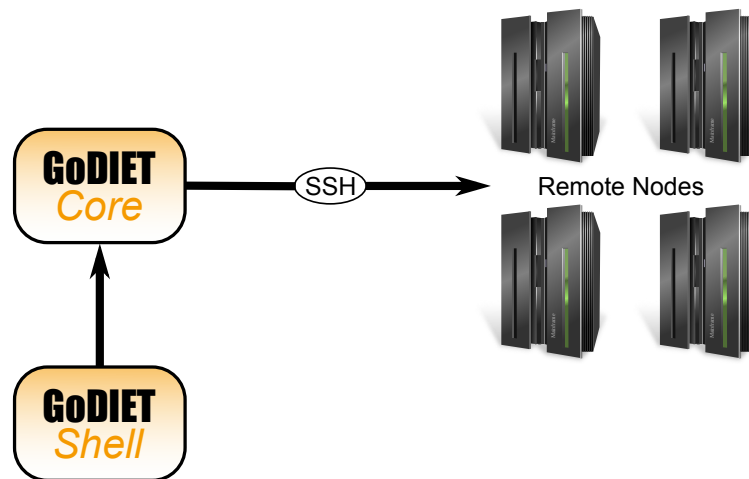


FIGURE 1.1 – Design principle of GoDIET.

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 respecter une grammaire, respectivement Configuration.xsd, Infrastructure.xsd et Diet.xsd. Des exemples de fichiers se trouvent dans le répertoire exemples 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 ressource 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. Il listera tous les éléments physiques indispensables au fonctionnement et au déploiement de DIET.

Les principaux éléments sont :

- **Domain** : définit un ensemble d'éléments (node ou gateway) qui peuvent communiquer directement (au sens IP). Dans le cas où des machines sont séparées par un dispositif de filtrage (mur en feu ou autres), il faut les décrire dans deux domaines différents
- **Storage** : définit un espace de stockage distant.
- **Node** : définit une machine physique sur laquelle sera exécuter les logiciels, typiquement les seds et les agents.
- **Gateway** : définit une passerelle réseau. Elle permet de créer les interconnexions entre les différents domaines.
- **Link** : Définit un lien directionnel par lequel deux passerelles pourront communiquer grâce au protocole SSH.

General infrastructure description layout :

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

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

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`.

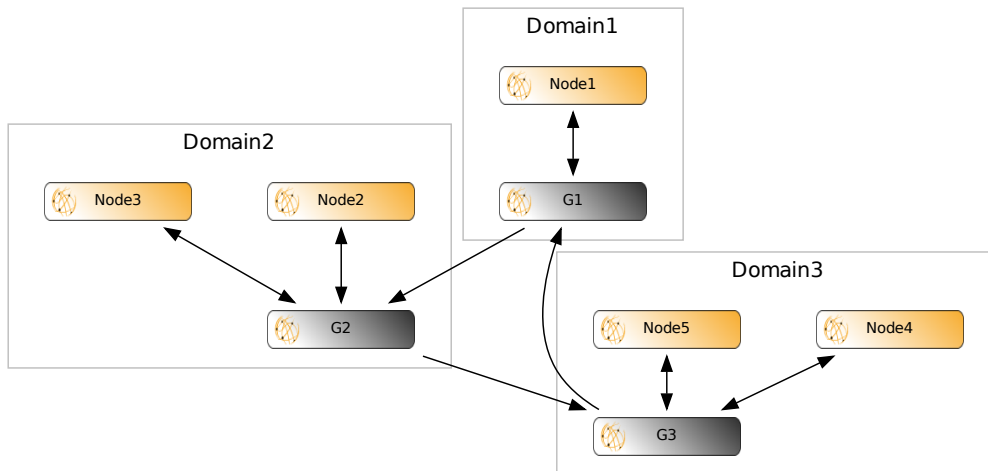


FIGURE 1.2 – Three domains infrastructure representation example

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

start command Start command options :

- **software** SoftwareName : Start one software given his name.
- **services** : Start all services
- **agents** : Start all agents
- **seeds** : Start all seeds
- **all** : Start all softwares register

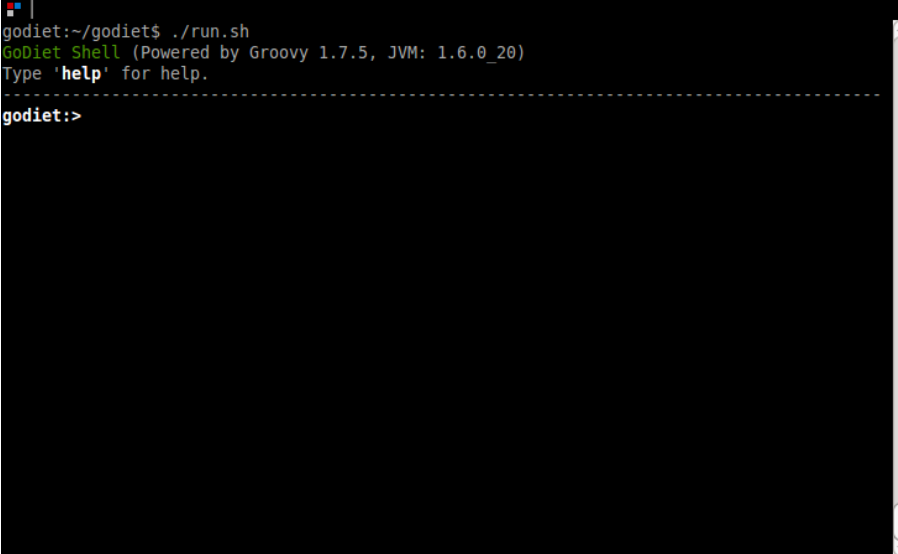
stop command takes the same parameters.

Status command

Status command options :

- **ma** : Display the masters agents status
- **la** : Display the locals agents status
- **seeds** : Display the servers daemons status
- **all** : Display all diet softwares status

La commande status affiche sous forme de tableau l'état des éléments gérés par GODIET. Un exemple d'exécution est affiché sur la figure 1.5. De gauche à droite les informations affichés



```
godiet:~/godiet$ ./run.sh
GoDiet Shell (Powered by Groovy 1.7.5, JVM: 1.6.0_20)
Type 'help' for help.
-----
godiet:>
```

FIGURE 1.3 – GoDIET shell startup.

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 exécuté ou est actuellement exécuté (selon son état).
- Le message d’information dans le cas où la ressource est dans l’état erreur.

Les différents états possibles sont

- **Incubate** : The component is correctly loaded in GoDIET.
- **Ready** : The configurations file are created in the local scratch directory and component is ready to start.
- **Up** : The component is Up. His state is periodically checked.
- **Down** : The component is Down. Typically if GoDIET user call stop on this component.
- **Error** : Could appears if there is an error in the description, if the

Draw command

Draw command options :

- **infrastructure** : Draw the infrastructure
- **platform** : Draw the platform infrastructure

Génère un fichier au format Dot. La figure 1.2 est le résultat de son exécution sur l’infrastructure décrite dans le fichier `godiet-infrastructure.xml`

```

phi@phi-laptop: ~/godiet 90x25
godiet:~/godiet$ ./run.sh
GoDiet Shell (Powered by Groovy 1.7.5, JVM: 1.6.0_20)
Type 'help' for help.

-----
godiet:> help

For information about Godiet, visit:
  http://www.sysfera.com

Available commands:
  help      (h ) Display this help message
  exit      (x ) Exit the shell
  quit      (q ) Alias to: exit
  loadPlatform (lp ) Load a platform file description
  loadDiet   (ld ) Load a diet file description.
  start      (run) Start software command
  stop       (st ) Stop software command
  status     (s ) Display softwares status
  ssh       (sh ) Display ssh status configure ssh keys

For help on a specific command type:
  help command

godiet:>

```

FIGURE 1.4 – The help command.

```

OmniNames Status (3) :
Label      Status      Since      Plugged      Cause
-----
omniNames1 ERROR      2011-05-06 12:07:46 Node1 null
omniNames2 UP        2011-05-06 12:07:53 Node2 -
omniNames3 UP        2011-05-06 12:08:01 Node4 -

Forwarders Status (6) :
Label      Status      Since      Plugged      Cause
-----
DietForwarder-G1-CLIENT INCUBATE      2011-05-06 12:07:25 G1 -
DietForwarder-G2-SERVER INCUBATE      2011-05-06 12:07:25 G2 -
DietForwarder-G2-CLIENT INCUBATE      2011-05-06 12:07:25 G2 -
DietForwarder-G3-SERVER INCUBATE      2011-05-06 12:07:25 G3 -
DietForwarder-G3-CLIENT INCUBATE      2011-05-06 12:07:25 G3 -
DietForwarder-G1-SERVER INCUBATE      2011-05-06 12:07:25 G1 -

Master agents Status (1) :
Label      Status      Since      Plugged      Cause
-----
MA1        INCUBATE      2011-05-06 12:07:25 Node1 -

No Local agents

Seds Status (3) :

```

FIGURE 1.5 – The 'status all' command.