# Nginxlib User Manual

Version 0.1 Alpha

by Mikes

yang1.zhang@cs2c.com.cn

http://blog.csdn.net/mikeszhang

Nginxlib is a java library for nginx server management. Nginxlib 0.1 is compatiable with nginx 1.4.*. Nginxlib depends on Orion-SSH 2 library.

Now the nginxlib uses will be intruduced by protical codes. All the library mian functions: monitor, configuration and control will be included.

## Get a library instance

The core of the nginxlib is the MiddlewareFactory class. One of the class instances bind with a specific remote nginx. After we get the instance, we can get to monitor, control or configure the nginx through interfaces provided by the MiddlewareFactory instance.

```
 import org.cs2c.nginlib.*;
1
2
3AuthInfo authInfo=MiddlewareFactory.newAuthInfo();
4authInfo.setHost("10.1.50.4");
5authInfo.setUsername("git");
6authInfo.setPassword("qwer1234");
7try{
8    MiddlewareFactory middleware=MiddlewareFactory.getInstance(authInfo
9, "/usr/local/nginx/");
10}
11catch(RemoteException e){
12    e.printStackTrace();
13}
```

On above code fragement, we must provide necessary authentication information and nginx home path in order to get the Middleware instance.

When you get the instance successfully, you can further get the necessary interfaces you need to operate the binding nginx.

```
1import org.cs2c.nginlib.*;
2import org.cs2c.nginlib.config.*;
3import org.cs2c.nginlib.ctl.*;
4import org.cs2c.nginlib.monitor.*;
5
6
7MiddlewareFactory middleware=MiddlewareFactory.getInstance(authInfo, "
8/usr/local/nginx/");
9/**Get the monitor for the remote nginx*/
10Monitor monitor=middleware.getMonitor();
11/**Get the configurator for configuring the remote nginx*/
```

```
Configurator conf=middleware.getConfigurator();
/**Get the controller for control the remote nginx*/
Controller ctl=middleware.getController();
```

## Control the nginx

When you get the Controller interface instance, you can use the rich fuctions in the interface to programmly control the remote nginx. The functions includes:

- Start: start the remote nginx.
- Restart: Restart the remote nginx.
- Reload: take effect the remote nginx configuration file, but not restart it.
- shutdown: shutdown the remote nginx processes.
- deploy: deploy web site file to the specific remote position. The file is in format of zip.

If any of the above methods executing fails, the RemoteException will be threw.

## Monitor the nginx

The interface of monitor provides two main functions, server monitor and nginx monitor. The server monitor function includes:

- CPUStatus: get the realtime cpu information of remote server.
- IOStatus: get the realtime io information of remote server, including any device.
- MemoryStatus: get the realtime memory and swap usage information of remote server.
- NetworkStatus: get the network transfer ratio of the remote server.

The nginx realtime status can also be gotten by NginxStatus. However, the function can be used only if nginx manager options have been configured.

With NginxStatus, we can know each nginx process resources taken, cpu and memory.

## Configure the nginx

The configuration functions are a litte more complex. First, you must be know nginx configuration file deeply. The nginx conf file cantains only two elements.

- Block:  a collection of many directives, surrend by "{}".
- Directive: implements a specific function of nginx, ended by ";". A directive can have zore or more than one parameter.

The library provides two accosiating entity beans, named Block and Directive. The interface of Configurator provide a lot of functions for operate conf file content, and the Block and Directive are used as arguments. The functions include:
Get: get the Block object in specific position in conf file.
Delete: delete specific block or directive in specific position.
Replace: replace the old element with the new one in the specific position.
Insert: instert a new element in after any element.
Append: add a new element in a specific block's end.

Detail for all method's parameters and usage can be find in library API document.