



Federation Management Suite

Keep the control on your AgentService Federation

The AgentService Team

Copyright @ 2009 – l.i.d.o. – DIST University of Genoa

Abstract

In this manual the AgentService user will learn to deploy an AgentService federation across a computer network. The user will be able to install AgentService as a windows service, configure the federation and platform parameters, and manage a distributed multi-agent application.

Index of Contents

Abstract	2
Index of Contents.....	3
1. Introduction	4
2. The windows service	5
2.1. Install the AgentService platform.....	5
2.2. Configure the AgentService platform	6
2.3. Configure the windows service	6
3. The Remote Platform Manager	8
3.1. Connection to the platform.....	9
3.2. Starting and stopping the platform.....	9
3.3. Upload files	9
3.4. Starting AgentService applications	11
3.5. Creating and activating agents.....	11
3.6. Activating agents.....	11
3.7. Activating agents.....	11
3.8. View logs.....	12
3.9. Windows Process Management	12
3.10. Menu.....	13
4. The Federation Manager	14
4.1. Connection to the federation.....	14
4.2. Submitting commands to the federation and platforms.....	16
4.3. Suggesting nodes to the federation.....	17

1. Introduction

The *Federation Management Suite* is a set of functionalities which enable users to fully exploit the possibility of deploying a federation of AgentService platforms, in an easy way.

We identify three main aspects of the *Federation Management Suite*:

- 1) The platform as a windows service: every local installation of AgentService is done as a local service.
- 2) The *Remote Platform Manager*: a windows form which allows users to manage a remote platform.
- 3) The *Federation Manager*: a windows form which offers a centralized point of view on the whole federation.

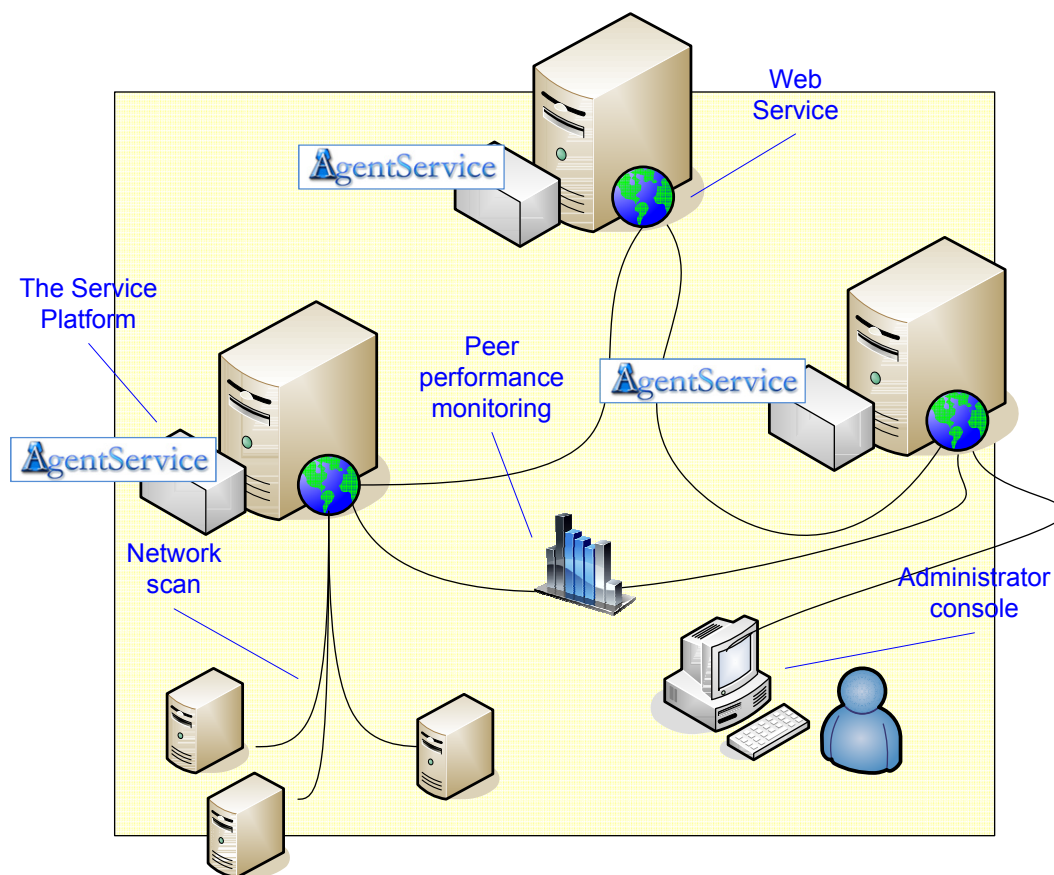


Figure 1: the Federation Management Suite.

As shown in Figure 1, remote nodes of the federation, interact through web service interfaces in order to exchange and share information about the federation topology and about the performances of each node. Every node scans the network in order to discover new AgentService nodes and to check the availability of the existent ones.

The aforementioned *Federation Manager* form interacts with the federation through whatever node and can manage agent instances, administering their whole life-cycle. In particular the user can start a new agent instance in two ways:

- By selecting a node and directly create a new agent instance here
- By delegating the federation to decide where the agent will be executed, according to the performances of every node.

Let's start to deploy the federation step by step.

2. The windows service

In order to use the graphical user interface for managing platforms and federations, an **AgentService platform must be installed as windows service**. In order to do this, run the installation file named *setup.exe* and follow the necessary steps. The infrastructure of your windows service will be installed in directory like this: *C:\Program Files\AgentService\Setup*, depending on your local operating system.

A new windows service appears in the services list. You can enter in your *control panel>administrative tools>services* and you will see a new service named *MyAgentService*. Here you can start, stop, or restart the service or make it automatic or manual.

2.1. Install the AgentService platform

Before to start the service, you have to install a usual AgentService platform (be sure that the platform version is 1.0.15 or higher) in the directory where the windows service infrastructure has just been installed (*C:\Program Files\AgentService\Setup*). In the *install.xml* file, you have to modify the installation directory (the *base-dir* tag) and modify the platform name (in the *description* tag, the *name* attribute). Keep in mind that the platform name must be unique, so, in your federation every platform has its specific name, for example, containing the computer name (e.g. *AgentService-ComputerDesktop003*).

Install the platform typing the usual command: `asdrv -i:install.xml`. (see the quick start guide for further information). Then go to the directory `C:\Program Files\AgentService\Setup` and launch `asdrv` without parameters. When the platform is running, press enter and stop it. This step is essential to create some configuration files.

2.2. Configure the AgentService platform

Open the file `modules\conf\msmq.conf` and set to `True` the two following parameters:

- `EnableRemoteSend`
- `EnableRemoteReceive`

Then, choose the port on which the platform will listen for incoming AgentService messages. The three parameters are aimed to open the platform to the rest of the federation for exchanging messages among remote agents.

2.3. Configure the windows service

Finally, you have only to create a configuration file for the windows service. Name it `config.xml` and copy into it the following lines:

```
<?xml version="1.0" encoding="utf-8"?>
<information>
  <install>
    <folder> C:\Program Files\AgentService\Setup </folder>
  </install>
  <logging>
    <generate>yes</generate>
    <append>no</append>
    <limited>yes</limited>
    <dimension>1</dimension>
  </logging>
  <platformName>
    <name>Gattobullli</name>
  </platformName>
  <servicePort>
    <port>8000</port>
  </servicePort>
  <remotingPort>
    <port>5000</port>
  </remotingPort>
  <periodicDiscoverPlatform>
    <second>120</second>
  </periodicDiscoverPlatform>
  <periodicPerformanceInterval>
    <second>30</second>
  </periodicPerformanceInterval>
  <priority>
    <masterPlatform>true</masterPlatform>
  </priority>
  <federation>
```

```
<network>130.251.22.</network>  
<mode>normal</mode>  
<available>always</available>  
</federation>  
</information>
```

Starting from the first tags, you have to set the folder where the AgentService platform has been installed (*folder* tag).

In the *logging* tag, you can set different parameters. Type *yes* if you want to generate a logging file (*generate* tag), *yes* or *no* in the *append* tag if you want to create logging file in append mode, every time the service is restarted. You can set also if the file will be limited, and in this case, you must set also the maximum size (in megabytes).

In *platformName*, set the name of the AgentService platform: the same string you wrote in the *install.xml* file.

Choose the port where the windows service is listening (*servicePort*), which is different from the port where the AgentService platform is listening for incoming agent messages (*remotingPort*). The value of *remotingPort* must be the same which is set in the *msmq.conf* file.

In the *periodicDiscoverPlatform* set the interval (in seconds) between to consequent checks for the node availability in the federation.

PeriodicPerformanceInterval sets the delay between to two readings of the performance parameters for the evaluation of every node. In fact, every node of the federation publishes its metrics about CPU usage, RAM available and number of executing agents, in order to allow the federation manager to decide where to execute new agents.

In *priority* we can state if the local node is a master node. In this case, the windows service is able to decide where to execute a new agent instance, evaluating the performances of other nodes. If you set *false*, the local node is only able to locally create agent instances. This setting is preferable in computers with few resources available.

In the last session, there are the federation parameters. So you must set up the network where the federation will be deployed (*network* tag). You can set the *mode*: *normal* if the network scan involves, for each online computer, only the standard port where a windows service is supposed to be in waiting (the default port is 8000, as shown in the *servicePort* tag). If you set *aggressive*, the local node will check all the 65000 ports for every computer. This mode is extremely slow.

The tag *available* states if the local platform can take part to a federation:

- *Always*: the node is open to every federation
- *Never*: the node is closed to whatever federation
- *Not always*: currently not supported.

Now, in order to try the *Federation Management Suite*, be sure to execute the local windows service named *MyAgentService*.

3. The Remote Platform Manager

The *Remote Platform Manager* is the tool aimed to manage a single platform running on a *windows service*, both locally and remotely. Through a web service interface the *Remote Platform Manager* directly interacts with the remote windows service and manages the whole AgentService platform, executing agents applications, or simply instantiating agents. As shown in Figure 2 the interface of the *Remote Platform Manager* is very simple. There is a toolbar where you can press buttons in order to submit commands to the platform. As we see later, in some case, additional forms will be opened in order to fill the details to perform the action. The form presents three main areas: the *console* where textual information is reported directly from the remote platform, the *properties of the connected platform* (top right), the *list of created agents* (bottom right).

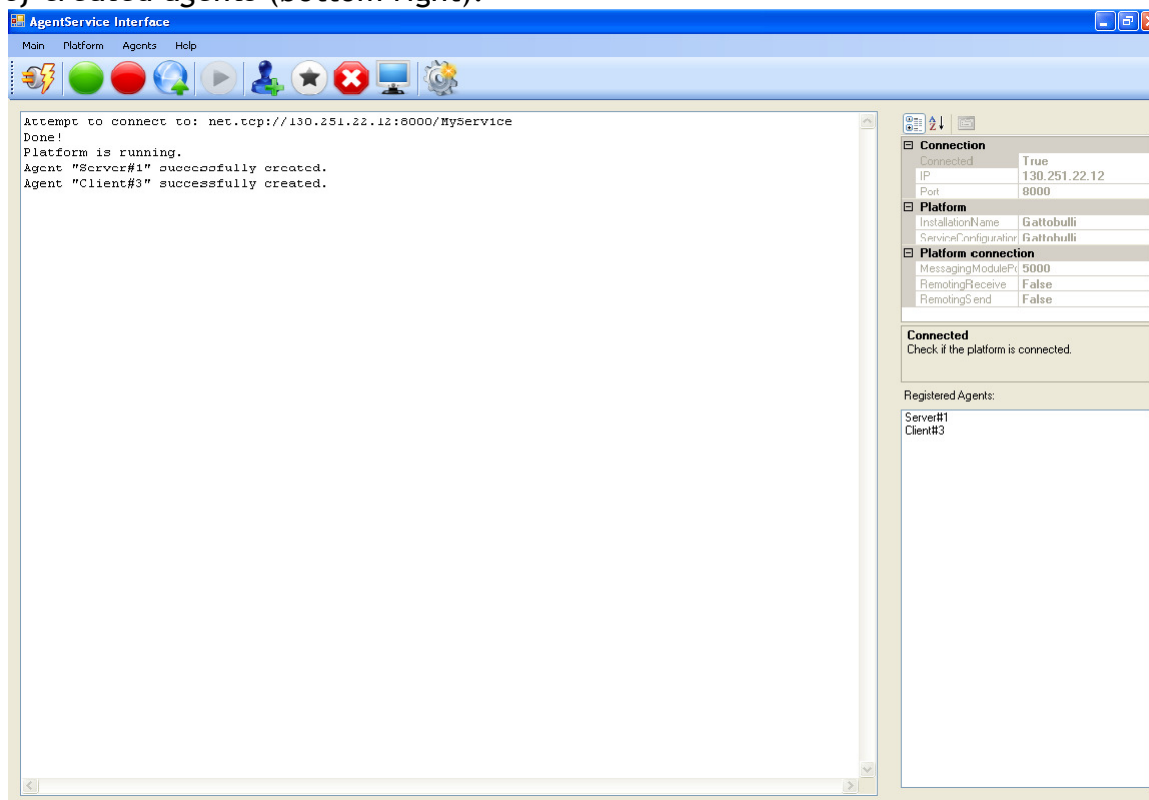


Figure 2: the main form of the Remote Platform Manager.

3.1. Connection to the platform

Click on the first button in the toolbar in order to open the Connection form (Figure 3). Here you can select a previously connected platform from the combo box or you can add or remove new connections. If you want to add a new connection, you have to insert a significant *label* for your connection. Then, insert the *IP address* and the *port*. These parameters are persistent and stored in a local file named *addresses.xml*.

3.2. Starting and stopping the platform

Now you are connected to the windows service, but the platform could be still stopped. Press the second button in the toolbar to start it, and the third button to stop it. If you reconnect later to the window service and the platform is still running, obviously you will see it as a started platform.

If you start a platform, the *run application* button is disabled, because you can start a multi-agent application (through an *.abs* file) only if the platform is stopped.

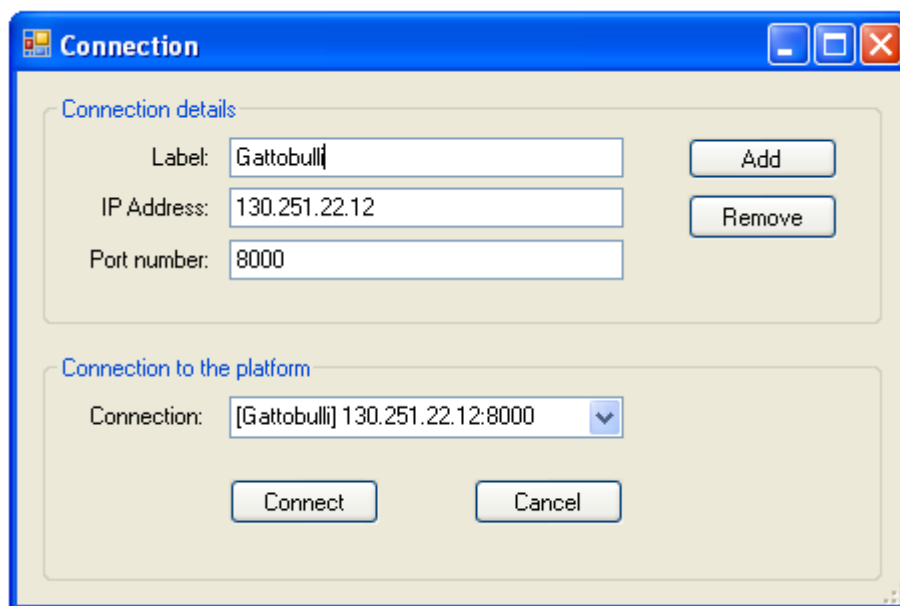


Figure 3: the connection form

3.3. Upload files

In order to upload files to the remote platform you can click on the button *Upload file* to send to the windows service one or more files. It is useful to upload assemblies, *.abs* files, configuration files, etc.

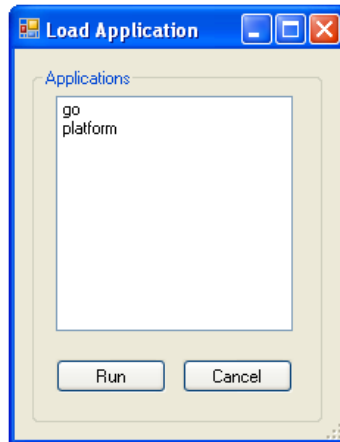


Figure 4: load application form.

If you replace an assembly which is loaded by the *Storage Module*, perhaps you must restart the service in order to unlock the assembly.

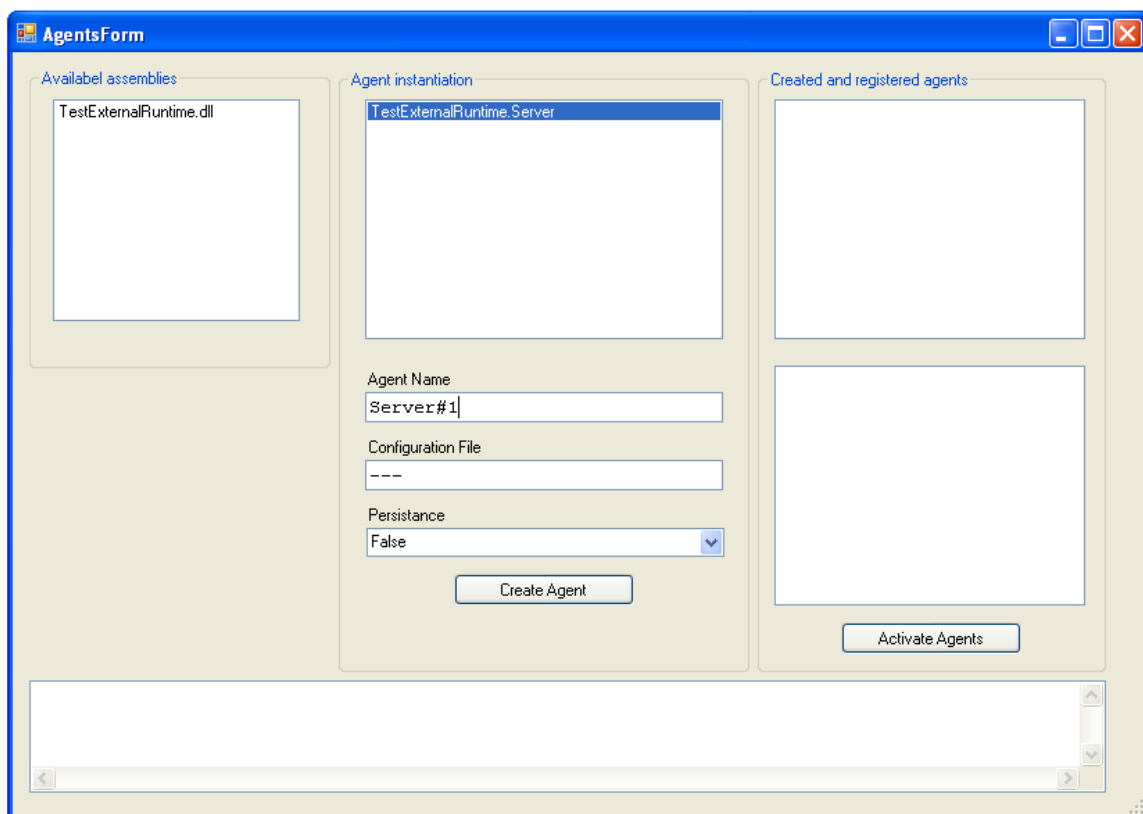


Figure 5: Agent instantiation.

3.4. Starting AgentService applications

The button *Run Application* allows you to execute a batch for an application, defined in its *.abs* file. By clicking the button you can choose the *.abs* file from the list (in Figure 4 we have two files: *go.abs* and *platform.abs*). Once the application is started, you can stop it, or add more agent instances.

3.5. Creating and activating agents

Press *Create Agent* to instantiate a new agent instance. In the new form (Figure 5) you can see all the assemblies containing agent definitions and you can select the agent template you want to instantiate. Once the agent type is selected, you must enter a unique name for the agent, insert an eventual configuration file name and set up its persistence mode. Then click on *Create Agent* in order to create an instance of the agent. Note that at the moment the new agent is not running. In order to start its behaviours, you must click the button *Activate agents* in this form, or, in a second moment, in the main toolbar.

3.6. Activating agents

The button *Activate agents* activates all the agents instantiated in the platform. You can perform this action also from the form for creating agent instances.

3.7. Activating agents

Click *Stop agents* in order to choose the agent instances to stop.

As shown in Figure 6, you can stop a single agent, or stop all the running agents.

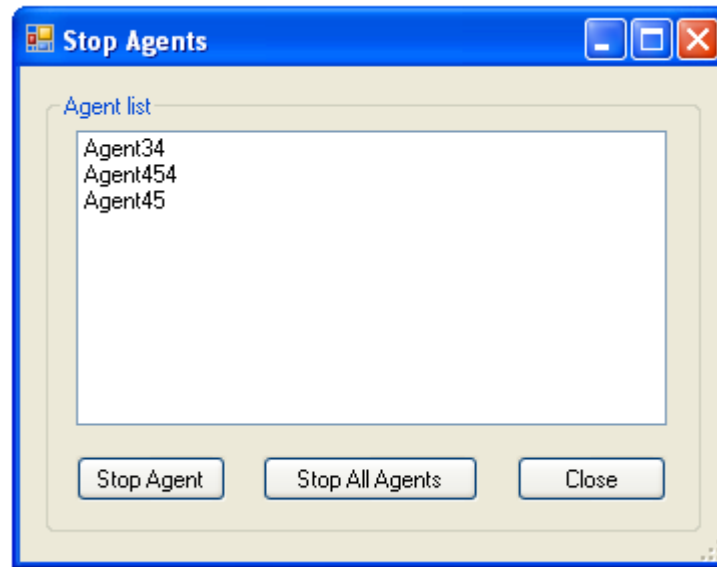


Figure 6: stopping agents.

3.8. View logs

You can view the logging files for the remote platform, by clicking on *View logs*. As shown in Figure 7 you can select from a list of log files the one you want to read.

View temporary log allows you to see the platform console during the last starting. By clicking *View log* you see the whole session log.

3.9. Windows Process Management

The last button allows you to manage your local windows services and in particular the AgentService Windows Service (by default: *myAgentService*). As shown in Figure 8 you can start or stop it.

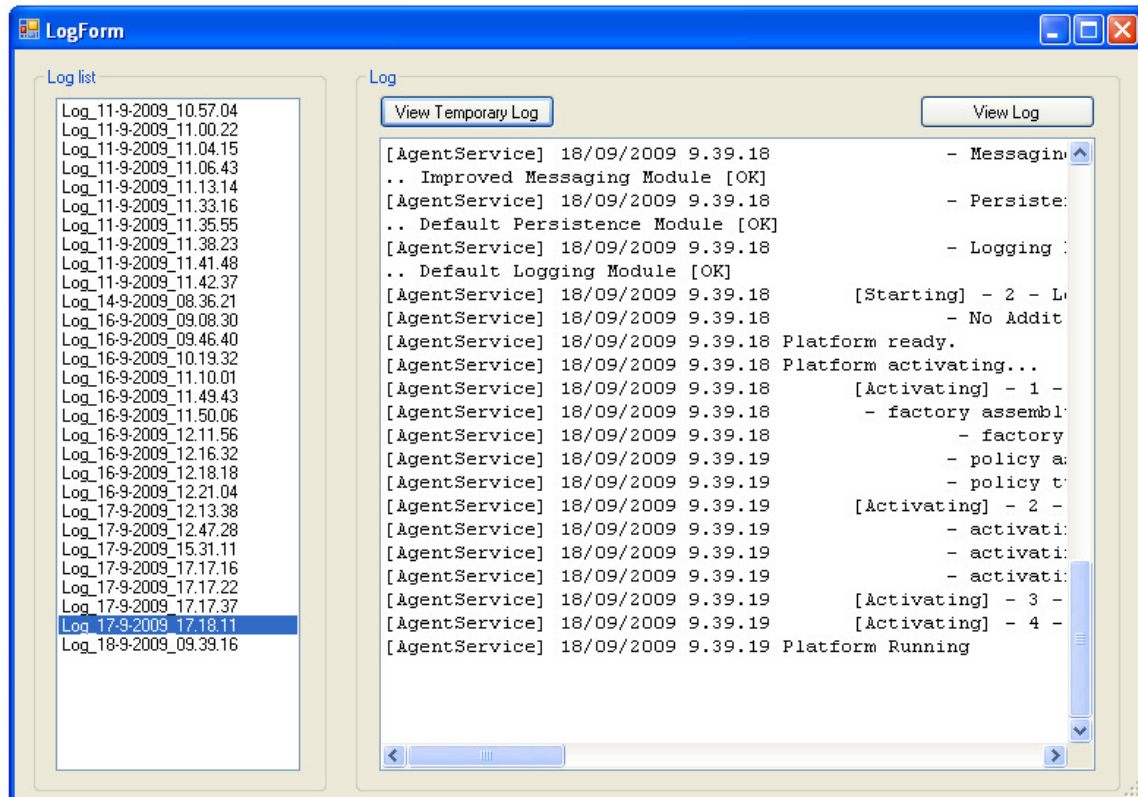


Figure 7: log form.

3.10. Menu

All the commands are replicated in the menus: main, platform, and agents.



Figure 8: starting and stopping a Windows Service.

agents), the perceptual CPU usage, and the available RAM. These parameters can be monitored also in another section of the form (Figure 11).

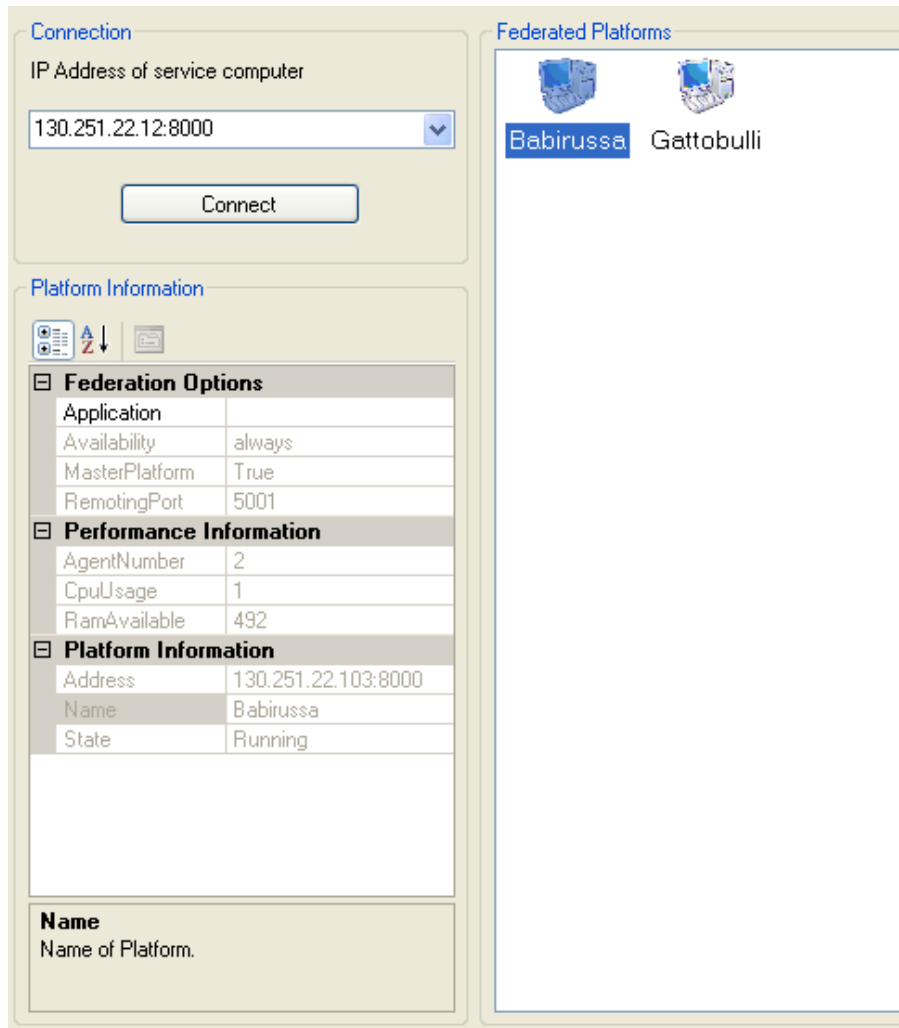


Figure 10: connection, list of federated nodes, and basic information.

Here you can find the parameters representing the status of each node: CPU usage, available memory, number of active agents, and the state of the platform.

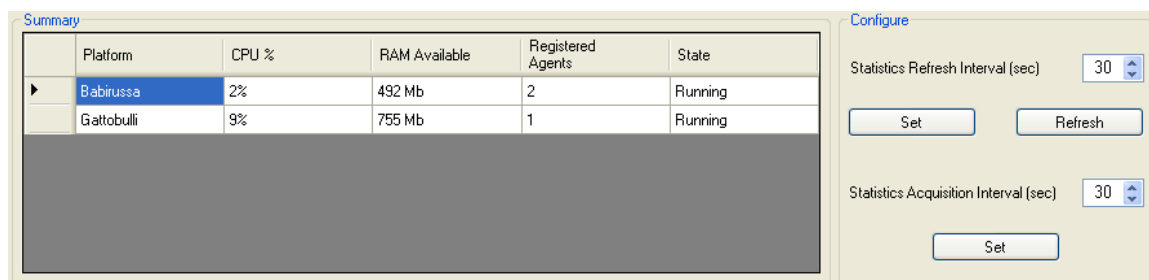


Figure 11: the synthesis of performance of each node.

These values are refreshed every 30 seconds, but it is possible to force the refresh or to change the timing interval.

4.2. Submitting commands to the federation and platforms

The Figure 12 shows the menu which appears if you right click on a node icon. You can perform actions of that node: starting and stopping the platform, view the active agents, or upload a file to the single platform.

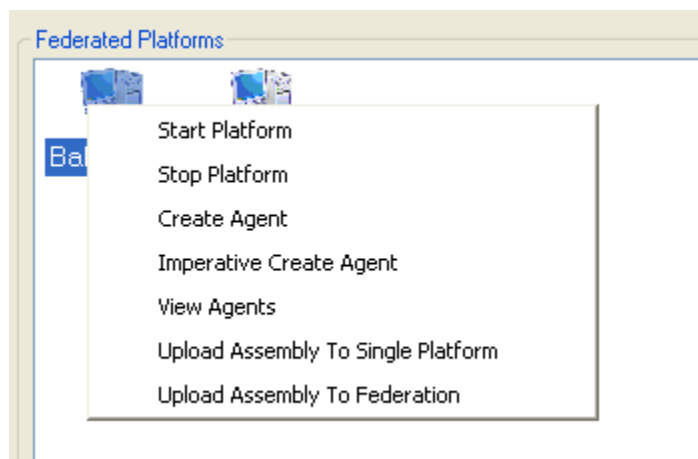


Figure 12: how to submit commands.

By clicking on *Create Agent* you submit the command on the whole federation, so the agent will be create on the best platform at the moment. *Imperative Create Agent*, instead, creates an agent on the selected node. In both cases, the input controls for the agent details will be enabled.

You can also upload an assembly or a file to the whole federation, in this case the file will send to each node of the platform.

Finally, you can see the list of created agents for each node, clicking on *View Agents*. The form presented in Figure 13 shows the list of agents; you can select an agent and stop it.

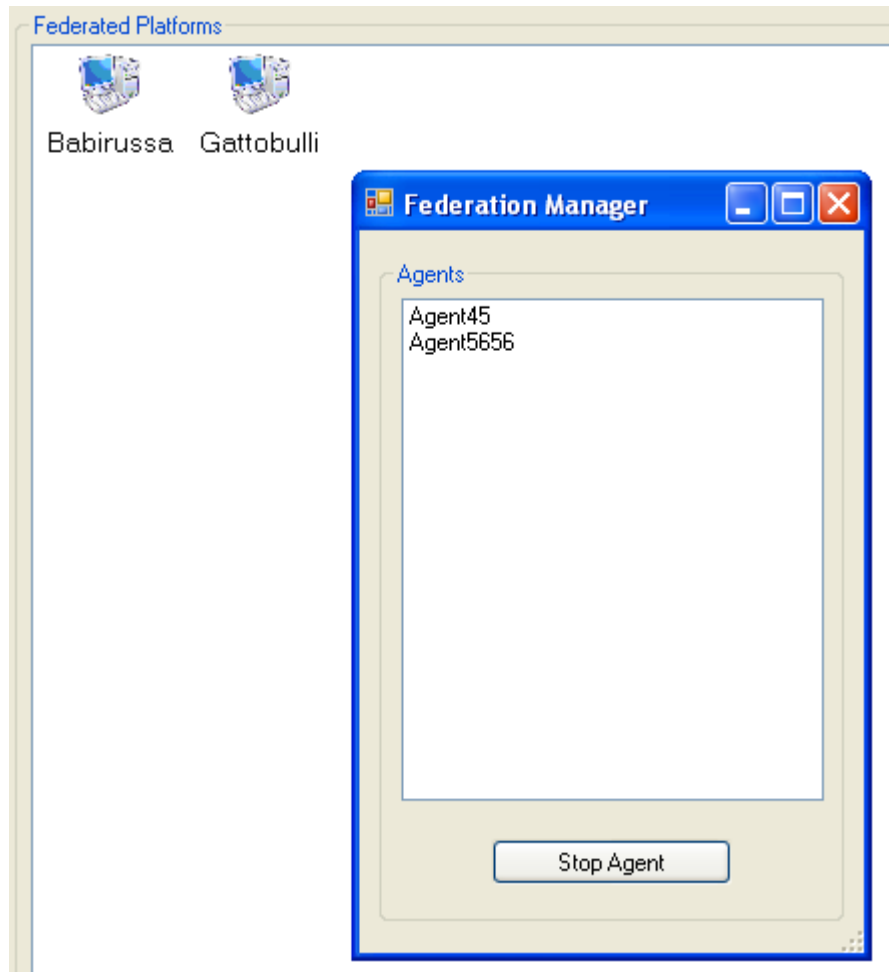


Figure 13: the list of agent.

4.3. Suggesting nodes to the federation

The process of discovering new nodes of the platform is a time-consuming task, so in order to accelerate the setup of a federation, you can suggest to a node an initial topology. In practice, you can suggest IP and port of remote nodes you know they must be federated with the current node.

Open the folder containing the Windows Service and open the file named `Service\Federation\PreviousPlatform.txt`. Here you can add new lines containing node information. For example, you can suggest a node in this mode:

```
130.251.22.13:8000=10
```

The `=10` simply states the number of attempts after which the node declare the remote platform missing.