NOVASCALE

Dynamic Domains for Applications

User's Guide

REFERENCE
86 A2 63ER 01

NOVASCALE

Dynamic Domains for Applications

User's Guide

Software

September 2006

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
86 A2 63ER 01

Printed in France

Suggestions and criticisms concerning the form, content, and presentation of this book are invited. A form is provided at the end of this book for this purpose.

To order additional copies of this book or other Bull Technical Publications, you are invited to use the Ordering Form also provided at the end of this book.

Trademarks and Acknowledgements

Table of contents

1. Introduction

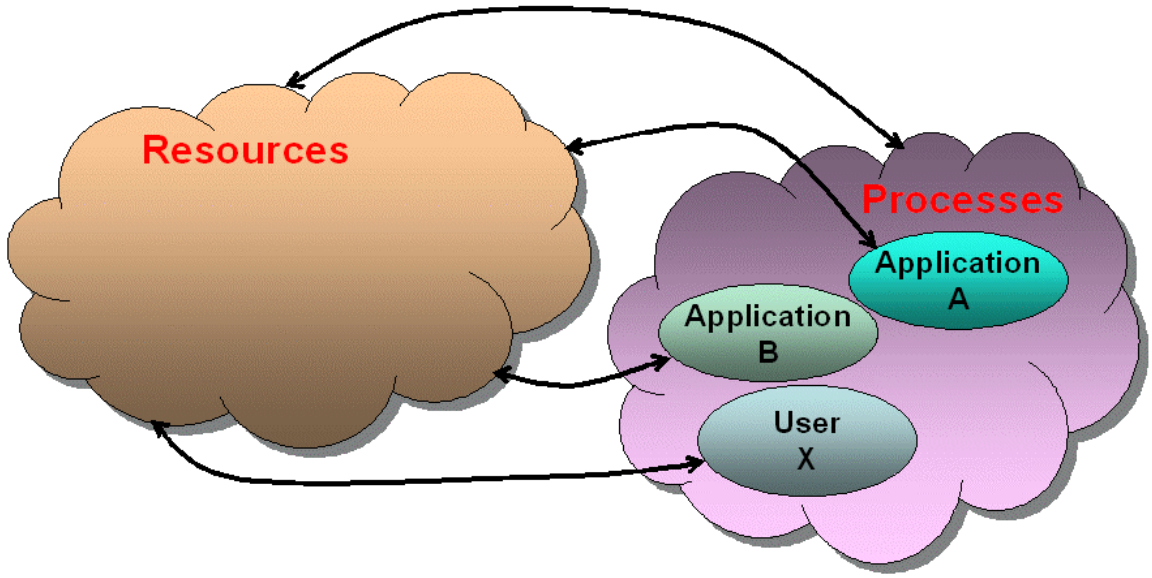
The "Dynamic Domains For Applications" software (simply referred to as "Dynamic Domains" in this manual) is a tool that can be used on the Linux operating system for simulating the partitioning of a multi-CPU machine at application level. "Dynamic Domains" can be used with the standard distributions (Red Hat RHEL 4, Suse SLES 9, Bull BAS 4, etc.). "Dynamic Domains" can be managed using the Webmin standard administration tool.

Description

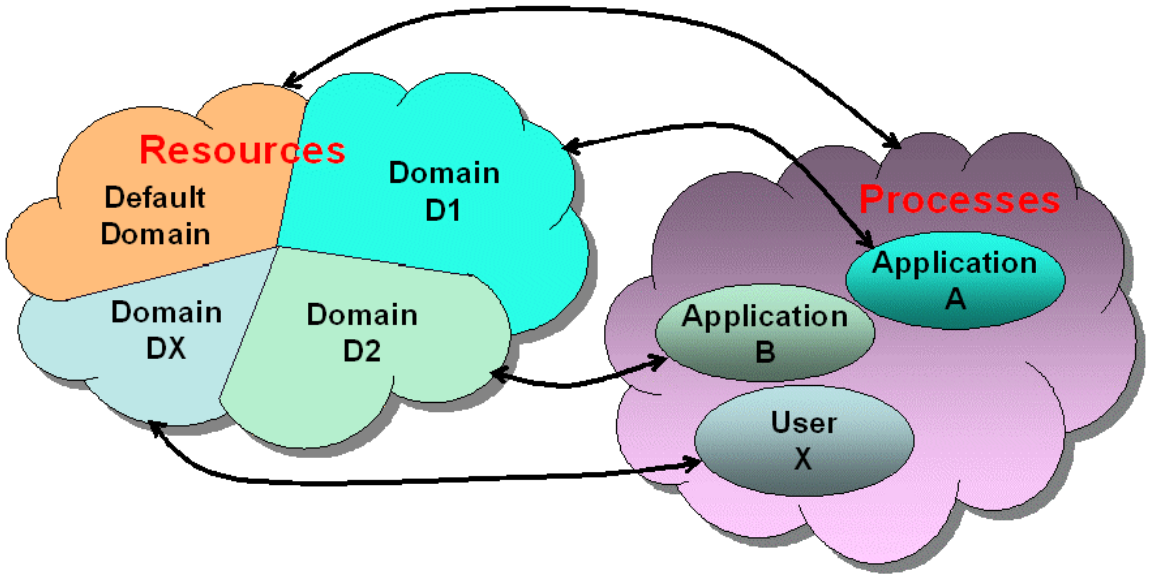
"Dynamic Domains" correspond to a breakdown of the resources available on a machine. A given domain is composed of a set of resources (one or more CPUs). On installation of the "Dynamic Domains" software, a default domain is automatically created. This domain will use all the machine's resources and all the machine's CPUs will be assigned to it. When a domain is created, the various resources required at domain level are recovered from the default domain.

Principle

The "Dynamic Domains" software enables resources to be reserved for a given application, user, etc., with resources usable by lower priority tasks being limited. In the example below, without domain, all resources will be shared among the various processes.

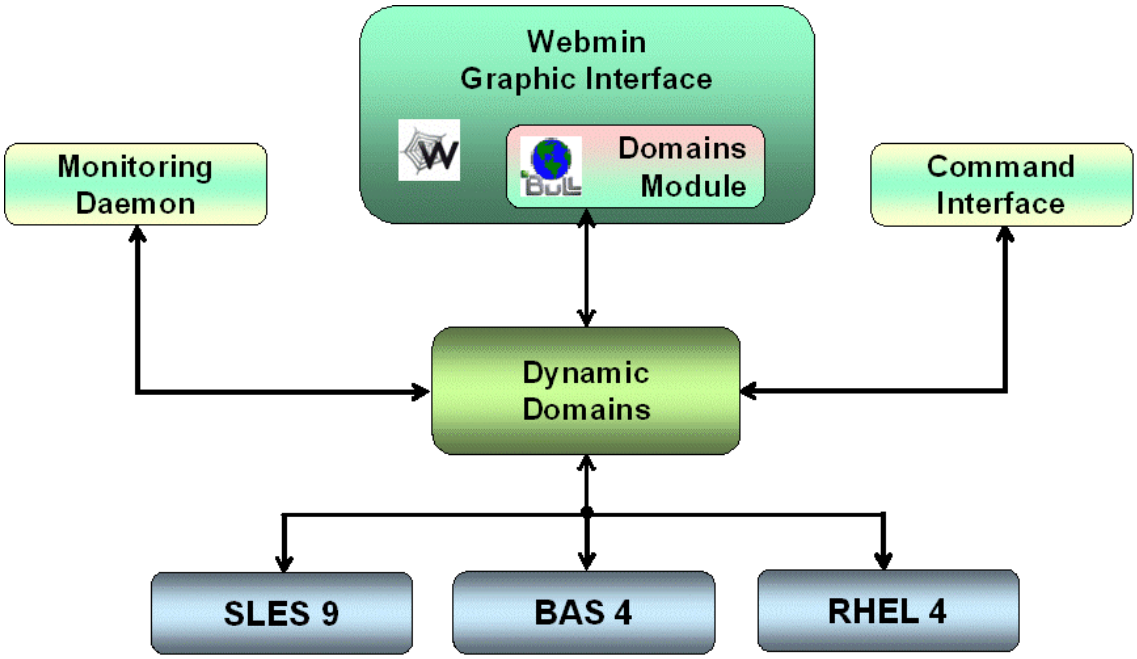


With domains, lower priority processes will share the resources of domain "default" while domain "D1" resources will be reserved for application "A", application "B" will be assigned to domain "D2", etc.



Resources can be dynamically assigned to the domains according to the usage level ("dynamic domains" notion). The dynamic management of domains is carried out by the monitoring daemon (see Appendix C: Simulation of dynamic domains).

Architecture



The graphic interface on Webmin enables the user to create new domains and assign processes to these domains by user name, task name or more sophisticated criteria in advanced mode. This interface can be accessed from a new Webmin module, via the "System" category.

This module includes three submodules, either in Normal mode (default configuration) or in Advanced mode (use of all software functions):

- "Domains Management" : Creation, edition, deletion of a domain.
- "Tasks Assignment" : Assignment of processes to the domains.
- "Configuration" : Configuration of "Dynamic Domains" module.

The monitoring daemon and the control interface are based on a set of tools installed by the product:

- ddadd : utility for creating a new domain.
- ddchg : utility for changing the characteristics of a domain.
- ddflt : utility for assigning processes to the domains using the multicriterion filter file.
- ddload : utility for simulating the load of a domain.
- ddls : utility for displaying the domains configuration and the processes assignment to domains.
- ddmon : dynamic domain management and domain monitoring daemon.
- ddrm : utility for destroying a domain.
- ddstat : utility for displaying the domains load status.
- ddtask : utility for assigning processes to the domains (equivalent to "taskset"

command for domains).

- ddtop : utility for displaying the CPUs loads and machine's processes (equivalent to commands "top" or "ps").
- ddtopd : daemon for periodically displaying the CPUs loads and machine's processes (daemon for the "ddtop" command).

See Chapter 5: "Commands"

Installation

The "Dynamic Domains" software is delivered in the form of an rpm file to be found on the following CD-ROM:

- Bull Extension Pack for RHEL4.

See Chapter 2: "Installation".

Checking the Installation

Once the prerequisites and "Dynamic Domains" software are installed, it is recommended to check the following:

- presence of domain "default",
- proper operation of the monitoring daemon,
- access to the graphic interface.

See Appendix A: "Check of installation".

Configuration and Use

The "Dynamic Domains" software is configured or administered using the "Webmin" tool ("System" category) via a browser (URL: http://<hostname>:10000/ or https://<hostname>:10000/).

See Chapter 3: "Configuration and use in Normal mode", Chapter 4 "Configuration and use in Advanced mode" and Appendix B "Example of use".

2. Installation

General

The "Dynamic Domains for Applications" software is delivered on the following CD-ROM:

- Bull Extension Pack for RHEL4.

This software requires a Linux Red Hat RHEL4 or Suse SLES9 distribution to be installed with the following prerequisites:

- shell sh,
- Webmin administration tool (rpm to be found on above CD-ROMs, in tools directory).

Installation of "Dynamic Domains" Software

- Install the prerequisites.
- Install the "Dynamic Domains" software rpm (example: ddomains-[vv.rr-cc].ia64.rpm), to be found on above CD-ROMs, in tools/ddfa directory:

```
rpm -ivh ddomains-[vv.rr-cc].ia64.rpm
```

The rpm installation checks that the required prerequisites are properly installed and prepares the software operating environment (configuration of Webmin administration tool).

Example

```
rpm -ivh ddomains-[vv.rr-cc].ia64.rpm
Preparing...                ########################################### [100%]
   1:ddomains               ########################################### [100%]
Stopping Webmin server in /usr/libexec/Webmin
Starting ddmon  OK
```

3. Configuration and use in Normal mode

General

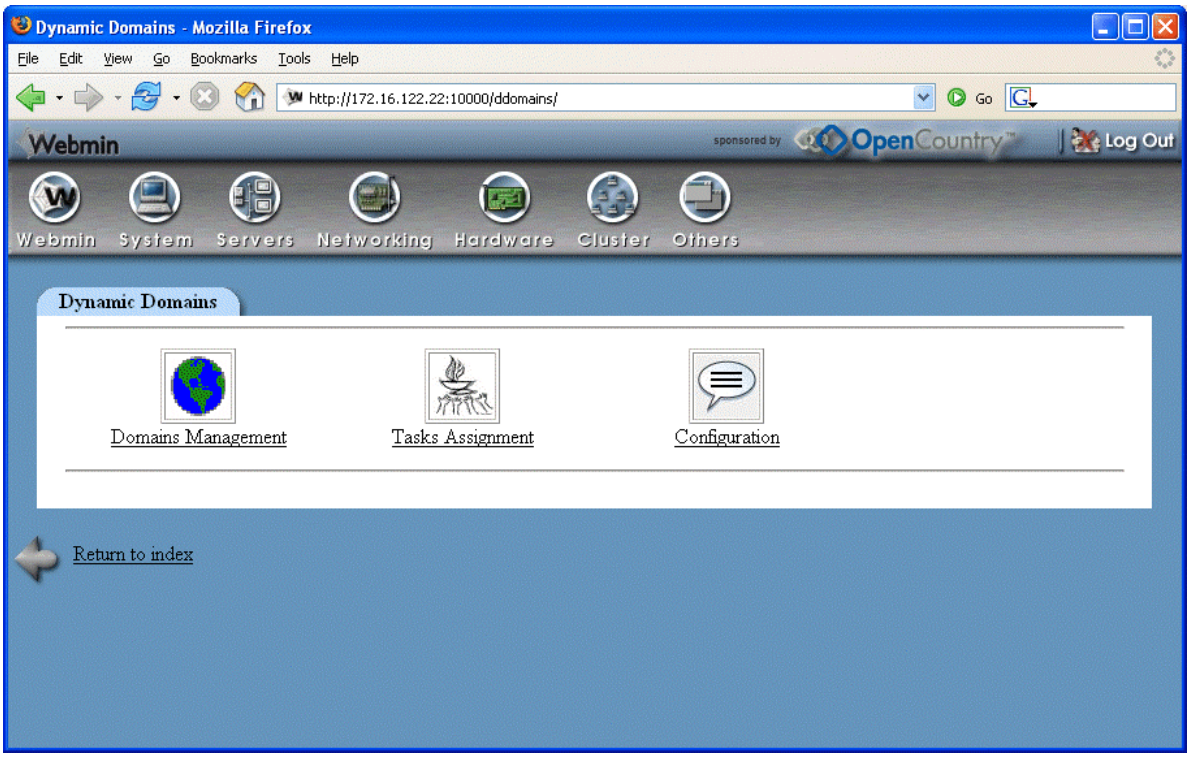
The "Dynamic Domains" software is configured or administered using the "Webmin" tool ("System" category) via a browser (URL: http://<hostname>:10000 or https://<hostname>:10000).
The administration tool can be used in two operating modes:

- a "Normal" mode (default configuration),
- an "Advanced" mode (giving access to all software functions).

The screens below give examples of software implementation.

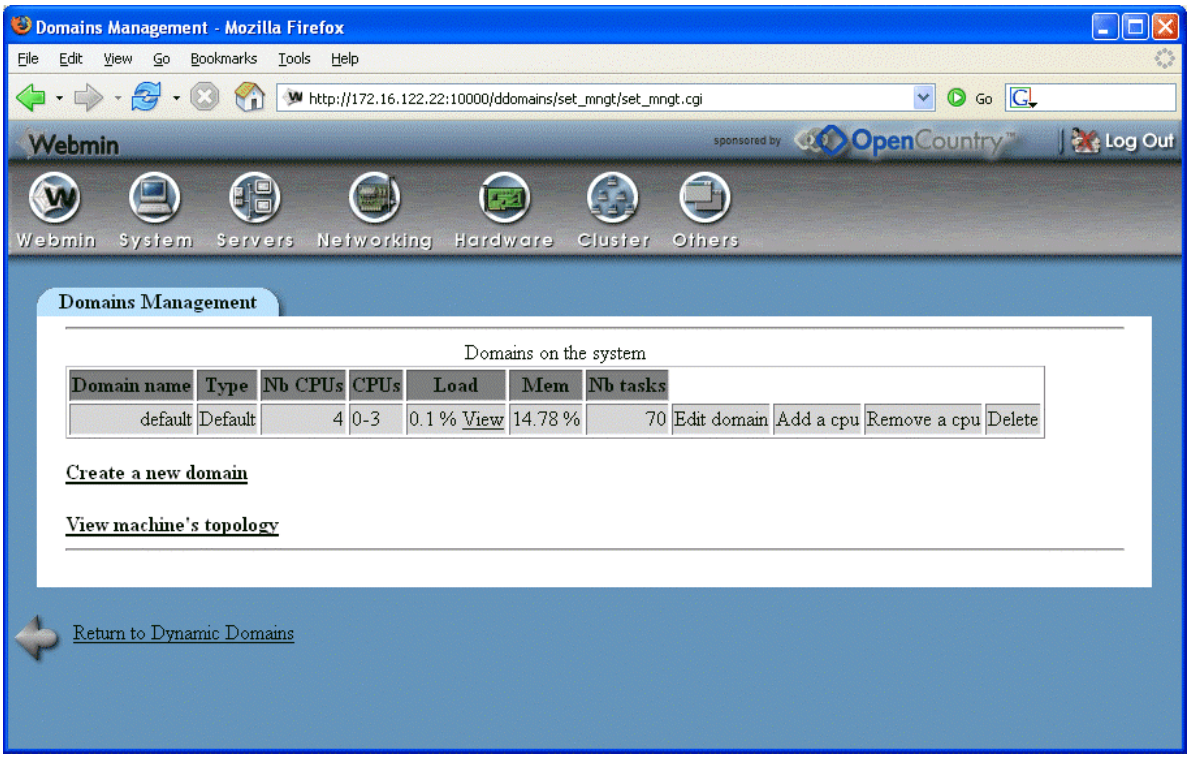
Home page

Access with Webmin after software installation: System->Dynamic Domains.



Managing a domain

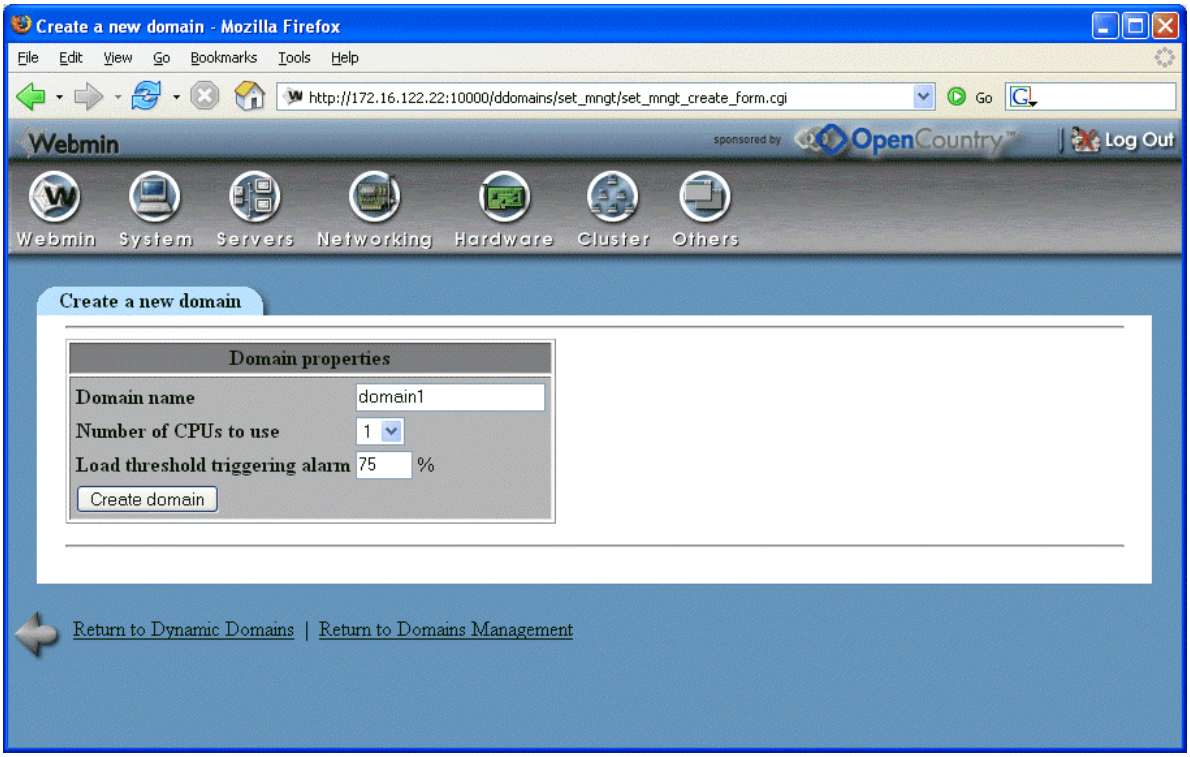
Click "**Domains Management**" to add, edit or delete a domain:



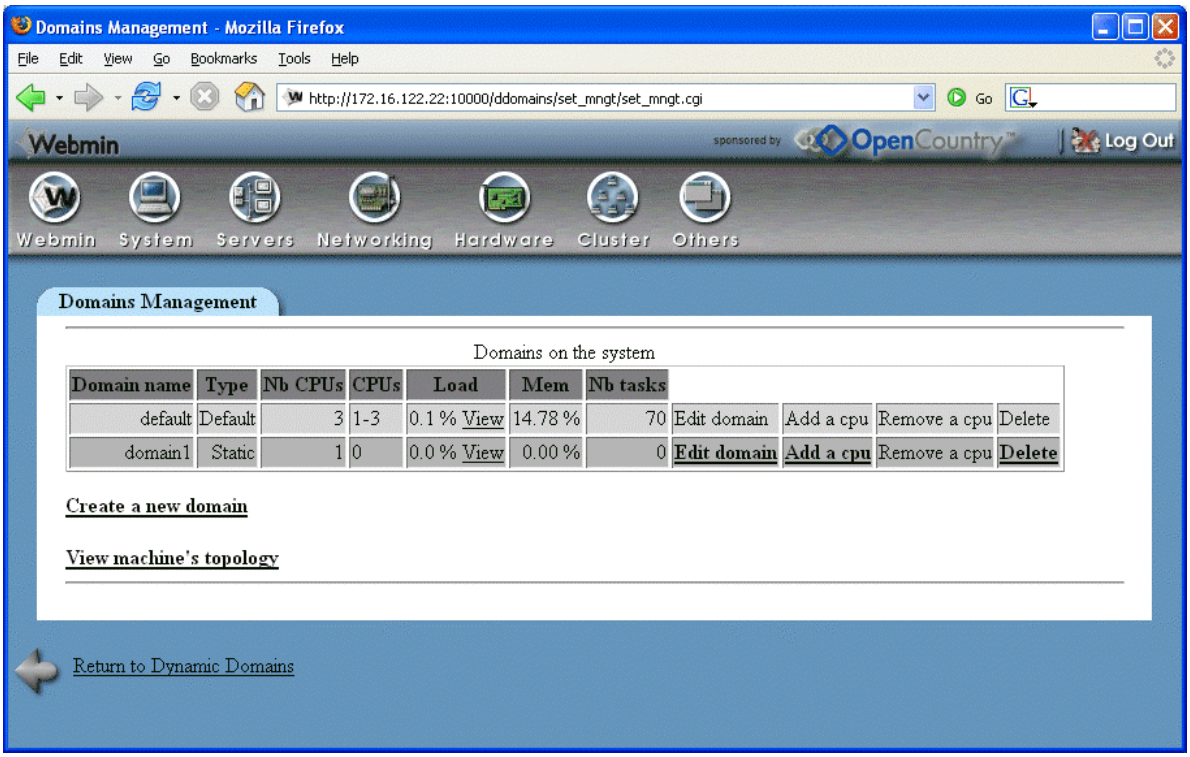
In the above example, domain "**default**" created on software installation uses all machine's CPUs (number of CPUs: **Nb CPUs = 4**, list of CPUs used: **CPUs = 0 to 3**), all tasks are assigned to it and its CPUs are not loaded (**Load ~ 0%**), the memory used by the domain (resident process memory) represents **~ 15%** of total memory. No action (edition, deletion, etc.) is possible on this domain.

Creating a new domain

Click "**Create a new domain**" to create a new domain.



- Enter the domain name (example: **domain1**).
- Select the number of CPUs for the domain (example: **1**),
- Select the alarm threshold for the domain (default: **75%**). When the domain load exceeds this threshold, an alarm message is issued by the monitoring daemon (see command ddmon).
- Click "**Create domain**".

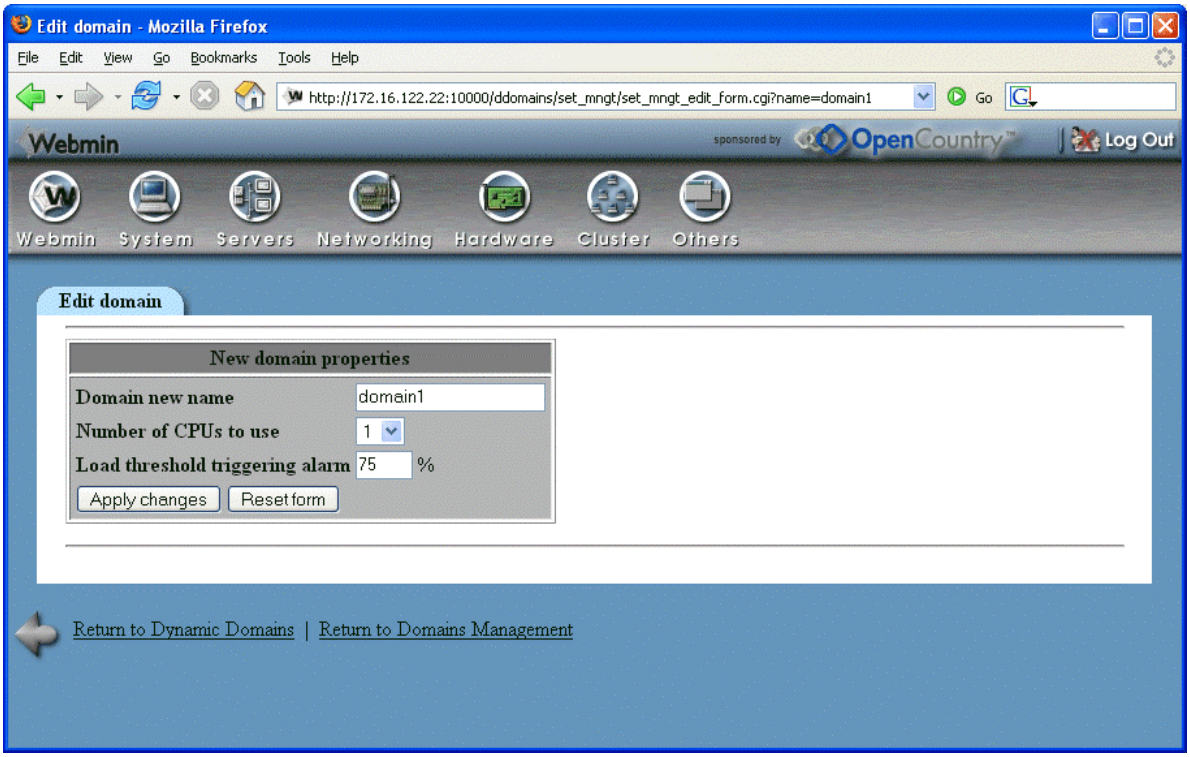


A new static domain has been created. This domain uses one CPU (CPU 0) and no task is assigned to it. The CPU used by this new domain has been taken from domain "**default**". Only three CPUs are now left to domain "**default**" (CPUs 1 to 3).

- Click on the following as applicable:
  - "**Add a cpu**" to add a CPU to the domain (this option is deactivated if no CPU is left). The added CPU is taken from domain "**default**".
  - "**Remove a cpu**" to remove a CPU from the domain (this option is deactivated if only one CPU is left to the domain). The removed CPU is added to domain "**default**".
  - "**Delete**" to delete the domain.
  - "**View**" in column "**Load**" to view the load of the domain CPUs.
  - "**Edit domain**" to edit the domain configuration.

Editing a domain

Click "**Edit domain**" for domain "**domain1**" in order to change its configuration.

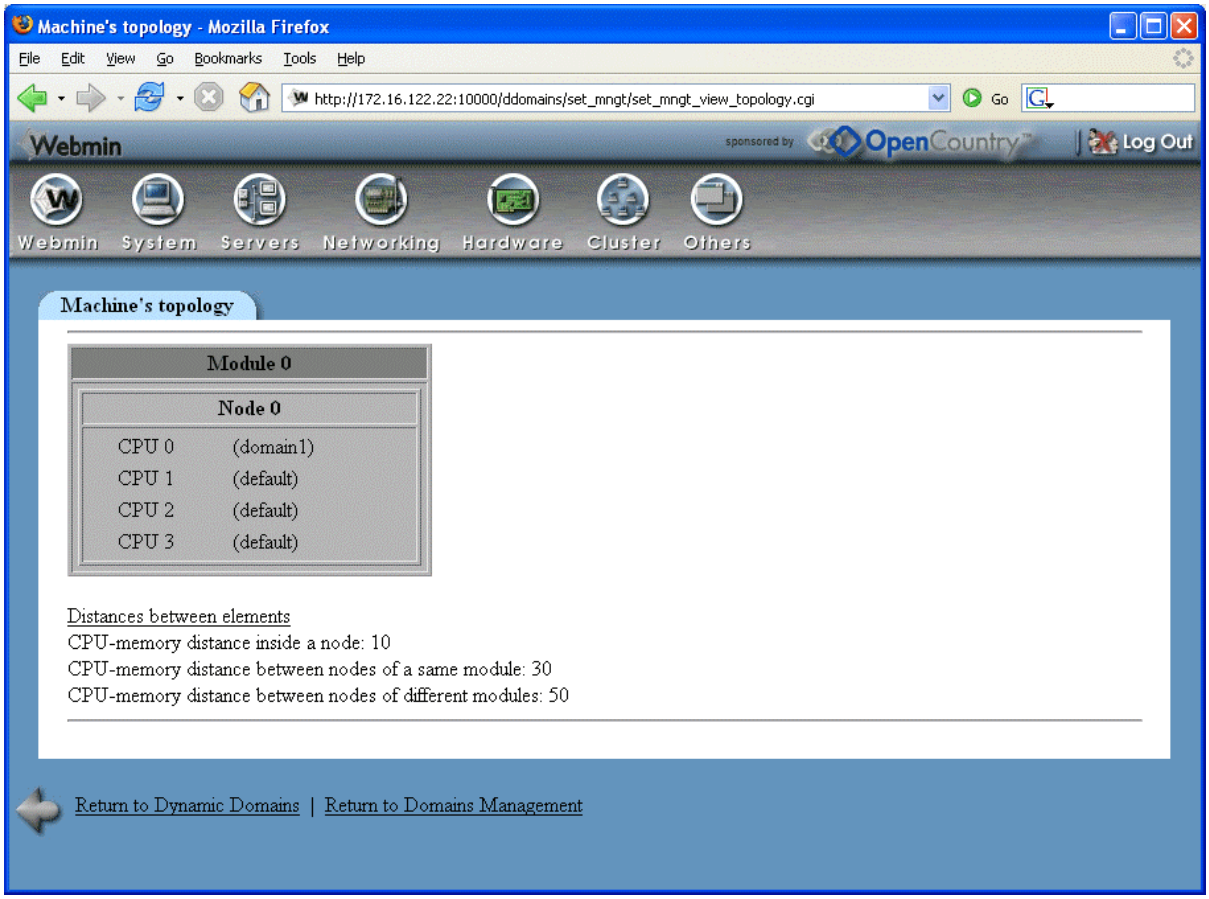


From this screen, the user can change the domain name, the number of CPUs used by the domain and the alarm threshold for the domain.

- Click "**Apply changes**" for changes to be taken into account or click "**Return to Domains Management**" to ignore the changes and return to domains management.

Viewing the new domain's topology

Click "**View machine's topology**" to view the domain's topology.

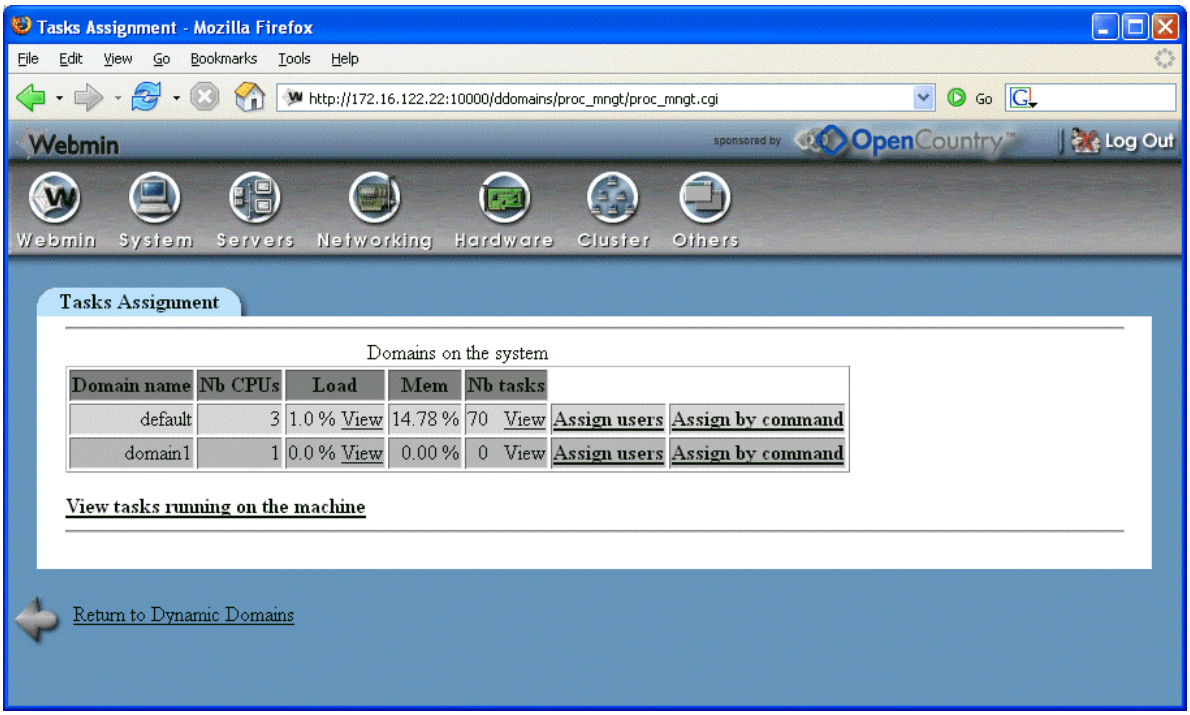


The machine consists in a single node including four CPUs. Domain "**domain1**" uses CPU 0. The distance between elements is provided by the Linux core. This distance represents the time required for accessing the memory of the same node, another node of the same module or another node for a given CPU (the NUMA [Non-Uniform Memory Access] factor). The CPUs used by a domain are selected with this distance being taken into account. The "**default**" domain uses the free CPUs (CPUs 1 to 3).

- Click "**Return to Dynamic Domains**" to return to the main screen.

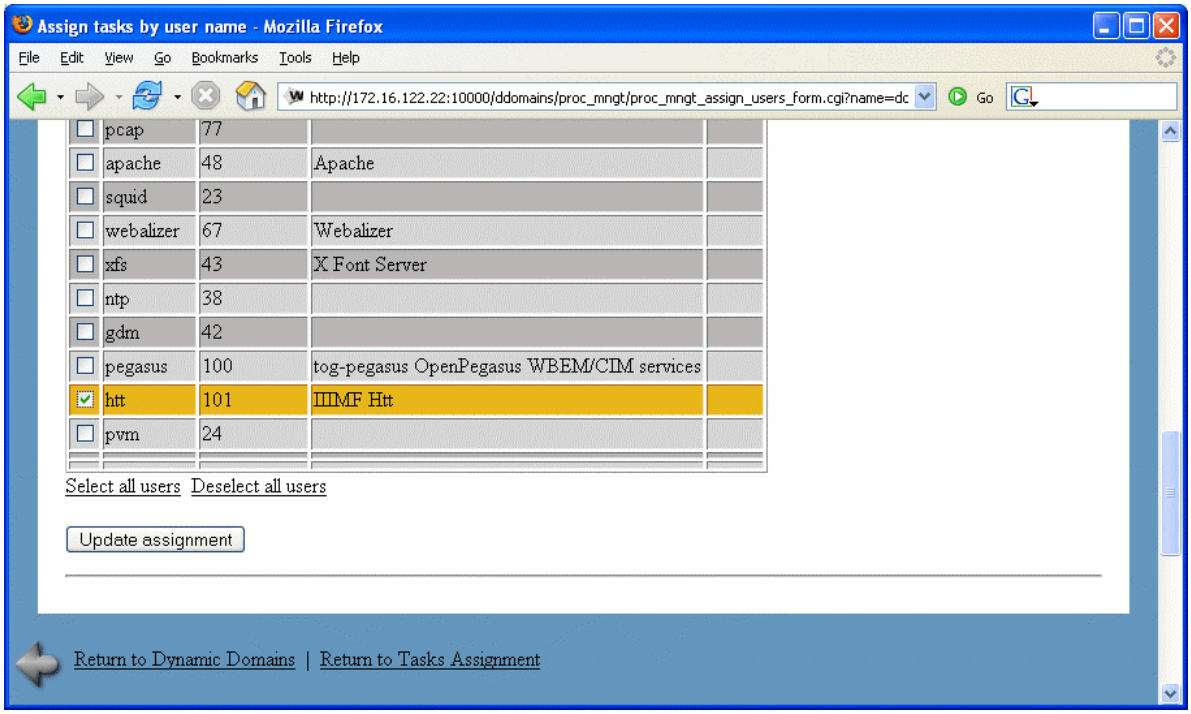
Adding or removing tasks for a domain

Click "**Tasks Assignment**" to add or remove a process from the domain.



Assigning users

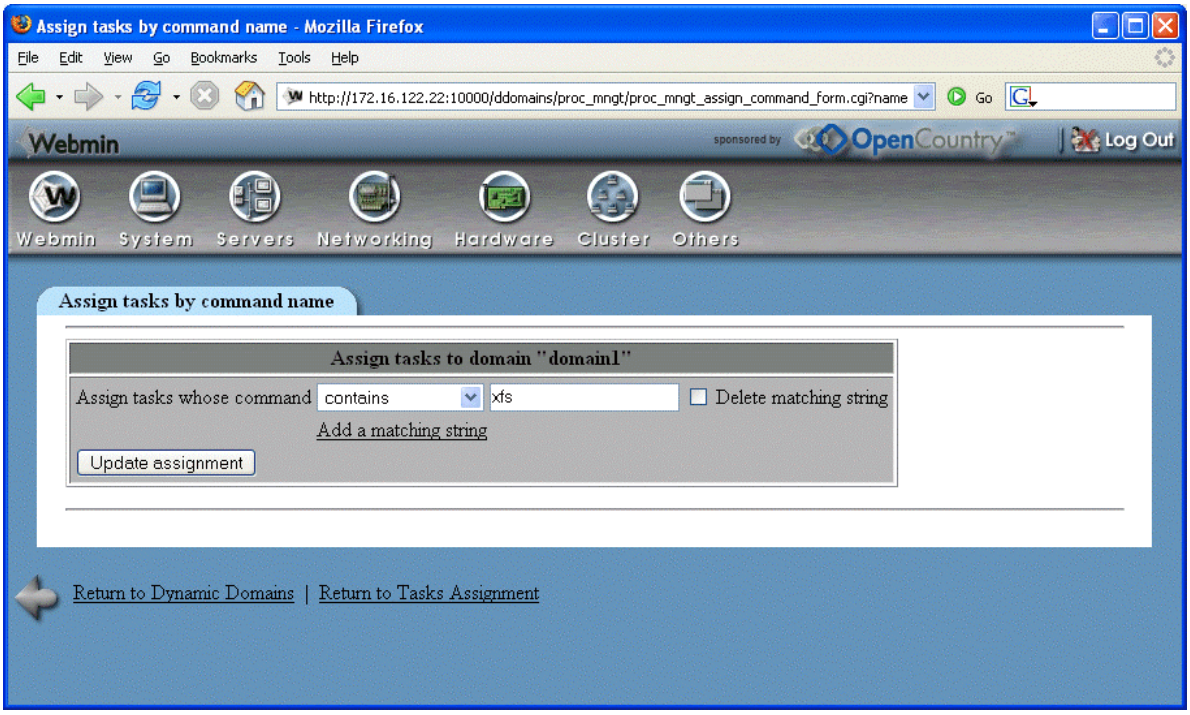
Click "**Assign users**" for domain "**domain1**" to assign one or more users' tasks to the domain.



- Check users whose tasks are to be assigned to the current domain (example: user "**htt**").
- Click "**Update assignment**" to validate the selections.

Assigning tasks by command name

Click "**Assign by command**" for domain "**domain1**" to assign tasks with a given character string in the associated command.



- Select the test mode and the content (example: assign tasks whose command name contains "**xfs**").
- Click "**Update assignment**" to validate the selections.

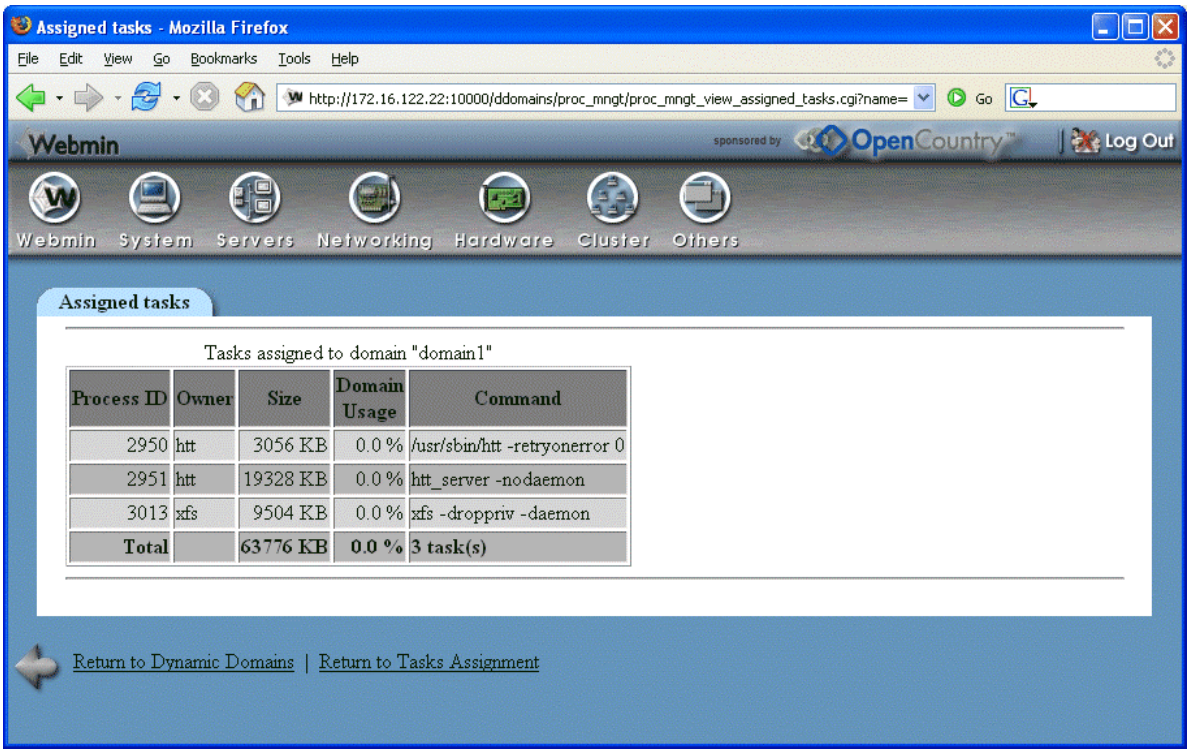
For assigning several tasks with different command names, use "**More matching strings**" to specify several character strings to be searched for.

Caution: With the graphic interface being used in "**normal**" mode, all tasks that do not meet at least one assignment criterion by user name or command name will be assigned to domain "**default**" when clicking on "**Update assignment**".

Click "**View tasks running on the machine**" to view the tasks assigned to the various domains.
Click "**View**" to view the domains loads or the list of tasks assigned to the domains.

Viewing the tasks assigned to a domain

Click "**View**" in the "**Nb tasks**" column for domain "**domain1**" to view the list of tasks assigned to this domain.

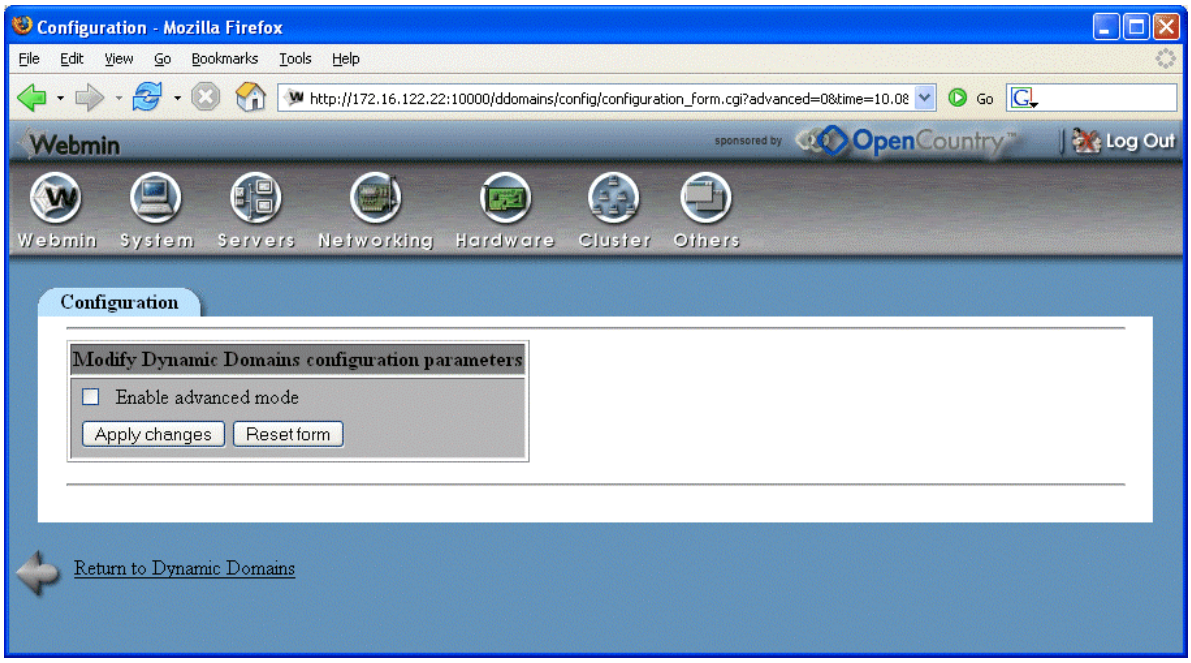


In the example above, two tasks of user "**htt**" and one task containing "**xfs**" in its command name are assigned to domain "**domain1**".

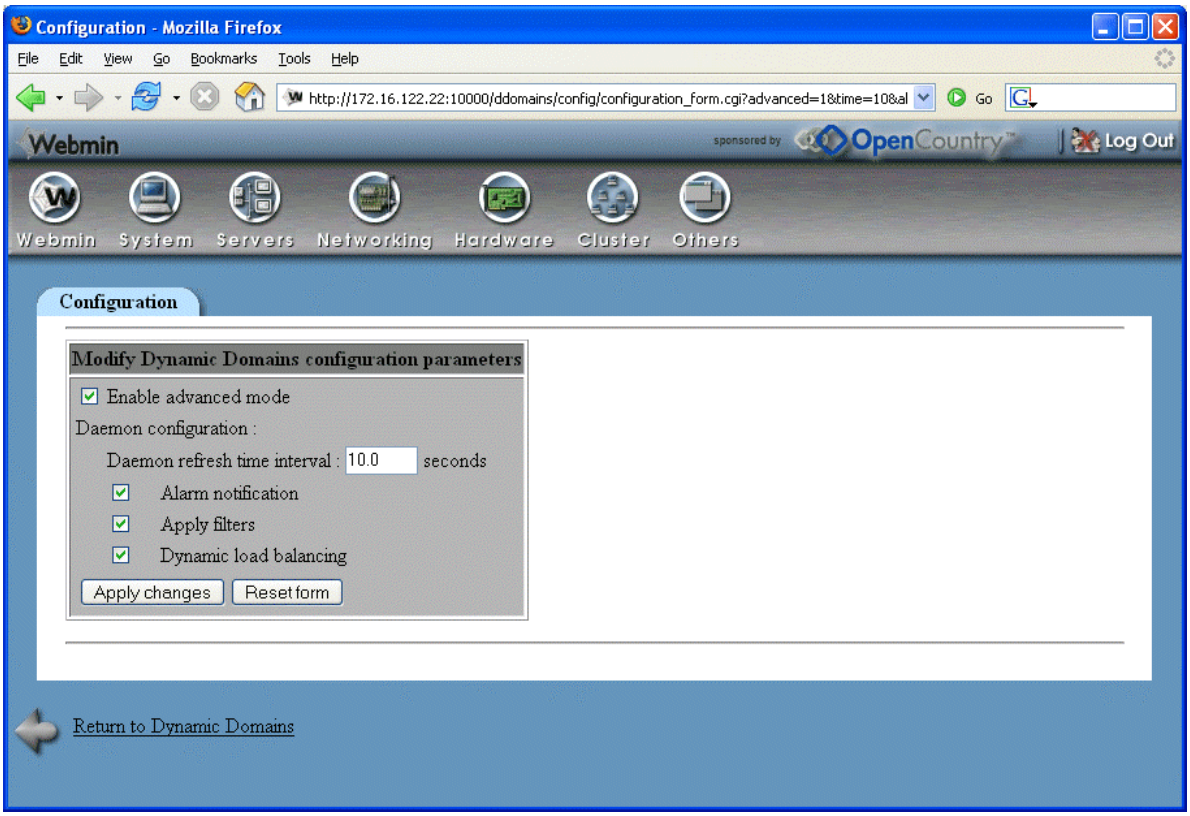
- Click "**Return to Dynamic Domains**" to return to the main screen.

Changing the interface operating mode

Click "**Configuration**" to change the operating mode of the graphic interface.



- Check the "**Enable advanced mode**" box to validate the advanced mode.



- If required, define the daemon parameters (polling period, whether or not alarm, filter and dynamic management are validated) or uncheck the "**Enable advanced mode**" box to validate the normal mode.
- Click the "**Apply changes**" button for changes to be taken into account or click "**Return to Dynamic Domains**" to ignore the changes and return to the main screen.

4. Configuration and Use in Advanced Mode

General

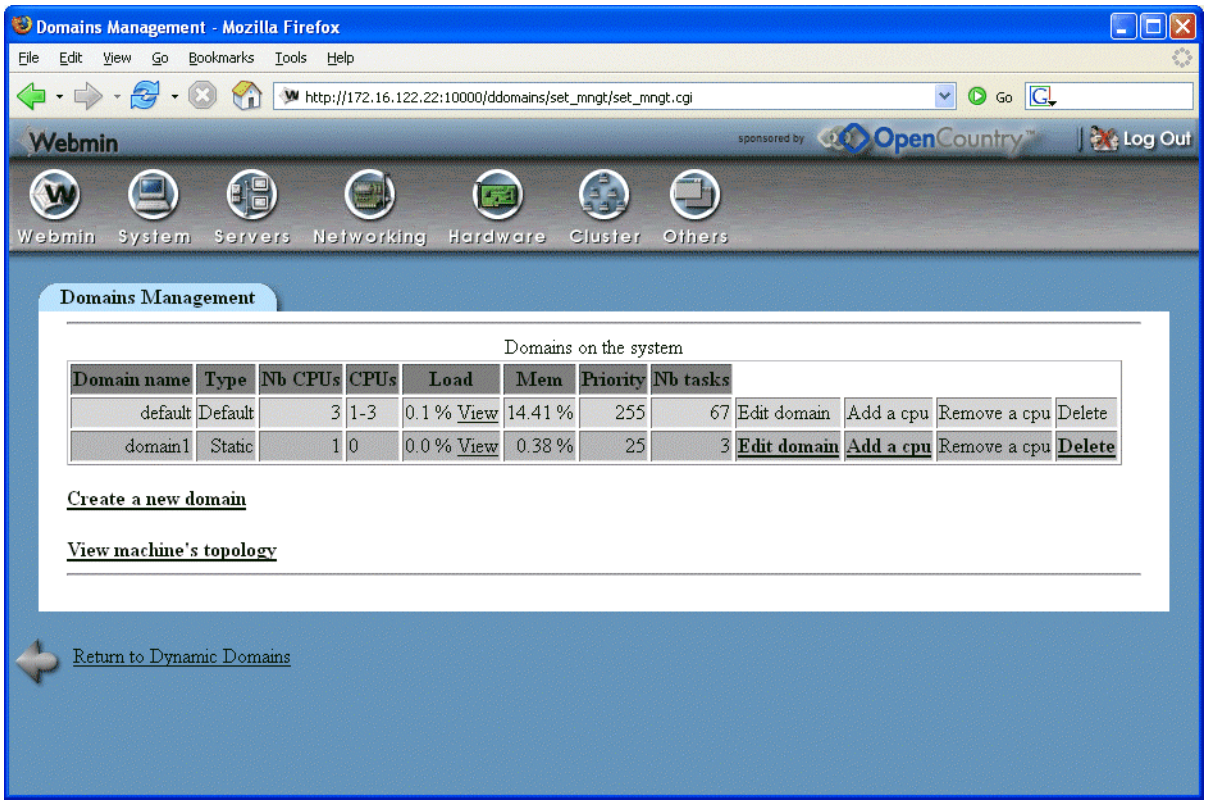
The "Dynamic Domains" software is configured or administered using the "Webmin" tool ("System" category) via a browser (URL: http://<hostname>:10000 or https://<hostname>:10000).
The administration tool can be used in two operating modes:

- a "Normal" mode (default configuration),
- an "Advanced" mode (giving access to all software functions).

Access with Webmin after software installation and "Advanced" mode validation from the configuration screen: System->Dynamic Domains.

Managing a domain

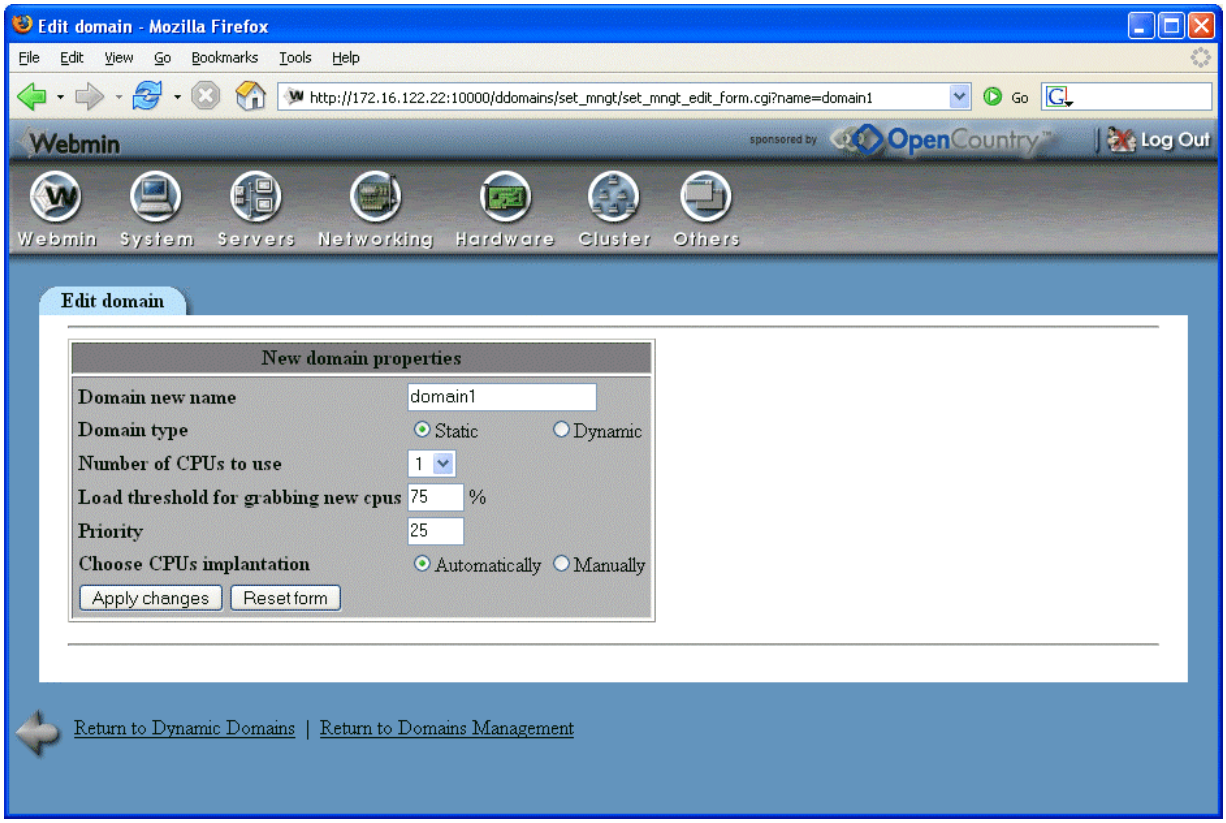
Click "**Domains Management**" to add, edit or delete a domain.



In advanced mode, domains management involves an additional notion: priority.

Changing a domain configuration

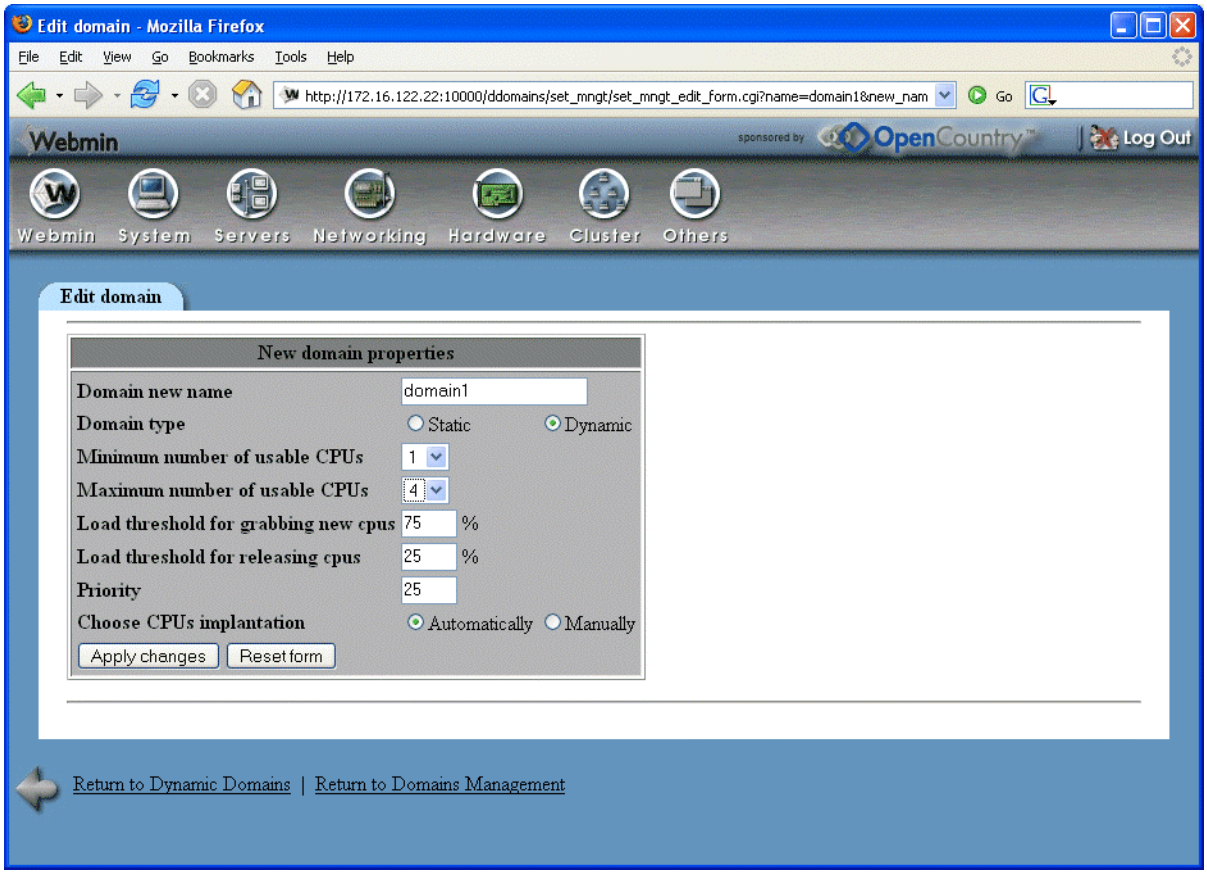
Click "**Edit domain**" for domain "**domain1**" in order to change its configuration.



From this screen, the user can change the domain name, the number of CPUs used, the alarm threshold, the priority, the CPUs used or the domain type (dynamic management).

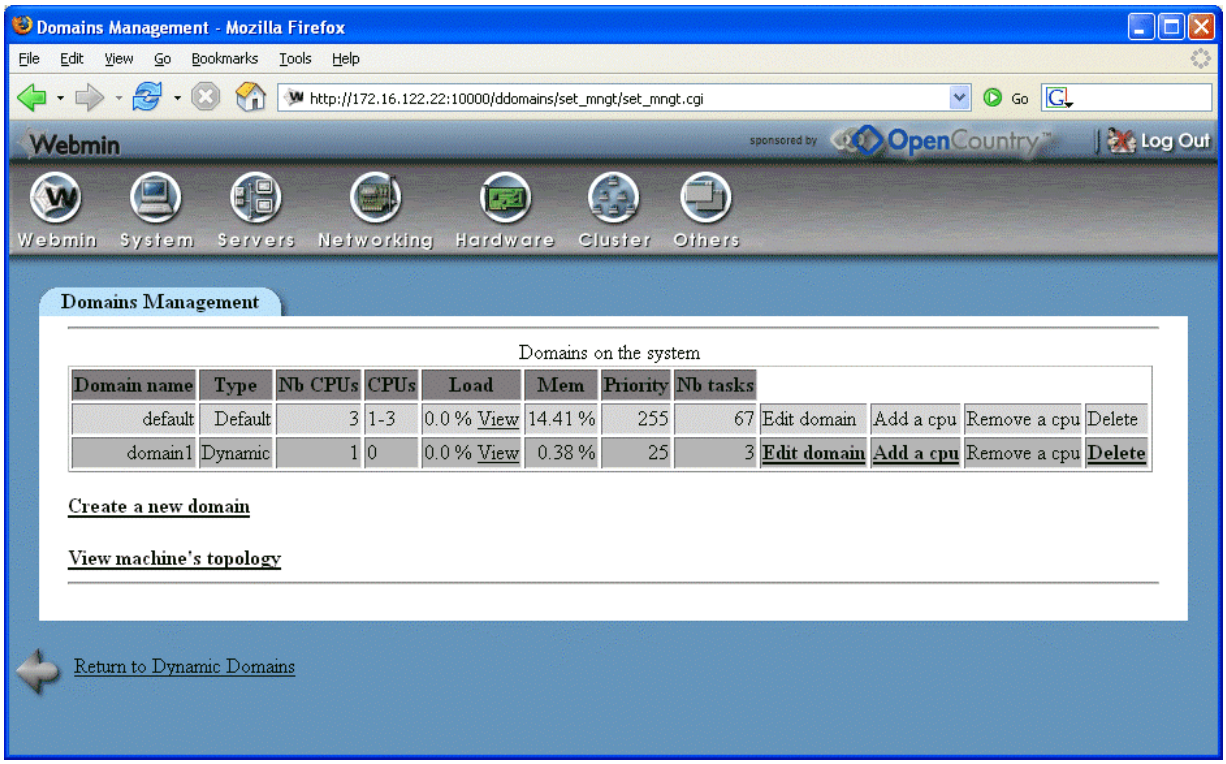
Dynamic management of a domain

Click "**Dynamic**" for dynamic domain management to be enabled.



From this screen, the user can change the domain name, the minimum number of CPUs used, the maximum number of usable CPUs, the alarm threshold or CPU load threshold, the CPU release threshold, the priority and the CPUs used, or return to static domain management.

- Change the "**Maximum number of usable CPUs**" (example: **4**).
- Click "**Apply changes**" for changes to be taken into account.



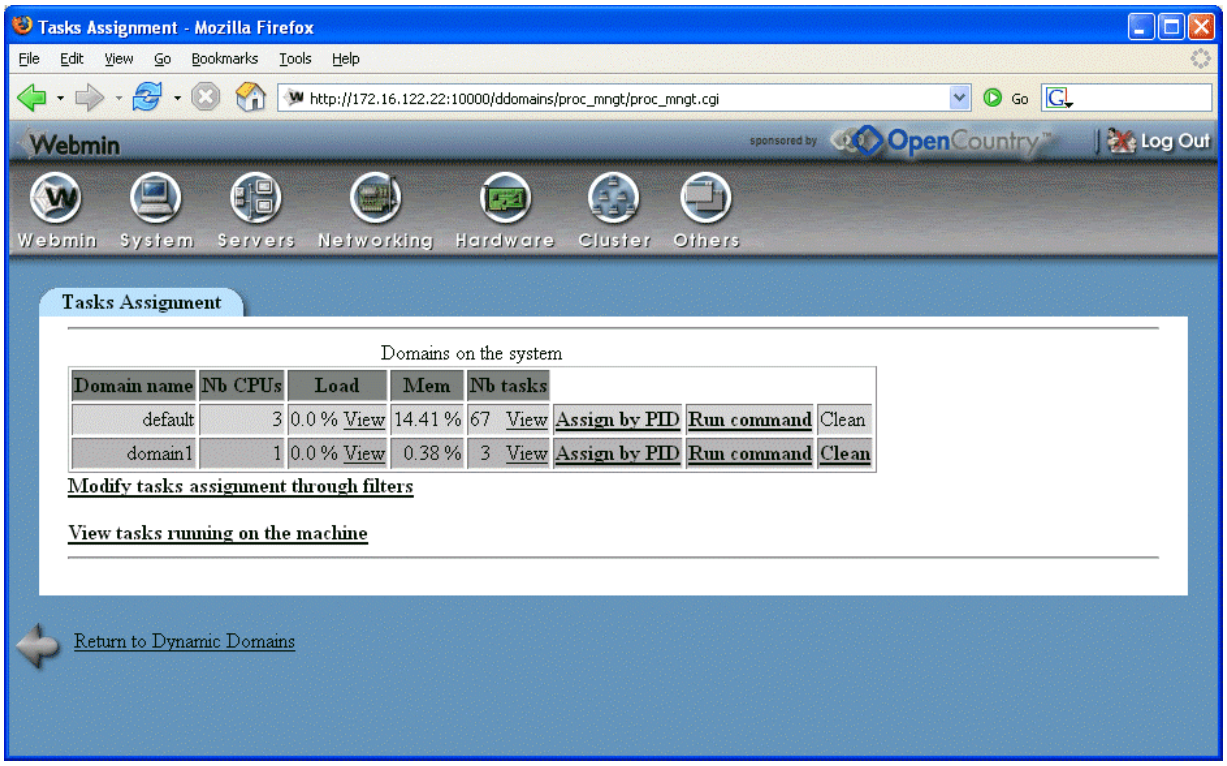
Domain "**domain1**" is now a dynamic domain. The daemon will add CPUs to this domain if its load happens to exceed the associated alarm threshold.

- Click on the following as applicable:
  - "**Add a cpu**" to add a CPU to the domain (for a dynamic domain, this option is deactivated if no CPU is left or if the domain has reached its maximum size).
  - "**Remove a cpu**" to remove a CPU from the domain (for a dynamic domain, this option is deactivated if the domain has reached its minimum size).
  - "**Delete**" to delete the domain.
  - "**Edit domain**" to edit the domain configuration.
  - "**View**" in column "**Load**" to view the load of the domain CPUs.
  - "**View machine's topology**" to check the domain's topology.

- Click "**Return to Dynamic Domains**" to return to the main screen.

Adding or removing tasks for a domain

Click "**Tasks Assignment**" to add or remove a process from the domain.



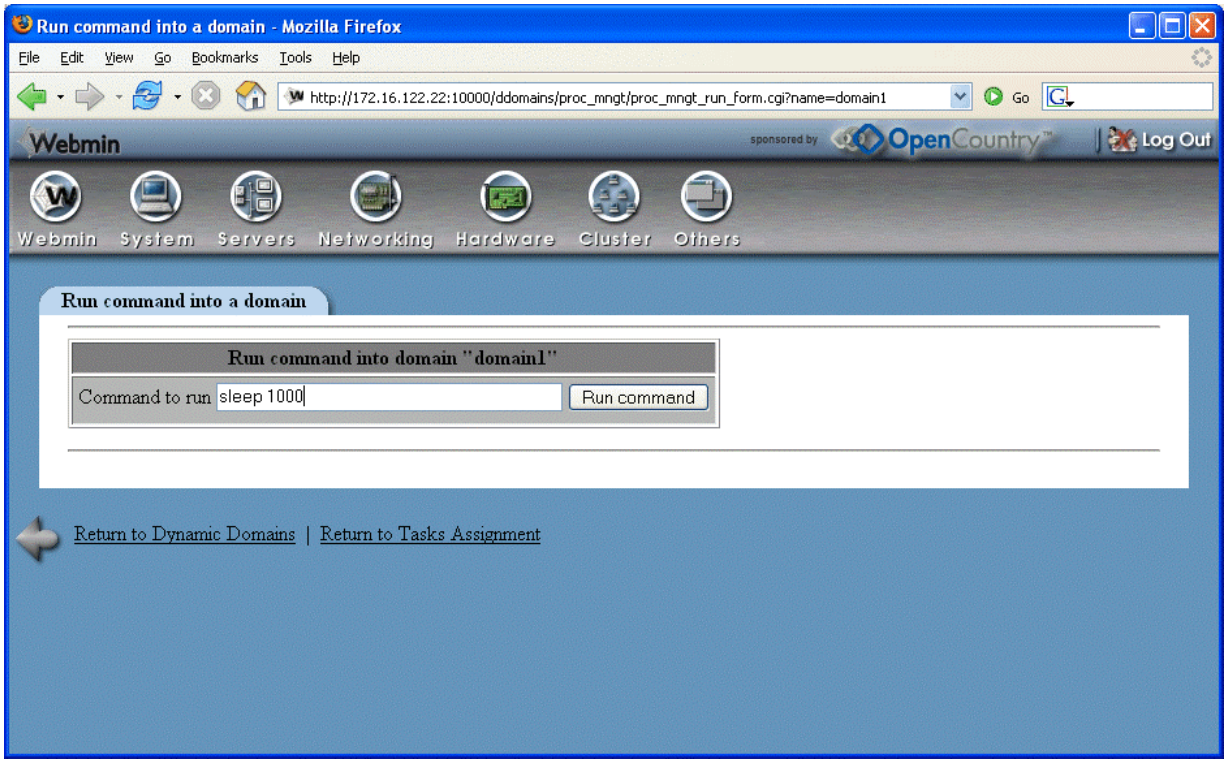
Tasks management in advanced mode can be used for assigning tasks by process number (**Assign by PID**), running a command in a given domain (**Run command**), cleaning a domain (**Clean**) and managing multicriterion filters (**Modify tasks assignment through filters**).

- Click on the following as applicable:

- "**Clean**" to assign to domain "**default**" any tasks in the domain that do not meet the filters criteria.
- "**Run command**" to run a task in the domain.
- "**Assign by PID**" to assign tasks to the domain by process number.
- "**View**" to view the domains loads or the list of tasks assigned to the domains.
- "**Modify tasks assignment through filters**" to define or modify the multicriterion filters.
- "**View tasks running on the machine**" to view the tasks assigned to the various domains.

Running a command in a domain

Click "**Run command**" of domain "**domain1**" to run a command in the domain.



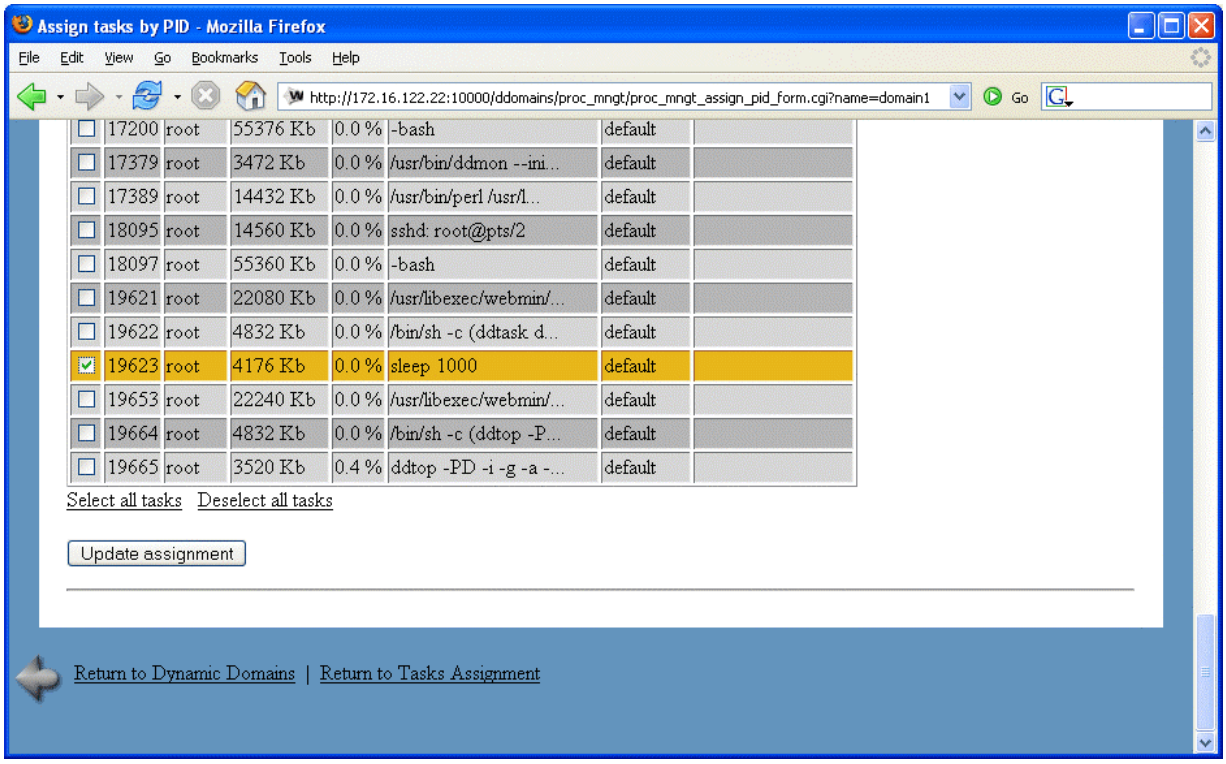
- Enter the name of the command to be run (example "**sleep 1000**").
- Click "**Run command**" to run this command in domain "**domain1**".

An additional task must be assigned to domain "**domain1**".

- Click "**Clean**".
  Task "**sleep 1000**" should have been assigned to domain "**default**" (no filter specifies that this task should be assigned to domain "**domain1**").

Assigning a task by process number

Click "**Assign by PID**" to reassign task "**sleep 1000**" to domain "**domain1**".



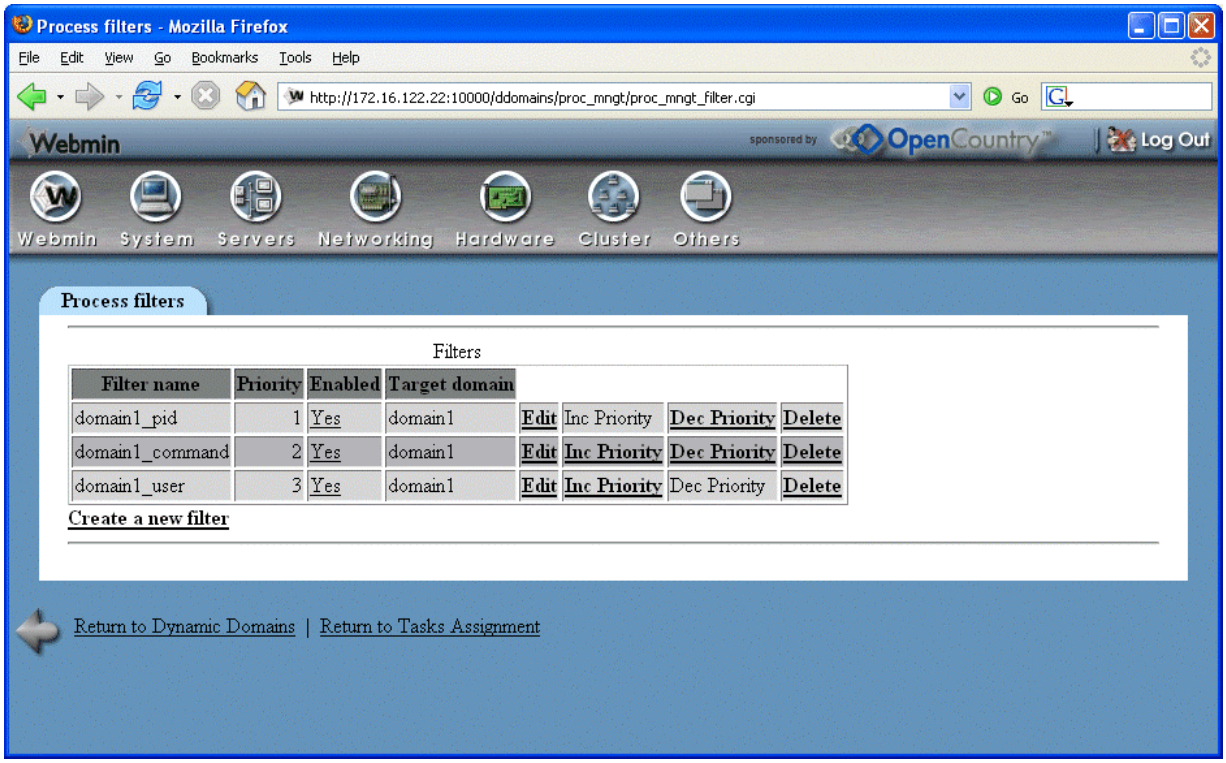
- Check the line corresponding to command "**sleep 1000**".
- Click "**Update assignment**" to validate the selection.

Task "**sleep 1000**" should now be assigned to domain "**domain1**".

- Click "**Clean**" and check that task "**sleep 1000**" is still assigned to domain "**domain1**".

Assigning a task by multicriterion filters

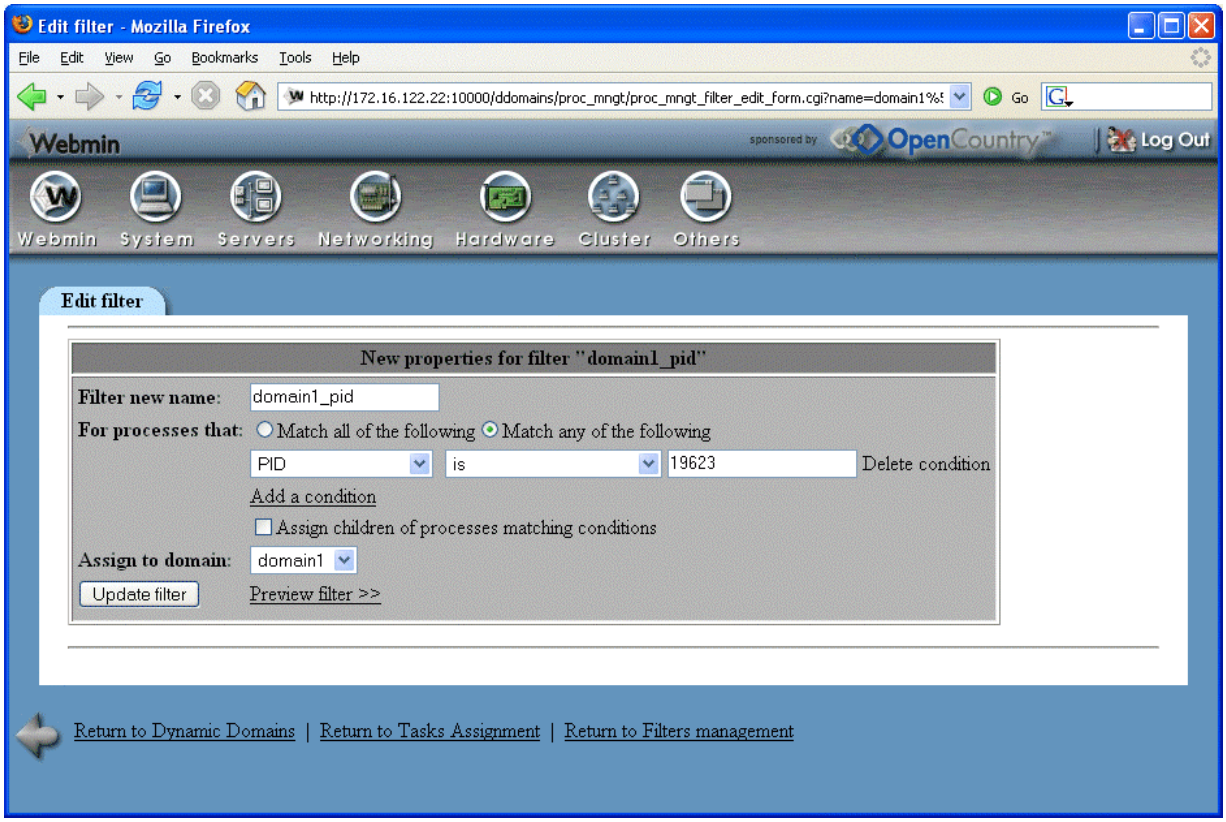
Click "**Modify tasks assignment through filters**" to define or modify the multicriterion filters.



These filters have been automatically created on the previous operations. The user can edit each of these filters (**Edit**), change its priority (**Inc[crease] Priority** or **Dec[crease] Priority**), disable it (**Enabled** = **Yes**) or delete it (**Delete**). The user can also add new filters (**Create a new filter**).

Modifying a filter

Click "**Edit**" for filter "**domain1_pid**", for example.

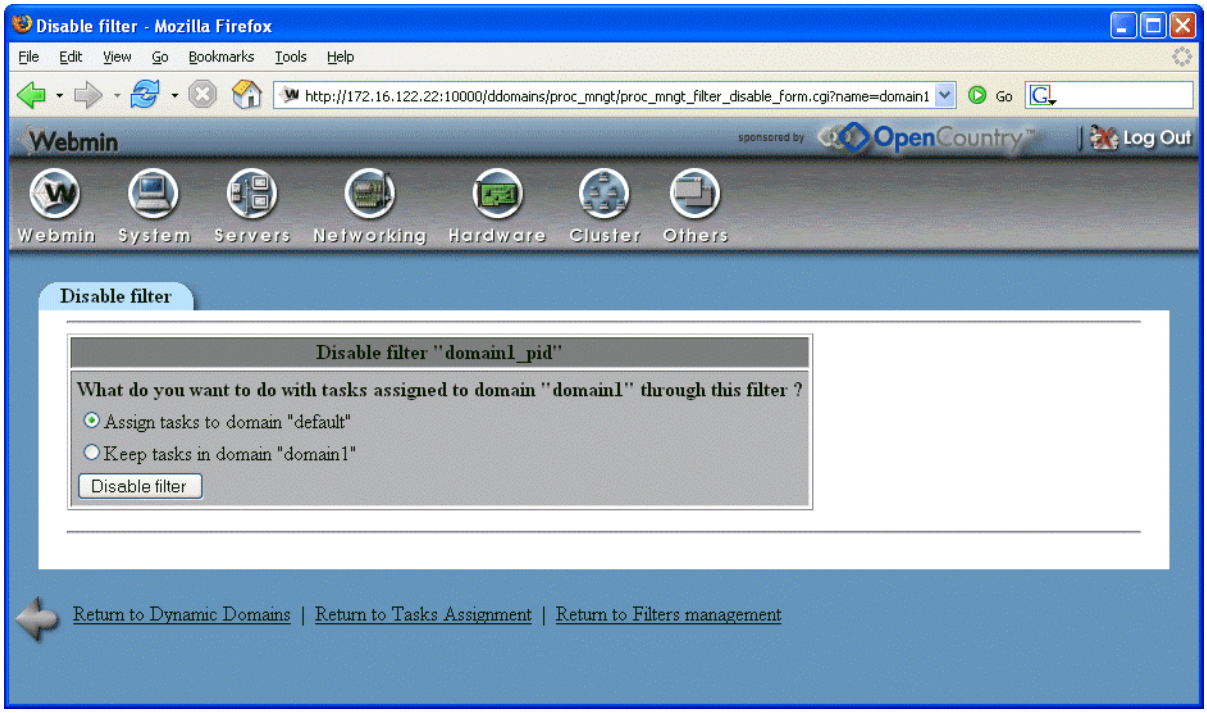


This is the filter defined when assigning tasks by process number (PID). The user can modify, add or delete matching conditions for this filter.

- Click "**Update filter**" for changes to be taken into account or click "**Return to Filters management**" to ignore the changes and return to filters management.

Disabling a filter

Click "**Yes**" in the "**Enabled**" column for filter "**domain1_pid**" in order to disable this filter.



- Choose whether the tasks assigned by the filter will be left in domain "**domain1**" or assigned to domain "**default**".
- Click "**Disable filter**" to disable the filter or click "**Return to Filters management**" to ignore the changes and return to filters management.

- Click "**Return to Dynamic Domains**" to return to the main screen.

5. Commands

The "Dynamic Domains" software installs the following commands:

- ddadd : utility for creating a new domain.
- ddchg : utility for changing the characteristics of a domain.
- ddflt : utility for assigning processes to the domains using the multicriterion filter file.
- ddload : utility for simulating the load of a domain.
- ddls : utility for displaying the domains configuration and the processes assigned to domains.
- ddmon : dynamic domain management and domain monitoring daemon.
- ddrm : utility for destroying a domain.
- ddstat : utility for displaying the domains load status.
- ddtask : utility for assigning processes to the domains (equivalent to "taskset" command for domains).
- ddtop : utility for displaying the CPUs loads and machine's processes (equivalent to commands "top" or "ps").
- ddtopd : daemon for periodically displaying the CPUs loads and machine's processes (daemon for the "ddtop" command).

ddadd

Description

The "ddadd" utility is used for creating a new domain and defining its characteristics (size, priority, thresholds, etc.). The names "-" and "all_cpus" are reserved and domain "default" has specific characteristics. Defining the characteristics of a domain comes to specifying its behaviour (static domain or dynamic domain).

Syntax

General format

```
/usr/bin/ddadd [[-d|--domain] domain] [[-u|--nbused] nbused] [[-n|--nbmin] nbmin]
               [[-m|--nbmax] nbmax] [[-p|--priority] priority]
               [[-t|--threshold] threshold] [[-s|--nbshar] nbshar]
               [[-M|--maxcpus] maxcpus] [[-U|--usedcpus] usedcpus]
               [[-S|--shareven] shareven] [-a] [-dc cnf] [-v]
```

Parameters

`[-d] domain`	Name of the domain to be created (by default: display of domains configuration).
`[-u] nbused`	Number of CPUs used (by default: 0 for domain "default", 1 for the other domains; 0 for using all usable CPUs).
`[-n] nbmin`	Minimum number of CPUs (by default: 1 or nbused).
`[-m] nbmax`	Maximum number of CPUs (by default: all CPUs for domain "default" and 1 for all other domains; 0 for all CPUs). [-p] priority
`[-p] priority`	Priority of the domain (by default: 255 for domain "default" and 25 for the other domains).
`[-t] threshold`	Alarm threshold for the domain's load (by default: 100 for domain "default" and 75 for the other domains).
`[-s] nbshar`	Number of sharable CPUs (by default: 0),
`[-M] maxcpus`	List of CPUs usable by the domain (by default: all CPUs).
`[-U] usedcpus`	List of CPUs used by the domain (by default: automatic selection).
`[-S] shareven`	Threshold for enabling the domain's CPUs to be released or shared (by default: 1/3 Threshold).
`[-a]`	In case of display, display of domains "all_cpus" and "other_cpus".
`[-dc cnf]`	Domains configuration file (by default: /var/ddomains/ddomains).
`[-v]`	Addition of traces.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

Creating the domain "default"

```
ddadd default
ddls
 Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default     1      4     4      255       100       33      0-3    0      39      43
```

Creating a dynamic domain with a size of 1 to 4 CPUs

```
ddadd domain1 1 1 4
ddls
 Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default     1      3     4      255       100       33      1-3    0      39      43
domain1     1      1     4       25        75       25        0    0       0       0
```

ddchg

Description

The "ddchg" utility is used for changing the characteristics of a domain (size, priority, thresholds, etc.). The names "-" and "all_cpus" are reserved and domain "default" has specific characteristics. Changing the characteristics of a domain may result in a change in its behaviour (a static domain may become a dynamic domain and vice versa).

Syntax

General format

```
/usr/bin/ddchg [[-d|--domain] domain] [[-u|--nbused] nbused] [[-n|--nbmin] nbmin]
               [[-m|--nbmax] nbmax] [[-p|--priority] priority]
               [[-t|--threshold] threshold] [[-s|--nbshar] nbshar]
               [[-S|--shareven] shareven] [[-A|--add] addcpus]
               [[-R|--release] release] [[-N|--newname] newname]
               [[-M|--maxcpus] maxcpus] [[-U|--usedcpus] usedcpus]
               [-a] [-dc cnf] [-v]
```

Parameters

`[-d] domain`	Name of the domain to be changed (by default: display of domains configuration).
`[-u] nbused`	Number of CPUs used (by default: no change; 0 for all usable CPUs).
`[-n] nbmin`	Minimum number of CPUs (by default: no change).
`[-m] nbmax`	Maximum number of CPUs (by default: no change; 0 for all CPUs). [-p] priority
`[-p] priority`	Priority of the domain (by default: no change).
`[-t] threshold`	Alarm threshold for the domain's load (by default: no change).
`[-s] nbshar`	Number of sharable CPUs (by default: no change).
`[-S] shareven`	Threshold for enabling the domain's CPUs to be released or shared (by default: no change).
`[-A] addcpus`	Number of CPUs to be added to the domain (by default: 0),
`[-R] release`	Number of CPUs to be removed from the domain (by default: 0),
`[-N] newname`	New name of the domain (by default: no change).
`[-M] maxcpus`	List of CPUs usable by the domain (by default: no change).
`[-U] usedcpus`	List of CPUs used by the domain (by default: no change).
`[-a]`	In case of display, display of domains "all_cpus" and "other_cpus".
`[-dc cnf]`	Domains configuration file (by default: /var/ddomains/ddomains).
`[-v]`	Addition of traces.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Example

Changing domain "domain1" (addition of a CPU)

```
     ddls
 Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default     1      3     4      255       100       33      1-3    0      39      43
domain1     1      1     4       25        75       25        0    0       0       0

     ddchg domain1 -A 1
     ddls
 Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default     1      2     4      255       100       33      2-3    0      39      43
domain1     1      2     4       25        75       25      0-1    0       0       0
```

ddflt

Description

Multicriterion filters are used for assigning tasks (processes or threads) to the domains as per conditions described in the filters configuration file. This file is read by the "ddflt" utility with the following results, as applicable:

- The enabled filters are interpreted and processes or threads that do not meet each filter's criteria are assigned to the domain specified by the filter.
- Tasks already assigned to domains that do not meet the filters' criteria are assigned to domain "default" (option: –default) or to all CPUs (option: –all_cpus).
- Tasks already assigned to a domain (option: -d domain) and that do not meet the filters' criteria are assigned to domain "default" (option: –default) or to all CPUs (option: –all_cpus).
- Tasks that meet the criteria of a filter (option : -F filter) are assigned to domain "default" (option: –default) or to all CPUs (option: –all_cpus).
- List the tasks that can be assigned by a filter (option -p filter).

By default, processes or threads that meet the filters' criteria will be assigned to the respective domains with no change in the assignment of tasks (processes and threads) that fail to meet such criteria.

Syntax

General format

```
/usr/bin/ddflt [-d|--domain domain] [-F|--filter filter] [-p|--preview filter]
               [-f fmt] [+f fmt] [-dc cnf] [-fc cnf] [-U] [-v] [-i|--noid]
               [-v] [--default] [--all_cpus]
```

Parameters

`[-d domain]`	Name of the domain whose tasks not taken into account by he filters will be assigned to domain "default" or to all CPUs (by default: all domains).
`[-F filter]`	Name of the filter whose relevant tasks will be assigned to domain "default" or to all CPUs.
`[-p filter]`	List the tasks that can be assigned by the "filter" filter,
`[-f fmt]`	Definition of parameters edited for the list of tasks listed by the "-p filter" parameter,
`[+f fmt]`	Add parameters edited for the list of tasks listed by the "-p filter" parameter, (format: pid[,...], type "ddls –format" for more details),
`[-dc cnf]`	Domains configuration file (by default: /var/ddomains/ddomains).

`[-fc cnf]`	Filter configuration file (by default: /var/ddomains/filters),
`[-U]`	Edit parameters with units.
`[-i]`	or [–noid]: No edit of parameter meanings.
`[-v]`	Addition of traces.
`[--default]`	Without parameter [-F filter]: tasks assigned to domain [-d domain] or to all domains and not taken into account by the filters are assigned to domain "default"; with parameter [-F filter]: tasks taken into account by the filter are assigned to domain "default".
`[--all_cpus]`	Without parameter [-F filter]: tasks not taken into account by the filters are assigned to all CPUs; with parameter [-F filter]: tasks taken into account by the filter are assigned to all CPUs.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

For tasks taken into account by the filters to be assigned to respective domains

```
ddflt
```

For tasks taken into account by the filters to be assigned to respective domains and for the other tasks to be assigned to domain "default"

```
ddflt –default
```

For tasks taken into account by the filters to be assigned to respective domains and for the other tasks of domain "domain1" to be assigned to domain "default"

```
ddflt domain1 –default
```

Multicriterion filters

Parameters

Multicriterion filters are used for conditioning the assignment of tasks (processes or threads) to domains. A filter is characterized by a set of parameters used for defining its name, the associated domain, its validity, its priority, its matching conditions, etc. The matching conditions of a filter will define criteria for assigning such or such tasks to such or such domain. Multiple conditions can be used.

Characteristics

Name	Name of the filter.
Domain	Name of the domain associated with the filter.
Enabled	Filter processing validity (0: disabled, 1: enabled).
Priority	Filter processing priority or processing order (1: highest priority).
Matching	Filter matching conditions (any: one of the conditions must be met, all: all conditions must be met).
Thread	Applicability to the process specified and to its threads (0: process only, 1: process + threads).
Children	Applicability to the process specified and to its children (0: process only, 1: process + children).
Daemon	Validity of filter processing by the daemon (0: disabled, 1: enabled).

Matching conditions

Field Name of the field involved in the condition:
- **pid** Process number,
- **tid** Thread number,
- **ppid** Children process number,
- **user** Name of task user,
- **group** Name of task group,
- **size** Memory size used by the task,
- **args** Arguments for the task command,

Condition Test condition:
- **eq** The field is equal to the value,
- **ne** The field is different from the value,
- **co** The field contains the value,
- **nc** The field does not contain the value,
- **be** The beginning of the field is equal to the value,
- **en** The end of the field is equal to the value,
- **lt** The field is lower than the value,
- **le** The field is lower than or equal to the value,
- **gt** The field is higher than the value,
- **ge** The field is higher than or equal to the value.

Value Expected value.

Example

For domain "domain1" to be assigned tasks that belong to user "bin", that use at least 512 K of memory and whose command contains the string "test", the filter defined must be associated with domain "domain" (domain = domain1, matching = all) with three conditions (user eq bin, size ge 512, and args co test).

ddload

Description

The "ddload" utility allows the load of a domain to be simulated.

Syntax

General format

```
/usr/bin/ddload [-i|--ident ident] [-d|-ld|--domain domain] [-t|--time time]
                [-l|--load load] [-p|--proc proc] [-m|--mem mem]
                [-dc cnf] [-v]
```

Parameters

[-i ident]	Name for identifying the programme (by default: no special identification),
[-d domain]	Name of the domain to be loaded (by default: no special assignment),
[-t time]	Duration of processing (by default: 10 seconds),
[-l load]	Load for each process (by default: 75% of the CPU used),
[-p proc]	Number of CPUs run (by default: 1 process),
[-m mem]	Size of memory used in bytes (by default: no special use of memory),
[-dc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-v]	Addition of traces.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

Running 4 processes in the domain "domain1" each using 20% of a CPU

```
ddload -d domain1 -l 20 -p 4 &
ddtop -s ddload -f domain,pid,%cpu,cmd
    Domain    Pid    %Cpu Command
   domain1   9242   19.41 "ddload -d domain1 -l 20 -p 4"
   domain1   9243   20.38 "ddload -d domain1 -l 20 -p 4"
   domain1   9244   20.38 "ddload -d domain1 -l 20 -p 4"
   domain1   9245   19.41 "ddload -d domain1 -l 20 -p 4"
```

Running a process in domain "domain1" with an identifier being defined:

```
ddload -i simulation -d domain1 &
ddtop -s simulat -f domain,pid,%cpu,cmd
    Domain    Pid    %Cpu Command
   domain1   9254   74.78 simulation
```

ddls

Description

The "ddls" utility is used for displaying the domains configuration, the list of processes assigned by domain and the multicriterion filters configuration.

Syntax

General format

```
/usr/bin/ddls [-d|ld [d...]] [-P|-p [p...]] [-s str] [-u user] [-F] [-f fmt]
              [+f fmt] [-z] [-a] [-g] [-dc cnf] [-fc cnf] [-U] [-v] [-i|--noid]
```

Parameters

`[-d|ld]`	Display of the domains configuration.
`[d...]`	List of domains involved (by default: all domains; format: domain_name1[,domain_name2][,...] or all).
`[-P]`	Display of processes and threads assigned to the domains.
`[-p]`	Display of processes (without threads) assigned to the domains.
`[p...]`	List of processes (PID) or threads (TID) involved (by default: all; format: pid[-pid][,pid[-pid]][,...] or all).
`[-s str]`	Display of processes containing string str in their command line.
`[-u user]`	Display of processes belonging to user "user".
`[-F]`	Display of filters configuration and assignment conditions.
`[-f fmt]`	Definition of parameters edited for the domains or processes.
`[+f fmt]`	Addition of parameters edited for the domains or processes (format: pid[,...], type "ddls –format" for more details).
`[-z]`	Display of processes using no memory.
`[-a]`	Display of processes not included in a domain.
`[-g]`	Display of global values of domains.
`[-dc cnf]`	Name of the domains configuration file (by default: /var/ddomains/ddomains).
`[-fc cnf]`	Name of the filters configuration file (by default: /var/ddomains/filters),
`[-U]`	Display of parameters with units.
`[-v]`	Addition of traces.
`[-i]` or `[--noid]`	No display of parameters meaning.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

Displaying the domains configuration

```
    ddls
 Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default      1      2     4      255       100       33      2-3   50      43      47
domain1      1      2     4       25        75       25      0-1    0       3       3
```

Displaying the configuration of domain "domain1"

```
    ddls -d domain1
 Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
domain1      1      2     4       25        75       25      0-1    0       3       3
```

Displaying filters configuration

```
    ddls -F
  Filter   Type Enabled Priority     Domain Matching  Threads Children Daemon
 filter1   user       1        1    domain1      any        0        0      0

  Filter    Field Condition Value
 filter1     user        eq  root
```

Displaying threads of domain "oracle"

```
    ddls -d domain1 -P
 Domain       Filter   Pid   Tid      Uid Command
domain1            -  9325  9325     root simulation
domain1            -  9327  9327     root simulation
domain1            -  9328  9328     root simulation
domain1            -  9329  9329     root simulation
domain1            -  9330  9330     root simulation
domain1            -  9331  9331     root simulation
```

Displaying processes of domain "domain1" with edited parameters being defined

```
    ddls -d domain1 -P -f pid,tid,uid,gid,cmd
  Pid   Tid      Uid      Gid Command
 9327  9327     root     root simulation
 9328  9328     root     root simulation
 9329  9329     root     root simulation
 9330  9330     root     root simulation
 9331  9331     root     root simulation
```

ddmon

Description

The monitoring daemon:

- coherency check between hardware configuration and domain configuration after rebooting of the machine.
- periodically assigns processes to domains according to the content of the multicriterion filters definition file,
- manages the alarms,
- dynamically allots resources to domains according to their load levels.

Coherency check

When the machine reboots, the daemon checks for coherency between the hardware configuration (quantity of CPUs and their numbers) and the domain configuration. If an error occurs in the coherency check and if the –R option is set, the daemon attempts to reconfigure the domains automatically.

Periodic assignment

On each daemon processing period, the multicriterion filters definition file is processed. All filters are checked and the various tasks (processes or threads) that are taken into account by filters enabled by the daemon (i.e. with parameter "Daemon" = 1) are assigned to the corresponding domains.

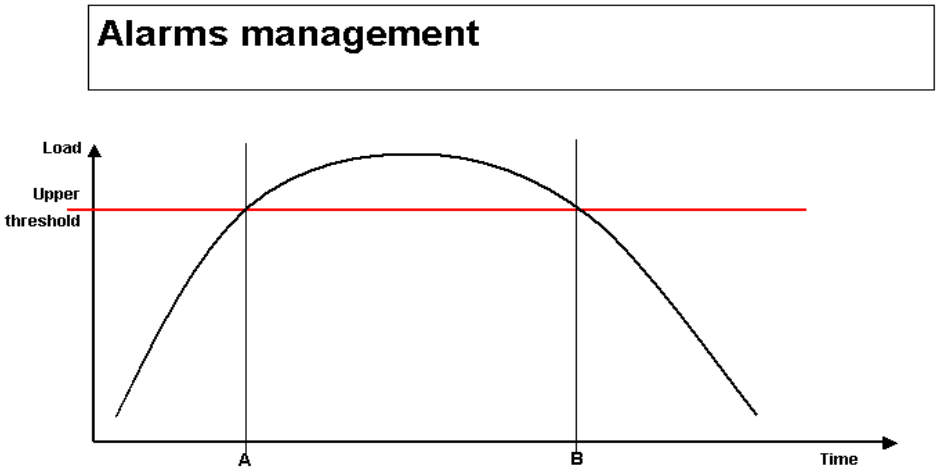
Alarms management

When the load level of a given domain exceeds its upper threshold, an alarm message is edited in the daemon trace file. On return to normal status (end of alarm conditions), an information message is edited in the trace file. The upper threshold value of a domain is defined by the characteristics of this domain.

Example of messages

```
06/01/30 10:07:59 ddmon*2 Domain domain1 : Alarm  threshold (>75%)
06/01/30 10:08:09 ddmon*4 Domain domain1 : Normal threshold
```

Principle

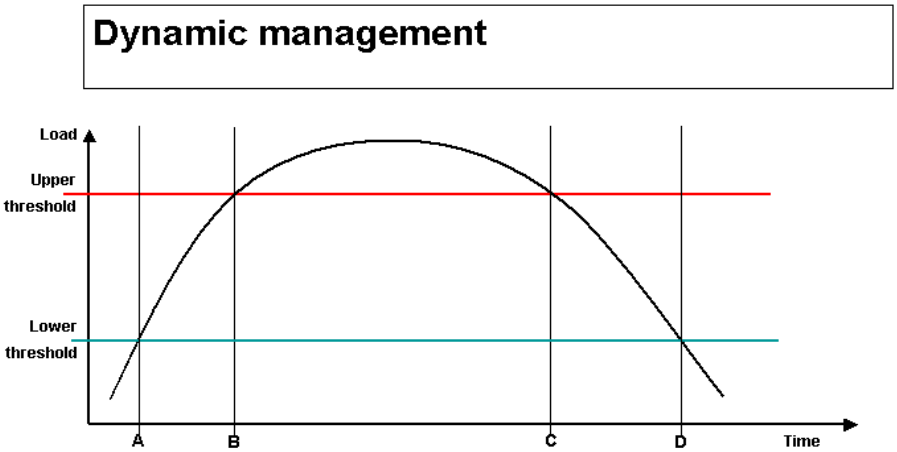


A. Upper threshold exceeded: beginning of alarm
B. Upper threshold no longer exceeded: end of alarm

4

Dynamic allotment of resources

Dynamic processing enables resources to be allotted to the domains or released according to the domains load. Only dynamic domains are managed by this feature (see "Characteristics of domains"). Resources are dynamically managed according to domains thresholds (upper and lower) and priority.



A. Lower threshold exceeded: resource pickup no longer authorized
B. Upper threshold exceeded: additional resources requested
C. Upper threshold no longer exceeded: no resources added
D. Lower threshold no longer exceeded: resource pickup authorized

5

See Appendix C: "Simulation of dynamic domains"

Syntax

General format

```
/usr/bin/ddmon [-t|--time time] [-F|--filters] [-A|--alarms] [-D|--dynamic]
               [-dc cnf] [-fc cnf] [-c cnf] [-h his] [-l log] [--init]
               [-nc cnf] [-R|--reconfig] [-v]
```

By default, command **ddmon** without parameter (/usr/bin/ddmon) is equivalent to /usr/bin/ddmon -F -A -D.

Parameters

[-t time]	Value of the daemon processing period (by default: 10 seconds),
[-F]	Processing of tasks assignment using the multicriterion filters file,
[-A]	Processing of alarms on threshold overshoot.
[-D]	Processing of dynamic domains management.
[-dc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-fc cnf]	Filters configuration file (by default: /var/ddomains/filters),
[-c cnf]	Daemon configuration file (by default: /var/ddtools/ddmon.conf).
[-h his]	Trace file (by default: console).
[-l log]	Trace display level (by default: 4),
[--init]	Creation of domains configuration file and domain "default", and assignment of tasks to domain "default" (by default: no creation, no assignment).
[-nc cnf]	Node configuration file (by default: /var/ddtools/node.conf).
[-R]	Automatic domain reconfiguration after coherency error (by default: no reconfiguration).
[-v]	Addition of traces.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Example

Running of daemon with configurations initialized

```
ddmon --init
ddmon : Initialization enabled
ddmon : Assign tasks to domain default
ddmon : Daemon started
ddmon : Time between measures : 10.000s
ddmon : Filters management started
ddmon : Alarms management started
ddmon : Dynamic management started
```

Daemon check service

When the product is installed, the **ddmon** daemon is activated via the **ddmon** service (file: /etc/init.d/ddmon). This service allows the operation of the daemon to be checked (starting it, stopping it or checking its status).

Running the daemon

```
/etc/init.d/ddmon start
Starting ddmon  OK
```

Status of the daemon

```
/etc/init.d/ddmon status
ddmon is running (pid=21264) since 06/06/20 08:25:35
```

Stopping the daemon

```
/etc/init.d/ddmon stop
Stopping ddmon  OK
```

Restarting the daemon (stopping and running)

```
/etc/init.d/ddmon restart
Stopping ddmon  OK
Starting ddmon  OK
```

Daemon configuration file

The configuration file defines the parameters and processing managed by the daemon. When the daemon is run, the configuration file (if it exists) defines the default values for the parameters of the daemon. At any time (daemon stopped or running), the user can edit some of the daemon's processing parameters via the configuration file (change the polling period, enable or disable a process, etc.).

Example of a configuration file

```
cat /var/ddtools/ddmon.conf
ddmon_enable = 1
ddmon_pid = 21264
start_time = 1150784735
log_file = /var/ddtools/ddmon.log
log_level = 4
init = 1
reconfig = 0
loop_time = 10.000
filters = 1
alarms = 1
dynamic = 1
ddmon_status = 1
```

Parameter meanings

ddmon_enable	Whether or not the daemon is enabled (value 0 or 1; 1: daemon enabled),
ddmon_pid	Daemon process number (do not change),
start_time	Daemon run date (in seconds since 01/01/1970; do not change),
log_file	Name of the daemon trace file (by default: /var/ddtools/ddmon.log),
log_level	Trace message edit level (by default: 4).
init	Process initialisation when run (value 0 or 1; 1: initialisation enabled).
reconfig	Process automatic domain reconfiguration when run (value 0 or 1; 1: reconfiguration enabled).
loop_time	Daemon process time in seconds (by default: 10 seconds).
filters	Periodic filter processing (value 0 or 1; 1: processing enabled).
alarms	Periodic alarm processing (value 0 or 1; 1: processing enabled).
dynamic	Periodic dynamic domain processing (value 0 or 1; 1: processing enabled).
ddmon_status	Status of the daemon (0: stopped, 1: running, -1: stopped after an error).

ddrm

Description

The "ddrm" utility is used for removing a domain in the domains configuration file. By default, a given domain can be removed only if no task (process or thread) is assigned to it. If tasks are still assigned to the domain to be removed, the user can force its removal (parameter: –remove or –rmonly) and can specify whether or not the assigned tasks will be reassigned to the "default" domain.

Syntax

General format

```
/usr/bin/ddrm [[-d|--domain] domain] [--remove] [--rmonly] [-a] [-dc cnf] [-v]
```

Parameters

[-d] domain	Name of the domain to be removed (by default: display of domains configuration).
`[--remove]`	Forcing of domain removal when tasks are still assigned to the domain.
`[--rmonly]`	Removal of the domain without reassigning its tasks to domain "default".
`[-a]`	In case of display, display of domains "all_cpus" and "other_cpus".
`[-dc cnf]`	Domains configuration file (by default: /var/ddomains/ddomains).
`[-v]`	Addition of traces.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Example

Removing domain "domain1"

```
        ddls
Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default    1      3     4      255       100       33      1-3    0      39      43
domain1    1      1     4       25        75       25        0    0       0       0

        ddrm domain1
        ddls
Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default    1      4     4      255       100       33      0-3    0      39      43
```

ddstat

Description

The "ddstat" utility is used for displaying the domains load status.

Syntax

General format

```
/usr/bin/ddstat [[-d|--domain] domain] [-t|--time time] [-a] [-dc cnf] [-v]
```

Parameters

[-d] domain	Name of the domain to be displayed (by default: all domains except the "default" domain),
[-t] time	Duration of load measurement (by default: 0.333 seconds).
[-a]	Display of domain "default" in addition to other domains, if required.
[-dc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-v]	Addition of traces.

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.
3 and above: one or more domains in alarm condition (number of domains involved = return code - 2).

Example

Displaying the status of domain "domain1"

```
ddstat domain1
ddstat : Domain    domain1 : Load value: 25% - Normal threshold
```

ddtask

Description

The "ddtask" utility is used for assigning tasks (processes or threads) to a domain. This can be done by list of process numbers (PID), by list of thread numbers (TID), by user name (UID) or by command name (character string contained in the process command line). This utility can also be used for directly running a task in a domain.

Syntax

General format

```
/usr/bin/ddtask domain [-dc cnf] [-v] [[-P|-p [p...]] [-s str] [-u user]]
      | [-c cmd[ args ...]]
```

Parameters

domain	Name of the target domain for assignment or running of the task,
[-dc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-v]	Addition of traces.
[-P]	Assignment of processes and associated threads.
[-p]	Assignment of processes only.
[p...]	List of processes (PID) or threads (TID) (by default: tous (all), format: pid[-pid][,pid[-pid]][,...] or all),
[-s str]	Assigning processes that contain the string "str" in the command, or ^str if the command starts with "str", or str$ if the command ends with "str", or ^str$ if the command is equal to "str",
[-u user]	Assignment of processes belonging to user "user".
[-c cmd]	Direct running of command "cmd" in a domain.
[args ...]	Arguments for command "cmd".

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

Assigning a process and its threads by process number

```
ddtask jonas -P 25695
```

Assigning processes and threads of a user

```
ddtask jonas -u invite
```

Running a command in a domain

```
ddtask jonas -c sleep 10
```

ddtop

Description

The "ddtop" utility is used for displaying various CPUs parameters as well as active nodes and processes of the machine. Parameters are displayed according to the level of detail requested (options: -d and -D) or can be redefined by the user (options: -f and +f). The expected results are displayed in chronological order (options: -C, -L, -P, -PM, -p, -pM, -M) or sorted by domain (options: -CD, -LD, -PD, -PMD, -pD, -pMD, -MD).

Syntax

General format

```
/usr/bin/ddtop [-t time] [-C[D]|-N[C]|-m][-L[D]|-P[M][D]|-p[M][D]|-M[D]]
               [v...] [-ld dom] [-s str] [-u user] [-f fmt] [+f fmt]
               [-g|-G] [-d|-D] [-U] [-z] [-a] [-n cpu] [-r] [-dc cnf]
               [-fc cnf] [-c cnf] [-v] [-i|--noid] [--gbld] [--gbl] [--all]
```

By default, command **ddtop** without parameter (**/usr/bin/ddtop**) is equivalent to **/usr/bin/ddtop -C -D -U**.

Parameters

[-t time]	Interval between two measurements or display (by default: 1.00 seconds).
[-C[D]]	display of CPUs load (CPU number x, y,...). D: sorted by domain name.
[-N[C]]	Display of nodes configuration (node number x, y,...). C: with CPU load.
[-m]	Display of memory used by node (node number x, y,...).
[-L[D]]	Display of list of threads by process (PID x, y,...). D: sorted by domain name.
[-P[M][D]]	Display of load by process and thread (PID x, y,...). M: addition of used memory. D: sorted by domain name.
[-p[M][D]]	Display of load by process without threads (PID x, y,...). M: addition of used memory. D: sorted by domain name.
[-M[D]]	Display of list of threads by process (PID x, y,...). D: sorted by domain name.
[v...]	List of CPU, node and process numbers (by default: all; format: a[-b][,c[-d]][,...] or all).
[-ld dom]	Display of parameters for the list of domains "dom" (by default: all; format: domain_name1[,domain_name2][,...] or all).
[-s str]	Display if the command line contains string "str".
[-u user]	Display for user "user".
[-f fmt]	Definition of parameters displayed for nodes, CPUs, etc.

[+f fmt]	Addition of parameters displayed for nodes, CPUs, etc. (format: pid[,...], type "ddtop –format" for more details)
[-g]	Global display of parameters (loads, memory, etc.) by item.
[-G]	Global display of parameters (loads, memory, etc.) for all items.
[-d]	Detailed display of parameters.
[-D]	Display with a maximum of parameters details.
[-U]	Display of parameters with units.
[-z]	Display of processes using no memory.
[-a]	Display of processes not included in a domain.
[-n cpu]	Number of CPUs used for computing loads (by default: 1; cpu = 1 or maximum number of machine's CPUs).
[-r]	Display every "-t time" seconds.
[-dc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-fc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-v]	Addition of traces.
[-i] or **[--noid]**	No display of parameters meaning.
[--gbld]	Display of sum for each domain.
[--gbl]	Display of global sum.
[--all]	Display of sum for the whole machine.
[stdin]	Value input on standard input (stdin) (same as [v...]).

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

Displaying domain name and load of CPUs 3 to 5

```
ddtop -CD 3-5
    Domain Cpus      Cpu  %Used  %Idle  %Wait
   domain4 3           3   0.00 100.00   0.00
   domain5 4           4   0.00 100.00   0.00
   domain6 5           5   0.00 100.00   0.00
```

Displaying nodes configuration

```
ddtop -NC
Node Cpus       %Cpu %MemU. Distance
   3 12-15      0.09   1.24 30,30,30,10
   2 8-11       0.09   1.76 30,30,10,30
   1 4-7        0.09   0.54 30,10,30,30
   0 0-3        0.09   5.42 10,30,30,30
```

Displaying CPUs 4 to 7 with selection of edited parameters

```
ddtop -CD 4-7 -f domain,cpu,%user,%syst,%idle
    Domain  Cpu  %User  %Syst  %Idle
   domain5    4   0.00   0.00 100.00
   domain6    5   0.00   0.00 100.00
   domain7    6   0.00   0.00 100.00
   domain8    7   0.00   0.00 100.00
```

Displaying processes whose command contains "tst", with addition of the command name

```
ddtop -pD -s tst +f cmd
    Domain   Pid   %Cpu Command
   domain8 18782   0.00 ./tst
   domain8 11519   0.00 ./tst
   domain8 11521   0.00 ./tst
  domain10 11520   0.00 ./tst
```

Displaying processes and associated threads whose command contains "tst", with addition of the command name

```
ddtop -PD -s tst +f cmd
    Domain   Pid   Tid   %Cpu Command
   domain7 11519 11522   0.00 -
   domain7 11521 11525   0.00 -
   domain8 18782 18782   0.00 ./tst
   domain8 11519 11519   0.00 ./tst
   domain8 11519 11524   0.00 -
   domain8 11521 11521   0.00 ./tst
   domain8 11521 11526   0.00 -
   domain9 11520 11523   0.00 -
  domain10 11520 11520   0.00 ./tst
  domain10 11520 11527   0.00 -
```

Displaying processes and associated threads with selection of PIDs

```
ddtop -PD 11534-11536,18782 +f cmd
    Domain   Pid   Tid   %Cpu Command
   domain7 11534 11537   0.00 -
   domain7 11536 11540   0.00 -
   domain8 18782 18782   0.00 ./tst
   domain8 11534 11534   0.00 ./tst
   domain8 11534 11539   0.00 -
   domain8 11536 11536   0.00 ./tst
   domain8 11536 11542   0.00 -
   domain9 11535 11538   0.00 -
  domain10 11535 11535   0.00 ./tst
  domain10 11535 11541   0.00 -
```

Displaying a list of threads for processes whose command contains "tst"

```
ddtop -LD -s tst -g
    Domain   Pid Tid ...
   default 18782
   default 24612 24617 24618
   default 24614 24619 24621
   default 24615 24622 24623
```

ddtopd

Description

The "ddtop" daemon allows you to list various parameters for CPUs, nodes and active processes on the machine to the console or in a trace file. Parameters are edited according to the level of detail requested (options: -d and -D) or can be redefined by the user (options: -f and +f). The results are edited in chronological order (options: -C, -L, -P, -PM, -p, -pM, -M) or classified by domain (options: -CD, -LD, -PD, -PMD, -pD, -pMD, -MD).

Syntax

General format

```
/usr/bin/ddtopd [-t time] [-C[D]|-N[C]|-m][-L[D]|-P[M][D]|-p[M][D]|-M[D]]
                [v...] [-ld dom] [-s str] [-u user] [-f fmt] [+f fmt]
                [-g|-G] [-d|-D] [-U] [-z] [-a] [-n cpu] [-dc cnf] [-fc cnf]
                [-c cnf] [-h hist] [-l log] [-i|--noid] [--gbld] [--gbl] [--all]
```

By default, command **ddtopd** without parameter (**/usr/bin/ddtopd**) is equivalent to **/usr/bin/ddtopd -C -D -U**.

Parameters

[-t time]	Interval between two measurements or display (by default: 1.00 seconds).
[-C[D]]	display of CPUs load (CPU number x, y,...). D: sorted by domain name.
[-N[C]]	Display of nodes configuration (node number x, y,...). C: with CPUs load.
[-m]	Display of memory used by node (node number x, y,...).
[-L[D]]	Display of list of threads by process (PID x, y,...). D: sorted by domain name.
[-P[M][D]]	Display of load by process and thread (PID x, y,...). M: addition of used memory. D: sorted by domain name.
[-p[M][D]]	Display of load by process without threads (PID x, y,...). M: addition of used memory. D: sorted by domain name.
[-M[D]]	Display of list of threads by process (PID x, y,...). D: sorted by domain name.
[v...]	List of CPU, node and process numbers (by default: all; format: a[-b][,c[-d]][,...] or all).
[-ld dom]	Display of parameters for the list of domains "dom" (by default: all, format: domain_name1[,domain_name2][,...] or all),
[-s str]	Display if the command line contains string "str".
[-u user]	Display for user "user".
[-f fmt]	Definition of parameters edited for nodes, CPUs, etc.

[+f fmt]	Addition of parameters displayed for nodes, CPUs, etc. (format: pid[,...], type "ddtop –format" for more details)
[-g]	Global display of parameters (loads, memory, etc.) by item.
[-G]	Global display of parameters (loads, memory, etc.) for all items.
[-d]	Detailed display of parameters.
[-D]	Display with a maximum of parameter details.
[-U]	Edit parameters with units.
[-z]	Display of processes using no memory.
[-a]	Display of processes not included in a domain.
[-n cpu]	Number of CPUs used for computing loads (by default: 1; cpu = 1 or maximum number of machine's CPUs).
[-dc cnf]	Name of the domains configuration file (by default: /var/ddomains/ddomains).
[-fc cnf]	Domains configuration file (by default: /var/ddomains/ddomains).
[-h hist]	Name of history trace file (by default: console).
[-l log]	Trace message edit level (by default: 4).
[-i] or **[--noid]**	No edit of parameter meanings.
[--gbld]	Display of sum for each domain.
[--gbl]	Display of global sum.
[--all]	Display of sum for the whole machine.
[stdin]	Value input on standard input (stdin) (same as [v...]).

Return codes

0 : Running OK.
1 : Access error.
2 : Error in a parameter.

Examples

Displaying a domain name and load of CPU 3 to 5 every 10 seconds

```
ddtopd -CD 3-5 -t 10
ddtopd: Daemon started
ddtopd: Time between measures : 10.000s
Domain   Cpus     Cpu      %Used    %Idle     %Wait
test     3        3        0.00     100.00     0.00
test1    4        4        0.00     100.00     0.00
test2    5        5        0.00     100.00     0.00

Domain   Cpus     Cpu      %Used    %Idle     %Wait
test     3        3        0.00     100.00     0.00
test1    4        4        0.00     100.00     0.00
test2    5        5        0.00     100.00     0.00
...
```

Displaying node configuration every 2 seconds

```
ddtopd -NC -t 2
ddtopd: Daemon started
ddtopd: Time between measures : 2.000s
Node     Cpus     %Cpu     %MemU.   Distance
0        0-15     0.23     44.05    10,30
1        16-31    0.00     0.34     30,10

Node     Cpus     %Cpu     %MemU.   Distance
0        0-15     0.09     44.05    10,30
1        16-31    0.00     0.34     30,10
...
```

Displaying total load for CPUs 0 to 7 every second

```
ddtopd -C 0-7 -g -t 1
ddtopd: Daemon started
ddtopd: Time between measures : 1.000s
cpu0     cpu1     cpu2     cpu3     cpu4     cpu5     cpu6     cpu7
0.00     1.37     0.00     0.00     0.00     0.00     0.10     0.00
0.00     0.58     0.00     0.00     0.00     0.00     0.58     0.00
...
```

Displaying the total domain load every second

```
ddtopd -CD -g -t 1
ddtopd: Daemon started
ddtopd: Time between measures : 1.000s
default  test     test1    test2
0.05     0.15     0.00     0.00
0.08     0.05     0.00     0.00
0.05     0.34     0.00     0.00
...
```

List to file for processes and their threads that have "test1" in their command with the addition of the command name

```
ddtopd -PD -s test1 +f cmd -t 1 -h trace.log &
tail -f trace.log
06/06/19 17:15:24 ddtopd: Daemon started
06/06/19 17:15:24 ddtopd: Time between measures : 1.000s
06/06/19 17:15:25 Domain   Pid     Tid     %Cpu  Command
06/06/19 17:15:25 test1   25357  25357    84.02  test1
06/06/19 17:15:26 test1   25357  25357    85.03  test1
06/06/19 17:15:27 test1   25357  25357    84.91  test1
06/06/19 17:15:28 test1   25357  25357    85.04  test1
06/06/19 17:15:29 test1   25357  25357    84.93  test1
...
```

Daemon check service

The **ddtopd** service allows the operation of the **ddtopd** daemon to be checked (starting it, stopping it or checking its status). The **ddtopd** daemon is not enabled when the product is installed.

Running the daemon

```
/etc/init.d/ddtopd start
Starting ddtopd  OK
```

Status of the daemon

```
/etc/init.d/ddtopd status
ddtopd is running (pid=21367) since 06/06/20 09:59:47
```

Stopping the daemon

```
/etc/init.d/ddtopd stop
Stopping ddtopd  OK
```

Restarting the daemon (stopping and running)

```
/etc/init.d/ddtopd restart
Stopping ddtopd  OK
Starting ddtopd  OK
```

Daemon configuration file

The configuration file defines the daemon's parameters. When the daemon is run, the configuration file (if it exists) defines the default values for the parameters of the daemon. At any time (daemon stopped or running), the user can edit some of the daemon's processing parameters via the configuration file (change the polling period, enable or disable a process)

Example of a configuration file

```
cat /var/ddtools/ddtopd.conf
ddtopd_enable = 1
ddtopd_pid = 26750
start_time = 1150782757
loop_time = 60.000
```

Parameter meanings

ddtopd_enable	Whether or not the daemon is enabled (value 0 or 1; 1: daemon enabled),
ddtopd_pid	Daemon process number (do not change),
start_time	Daemon run date (in seconds since 01/01/1970; do not change),
loop_time	Daemon process time in seconds (by default: 60 seconds).

6. Characteristics of domains

Parameters of a domain

A domain is characterized by a set of parameters used for defining its type (static or dynamic), size, priority, etc.

Domain size

NbMin: Minimum number of resources guaranteed for the domain.
NbUsed: Number of resources used by the domain.
NbMax: Maximum number of resources usable by the domain.

Domain priority

Priority: Priority in resources allotment to the domain
(1: highest priority, 255: lowest priority).

Domain thresholds

Threshold: Upper threshold for alarm management and adding resources.
SharEven: Lower threshold for authorizing resources pickup.

CPUs mask

UsedCpus: List of CPUs used by the domain.
MaxCpus: List of CPUs usable by the domain.

Static domain

A domain is static when its parameters NbMin and NbMax have the same value (no change in its size).

Dynamic domain

A domain is dynamic when its parameters NbMin and NbMax are different. These parameters allow the number of domain resources to be varied. The dynamic aspect of a domain is managed by the daemon (ddmon) in the following conditions:

- The domain load level exceeds its upper threshold.
- The domain size (number of resources) is lower than its maximum size (number of CPUs used < maximum number of CPUs).
- Another domain with lower priority is able to release some resources (number of CPUs used > minimum number of CPUs).
- A domain with an equal or higher priority and with a load level lower than its lower threshold is able to release some resources.

Domain "default"

Domain "default" is similar to the other domains as regards the following characteristics: priorities management, load thresholds management, dynamic management or not. It does not behave like the other domains in the following cases:

- On tasks assignment, it may be automatically assigned tasks that are not taken into account by other domains.
- In case resources are reduced for a given domain (release of a CPU), all released resources are allotted to domain "default".
- If minimum resources are guaranteed for domain "default", no resource is reserved for it (another domain can take any resource from domain "default" within the limit of its guaranteed minimum).

Domain "default" cannot be removed using the graphics interface.

A. Checking the installation

Checking the domain "default"

Once the "Dynamic Domains" software is installed, a domain called "default" should be created automatically. This domain must use all the machine's CPUs.
To check the configuration of domain "default", enter the following:

```
ddls
Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
default    1      4     4      255       100       33      0-3    0      68      82
```

Checking the daemon

Once the "Dynamic Domains" software is installed, the monitoring daemon ("ddmon") must be running.
To display the daemon status, enter the following:

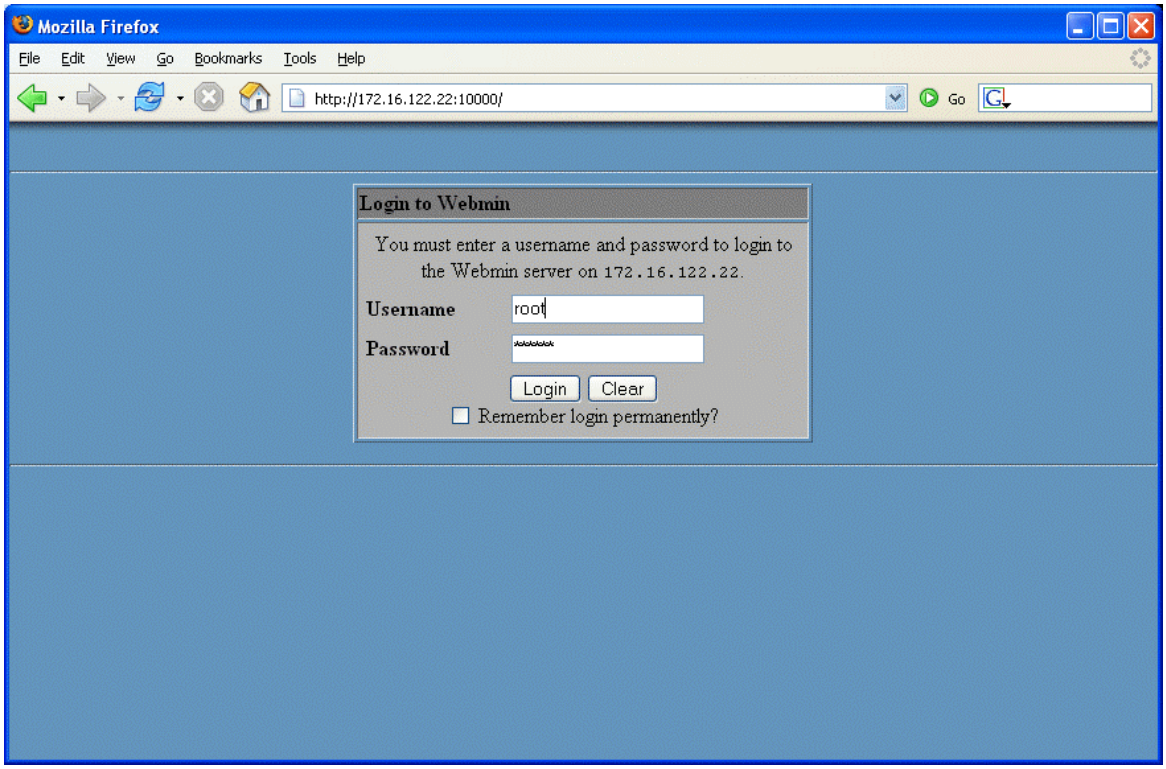
```
/etc/init.d/ddmon status
ddmon is running (pid=17379) since 06/02/07 07:46:18
```

To display the daemon trace file, enter the following:

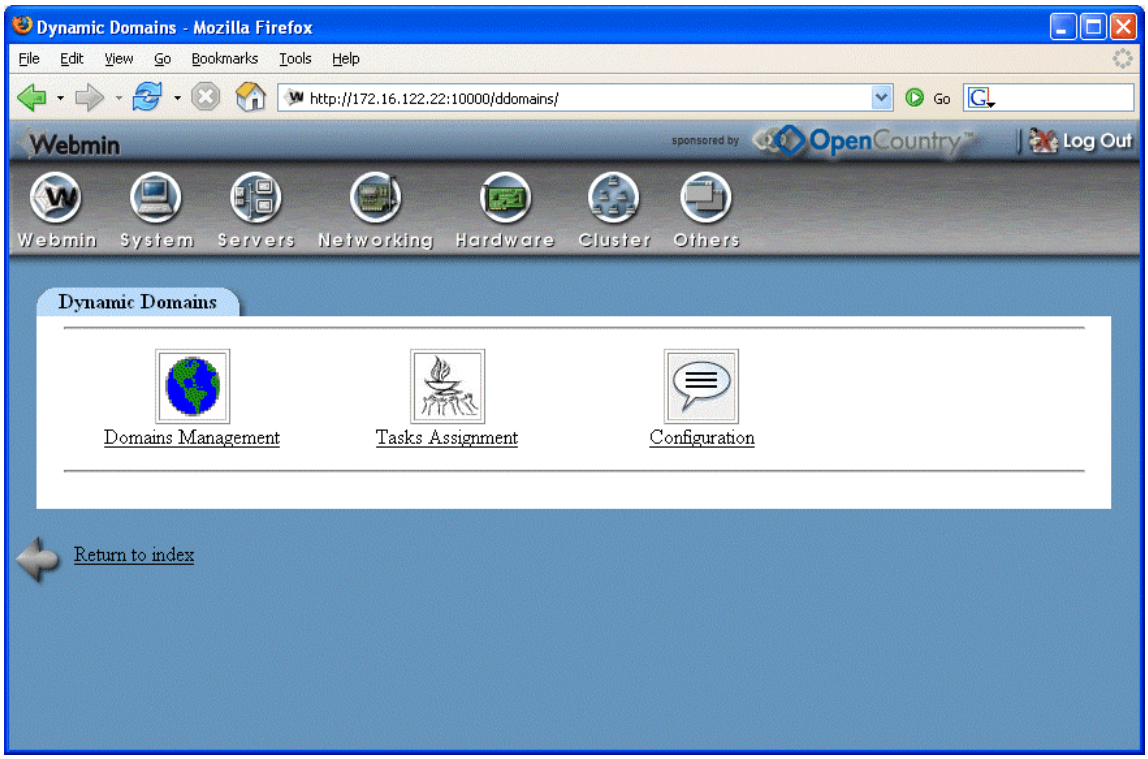
```
tail /var/ddtools/ddmon.log
06/06/19 07:46:18 ddmon : Initialization enabled
06/06/19 07:46:18 ddmon : Create domains configuration file
                          /var/ddomains/ddomains
06/06/19 07:46:18 ddmon : Create new domain default
06/06/19 07:46:18 ddmon : Assign tasks to domain default
06/06/19 07:46:18 ddmon : Create filters configuration file
                          /var/ddomains/filters
06/06/19 07:46:18 ddmon : Daemon started
06/06/19 07:46:18 ddmon : Time between measures : 10.000s
06/06/19 07:46:18 ddmon : Filters management started
06/06/19 07:46:18 ddmon : Alarms management started
06/06/19 07:46:18 ddmon : Dynamic management started
```

Checking access via Webmin

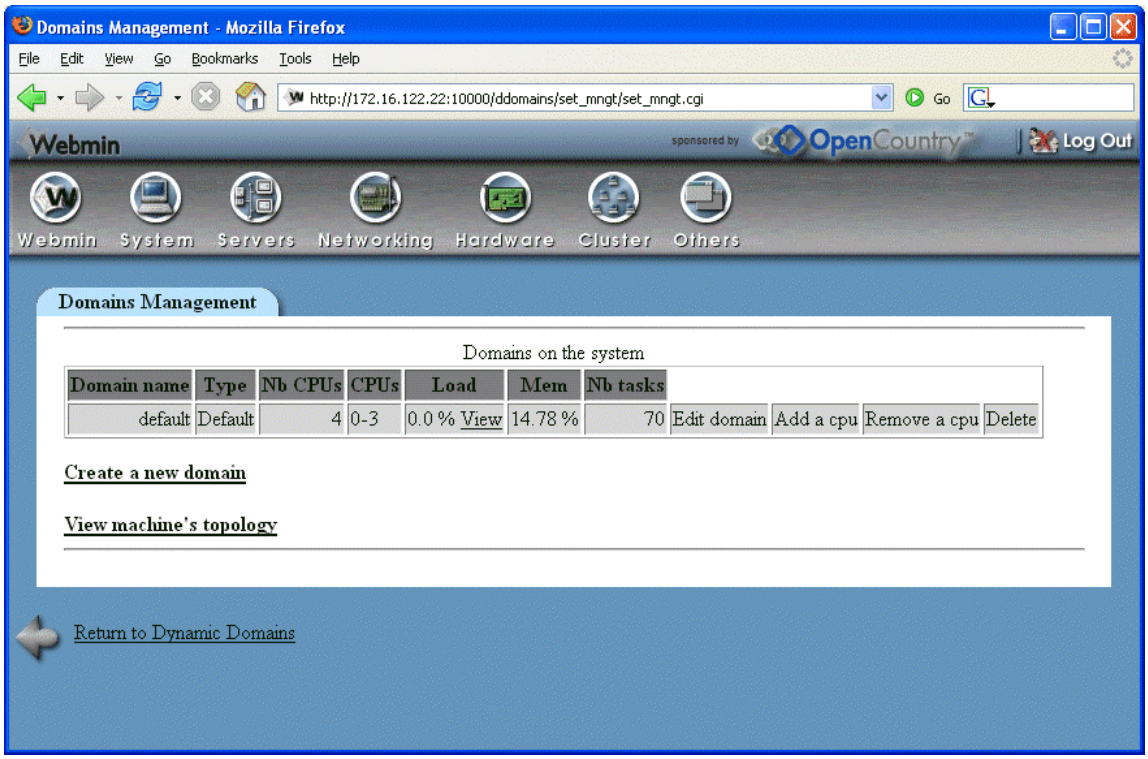
The "Dynamic Domains" software is configured or administered using the "Webmin" tool ("System" category) via a browser (URL: http://<nomhote>:10000 or https://<nomhote>:10000).



- Enter user name "root" with its password.
- Click "**Login**" to validate.
- Select group "**System**".
- Click the icon for the "**Dynamic Domains**" module.



- Click the "**Domains Management**" icon to check the configuration of domain "default".



Domain "default" must use all the machine's CPUs. No action (edition, deletion, etc.) is possible on this domain. With this screen, the user can create new domains (see chapter 3: "Configuration et utilisation en mode Normal" or chapter 4 "Configuration et utilisation en mode Expert").

B. Example of use

Running a process to simulate the load

This involves running programmes that use CPU resource, using the "ddload" utility. Type:

```
ddload -i load_2x45% -t 3600 -l 45 -p 2 &
ddload -i load_85% -t 3600 -l 85 &
```

Checking operation of the load simulation processes

The "load_xx" must be assigned to domain "default" and can use any machine's CPUs. To check the assignment of the "load_xx" processes, enter the following:

```
ddls -s load_
    Domain       Filter   Pid      Uid Command
   default            - 18046     root load_2x45%
   default            - 18047     root load_2x45%
   default            - 18048     root load_85%
```

The "load_xx" processes can periodically change CPUs. To check any possible change in the CPUs used by the "load_xx" processes, enter the following repeatedly:

```
ddls -s load_ -f domain,pid,lcpu,cmd
    Domain   Pid  C Command
   default 18046  3 load_2x45%
   default 18047  2 load_2x45%
   default 18048  3 load_85%
```

where "C" is the number of the last CPU used (lcpu).

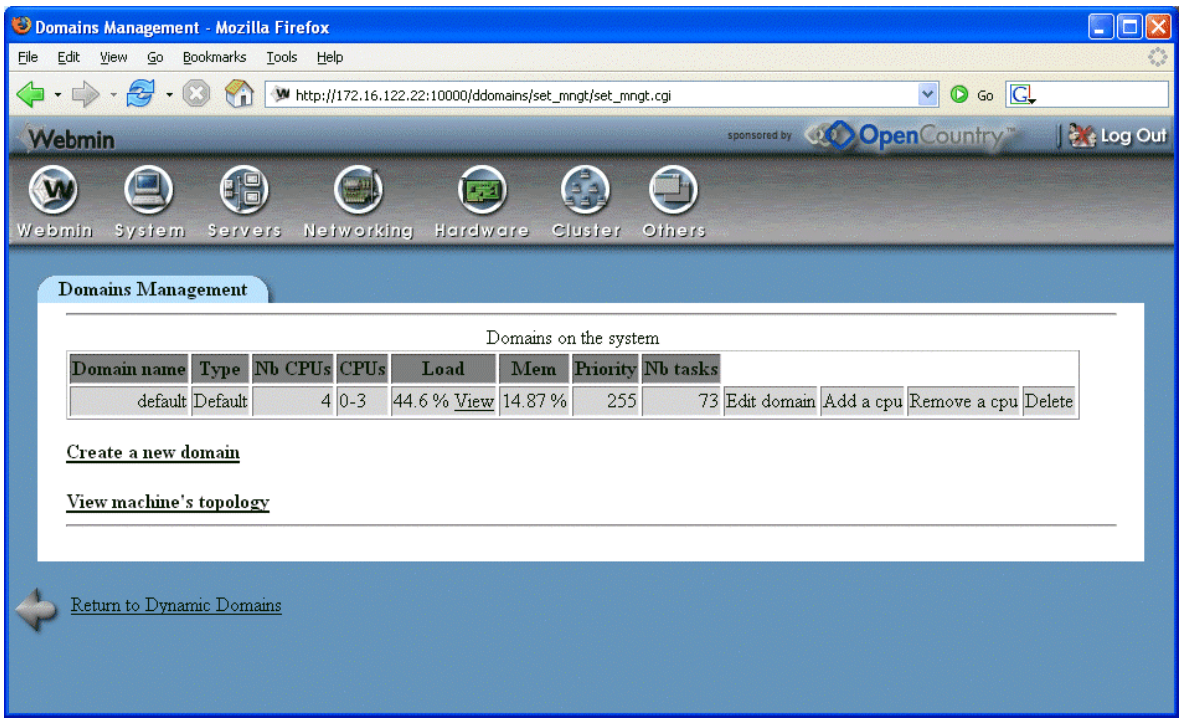
Each "load_xx" process uses resources from its CPU. To check the CPU load of the "load_xx" processes, enter the following:

```
ddtop -P -s load_ +f cmd
  Pid   Tid   %Cpu Command
18046 18046  44.49 load_2x45%
18047 18047  44.68 load_2x45%
18048 18048  84.55 load_85%
```

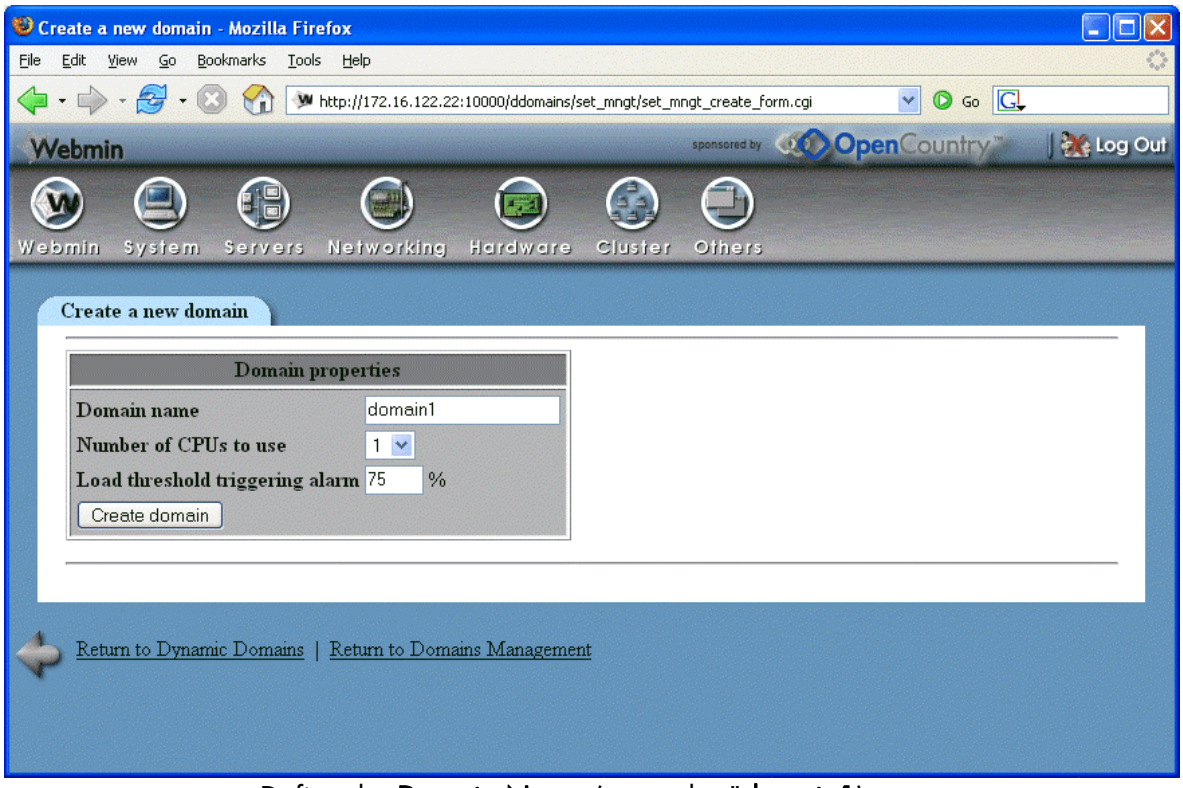
Configuring a domain

With Webmin: System->Dynamic Domains

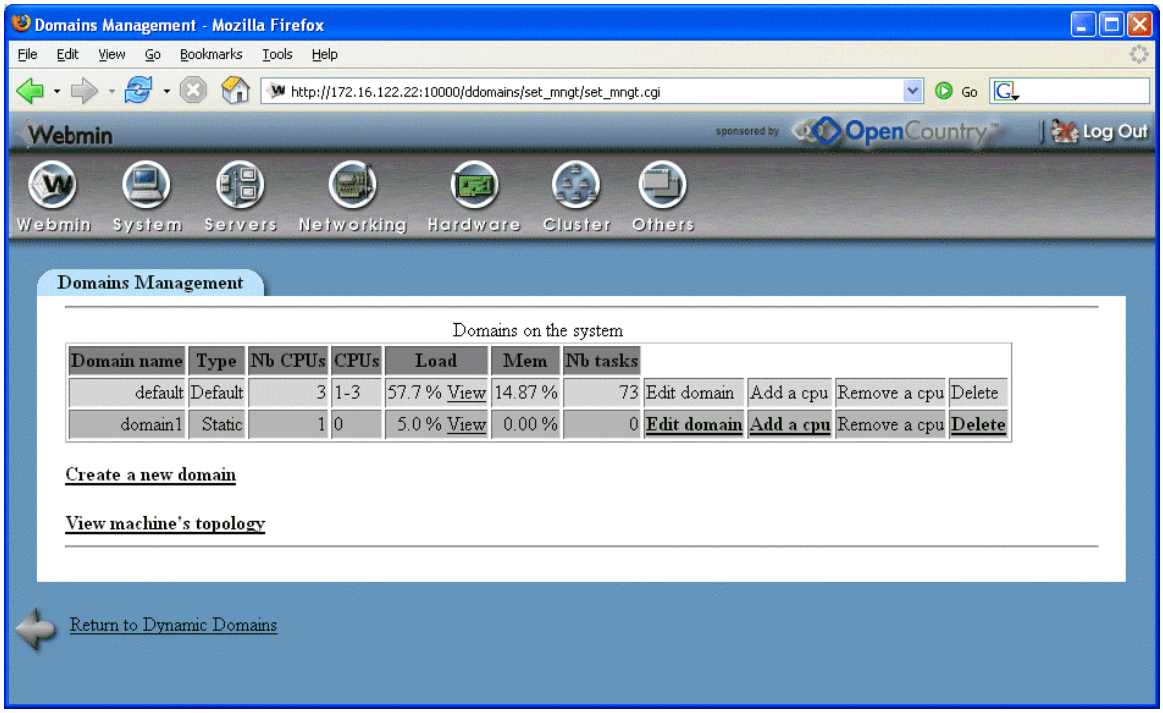
- Click "**Domains Management**" to configure a new domain.



  - Click "**Create a new domain**" to create a new domain.



- Define the **Domain Name** (example: "**domain1**).
- Define the number of CPUs to be reserved for the domain (**Number Of CPUs to use**) (example: "**1**".
- Define the alarm threshold for the domain (**Load threshold triggering alarm**) (example: "**75**".
- Click "**Create domain**".



Domain "**domain1**" is added to the list of domains and domain "**default**" no longer uses all the machine's CPUs.

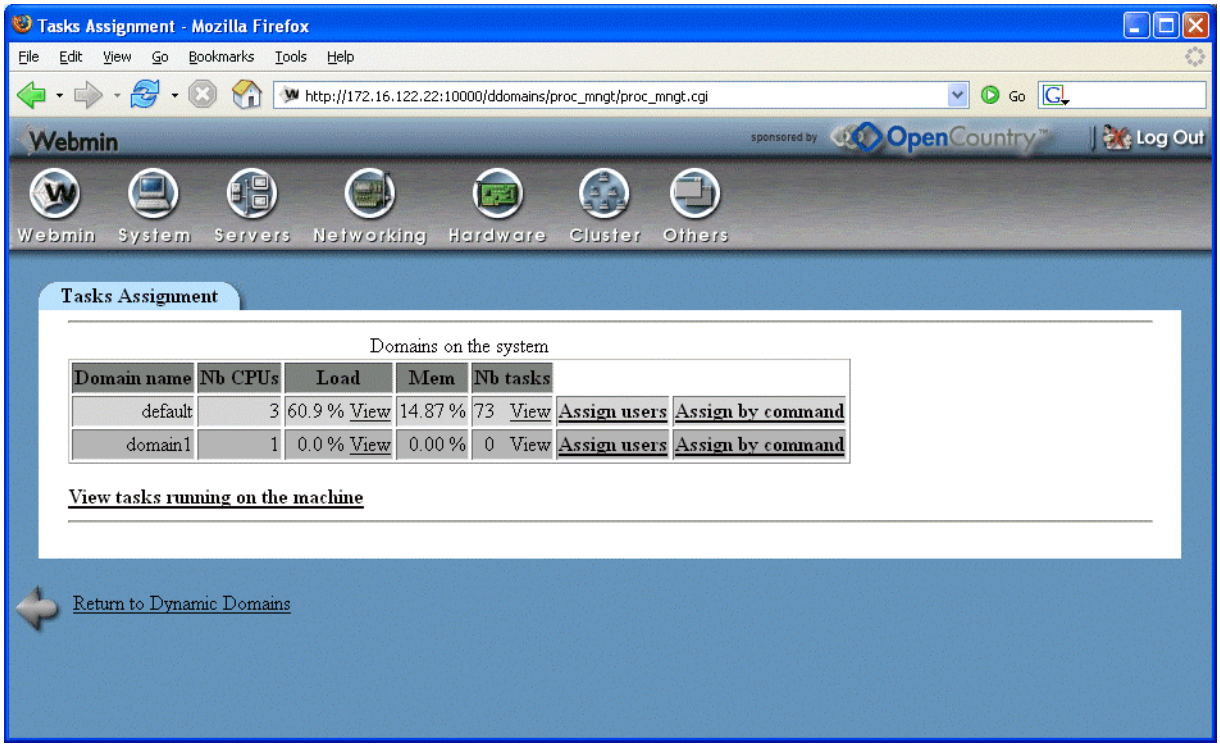
Check of the new domain configuration:

```
ddls
    Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
   default     1      3     4      255       100       33      1-3   53      71       85
   domain1     1      1     1       25        75       25        0    0       0        0
```

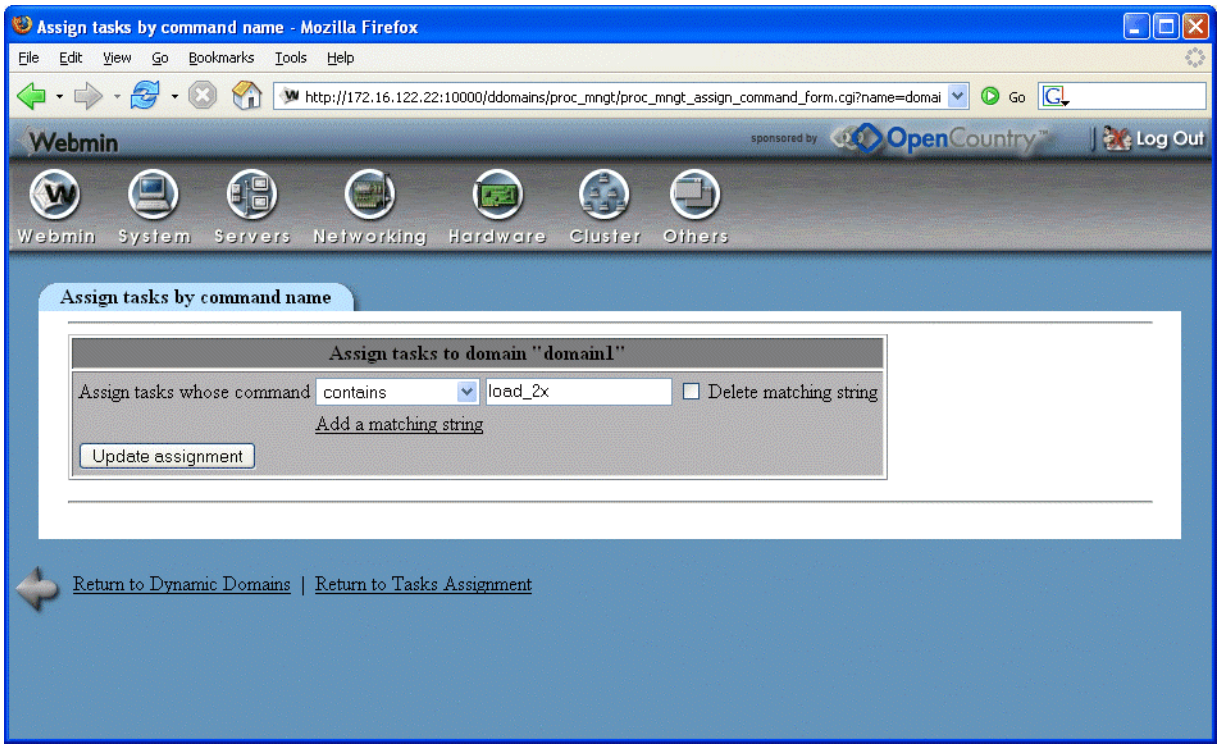
Adding a process to the domain

With Webmin: System->Dynamic Domains

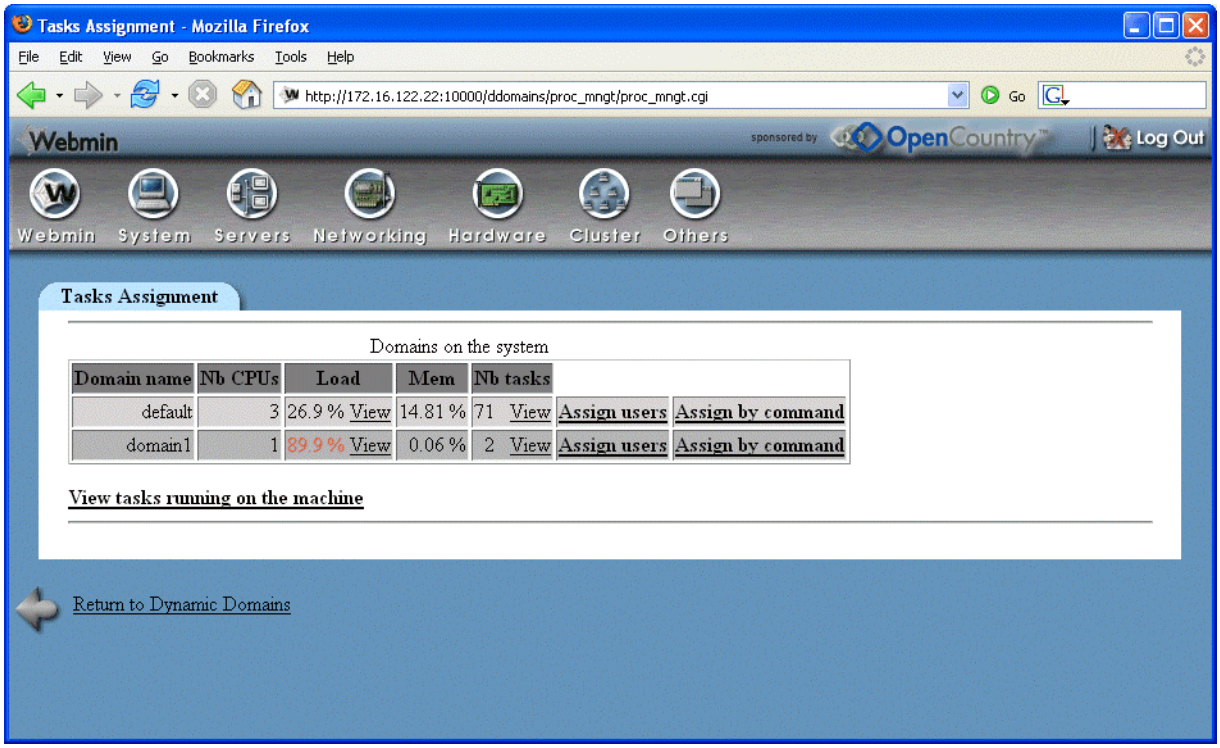
- Click "**Tasks Assignment**" to add a process to the domain.



- Click "**Assign by command**" for domain "**domain1**" in order to add a process to the domain.

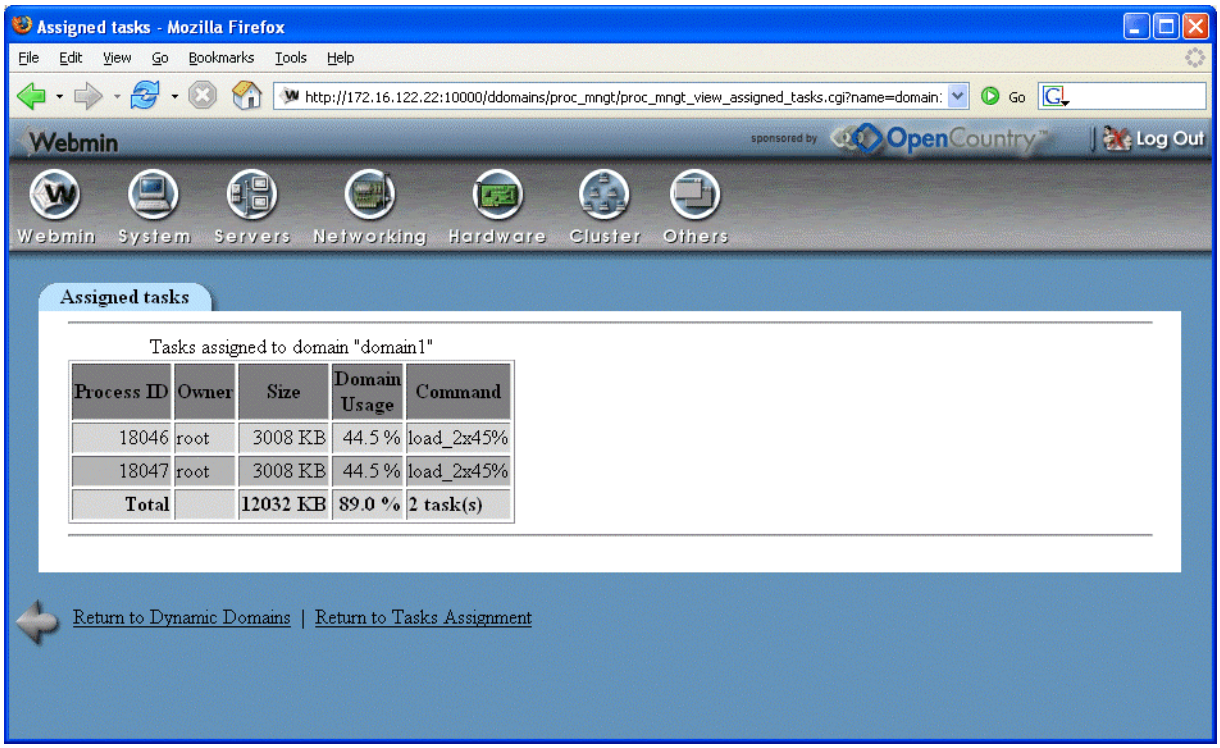


- Select condition "**contains**" and value "**load_2x**".
- Click "**Update assignment**".



Domain "**domain1**" has to be loaded (load > 75%).

o Click "**View**" in "**task**" for domain "**domain1**".



The "**load_2x45%**" tasks must be in domain "**domain1**" and load the domain ("**Domain Usage**" > 75%).

- Click "**Return to Dynamic Domains**" to return to the main screen.

Check that the processes "**load_2x45%**" are assigned to domain "**domain1**", that the CPU used corresponds to the one defined for the domain and that process "**load_85%**" is still assigned to domain "**default**".

To check the assignment of the "**load_xx**" processes, enter the following:

```
ddls -s load_ -f domain,pid,lcpu,cmd
    Domain   Pid  C Command
   default 18048  1 load_85%
   domain1 18046  0 load_2x45%
   domain1 18047  0 load_2x45%
```

If the daemon "ddmon" is active, an alarm message must be added to its trace file (file: **/var/ddtools/ddmon.log**)

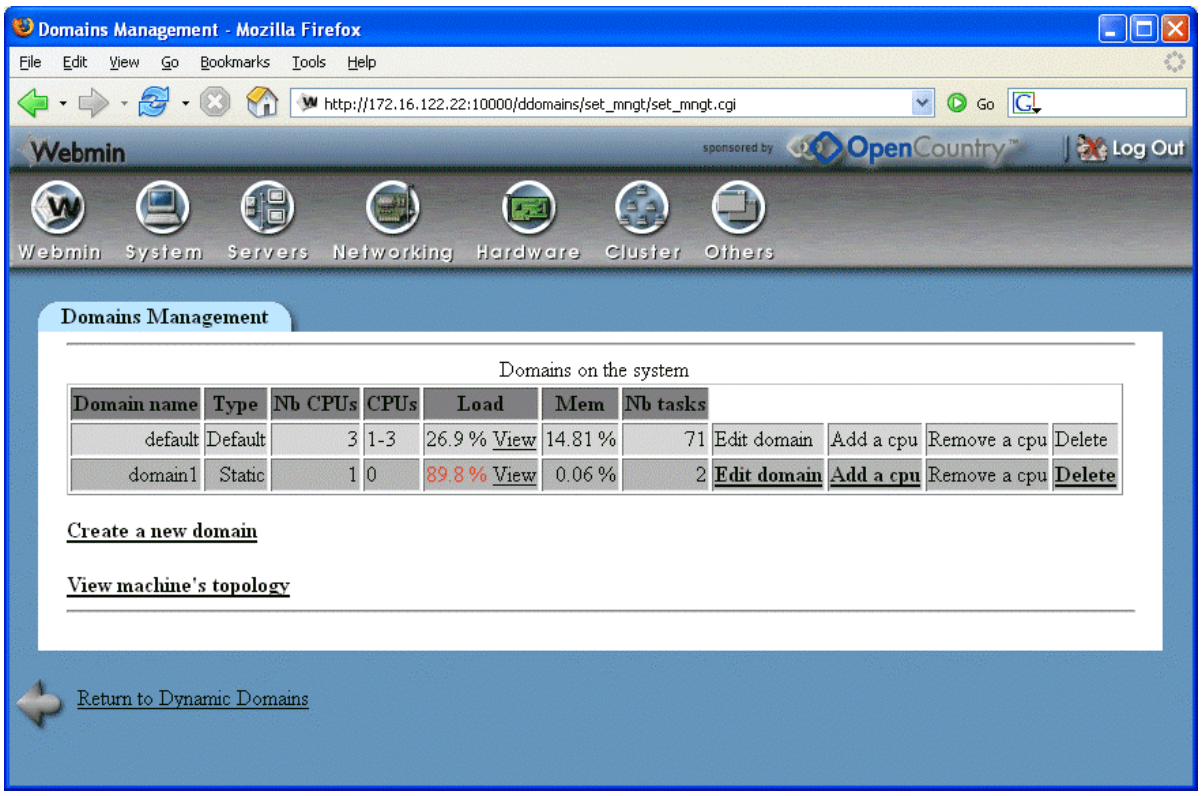
Example:

```
06/06/19 08:49:43 ddmon*2 Domain domain1 : Alarm  threshold (>75%)
```

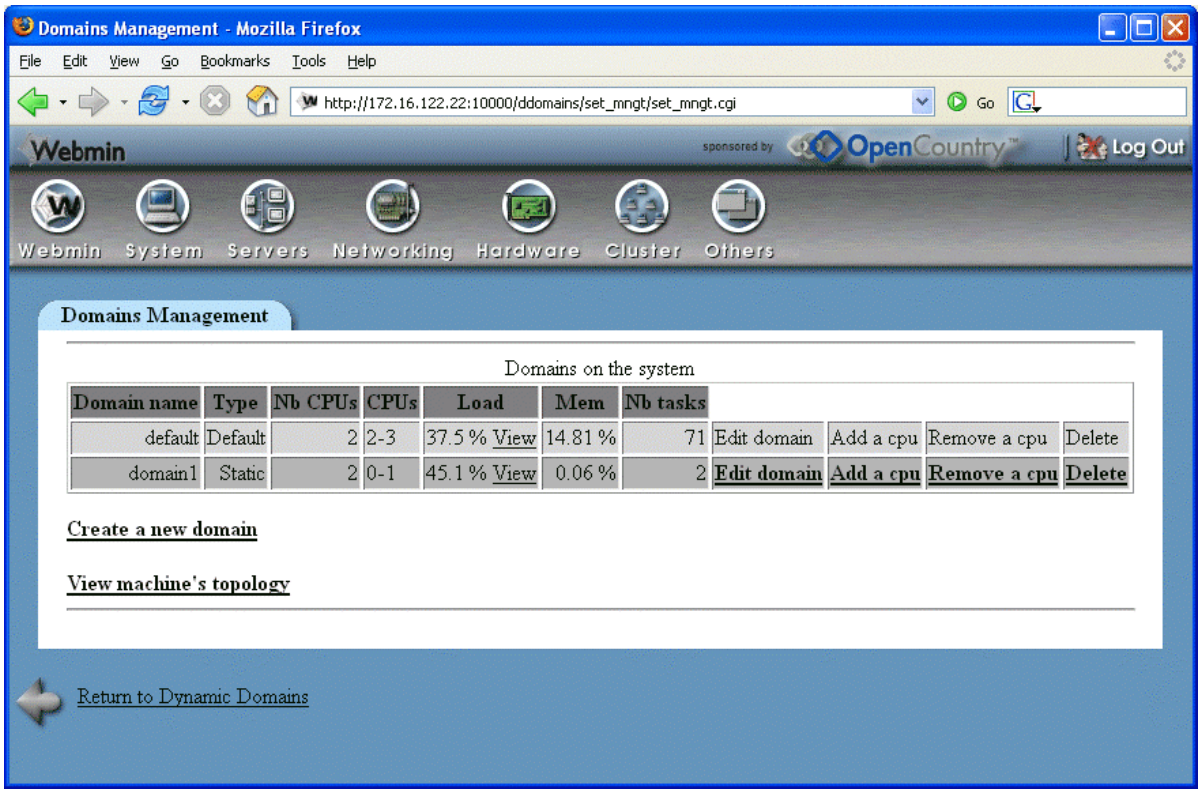
Adding a CPU to a domain

With Webmin: System->Dynamic Domains

- Click "**Domains Management**" to change a domain.



- Click "**Add a cpu**" of domain "**domain1**" to add a CPU to the domain.



Domain "**domain1**" uses 2 CPUs and no longer has an alarm.

- Click "**Return to Dynamic Domains**" to return to the main screen.

Check that the "**load_2x45%**" processes that are assigned to domain "**domain1**" use one of the domain's CPUs. To check the assignment of the "**load_xx**" processes, enter the following:

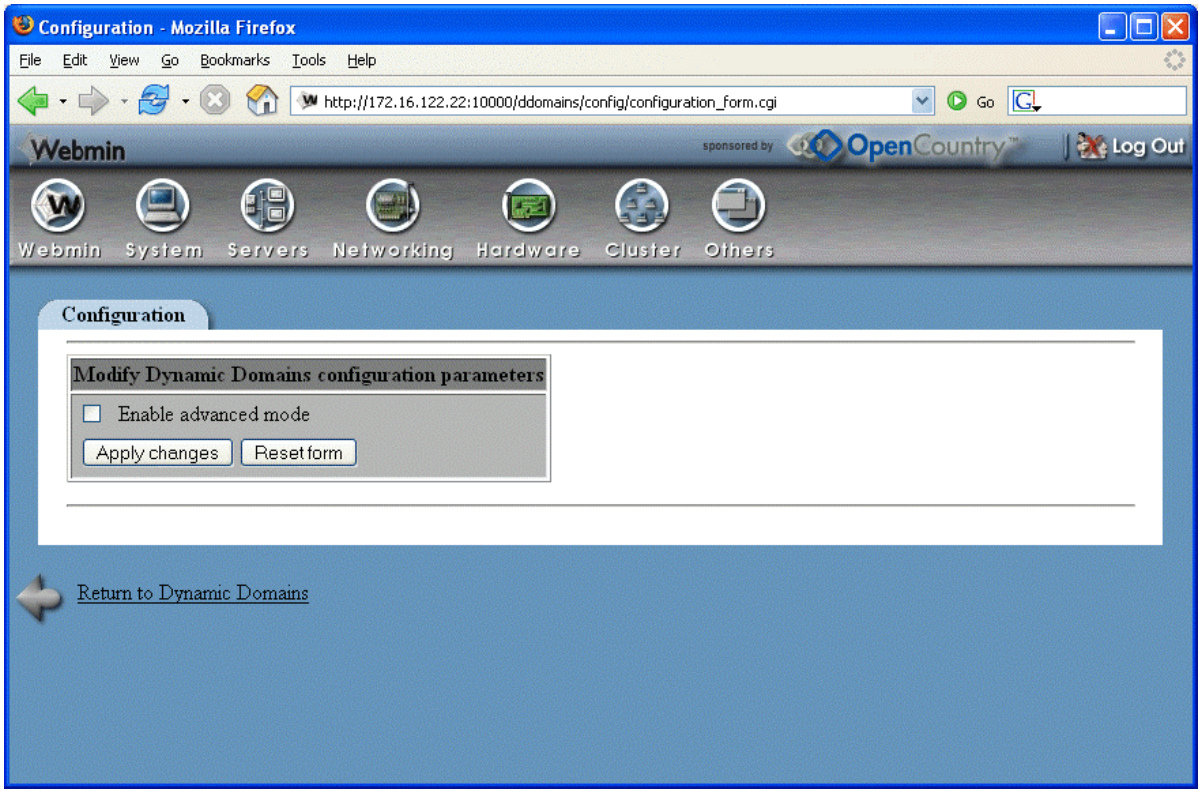
```
ddls -s load_ -f domain,pid,lcpu,cmd
    Domain   Pid  C Command
   default 18487  3 load_85%
   domain1 18485  0 load_2x45%
   domain1 18486  1 load_2x45%
```

Changing a domain (switching to dynamic mode)

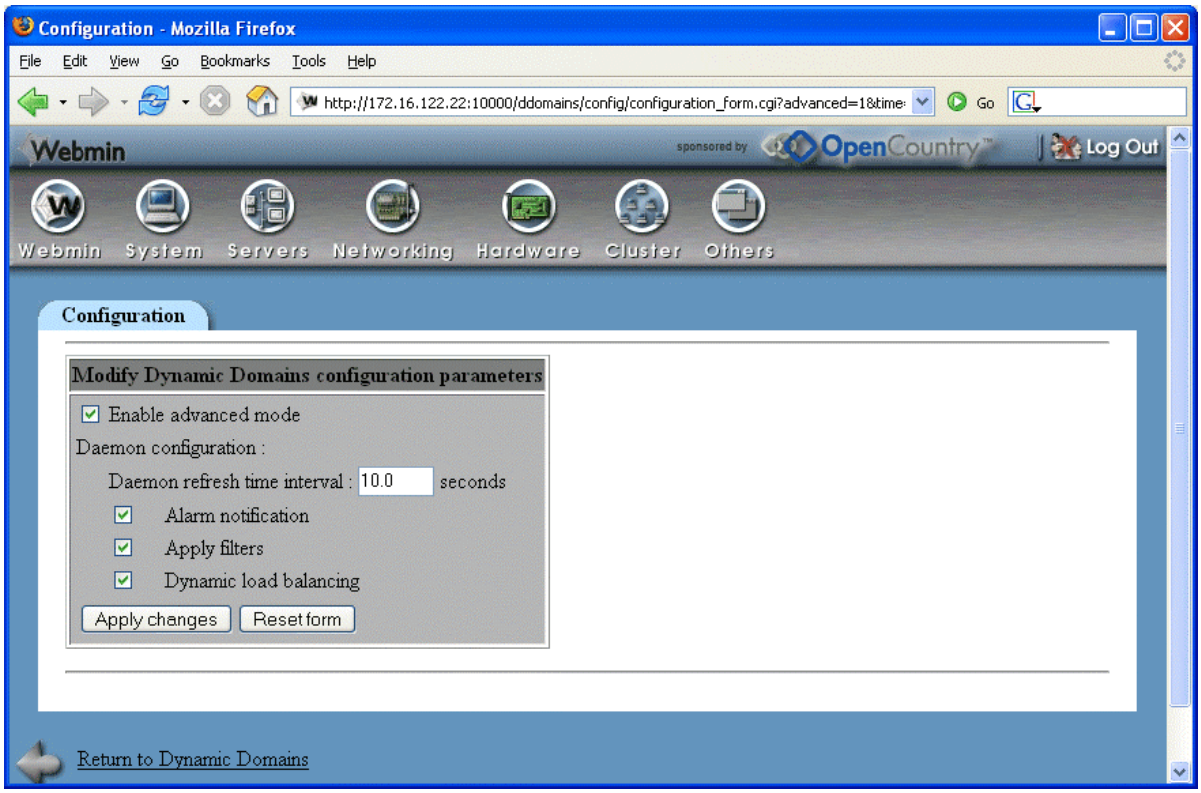
By default, once the "Dynamic Domains" software is installed, the operating interface on Webmin is configured in normal mode (simplified access to software functionalities). For dynamic domains configuration to be possible, the operating interface must be configured in advanced mode (full access to software functionalities).

With: Webmin->System->Dynamic Domains

- Clicking on "**Configuration**" to switch to advanced mode.

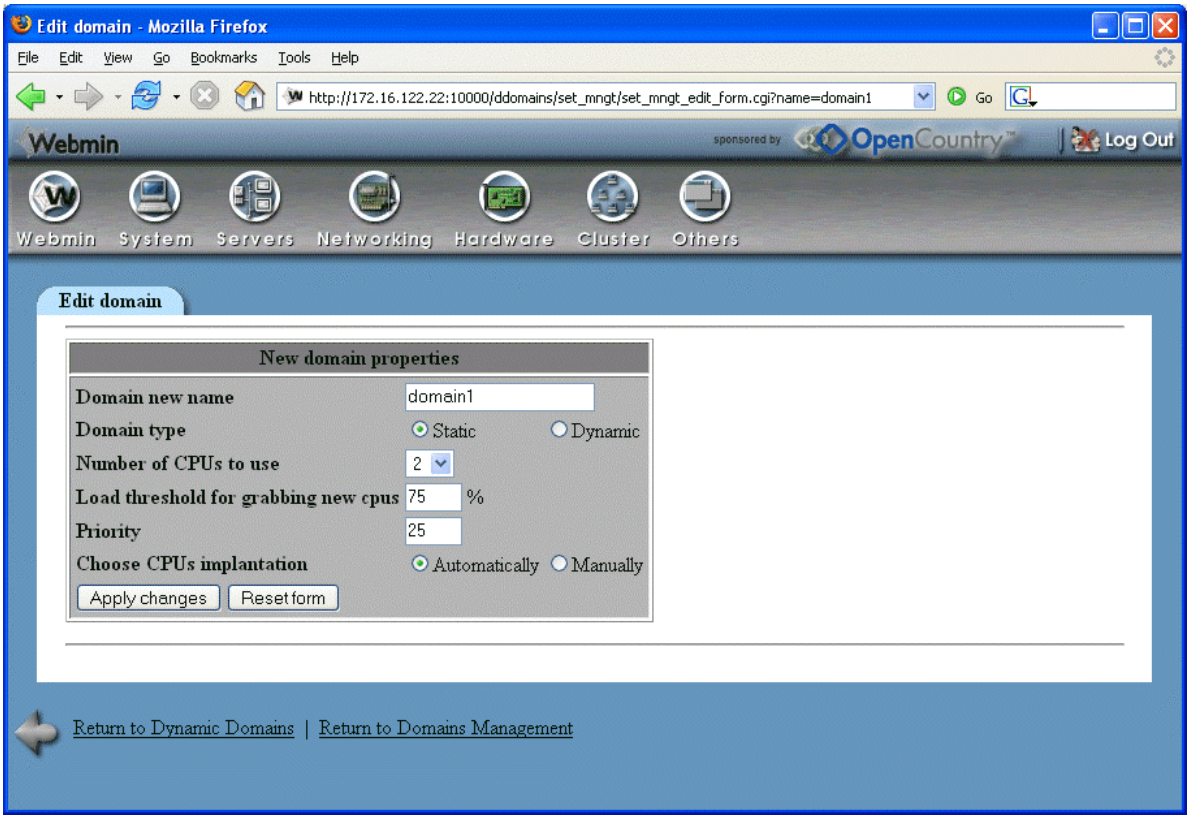


o Check the "**Enable advanced mode**" box to validate the advanced mode.

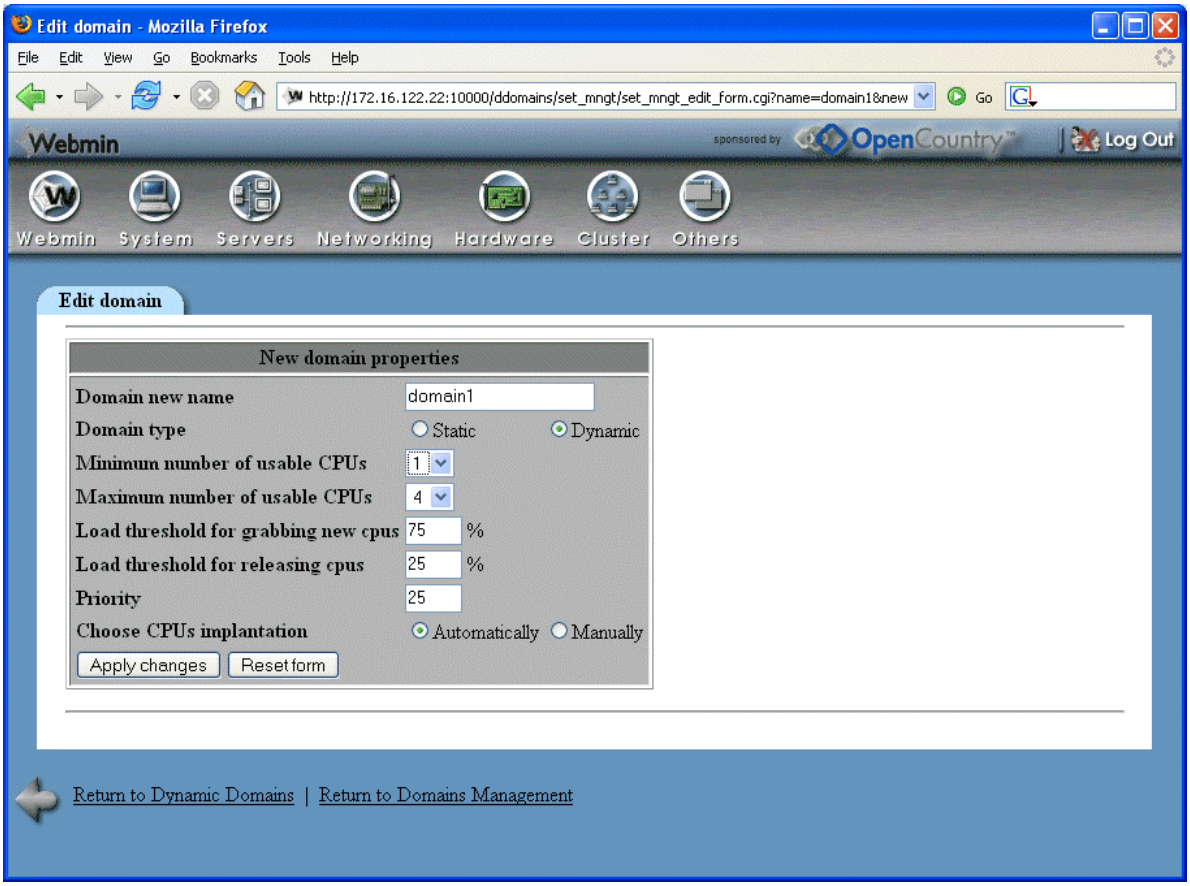


- Click "**Apply changes**".

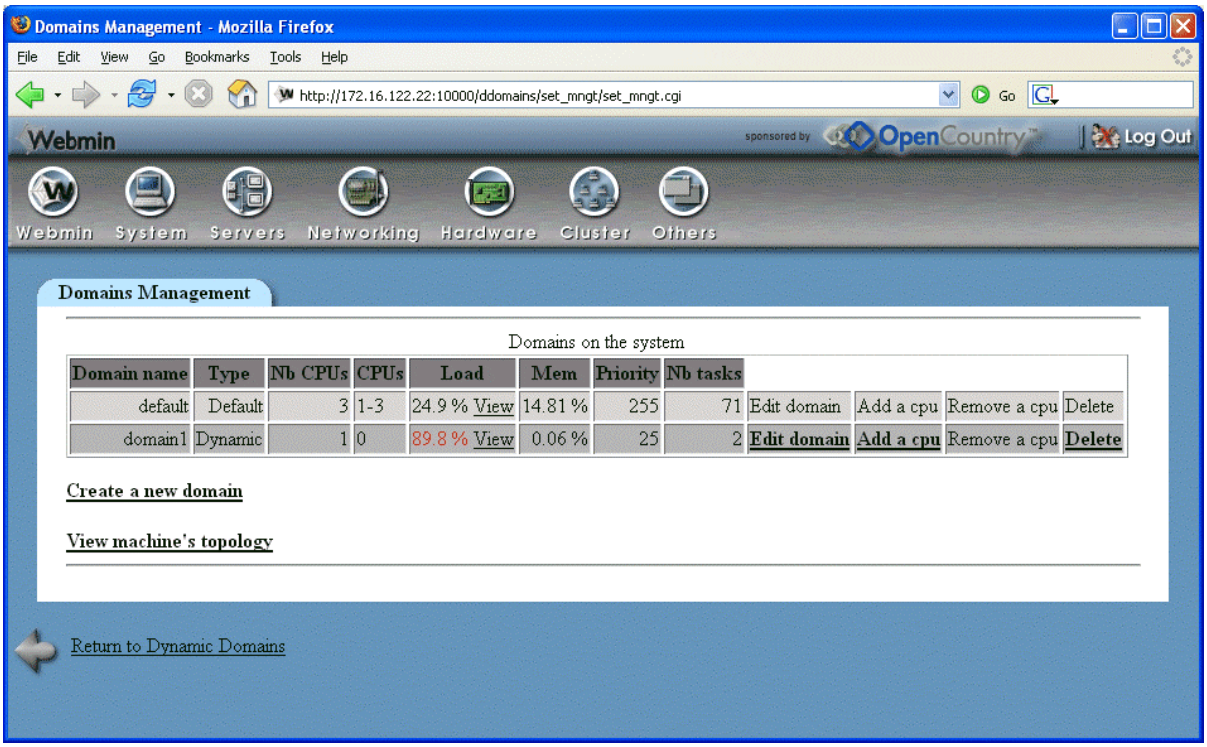
- Click "**Domains Management**" to change a domain.
  - Click "**Edit domain**" for domain "**domain1**" in order to change its characteristics.



- Click domain type "**Dynamic**".

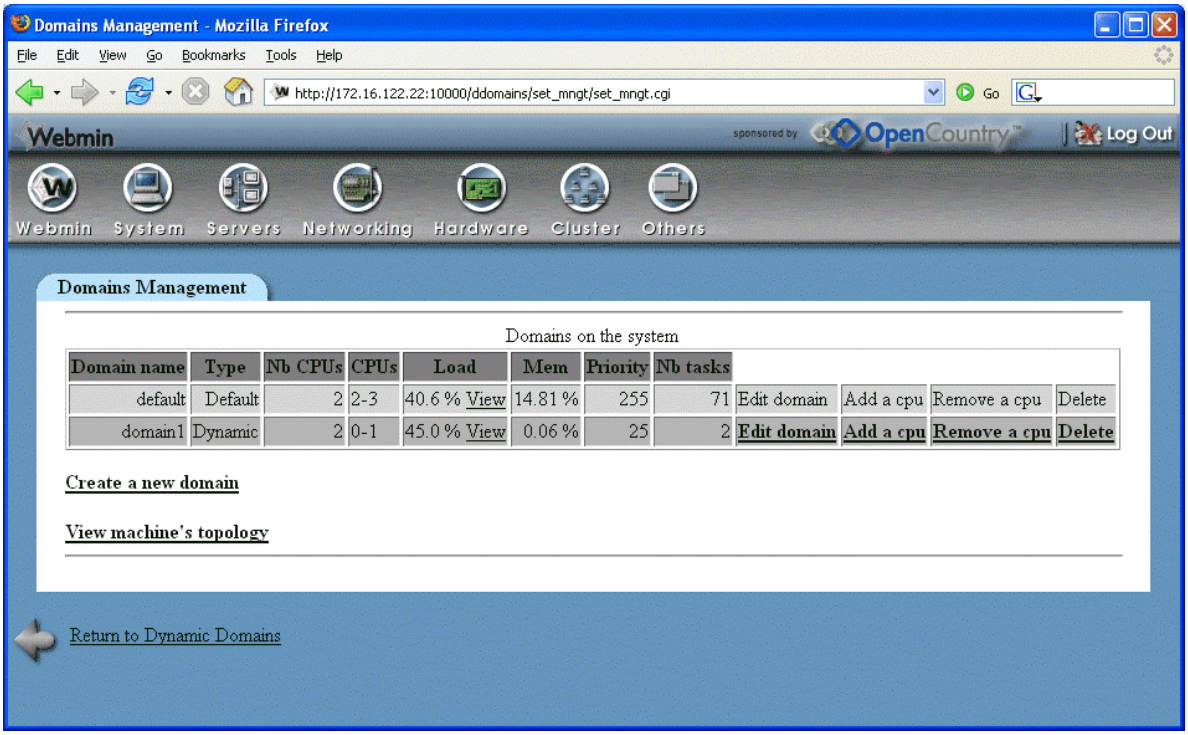


- Change the maximum number of CPUs for the domain (**Maximum number of usable CPUs**) example: "**1**".
- Change the maximum number of CPUs for the domain (**Maximum number of usable CPUs**) (example: "**4**".
- Click "**Apply changes**".



Domain "domain1" is now a dynamic domain. It uses a single CPU but, as the domain's load exceeds its upper threshold (alarm threshold), the daemon has to automatically add resources to it (add a new CPU to domain "**domain1**").

- Click "**Reload the page**" to check that CPU has been assigned automatically to domain "**domain1**".



- Click "**Return to Dynamic Domains**" to return to the main screen.

To check the addition of a resource (number of CPUs used = 2) using the command interface, enter the following:

```
ddls
    Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
   default     1      2     4      255       100       33      2-3   42      69      83
   domain1     1      2     4       25        75       25      0-1   44       2       2
```

Since domain "**domain1**" no longer has an alarm, the daemon no longer adds any additional resources to it. To add a resource (CPU) to domain "**domain1**" using the command interface, enter the following:

```
ddchg domain1 --add 1
```

Check with:

```
ddls
    Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
   default     1      1     4      255       100       33         3   90      69      83
   domain1     1      3     4       25        75       25      0-2   31       2       2
```

The process load for domain "**domain1**" is badly distributed. To check the load distribution on the domain CPUs, enter the following:

```
ddtop -CD
    Domain      Cpus  Cpu  %Used  %Idle  %Wait
   default         3    3  87.04  12.96   0.00
   domain1         0    0  43.90  56.10   0.00
   domain1         1    1  38.05  61.95   0.00
   domain1         2    2   0.00 100.00   0.00
```

In this example, CPU 2 of domain "domain1" is not used. The domain resources can be easily limited to the minimum required, for instance by allotting the minimum resources to the domain; the daemon will add new resources as required according to the domain load. Allotment of minimum resources (domain "**domain1**": **Nbused=1**):

```
ddchg domain1 1
```

After processing, the daemon must add a CPU to domain "domain1":

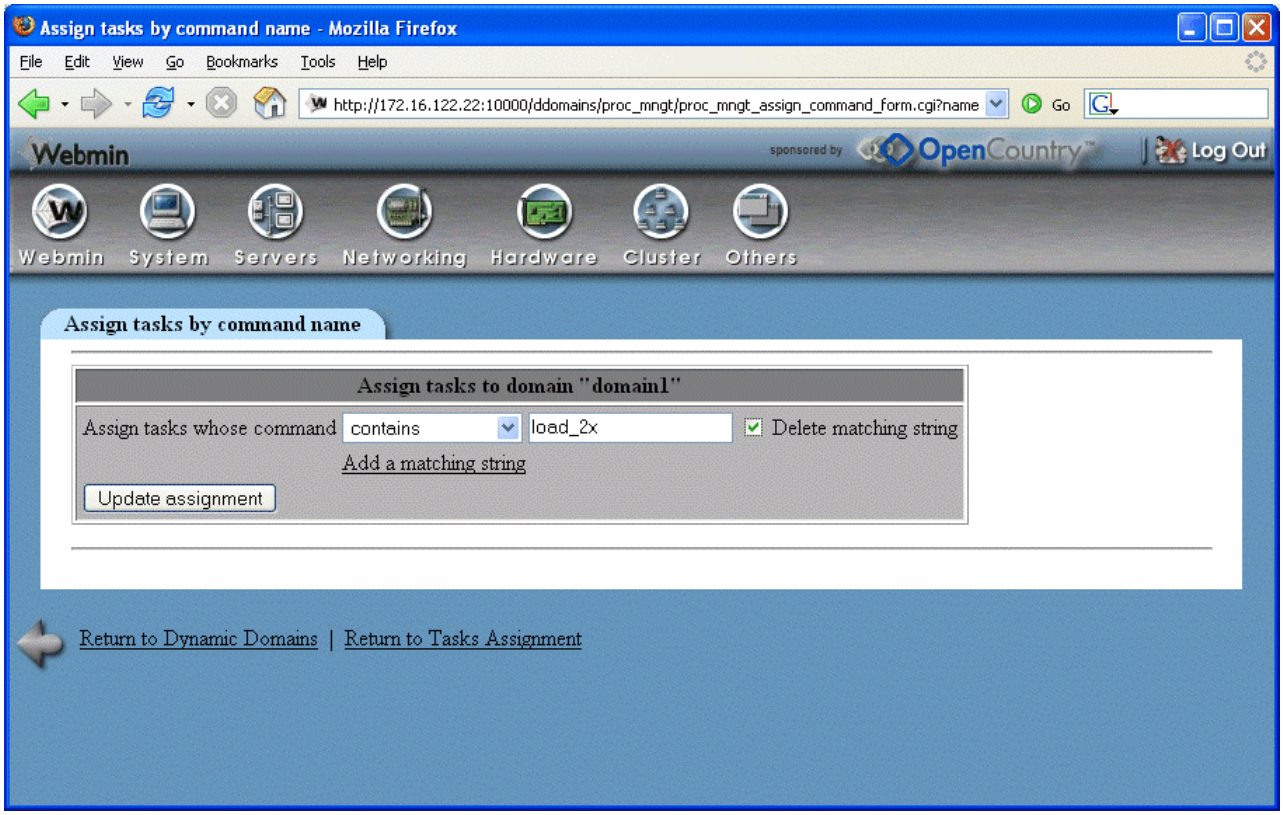
```
ddls
    Domain NbMin NbUsed NbMax Priority Threshold SharEven UsedCpus Load NbProcs NbThrds
   default     1      2     4      255       100       33      2-3   42      69      83
   domain1     1      2     4       25        75       25      0-1   36       2       2
```

Resuming normal mode and removing domain "domain1"

With: Webmin->System->Dynamic Domains

- Click "**Configuration**" to switch to advanced mode.
  - Uncheck the "**Enable advanced mode**" box to disable advanced mode.
  - Click "**Apply changes**".

- Click "**Tasks Assignment**" to add a process to the domain.
  - Click "**Assign by command**" of domain "**domain1**" to remove task assignment.



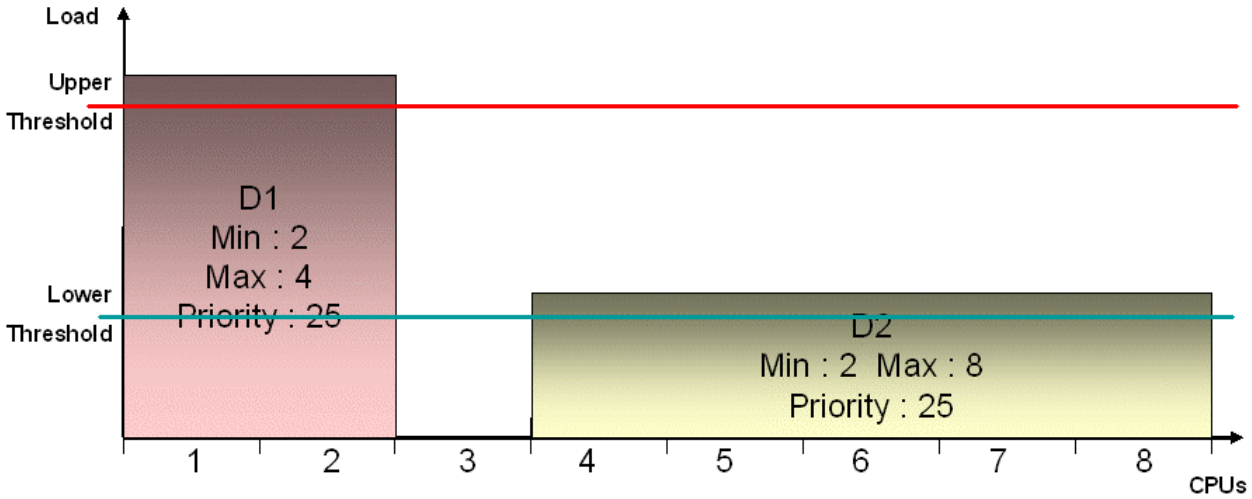
  - Check the "**Delete matching string**" box.
  - Click "**Update Assignment**".

- Click "**Return to Dynamic Domains**" to return to the main screen.

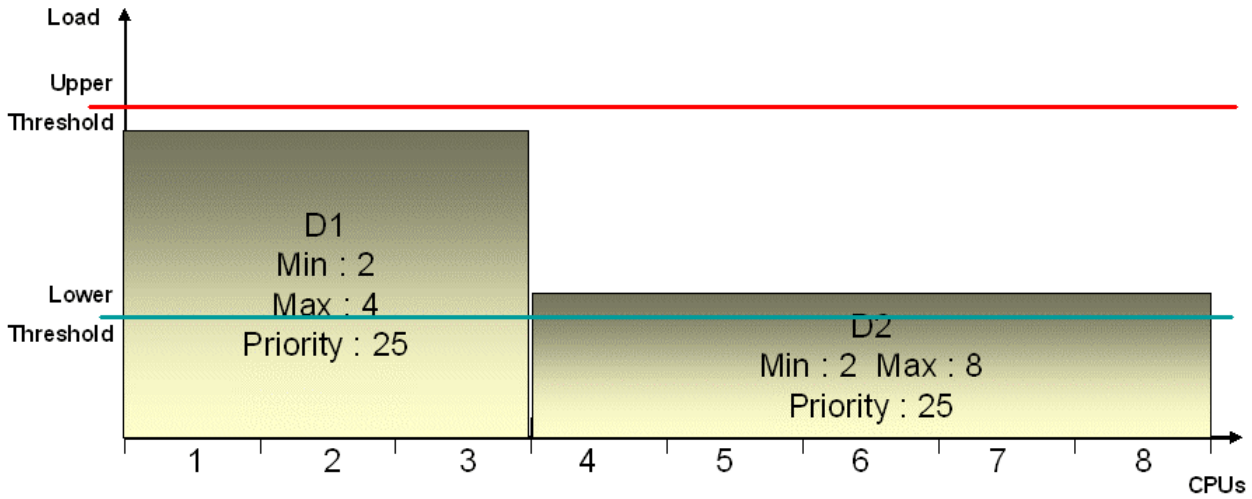
- Click "**Domains Management**" to remove domain "**domain1**".
  - Click "**Delete**" and confirm the action.

- Click "**Return to Dynamic Domains**" to return to the main screen.

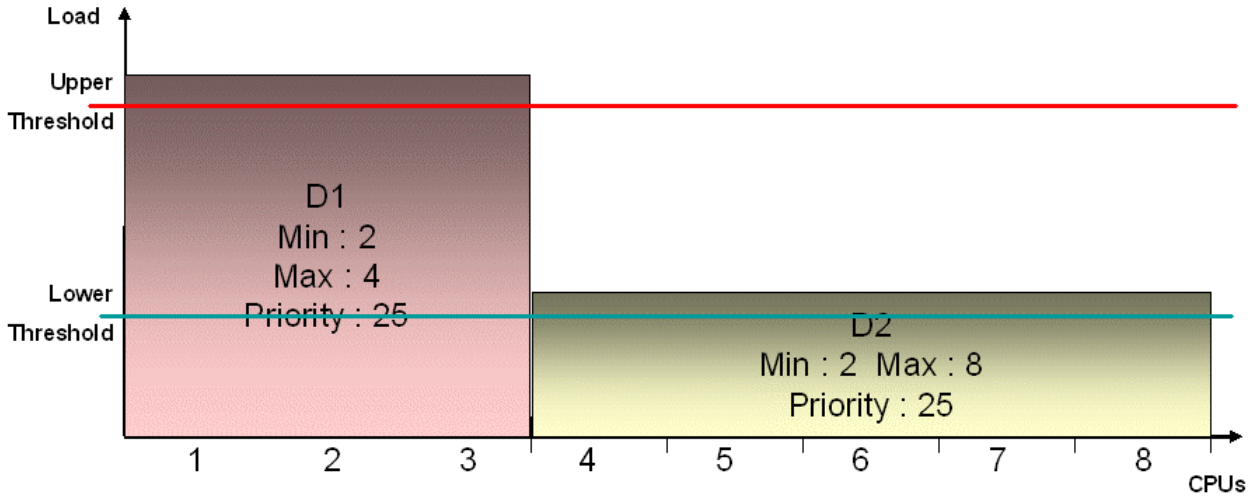
C. Simulating dynamic domains



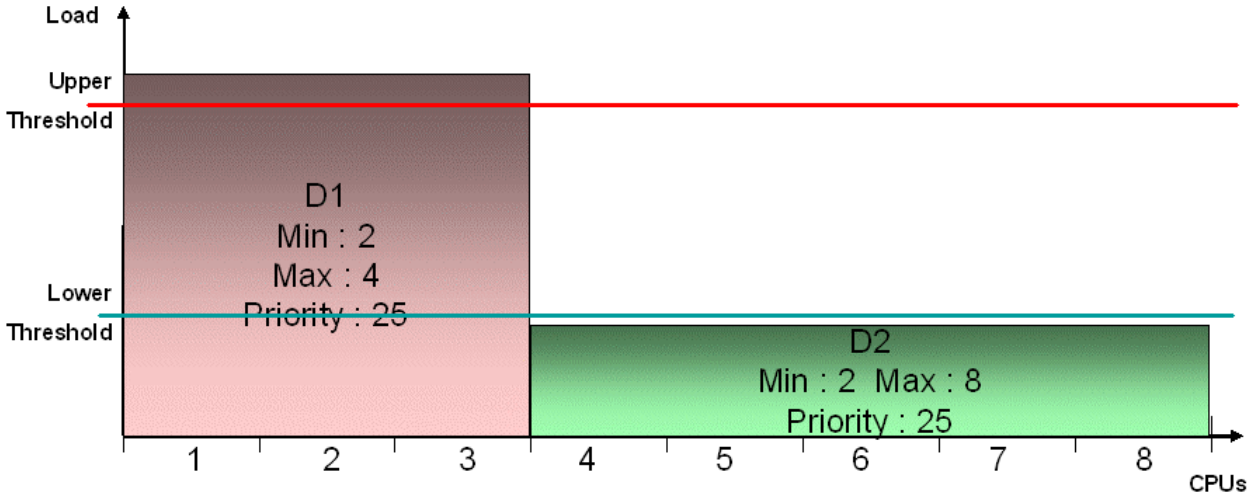
1. 2 domains with domain D1 in Alarm condition



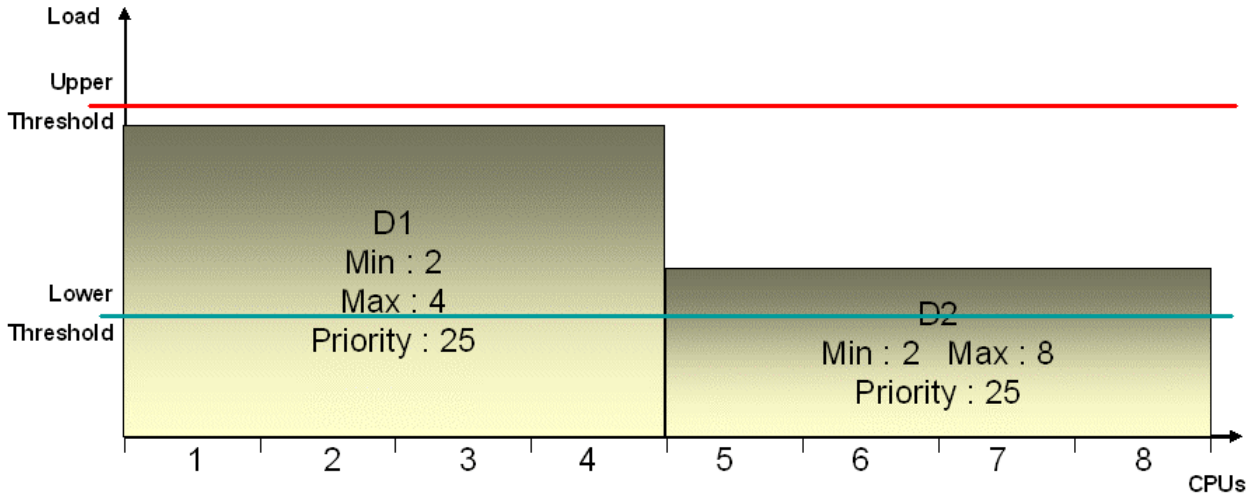
1. 2 domains with domain D1 in Alarm condition
2. Addition of a CPU to domain D1



1. 2 domains with domain D1 in Alarm condition
2. Addition of a CPU to domain D1
3. Alarm condition entered for domain D1



1. 2 domains with domain D1 in Alarm condition
2. Addition of a CPU to domain D1
3. Alarm condition entered for domain D1
4. Passage below the lower threshold for domain D2



1. 2 domains with domain D1 in Alarm condition
2. Addition of a CPU to domain D1
3. Alarm condition entered for domain D1
4. Passage below the lower threshold for domain D2
5. Addition of a CPU (of domain D2) to domain D1

Index

Technical publication remarks form

Title:	NOVASCALE Dynamic Domains for Applications User's Guide

Reference No.:	86 A2 63ER 01

Date:	September 2006

ERRORS IN PUBLICATION

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Your comments will be promptly investigated by qualified technical personnel and action will be taken as required.
If you require a written reply, please include your complete mailing address below.

NAME: _____ Date: _____
COMPANY: _____
ADDRESS: _____
_____

Please give this technical publication remarks form to your BULL representative or mail to:

Bull - Documentation Dept.
1 Rue de Provence
BP 208
38432 ECHIROLLES CEDEX
FRANCE
info@frec.bull.fr

Technical publications ordering form

To order additional publications, please fill in a copy of this form and send it via mail to:

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Phone: +33 (0) 2 41 73 72 66
FAX: +33 (0) 2 41 73 70 66
E-Mail: srv.Dupilcopy@bull.net

CEDOC Reference #	Designation	Qty
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
[_ _] : The latest revision will be provided if no revision number is given.		

NAME: ______________________________ Date: ______________

COMPANY: ______________________________

ADDRESS: ______________________________

____

PHONE: ______________________ FAX: ______________________

E-MAIL: ______________________________

For Bull Subsidiaries:
Identification: ______________________________

For Bull Affiliated Customers:
Customer Code: ______________________________

For Bull Internal Customers:
Budgetary Section: ______________________________

For Others: Please ask your Bull representative.

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE No.
86 A2 63ER 01