Requirement Expression Syntax

The syntax of the requirement expressions is defined as:

```
stmt::= expr
expr::= VARIABLE '=' INTEGER
        | VARIABLE '>' INTEGER
        | VARIABLE '<' INTEGER
        | VARIABLE '=' STRING
        | expr '&' expr
        | expr '|' expr
        | '!' expr
        | '(' expr ')'
```

Each expression is evaluated to 1 (TRUE) or 0 (FALSE). Only those hosts for which the requirement expression is evaluated to TRUE will be considered to execute the job.

Logical operators are as expected (less '<', greater '>', '&' AND, '|' OR, '!' NOT), '=' means equals with integers. When you use '=' operator with strings, it performs a shell wildcard pattern matching.

Examples:

```
REQUIREMENTS = 'LRMS_TYPE = "pbs"'       # Only use pbs
REQUIREMENTS = 'HOSTNAME = "*.es"'       # Only hosts ending in ".es"
REQUIREMENTS = 'HOSTNAME = "mycomputer"' # Only use mycomputer
REQUIREMENTS = 'ARCH = "x86_64"'         # Only host x86_64 architecture
REQUIREMENTS = 'ARCH = "x86_64" & HOSTNAME = "*.es"' # Only hosts ending in ".es" and have x86_64 architecture
```

You can check the resources that match your requirements with `wfr4g_resources -m GW_identification`.

```
[user@mycomputer~]$ wrf4g_status -l
Realization        GW  Stat Chunks Comp.Res   WN         Run.Sta       ext   %
test               0     R  1/3    mycomputer ciclon     WRF             - 0.00
[user@mycomputer~]$ wrf4g_resources -m 0
HID QNAME      RANK  PRI  SLOTS HOSTNAME
0   default    0     1     0     mycomputer
```

Requirement Variables

If you want to see more information about a single resource, use the `wrf4g_resources` command followed by the host identification (HID):

```
[user@mycomputer~]$ wrf4g_resources
HID PRI  OS              ARCH   NODES(U/F/T)  LRMS         HOSTNAME
0   1   GNU/Linux2.6.18 x86_64         0/1/1  FORK         mycomputer
[user@mycomputer~]$ wrf4g_resources 0
HID PRI  OS              ARCH   NODES(U/F/T) LRMS         HOSTNAME   QUEUENAME        SL(F/T) WALLT CPUT  COUNT MAXR  MAXQ
0   1   GNU/Linux2.6.18 x86_64        0/1/1  FORK         mycomputer   default           1/1  0     0     0     0     0
```

The REQUIREMENTS values are:

Variable	Description
HOSTNAME	Execution host (e.g. mycomputer)
ARCH	Architecture of the execution host (e.g. i686, x86_64)
OS_NAME	Operating System name of the execution host (e.g. Linux, SL)
OS_VERSION	Operating System version of the execution host (e.g. 2.6.9-1.66, 3)
CPU_MODEL	CPU model of the execution host (e.g. Intel(R) Pentium(R) 4 CPU 2, PIV)
CPU_MHZ	CPU speed in MHz of the execution host
CPU_FREE	Percentage of free CPU of the execution host
NODECOUNT	Total number of cores of the execution host
SIZE_MEM_MB	Total memory size in MB of the execution host
FREE_MEM_MB	Free memory in MB of the execution hosts
SIZE_DISK_MB	Total disk space in MB of the execution hosts
FREE_DISK_MB	Free disk space in MB of the execution hosts
LRMS_TYPE	Type of local DRM system for execution (e.g. pbs, sge)
QUEUE_NAME	Name of the queue (e.g. default, short)
QUEUE_NODECOUNT	Total node count of the queue
QUEUE_FREENODECOUNT	Free node count of the queue
QUEUE_MAXTIME	Maximum wall time of jobs in the queue
QUEUE_MAXCPUTIME	Maximum CPU time of jobs in the queue
QUEUE_MAXCOUNT	Maximum count of jobs that can be submitted in one request to the queue
QUEUE_MAXRUNNINGJOBS	Maximum number of running jobs in the queue
QUEUE_MAXJOBSINQUEUE	Maximum number of queued jobs in the queue
QUEUE_DISPATCHTYPE	Dispatch type of the queue (e.g. batch, inmediate)
QUEUE_PRIORITY	Priority of the queue
QUEUE_STATUS	Status of the queue (e.g. active, production)

Environment Expression Syntax

The syntax of the environment expressions is specified in a comma-separated, source/destination pair.

```
stmt::= VARIABLE = VALUE, VARIABLE = VALUE, ...
```

Examples:

```
ENVIRONMENT = 'WALLTIME = 00:01:00' # 60 seconds of max walltime
ENVIRONMENT = 'WALLTIME = 00:01:00, MEMORY = 2000' # 60 seconds of max walltime and 2 GB of RAM memory
```

Environment Variables

The variables defined in the ENVIRONMENT are:

Variable	Description
CPUTIME	Maximum amount of CPU time used by all processes in the job (**HH:MM:SS**)
WALLTIME	Maximum amount of real time during which the job can be in the running state (**HH:MM:SS**)
MEMORY	Maximum amount of physical memory used by the job (**MB**)