

---

# Using Platform™ LSF™ on Windows®

Version 7 Update 2  
Release date: November 2007  
Last modified: January 31, 2008  
Comments to: [doc@platform.com](mailto:doc@platform.com)  
Support: [support@platform.com](mailto:support@platform.com)



---

## Copyright

© 1994-2008, Platform Computing Inc.

Although the information in this document has been carefully reviewed, Platform Computing Inc. (“Platform”) does not warrant it to be free of errors or omissions. Platform reserves the right to make corrections, updates, revisions or changes to the information in this document.

UNLESS OTHERWISE EXPRESSLY STATED BY PLATFORM, THE PROGRAM DESCRIBED IN THIS DOCUMENT IS PROVIDED “AS IS” AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL PLATFORM COMPUTING BE LIABLE TO ANYONE FOR SPECIAL, COLLATERAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING WITHOUT LIMITATION ANY LOST PROFITS, DATA, OR SAVINGS, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS PROGRAM.

## We’d like to hear from you

You can help us make this document better by telling us what you think of the content, organization, and usefulness of the information. If you find an error, or just want to make a suggestion for improving this document, please address your comments to [doc@platform.com](mailto:doc@platform.com).

Your comments should pertain only to Platform documentation. For product support, contact [support@platform.com](mailto:support@platform.com).

## Document redistribution and translation

This document is protected by copyright and you may not redistribute or translate it into another language, in part or in whole.

## Internal redistribution

You may only redistribute this document internally within your organization (for example, on an intranet) provided that you continue to check the Platform Web site for updates and update your version of the documentation. You may not make it available to your organization over the Internet.

## Trademarks

LSF is a registered trademark of Platform Computing Inc. in the United States and in other jurisdictions.

ACCELERATING INTELLIGENCE, PLATFORM COMPUTING, PLATFORM SYMPHONY, PLATFORM JOBSCHEDULER, PLATFORM ENTERPRISE GRID ORCHESTRATOR, PLATFORM EGO, and the PLATFORM and PLATFORM LSF logos are trademarks of Platform Computing Inc. in the United States and in other jurisdictions.

UNIX is a registered trademark of The Open Group in the United States and in other jurisdictions.

Microsoft is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Other products or services mentioned in this document are identified by the trademarks or service marks of their respective owners.

## Third-party license agreements

<http://www.platform.com/Company/third.part.license.htm>

## Third-party copyright notices

<http://www.platform.com/Company/Third.Party.Copyright.htm>

---

# Contents

1	Testing Your LSF Installation	5
	Contents	5
	Checking the license server (permanent LSF license)	6
	Checking the cluster	7
	Checking the LSF batch system	13
	Test the Platform Management Console (PMC)	15
2	Permanent LSF Licenses and FLEXlm on Windows hosts	17
	Contents	17
	Types of LSF licenses	18
	FLEXlm License Management	18
	LSF license file (license.dat)	19
	Obtaining an LSF license	19
	Updating licenses	20
	Creating the FLEXlm service	22
	Starting the license server on Windows	23
	Checking the license server status	23
	Installing a new permanent license	24
3	LSF Default User Mapping	25
	Contents	25
	About LSF default user mapping	25
	Specifying user names	27
	Configuring LSF default user mapping	28
	Syntax substitution for Windows user names	29
4	Environment	31
	Contents	31
	Job execution environment	31
	Controlling execution environment using job starters	32
5	Charting Resources with Windows Performance Monitor	33
	Contents	33
	LSF Monitor statistics	33
	Installing LSF Monitor	34
	Configuring LSF Monitor	35
	Using LSF Monitor	36
	Uninstalling LSF Monitor	36

6	Dynamic IP Addressing for LSF Hosts	37
	Contents	37
	About Dynamic Host Configuration Protocol (DHCP)	37
	How LSF works with dynamic IP addressing	37
	Setup	38
	Requirements	38
	Configuring WINS	38
	Configuring dynamic IP addressing	38
7	Displaying GUIs in LSF with Microsoft Terminal Services	41
	Contents	41
	How LSF Works with Terminal Services	42
	Requirements	43
	Configuring Terminal Services for LSF	44
	Configuring LSF to run Terminal Services jobs	45
	Submitting LSF jobs to Terminal Services hosts (tssub)	46
	Limiting the number of Terminal Services jobs on a host	47
	Submitting LSF jobs to Terminal Services hosts from UNIX	48
8	Installing LSF in a Mixed Cluster	49
	Contents	49
	Setting up a Linux cluster with Windows compute nodes	49
	Setting up a Windows cluster with Linux compute nodes	53
	Index	55

## Testing Your LSF Installation

Before you make LSF available to users, you should make sure LSF is installed and operating correctly. This chapter describes how to use some basic LSF commands to do the following:

- ◆ Check the cluster configuration
- ◆ Start the LSF daemons (LSF services)
- ◆ Verify that your new cluster is operating correctly

If you have a mixed UNIX and Windows cluster, make sure you can perform operations from both UNIX and Windows hosts.

### Contents

- ◆ [Checking the license server \(permanent LSF license\)](#) on page 6
- ◆ [Checking the cluster](#) on page 7
- ◆ [Checking the LSF batch system](#) on page 13

Checking the license server (permanent LSF license)

## Checking the license server (permanent LSF license)

If you are using a DEMO license, proceed to [Checking the cluster](#) on page 7.

If you are using a permanent LSF license, perform the steps indicated to check the license server.

### Check the License Server is started

The FLEXlm License Server service is installed as a Windows service to start automatically.

To check the License Server is started:

- ◆ Select **Start > Settings > Control Panel > Services** and make sure the FLEXlm License Server service is started.

### Display license server status

#### The lmstat command

Use the `lmstat` command to check the License Server status and display the number of licenses available. You must use the `-c` option to specify the path to the LSF license file.

For example, depending on the LSF features installed, the output of the command should look something like the following:

```
C:\lsf\7.0\etc> lmutil lmstat -a -c %LSF_ENVDIR%/license.dat
lmutil - Copyright (C) 1989-2000 Globetrotter Software, Inc.
Flexible License Manager status on Fri 05/24/2002 13:23
License server status: 1711@hostA
    License file(s) on hostA: f:\winnt\system32\\hostA\c$\flexlm\license.dat:

    hostA: license server UP (MASTER) v7.0

Vendor daemon status (on hostA):

    lsf_ld: UP v7.0

Feature usage info:

Users of lsf_base: (Total of 2 licenses available)

Users of lsf_manager: (Total of 2 licenses available)

Users of lsf_sched_fairshare: (Total of 2 licenses available)
...
```



## Checking the cluster

```
Dec 21 08:38:59 2007 4196:1492 6 7.02 Master LIM is not running in
EGO_DISABLE_UNRESOLVABLE_HOST mode.
Dec 21 08:38:59 2007 4196:1492 5 7.02 C:\LSF_7.0\7.0\etc\lim.exe -C
Dec 21 08:38:59 2007 4196:1492 7 7.02 setMyClusterName: searching cluster files...
Dec 21 08:38:59 2007 4196:1492 7 7.02 setMyClusterName: local host hostA belongs to
cluster cluster1
Dec 21 08:38:59 2007 4196:1492 3 7.02 domanager(): C:\LSF_7.0\
conf\lsf.cluster.cluster1(13):
The cluster manager is the invoker <LSF\lsfadmin> in debug mode
Dec 21 08:38:59 2007 4196:1492 6 7.02 reCheckClass: numhosts 1 so reset exchIntvl to 15.00
Dec 21 08:38:59 2007 4196:1492 7 7.02 getDesktopWindow: no Desktop time window configured
Dec 21 08:38:59 2007 4196:1492 6 7.02 Checking Done.
-----
No errors found.
```

The messages shown are typical of normal output from `lsadmin ckconfig -v`. Other messages may indicate problems with the Platform LSF configuration. See the *Platform LSF Reference* for help with some common configuration errors.

## Start the cluster

When you first start the cluster, it takes LSF some time to select an LSF master host. During this time (approximately 20 seconds) the cluster may not be able to locate the master host.

Use the following command to start the LSF cluster:

```
C:\lsf\7.0\bin> lsfstartup
```

This command starts the LSF services, Platform LIM, Platform RES, and Platform SBD on all LSF Windows hosts.

## Mixed cluster

If you have a mixed UNIX-Windows cluster, you will need to log on to a UNIX host and start the UNIX daemons with `lsfstartup`, and then log on to a Windows host and use `lsfstartup` from a Windows host to start LSF services on all Windows hosts.

## Check the Load Information Manager (LIM)

If all the following commands display correct output, the LIMs are running correctly.

### The `lsid` command

The `lsid` command displays the cluster name and master host name.

The master name displayed by `lsid` may vary, but it is usually the first host configured in the `Hosts` section of the `LSF_CONFDIR\lsf.cluster.cluster_name` file.

```
lsid
Platform LSF 7 Update 2 Jan 31 2008
Copyright 1992-2007 Platform Computing Corporation
My cluster name is cluster1
My master name is hostA.platform.com
```

### The `lsinfo` command

The `lsinfo` command displays cluster configuration information about resources, host types, and host models. The information displayed by `lsinfo` is configured in `LSF_CONFDIR\lsf.shared`.

Depending on the LSF products installed, and the host types configured in your cluster, the output of the command should look something like the following. The ellipsis (...) indicates where the full output has been shortened for appearance.

In this example, only built-in resources are shown. Refer to *Administering Platform LSF* for information about configuring custom resources.

```
lshinfo
RESOURCE_NAME  TYPE  ORDER  DESCRIPTION
r15s           Numeric  Inc    15-second CPU run queue length
r1m            Numeric  Inc    1-minute CPU run queue length (alias: cpu)
r15m          Numeric  Inc    15-minute CPU run queue length
ut             Numeric  Inc    1-minute CPU utilization (0.0 to 1.0)
pg            Numeric  Inc    Paging rate (pages/second)
io            Numeric  Inc    Disk IO rate (Kbytes/second)
ls            Numeric  Inc    Number of login sessions (alias: login)
it            Numeric  Dec    Idle time (minutes) (alias: idle)
tmp           Numeric  Dec    Disk space in /tmp (Mbytes)
swp           Numeric  Dec    Available swap space (Mbytes) (alias: swap)
mem           Numeric  Dec    Available memory (Mbytes)
```

...

```
TYPE_NAME
UNKNOWN_AUTO_DETECT
DEFAULT
DigitalUNIX
HPPA
IBMAIX3
NTX86
NTALPHA
SGI6
SUNSOL
WIN95
```

...

```
MODEL_NAME      CPU_FACTOR  ARCHITECTURE
Ultra5S          10.30      SUNWUltra510_270_sparcv9
HP300            1.00
PENT_100         7.00
PC450            13.20     i686_448
NEWS5000         7.00
INDIGOXS24       7.00
SunSparc         12.00
```

...

## The lshosts command

The `lshosts` command displays configuration information and status of LSF hosts.

The output contains one line for each host in the cluster. Type, model, and resource information is configured in the `LSF_CONFDIR\lsf.cluster.cluster_name` file.

The `cpuf` matches the CPU factor given for the host model in `LSF_CONFDIR\lsf.shared`.

```
lshosts
HOST_NAME  type  model  cpuf  ncpus  maxmem  maxswp  server  RESOURCES
HostA     NTX86  PC450  13.2  1      127M    514M    Yes    (win2k)
HostB     SUNSOL5  DEFAULT  1.0  4      1024M   1934M   Yes    ( )
```

## Checking the cluster

```
HostC   SGI6   DEFAULT  1.0   -   -   -   Yes   (  
HostD   HPPA   DEFAULT  1.0   1   108M  256M  Yes   (  
)
```

## The lsload command

The `lsload` command displays the current load levels of the cluster.

The output contains one line for each host in the cluster. The status should be `ok` for all hosts in your cluster.

```
lsload  
HOST_NAME  status  r15s  r1m  r15m  ut   pg  ls   it   tmp   swp   mem  
HostA      ok      0.0   0.0   0.0   6%  0.2  2   1365  97M  65M  29M  
HostB      ok      0.1   0.1   0.2   9%  0.0  4    1   130M  319M  12M  
HostC      ok      2.5   2.2   1.9  64%  56.7  50   0   929M  931M  4000M  
HostD      ok      0.2   0.2   0.2   1%  0.0  0    0   93M   86M  50M
```

## Check the Remote Execution Server (RES)

Make sure you have input your user password using `lspasswd`.

If all the following commands display correct output, RES on all hosts is running correctly.

**The lsrun command** The `lsrun` command runs a command on one LSF host through RES. For example, the following command runs the `hostname` command on the remote host `hostA`:

```
lsrun -v -m hostA hostname  
<<Execute hostname on remote host hostA>>  
hostA
```

## The lsgrun command

The `lsgrun` command runs a command on a group of hosts through RES. For example, the following command runs the `hostname` command on three remote hosts:

```
lsgrun -v -m "hostA hostB hostC" hostname  
<<Executing hostname on hostA>>  
hostA  
<<Executing hostname on hostB>>  
hostB  
<<Executing hostname on hostC>>  
hostC  
<<Executing hostname on hostD>>  
hostD
```

## The lsclusters command

The `lsclusters` command displays cross-cluster configuration information. The status should be `ok` for your cluster.

```
lsclusters -l  
CLUSTER_NAME  STATUS  MASTER_HOST  ADMIN  HOSTS  SERVERS  
cluster1      ok      HostA        lsfadmin  4      4  
LSF administrators: lsfadmin  
Available resources: win2k  
Available host types: WINX86  
Available host models: UNKNOWN_AUTO_DETECT PC450  
Accept jobs from this cluster: yes  
Send jobs to this cluster: yes
```

## For more information

- ◆ For more information about LSF commands, refer to *Administering Platform LSF* and the *Platform LSF Reference*.

## LSF on Platform EGO

LSF on Platform EGO allows EGO to serve as the central resource broker, enabling enterprise applications to benefit from sharing of resources across the enterprise grid.

See *Administering Platform LSF* for more information about LSF on Platform EGO.

See *Administering and Using Platform EGO* for detailed information about EGO administration.

## How to handle parameters in `lsf.conf` with corresponding parameters in `ego.conf`

When EGO is enabled, existing LSF parameters (parameter names beginning with `LSB_` or `LSF_`) that are set only in `lsf.conf` operate as usual because LSF daemons and commands read both `lsf.conf` and `ego.conf`.

Some existing LSF parameters have corresponding EGO parameter names in `ego.conf` (`LSF_CONFDIR\lsf.conf` is a separate file from `LSF_CONFDIR\ego\cluster_name\kernel\ego.conf`). You can keep your existing LSF parameters in `lsf.conf`, or you can set the corresponding EGO parameters in `ego.conf` that have not already been set in `lsf.conf`.

You cannot set LSF parameters in `ego.conf`, but you can set the following EGO parameters related to LIM, PIM, and ELIM in either `lsf.conf` or `ego.conf`:

- ◆ `EGO_DAEMONS_CPUS`
- ◆ `EGO_DEFINE_NCPUS`
- ◆ `EGO_SLAVE_CTRL_REMOTE_HOST`
- ◆ `EGO_WORKDIR`
- ◆ `EGO_PIM_SWAP_REPORT`

You cannot set any other EGO parameters (parameter names beginning with `EGO_`) in `lsf.conf`. If EGO is not enabled, you can only set these parameters in `lsf.conf`.

---

**NOTE:** If you specify a parameter in `lsf.conf` and you also specify the corresponding parameter in `ego.conf`, the parameter value in `ego.conf` takes precedence over the conflicting parameter in `lsf.conf`.

If the parameter is not set in either `lsf.conf` or `ego.conf`, the default takes effect depends on whether EGO is enabled. If EGO is not enabled, then the LSF default takes effect. If EGO is enabled, the EGO default takes effect. In most cases, the default is the same.

Some parameters in `lsf.conf` do not have exactly the same behaviour, valid values, syntax, or default value as the corresponding parameter in `ego.conf`, so in general, you should not set them in both files. If you need LSF parameters for backwards compatibility, you should set them only in `lsf.conf`.

---

If you have LSF 6.2 hosts in your cluster, they can only read `lsf.conf`, so you must set LSF parameters only in `lsf.conf`.

### LSF and EGO corresponding parameters

The following table summarizes existing LSF parameters that have corresponding EGO parameter names. You must continue to set other LSF parameters in `lsf.conf`.

lsf.conf parameter	ego.conf parameter
LSF_API_CONNTIMEOUT	EGO_LIM_CONNTIMEOUT
LSF_API_RECVTIMEOUT	EGO_LIM_RECVTIMEOUT
LSF_CLUSTER_ID (Windows)	EGO_CLUSTER_ID (Windows)
LSF_CONF_RETRY_INT	EGO_CONF_RETRY_INT
LSF_CONF_RETRY_MAX	EGO_CONF_RETRY_MAX
LSF_DEBUG_LIM	EGO_DEBUG_LIM
LSF_DHPC_ENV	EGO_DHPC_ENV
LSF_DYNAMIC_HOST_TIMEOUT	EGO_DYNAMIC_HOST_TIMEOUT
LSF_DYNAMIC_HOST_WAIT_TIME	EGO_DYNAMIC_HOST_WAIT_TIME
LSF_ENABLE_DUALCORE	EGO_ENABLE_DUALCORE
LSF_GET_CONF	EGO_GET_CONF
LSF_GETCONF_MAX	EGO_GETCONF_MAX
LSF_LIM_DEBUG	EGO_LIM_DEBUG
LSF_LIM_PORT	EGO_LIM_PORT
LSF_LOCAL_RESOURCES	EGO_LOCAL_RESOURCES
LSF_LOG_MASK	EGO_LOG_MASK
LSF_MASTER_LIST	EGO_MASTER_LIST
LSF_PIM_INFODIR	EGO_PIM_INFODIR
LSF_PIM_SLEEPTIME	EGO_PIM_SLEEPTIME
LSF_PIM_SLEEPTIME_UPDATE	EGO_PIM_SLEEPTIME_UPDATE
LSF_RSH	EGO_RSH
LSF_STRIP_DOMAIN	EGO_STRIP_DOMAIN
LSF_TIME_LIM	EGO_TIME_LIM

#### Parameters that have changed in LSF

The default for `LSF_LIM_PORT` has changed to accommodate EGO default port configuration. On EGO, default ports start with `lim` at 7869, and are numbered consecutively for `pem`, `vemkd`, and `egosc`.

This is different from previous LSF releases where the default `LSF_LIM_PORT` was 6879. `res`, `sbatchd`, and `mbatchd` continue to use the default pre-version 7 ports 6878, 6881, and 6882.

Upgrade installation preserves existing port settings for `lim`, `res`, `sbatchd`, and `mbatchd`. EGO `pem`, `vemkd`, and `egosc` use default EGO ports starting at 7870, if they do not conflict with existing `lim`, `res`, `sbatchd`, and `mbatchd` ports.

#### EGO connection ports and base port

On every host, a set of connection ports must be free for use by LSF and EGO components.

LSF and EGO require exclusive use of certain ports for communication. EGO uses the same four consecutive ports on every host in the cluster. The first of these is called the base port.

The default EGO base connection port is 7869. By default, EGO uses four consecutive ports starting from the base port. By default, EGO uses ports 7869-7872.

The ports can be customized by customizing the base port. For example, if the base port is 6880, EGO uses ports 6880-6883.

LSF and EGO needs the same ports on every host, so you must specify the same base port on every host.

## Checking the LSF batch system

To check the LSF batch system, complete the following steps:

- 1 Verify the LSF batch daemon configuration using the `badmin` command.
- 2 Check the LSF batch system by running a few basic commands: `bhosts`, `bqueues`, `bsub`, `bjobs`.

To perform these checks, LIM and `mbatchd` must be running on the master host and on the submission host, which is the host from which you are running the command. See [Start the cluster](#) on page 8 for information about starting LSF services.

Refer to the *LSF Reference* for an explanation of the output for the LSF commands discussed in this section.

### Verify the LSF batch daemon configuration

#### The `badmin` command

The `badmin` command controls and monitors the operation of the LSF Batch system. Use the `badmin ckconfig` command to check the LSF Batch configuration files. The `-v` option displays detailed information about the configuration:

```
C:\LSF_7.0>badmin ckconfig -v
```

```
Checking configuration files ...
```

```
-----  
No errors found.
```

The messages shown above are the normal output from `badmin ckconfig -v`. Other messages may indicate problems with the Platform LSF Batch configuration. Refer to the *Platform LSF Reference* for help with some common configuration errors.

### Display batch hosts

#### The `bhosts` command

The `bhosts` command displays the status of batch server hosts in the cluster. The status should be `ok` for all hosts in your cluster.

```
C:\lsf\bin>bhosts
```

HOST_NAME	STATUS	JL/U	MAX	NJOBS	RUN	SSUSP	USUSP	RSV
hostA	ok	-	-	0	0	0	0	0
hostB	ok	-	-	0	0	0	0	0
hostC	ok	-	-	0	0	0	0	0
hostD	ok	-	-	0	0	0	0	0

## Display batch queues

### The bqueues command

The `bqueues` command checks available queues and their configuration parameters. For a queue to accept and dispatch jobs, the status should be Open:Active. Queue information displayed by `bqueues` is configured in `LSB_CONFDIR\cluster_name\configdir\lsb.queues`.

```
C:\lsf\bin>bqueues
QUEUE_NAME      PRIO  STATUS          MAX  JL/U  JL/P  JL/H  NJOBS  PEND  RUN  SUSP
owners           43    Open:Active      -    6    -    -    0      0     0    0
priority         43    Open:Active      -    -    -    -    0      0     0    0
night            40    Open:Active      -    -    -    -    0      0     0    0
chkpnt_rerun_qu 40    Open:Active      -    -    -    -    0      0     0    0
short            35    Open:Active      -    -    -    -    0      0     0    0
license          33    Open:Active      -    -    -    -    0      0     0    0
normal           30    Open:Active      -    -    -    -    0      0     0    0
idle             20    Open:Active      -    -    -    -    0      0     0    0
```

## Display the default batch queue

### The bparams command

The `bparams` command displays information about the LSF Batch configuration parameters. Use `bparams` to display the name of the default queue:

```
C:\lsf\bin>bparams
Default Queues:  normal
Job Dispatch Interval: 20 seconds
Job Checking Interval: 15 seconds
Job Accepting Interval: 20 seconds
```

The `DEFAULT_QUEUE` parameter in `LSB_CONFDIR\cluster_name\configdir\lsb.params` defines which queue is the default queue.

## Submit a test job

### The bsub command

The `bsub` command submits jobs to LSF queues.

For example, the following command submits a sleep job to the default queue named `normal`:

```
C:\lsf\7.0\bin> bsub sleep 60
Job <1> is submitted to default queue <normal>.
```

## Display batch jobs

### The bjobs command

The `bjobs` command displays the job status. The `bjobs -l` option displays a long format of jobs running in the batch system. Use `bjobs -w` to display the full user name, including domain name.

```
C:\lsf\7.0\bin> bjobs
JOBID USER      STAT  QUEUE    FROM_HOST  EXEC_HOST  JOB_NAME  SUBMIT_TIME
1    lsfadmin  RUN   normal   hostA      hostB      sleep 60  Jan 5 17:39:58
```

If all hosts are busy, the job is not started immediately and the `STAT` column says `PEND`. The job `sleep 60` should take one minute to run. When the job completes, LSF sends mail reporting the job completion.

## For more information

- ◆ For more information about LSF commands, refer to *Administering Platform LSF* and *Platform LSF Reference*.

## Test the Platform Management Console (PMC)

- 1 Browse to the web server URL and log in to the PMC as user Admin with password Admin.
  - ❖ If you have only one management host (the master host), the web server URL is `http://master_host:8080/platform`.
  - ❖ If you have multiple management hosts, locate the web server:
    - 1 Log on as `lsfadmin` and run `egosh client view`.  
This command locates the PMC. It is only needed if EGO is enabled.
    - 2 Scan the client list for a name preceded by `GUIURL`, such as `GUIURL_HostW`.
    - 3 The additional information shows the web server URL; for example, `http://Host_W:8080/platform`.
- 2 As a security measure, use the PMC to change the Admin and Guest account passwords from the simple default passwords, Admin and Guest.

Test the Platform Management Console (PMC)

## Permanent LSF Licenses and FLEXlm on Windows hosts

Managing LSF licenses is described in *Administering Platform LSF*. This chapter includes additional information that is necessary to license a cluster that includes Windows hosts. You must have a valid license to run LSF.

If you install a permanent license using the LSF installer, and you do not already use FLEXlm software, you must install and configure FLEXlm before running the LSF installer.

### Contents

- ◆ [Types of LSF licenses](#) on page 18
- ◆ [FLEXlm License Management](#) on page 18
- ◆ [LSF license file \(license.dat\)](#) on page 19
- ◆ [Obtaining an LSF license](#) on page 19
- ◆ [Updating licenses](#) on page 20
- ◆ [Starting the license server on Windows](#) on page 23
- ◆ [Checking the license server status](#) on page 23
- ◆ [Installing a new permanent license](#) on page 24

## Types of LSF licenses

You can run LSF with either an DEMO (evaluation) or a permanent license.

**DEMO (evaluation) licenses** If you have not yet purchased LSF, get a demo license key. Typically, a demo license licenses all LSF products and expires after 30 days. If you purchase LSF during this time, you can switch to a permanent license with no interruption in service.

**Permanent licenses** A permanent license restricts the total number of hosts in the cluster and defines which LSF products will be used. It defines which hosts in your network will run LSF, and how they are arranged into clusters. This type of license does not expire, but has to be updated if you add more hosts or add another LSF product.

## FLEXlm License Management

Platform LSF uses the FLEXlm license management software from Macrovision Corporation.

LSF is compatible with FLEXlm version 7.0 or later. If you run an older version, you must upgrade FLEXlm before you can use LSF.

### FLEXlm license server

A FLEXlm license server is required for permanent licenses only. Permanent LSF licenses are managed by the FLEXlm license server daemon (`lmgrd`) running on one or three hosts in your network.

If you install a permanent license using the LSF installer, and you do not already use FLEXlm software, you must install and configure FLEXlm before running the LSF installer.

To get a permanent license, you must provide information about the license server host. Platform Computing requires the hardware host name and host identifier for each license server host at your site.

Choose a reliable host as the FLEXlm license server to ensure that the LSF licenses are always available. In many installations, the license server runs on the host that is the dedicated file server for the Platform LSF software. This permits the licenses to be available whenever LSF is available.

You can specify 3 hosts to be license servers instead of just one. For more information about using multiple license servers for licensing Platform LSF, see in *Administering Platform LSF*.

### Existing FLEXlm server

If you already have a FLEXlm license server installed, you can merge the contents of your LSF license file into your existing FLEXlm license file. See [Updating a FLEXlm License](#) on page 21 for steps.

### FLEXlm license management tools

*Administering Platform LSF* describes FLEXlm commands and utilities that can help you manage your FLEXlm licenses. On Windows, these FLEXlm commands are subcommands of `lmutil`.

For example, to retrieve the FLEXlm host ID, run `lmutil lmhostid` (on UNIX, run `lmhostid`). To check the license server, run `lmutil lmstat` (on UNIX, run `lmstat`).

## LSF license file (license.dat)

The LSF license is stored in a text file. By default, the file name is `license.dat`. Your permanent license file must contain all the products that are defined in `LSF_CONFDIR\lsf.cluster.cluster_name`.

For a description of the license file format, see *Administering Platform LSF*.

### Locations of the license file

By default, there are two copies of your permanent license file. The installer copies your LSF license to the LSF directory, but you must update the FLEXlm directory manually. After the permanent LSF license is installed, you have to update both copies when you make changes to your LSF licensing.

- ◆ LSF uses the file specified by `LSF_LICENSE_FILE` in `lsf.conf`. By default, the file is `license.dat` in `LSF_CONFDIR`. For example, if your LSF installation directory is `lsfshare\lsf` on `HostA`, your `lsf.conf` file might include the following line:

```
LSF_LICENSE_FILE=\\HostA\lsfshare\lsf\conf\license.dat
```

- ◆ The FLEXlm license manager uses the file in the FLEXlm installation directory on the host that is the FLEXlm license server. By default, the file is `license.dat` in `C:\flexlm`. For example, your license server might include the following file:  
`C:\flexlm\license.dat`

### Using FLEXlm to manage other software

If you use FLEXlm to manage other software, the `license.dat` file used by FLEXlm contains information that is not related to LSF, but the `license.dat` file used by LSF only contains LSF information. Even if the files in both locations are not identical, the lines relating to LSF must be exactly the same.

### Modifying the LSF license

- ◆ On Windows, you must always modify the file used by FLEXlm manually. The installer does not overwrite the file `C:\flexlm\license.dat` if it already exists.
- ◆ If you move the license file used by LSF, you must update the `LSF_LICENSE_FILE` parameter in `lsf.conf`.

## Obtaining an LSF license

Where to get it:

- ◆ Contact your LSF vendor.
- ◆ Complete the evaluation form on the Platform Web site ([www.platform.com](http://www.platform.com)).

To allow Platform or your LSF vendor to create or update a permanent license, you must provide the following information:

- ◆ Host name of the license server
- ◆ FLEXlm host ID of the license server
- ◆ Number of LSF license units

## Get the host name

To retrieve the host name of the license server:

- 1 Log on to the host that is going to be your license server.
- 2 Open a command prompt (Start | Programs | Command Prompt).
- 3 Type the `hostname` command:

```
c:\>hostname
```

Windows displays the hardware host name.

## Get the FLEXlm host ID

To retrieve the FLEXlm host ID of the license server, run the `lmutil lmhostid` command at an MS-DOS prompt:

```
C:\temp>%LSF_SERVERDIR%\lmutil lmhostid
```

```
lmutil - Copyright (C) 1989-1997 Globetrotter Software, Inc.
```

```
The FLEXlm host ID of this machine is "0090273eeb99"
```

## Updating licenses

Once you have created an LSF cluster, you can update the licensing at any time. The procedure varies:

- ◆ [Update an evaluation cluster](#) on page 20
- ◆ [Expand an existing cluster](#) on page 20
- ◆ [Upgrading to a new version of LSF](#) on page 21

### Update an evaluation cluster

If you have a demo license, you must update it to a permanent license when you finish the evaluation and purchase LSF. LSF permanent licenses use FLEXlm license management software.

- 1 If you already use FLEXlm, check the version. LSF is compatible with FLEXlm version or later. If you have an older version, you must upgrade FLEXlm before installing the permanent LSF license.
- 2 If you do not have FLEXlm installed, see [FLEXlm License Management](#) on page 18 to learn about FLEXlm and permanent licensing.
- 3 Obtain a new permanent license. See [Obtaining an LSF license](#) on page 19.
- 4 If you already have FLEXlm installed, you must update your FLEXlm license file manually. Append the entire contents of the new LSF license to your existing file. See [Updating a FLEXlm License](#) on page 21.
- 5 If you have not yet installed FLEXlm, you must install FLEXlm, then set up the permanent license file used by FLEXlm. See [Installing a new permanent license](#) on page 24.

### Expand an existing cluster

When you have a permanent LSF license, the number of hosts and selection of LSF products and features is controlled by licensing.

- ◆ To expand the size of the cluster by adding hosts, you probably need to purchase additional licenses for the additional hosts.
- ◆ To enable new LSF products or features such as LSF MultiCluster, you need to purchase additional licenses. Sometimes you also need to install additional software. Update your license before you update the PRODUCTS line in the Parameters section of `LSF_CONFDIR\lsf.cluster.cluster_name`.

To make a change that involves additional licensing, the first step is to manually update your existing license files. In the license file used by FLEXlm, the lines related to LSF must be identical to the contents of the license file used by LSF.

- 1 Obtain a new permanent license. See [Obtaining an LSF license](#) on page 19. Instead of getting an entire new license, you will probably get an LSF INCREMENT license key or a new FEATURE line. Append the new lines to the existing LSF content without overwriting or deleting any information. If you have INCREMENT lines, you must add them immediately after the FEATURE line for the product.
- 2 Update FLEXlm. See [Updating a FLEXlm License](#) on page 21.
- 3 Update LSF. See [Updating LSF license features](#) on page 21.

## Upgrading to a new version of LSF

Sometimes a minor upgrade or patch does not require any changes to the licensing. If the upgrade requires a new license, update the FLEXlm license manually and follow the steps in *Migrate Your Windows Cluster Platform LSF Version 7*.

## Updating LSF license features

- 1 Open the license file using a text editor such as Notepad. The license file used by LSF is specified by `LSF_LICENSE_FILE` in `lsf.conf`.
- 2 Make the required changes. This normally involves appending the new lines without overwriting or deleting existing content.
- 3 Save and close the file.
- 4 Reconfigure LSF:
 

```
lsadmin reconfig
```

## Updating a FLEXlm License

- 1 Open the license file using a text editor. The license file used by FLEXlm is normally `c:\flexlm\license.dat` on the license server host.
- 2 Make the required changes. This normally involves adding the new lines to your FLEXlm license. The only time you replace existing lines is when you upgrade from one version of LSF to another. Do not overwrite or delete any content that is not related to LSF.
- 3 Save and close the file.

- 4 Update FLEXlm. Log on to the license server host and re-read the updated license file:

```
c:\flexlm\bin> lmutil lmreread -c c:\flexlm\license.dat
```

If the command is successful, you should see a message similar to the following:

```
lmreread - Copyright (C) 1989-1997 Globetrotter Software, Inc.  
lmreread successful
```

## Creating the FLEXlm service

If you are using a permanent LSF license, you must run `installs.exe` to install or remove the FLEXlm license manager as a Windows Service. `installs.exe` is located under `LSF_TOP\7.0\etc`.

The `installs` command has the following syntax:

```
installs -c license_file_path -e path_to_lmgrd.exe -l log_file_path  
-n service_name [-k lmgrd_parameters]
```

If `-n` is not specified, FLEXlm License Manager is used as the service name.

The `-k` switch is optional and is used to pass one or more startup command-line parameters (`-local`, `-x lmdown`, and `-x lmremove`) to `lmgrd`.

See the *FLEXlm Reference Manual* for more information about the `installs` command.

## Install the FLEXlm service

Run `installs.exe` to install or remove the FLEXlm license manager as a Windows Service.

- 1 Open a Windows Command Prompt.
- 2 Run the `installs` command.

```
LSF_TOP\7.0\etc\installs -c license_file_path -e path_to_lmgrd.exe
```

For example:

```
C:\LSF_7Update2\7.0\etc\installs -c c:\sharetest\license.dat  
-e c:\LSF_7Update2\7.0\etc\lmgrd.exe
```

The `lmgrd.exe` is located under `LSF_TOP\7.0\etc`.

## Remove the FLEXlm service

If you need to remove the FLEXlm license manager service, run `installs.exe`.

- 1 Open a Windows Command Prompt.
- 2 Run the `installs` command.

```
LSF_TOP\7.0\etc\installs.exe -r -n "service_name"
```

For example:

```
LSF_TOP\7.0\etc\installs.exe -r -n "FLEXlm License Manager"
```

If you did not specify `-n` when you installed the license manager service, FLEXlm License Manager is used as the default service name.

## Starting the license server on Windows

### Start FLEXlm automatically

- 1 Restart the license server host.

### Start FLEXlm manually

- 1 Log on to the license server host.
- 2 Choose **Services** in the Windows Control Panel.
- 3 Choose **Services** in the Windows Control Panel.

## Checking the license server status

If you are using a permanent LSF license, check the license server by starting the license manager and displaying the status with the `lmutil lmstat` command on Windows.

If you are using a demo license, you do not have to perform this check because a demo license does not run a license server daemon.

### The `lmutil lmstat` command

After starting the license server, use the `c:\flexlm\bin\lmutil lmstat` command on Windows to check its status and display the number of licenses available. You must use the `-c` option of `lmstat` to specify the path to the LSF license file.

The output of `lmstat` gives the status of the license server together with the name of the LSF vendor daemon (`lsf_ld`). The usage information for each product in the license file is provided.

For example, depending on the LSF features installed, the output of the command should look something like the following:

```
C:\flexlm\bin> lmutil lmstat -a -c c:\flexlm\license.dat
lmutil - Copyright (C) 1989-2000 Globetrotter Software, Inc.
Flexible License Manager status on Fri 05/24/2002 13:23
License server status: 1711@hostA
License file(s) on hostA: f:\winnt\system32\\hostA\c$\flexlm\license.dat:

hostA: license server UP (MASTER) v7.0

Vendor daemon status (on hostA):

lsf_ld: UP v7.0

Feature usage info:
```

## Installing a new permanent license

Users of lsf\_base: (Total of 2 licenses available)

Users of lsf\_manager: (Total of 2 licenses available)

Users of lsf\_sched\_fairshare: (Total of 2 licenses available)

...

## Installing a new permanent license

**Prerequisites:** If you do not already use FLEXlm, you must make sure FLEXlm is installed and running properly.

Complete the following steps to replace a temporary demo license used by LSF with a permanent license.

- 1 Shut down the master host.

```
lsfshutdown -m master_host_name
```

- 2 Copy the path to the permanent license to LSF\_LICENSE\_FILE defined in lsf.conf.

If you installed LSF with a default installation, the license file is installed in the LSF configuration directory (`LSF_CONFDIR\license.dat`).

**TIP:** You can put the permanent license anywhere you want, as long as you update the LSF\_LICENSE\_FILE parameter in `$LSF_CONFDIR\lsf.conf`.

- 3 Start the master host.

```
lsfstartup -m master_host_name
```

## LSF Default User Mapping

The default user mapping in LSF has no effect on a UNIX-only cluster. You do not need to understand this feature unless your cluster includes Windows hosts.

### Contents

- ◆ [About LSF default user mapping](#) on page 25
- ◆ [Specifying user names](#) on page 27
- ◆ [Configuring LSF default user mapping](#) on page 28
- ◆ [Syntax substitution for Windows user names](#) on page 29

### About LSF default user mapping

The default user mapping determines whether you can specify a Windows user in LSF by the user name alone. In a mixed cluster, it also specifies whether a Windows user account maps to a UNIX account of the same name, to allow cross-platform operation.

### How LSF default user mapping works

If you specify an LSF user domain, the default user mapping is enabled. For a multiple-domain Windows environment on a UNIX-Windows mixed cluster, you can specify an unlimited number of Windows domains as the LSF user domain.

When the default user mapping is enabled,

- ◆ A user name specified without a domain is interpreted (on a Windows host) as belonging to the LSF user domain
- ◆ A user name specified with the domain name of the LSF user domain is stripped of the domain name

### Mixed cluster

In a mixed UNIX-Windows environment, if your Windows account in the LSF user domain has the same user name as your UNIX account, LSF's default user mapping lets LSF schedule and track jobs from both accounts as if they belong to a single user. On the execution host, LSF automatically runs the job using whichever of the two accounts is appropriate for that host.

## About LSF default user mapping

To submit cross-platform jobs when your accounts have different user names in different environments, you should configure user account mapping for individual users. For more information, see *Administering Platform LSF*.

### Multiple domain accounts

To run jobs, the existing domain trust relationships apply in LSF, so if the execution domain trusts the submission domain, your job can run in the execution domain under your submission account.

If a user domain is...	Then LSF treats the Windows and UNIX user as...
specified by the parameter LSF_USER_DOMAIN	the same user
not specified by the parameter LSF_USER_DOMAIN	different users

Accounts with the same user name in different domains are still treated as separate users by LSF.

You can use the environment variable LSF\_EXECUTE\_DOMAIN to specify only one of the domains listed in LSF\_USER\_DOMAIN. When you specify an execution domain, LSF runs the job using the specified domain user account, without trying all of the domain accounts in the order listed in LSF\_USER\_DOMAIN.

### Local accounts

If your local account has the same user name and password on every Windows host, LSF's default user mapping lets LSF schedule and track jobs from all hosts as if they belong to a single user. On the execution host, LSF automatically runs the job using the local user account.

If your accounts have different user names in different environments, you should configure user account mapping. For more information, see *Administering Platform LSF*.

## Installation examples

In the following examples, assume you are User1, and you have a valid user account in 3 Windows domains as well as a valid UNIX account. Not all the accounts can be used with LSF. Depending on the type of cluster, and the way you install the cluster, here are the different ways that LSF is configured:

### Install or upgrade a UNIX-only cluster

No mapping. You have one UNIX account, and LSF recognizes 1 user:

- ◆ user1 (UNIX account)

### Install a new Windows-only cluster

No mapping. You have 3 Windows accounts. For purposes of fairshare, per-user job slot limits, displaying statistical data, and so on, LSF recognizes 3 separate users:

- ◆ DOMAIN\_A\user1
- ◆ DOMAIN\_B\user1
- ◆ DOMAIN\_C\user1

### Create a new UNIX-Windows cluster

You must enable default user mapping for one of your Windows accounts (such as Domain A) so that you can run cross-platform jobs between UNIX and Windows. LSF recognizes 3 separate users:

- ◆ user1 (your UNIX and Domain A accounts are treated as a single LSF user)
- ◆ DOMAIN\_B\user1

- ◆ DOMAINC\user1

If you never run cross-platform jobs, you might choose to disable default user mapping by not specifying an LSF user domain. LSF then recognizes 4 separate users:

- ◆ user1 (UNIX account)
- ◆ DOMAINA\user1
- ◆ DOMAINB\user1
- ◆ DOMAINC\user1

You can specify multiple domains when you define `LSF_USER_DOMAIN`, which will allow users to submit jobs from a UNIX host in a multiple-domain Windows environment.

## Specifying user names

In a Windows cluster or mixed UNIX-Windows cluster, in a domain environment, LSF users in different Windows domains might have the same user name. Because of this, LSF uses the Windows domain name with the user name, to differentiate the users.

### User name only

When the default mapping is enabled, the user name alone specifies a user in the LSF user domain. The combination of a user name plus the domain name of the LSF user domain is not used in LSF.

### Domain name with user name

#### Default mapping disabled

All Windows user accounts are specified using the domain name with the user name. There is no LSF user domain.

#### Default mapping enabled

User accounts in all domains except for the LSF user domain are specified using the domain name with the user name.

### How to specify a user name with a domain name

Unless a Windows user account belongs to the LSF user domain (`LSF_USER_DOMAIN` in `lsf.conf`), the combination of domain name and user name specifies a Windows domain user in LSF. The syntax is:

```
[DOMAIN_NAME|.]\user_name
```

Type the domain name in capital letters. Use a period (.) instead of a domain name to specify a local account instead of a domain account.

UNIX systems interpret the single backslash as a special character, so on UNIX you have to use a double backslash to specify the domain name in the command line:

#### Windows

```
bjobs -u MYDOMAIN\user1
```

#### UNIX

```
bjobs -u MYDOMAIN\\user1
```

## Viewing user names

Use `bjobs -w` to view information about jobs and see the full name of a Windows user, including domain name.

When you run `bjobs`, the default is to truncate user names, and display the names of Windows users without the domain name.

## Windows user authentication

LSF recognizes UNIX and Windows authentication environments, including different Windows domains and individual Windows workgroup hosts.

- ◆ In a Windows domain environment, user accounts are validated at the domain level, and your user account is valid on all hosts in your domain (and might be valid in other domains, if there is a trust relationship).
- ◆ In a Windows workgroup environment, each host authenticates the user account, so your local account is only valid on one host.

See *Administering Platform LSF* for more information about user authentication in LSF.

## lspasswd command

You must use `lspasswd` or `wgpaswd` to register and update user names and passwords. You must run `lspasswd` from an LSF server host. You cannot run the command from an LSF client host. The password must be 31 characters or less.

You can run `lspasswd` on Windows in a non-shared file system environment. You must define the parameter `LSF_MASTER_LIST` in `lsf.conf` so that jobs will run with the correct permissions. If this parameter is not defined, LSF assumes that the cluster uses a shared file system environment.

You can also run `lspasswd` to check that the password is valid for the specified user, or to remove a user entry from the password database.

See the Platform LSF Reference for more information about `lspasswd` and `wgpaswd`.

## Password problem notification on Windows

A Windows job may not be able to run because of a problem with the user's LSF password (entered and updated using `lspasswd`). If LSF does not recognize the password, the problem could be:

- ◆ The Windows user account password was never registered with LSF with `lspasswd`.
- ◆ The password in Windows changed but was not updated in LSF with `lspasswd`.

If a job is in PENDING state and LSF cannot run it because of a password problem, by default, LSF puts the job into USUSP and then notifies the user via email. The user can fix the problem, and then use `bresume` to release the job from USUSP.

## Configuring LSF default user mapping

- ◆ To enable or modify default user mapping after you install LSF, set `LSF_USER_DOMAIN` in `lsf.conf` and specify the LSF user domain:

```
LSF_USER_DOMAIN=DomainA
```

- ◆ You can also specify multiple domains:

```
LSF_USER_DOMAIN=DomainA:DomainB:DomainC
```

- ◆ Depending on the cluster configuration, you might have to redefine the service accounts, cluster administrators, queue administrators, user group memberships, and so on, so that your cluster remains operational after you restart the cluster.
- ◆ Whenever you make any change to default user mapping, you affect users in the old LSF user domain and in the new LSF user domain. If you specify a new LSF user domain, users in both domains will have to use `lspasswd` to register their new names and passwords.

If users in the old and new LSF user domain have the same user name (such as `olddomain\user1` and `newdomain\user1`), then the `user1` account is already registered with LSF, and the user from the new LSF user domain has to change the password. To change the password, he must input the current password, which was set by the old user.

## Syntax substitution for Windows user names

In *Administering Platform LSF* and other LSF documentation, a user name is represented by the syntax:

*user\_name*

If your cluster includes Windows hosts, the full syntax for a user account on Windows is:

`[DOMAIN_NAME\ | .\]user_name`

Always type the domain name in capital letters.

## LSF commands

In the following LSF commands, use the full syntax to specify a user name.

- ◆ `bchkpnt`
- ◆ `bdel`
- ◆ `bhist`
- ◆ `bjobs`
- ◆ `bkill`
- ◆ `bmig`
- ◆ `bmod`
- ◆ `brequeue`
- ◆ `bresume`
- ◆ `bstop`
- ◆ `bsub`
- ◆ `bswitch`
- ◆ `busers`
- ◆ `lsacct`
- ◆ `lspasswd`

## LSF files

In the following configuration files and parameters, use the full syntax to specify a user name.

## Syntax substitution for Windows user names

- ◆ `lsb.hosts`
  - ❖ `USER_SHARES`
- ◆ `lsb.params`
  - ❖ `SYSTEM_MAPPING_ACCOUNT`
- ◆ `lsb.queues`
  - ❖ `ADMINISTRATORS`
  - ❖ `FAIRSHARE`
  - ❖ `USERS`
- ◆ `lsb.users`
  - ❖ `GROUP_MEMBER`
  - ❖ `USER_SHARES`
  - ❖ `USER_NAME`
  - ❖ `LOCAL`
  - ❖ `REMOTE`
- `lsf.cluster.cluster_name`
  - ❖ `ADMINISTRATORS`
- ◆ `lsf.conf`
  - ❖ `LSF_SHELL_AT_USERS`
- ◆ `lsf.sudoers`
  - ❖ `LSF_EAUTH_USER`
  - ❖ `LSF_EEXEC_USER`
  - ❖ `LSF_STARTUP_USERS`
  - ❖ `LSB_PRE_POST_EXEC_USER`

## Environment

This chapter describes how LSF sets the Windows environment variables.

### Contents

- ◆ [Job execution environment](#) on page 31 (describing the default option)
- ◆ [Controlling execution environment using job starters](#) on page 32 (describing 2 alternative options)

### Job execution environment

By default, Platform LSF transfers environment variables from the submission to the execution host. However, some environment variables do not make sense when transferred.

#### How LSF sets the job execution environment

When submitting a job from a Windows to a UNIX machine, the `-L` option of `bsub` can be used to reinitialize the environment variables. If submitting a job from a UNIX machine to a Windows machine, you can set the environment variables explicitly in your job script.

#### PATH environment variable on UNIX and Windows

Platform LSF automatically resets the `PATH` on the execution host if the submission host is of a different type. If the submission host is Windows and the execution host is UNIX, the `PATH` variable is set to `/bin:/usr/bin:/sbin:/usr/sbin` and the path to the LSF `bin` directory is appended to it.

If the submission host is UNIX and the execution host is Windows, the `PATH` variable is set to the system `PATH` variable with the path to the LSF `bin` directory appended to it. Platform LSF looks for the presence of the `WINDIR` variable in the job's environment to determine whether the job was submitted from a Windows or UNIX host. If `WINDIR` is present, it is assumed that the submission host was Windows; otherwise, the submission host is assumed to be a UNIX machine.

#### Environment variable handling on Windows

The following Windows environment variables are overridden based on the values on the execution host:

## Controlling execution environment using job starters

- ◆ COMPSPEC
- ◆ COMPUTERNAME
- ◆ NTRESKIT
- ◆ OS2LIBPATH
- ◆ PROCESSOR\_ARCHITECTURE
- ◆ PROCESSOR\_LEVEL
- ◆ SYSTEMDRIVE
- ◆ SYSTEMROOT
- ◆ WINDIR

If the WINDIR on the submission and execution host are different, then the system PATH variable on the execution host is used instead of that from the submission host.

Avoid using drive names in environment variables (especially the PATH variable) for drives that are connected over the network. It is preferable to use the UNC form of the path. This is because drive maps are shared between all users logged on to a particular machine. For example, if an interactive user has drive F: mapped to \\serverX\share, then any batch job will also see drive F: mapped to \\serverX\share. However, drive F: might have been mapped to a different share on the submission host of the job.

Job starters can be used to perform more site-specific handling of environment variables.

## Controlling execution environment using job starters

The command `bsub -L` cannot be used for a Windows execution host. LSF provides the following two job starters:

- ◆ `preservestarter` — preserves the default user environment of the execution host; does not include any submission host settings
- ◆ `augmentstarter` — augments the default user environment of the execution host by adding settings from the submission host that are not already defined on the execution host

### Where the job starter executables are located

By default, the job starter executables are installed in `LSF_BINDIR`. If you prefer to store them elsewhere, make sure they are in a directory that is included in the default PATH on the execution host.

For example, on Windows, put the job starter under `%WINDIR%`.

### Source code for the job starters

The source code for the job starters is installed in `LSF_TOP\7.0\examples`.

# Charting Resources with Windows Performance Monitor

LSF integrates with Windows Performance Monitor, so you can chart LSF cluster, host, queue, and job performance information. Windows Performance Monitor can also be used to trigger external commands when specified thresholds are exceeded.

A service called LSF Monitor passes information from LSF to the Windows Performance Monitor. LSF Monitor must be installed separately. On UNIX, you can use `xlsmon` to chart load information or you can use Platform SiteAssure to manage LSF resources on UNIX. To chart historical data, you can use Platform Analyzer.

This chapter provides instructions for installing, configuring, and using LSF Monitor.

## Contents

- ◆ [LSF Monitor statistics](#) on page 33
- ◆ [Installing LSF Monitor](#) on page 34
- ◆ [Configuring LSF Monitor](#) on page 35
- ◆ [Using LSF Monitor](#) on page 36
- ◆ [Uninstalling LSF Monitor](#) on page 36

## LSF Monitor statistics

Once installed, LSF Monitor automatically sends information to the Windows Performance Monitor. Use the Windows Performance Monitor to chart LSF performance information.

The host, queue, and job objects support multiple instances.

The following LSF information is available:

- ◆ [Cluster information](#) on page 34
- ◆ [Host information](#) on page 34
- ◆ [Queue information](#) on page 34
- ◆ [Job information](#) on page 34

- ◆ [External information](#) on page 34

## Cluster information

- ◆ Number of available servers
- ◆ Number of unavailable servers
- ◆ Number of servers where an LSF daemon (sbatchd or RES service) is down
- ◆ Number of unlicensed servers
- ◆ Number of pending jobs in the cluster
- ◆ Number of running jobs in the cluster
- ◆ Number of suspended jobs in the cluster
- ◆ Number of sick jobs (jobs submitted with no password, jobs with job dependency never satisfied, and jobs pending more than 3 days)
- ◆ Response time of LIM (as measured by the time to make an `ls_load` call)
- ◆ Response time of `mbatchd` (as measured by the time to make an `lsb_queueinfo` call)

## Host information

- ◆ Load indices: `r15s`, `r15m`, `mem`, `swap`, `pg`, `ut`
- ◆ Number of running jobs
- ◆ Number of suspended jobs
- ◆ Number of reserved job slots
- ◆ External load Indices

## Queue information

- ◆ Number of pending jobs
- ◆ Number of running jobs
- ◆ Number of suspended jobs
- ◆ Number of reserved job slots

## Job information

- ◆ CPU time used by the job
- ◆ Memory used by the job (for jobs running on UNIX only)
- ◆ Swap space used by the job (for jobs running on UNIX only)

## External information

- ◆ Values of one or two external load indices (configured by the LSF administrator)

# Installing LSF Monitor

## Requirements

**LSF version**            You must have a cluster running LSF version 4.0 or higher.

<b>Host</b>	Install LSF Monitor on any LSF server or client host running Windows.
<b>Cluster</b>	The cluster can include UNIX hosts.
<b>Account</b>	The LSF Monitor service runs under the account of an LSF cluster administrator. You will need to provide an LSF cluster administrator account and password.

## Install LSF Monitor

The LSF Monitor setup program is installed with LSF. Use `lsfmon -install` to actually install the LSF Monitor service:

- 1 Log on to a Windows host as an LSF user in an existing LSF cluster.
- 2 In a command prompt, type:  

```
lsfmon -install
```

LSF Monitor is installed.
- 3 On the Windows Control Panel, click **Services**.  
The Services window opens.
- 4 Right-click **LSF Monitor** and click **Properties**.
- 5 In the Log On As section, deselect **System Account**, select **This Account**, and specify an LSF cluster administrator account (such as Administrator).
- 6 Type in the password twice and click **OK**.
- 7 In the Services window, select **LSF Monitor** and click **Start** to start the service.

## Configuring LSF Monitor

You can configure sample intervals for host, queue and job information along with external load indices.

### In this section

- ◆ [Configure sample intervals](#) on page 35
- ◆ [Configure external load indices](#) on page 36

## Configure sample intervals

<b>Overview</b>	LSF Monitor periodically samples information from LSF and updates the Windows Performance Monitor.
<b>Default sample intervals</b>	By default, information is sampled at the following intervals: <ul style="list-style-type: none"> <li>◆ Host information = 30 seconds</li> <li>◆ Queue information = 45 seconds</li> <li>◆ Job information = 60 seconds</li> </ul>
<b>Change sample intervals</b>	To change the sample intervals for LSF host, job, or queue information, modify the Windows Registry settings. Back up your registry before you make any changes.

To modify the registry:

- 1 Select the Registry subkey:  
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LSFMonitor`
- 2 Edit the appropriate value, and specify the new sample interval in seconds:
  - ❖ `SampleIntervalHost`
  - ❖ `SampleIntervalJob`
  - ❖ `SampleIntervalQueue`

## Configure external load indices

You can use the Windows Performance Monitor to chart up to 2 external load indices, which you specify.

To configure LSF Monitor to monitor external load indices, modify the Windows Registry settings. Update your backups before you modify the Registry.

- 
- 1 Go to the Registry subkey `:HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LSFMonitor`
  - 2 Specify the appropriate value, and type the name of an external load index that is configured in your cluster:
    - ◆ `ExternalLoadIndex1`
    - ◆ `ExternalLoadIndex2`
- 

## Using LSF Monitor

### Start and stop LSF Monitor

Use the Windows Control Panel to start or stop the LSF Monitor service.

### View information about LSF Monitor

Errors related to LSF API calls and the operation of LSF services will be logged to the Windows event log. You can use the Windows Event Viewer to view this information.

## Uninstalling LSF Monitor

Use `lsfmon -remove` to uninstall LSF Monitor. This command will first stop the LSF Monitor service if it is running, then remove it and remove related information from the Windows Registry.

### Uninstall LSF Monitor

To uninstall LSF Monitor:

- ◆ In a DOS window, type:  
`lsfmon -remove`

## Dynamic IP Addressing for LSF Hosts

This chapter describes how to use dynamic IP addressing with an LSF cluster.

### Contents

- ◆ [About Dynamic Host Configuration Protocol \(DHCP\)](#) on page 37.
- ◆ [How LSF works with dynamic IP addressing](#) on page 37.
- ◆ [Setup](#) on page 38.
- ◆ [Requirements](#) on page 38.
- ◆ [Configuring WINS](#) on page 38.
- ◆ [Configuring dynamic IP addressing](#) on page 38.

### About Dynamic Host Configuration Protocol (DHCP)

DHCP (Dynamic Host Configuration Protocol) enables individual computers on an IP network to extract their configurations from particular machines (DHCP servers) that have no exact information about the individual computers until they request the information. This reduces the work necessary to administer a large IP network. The most significant piece of information distributed in this manner is the IP address.

### How LSF works with dynamic IP addressing

LSF hosts running Windows can be configured as DHCP clients, which means their IP address is dynamic. Users who dial in or connect from a remote location might be assigned a different IP address with each connection.

The DHCP server issues an IP address to the LSF host, but LSF gets the IP address from DNS (Domain Name System). A WINS (Windows Internet Naming Service) server synchronizes information between the DHCP and DNS servers.

The IP address should not be changed while there are active TCP/IP connections with the host, for example, while installing LSF or running LSF commands. Normally, the IP address is maintained until the host is restarted or until the network connection is broken.

## Setup

If an LSF client host is assigned a new IP address, you should wait for WINS to update DNS before you use that host to run LSF.

### LSF client hosts

LSF client hosts can be DHCP clients and can change their IP addresses anytime in a running cluster.

### LSF server hosts

Installing dynamic hosts on Windows allows support for dynamic IP addressing for LSF server hosts using DHCP.

LSF server hosts can be DHCP clients and can change their IP addresses anytime in a running cluster. The master host also saves the slave host IP address.

## Setup

To run LSF on hosts that are DHCP clients, take these steps.

- 1 Make sure your system uses DHCP, DNS, and WINS. See [Requirements](#) on page 38.
- 2 Configure a short cache timeout value on the WINS server. See [Configuring WINS](#) on page 38.
- 3 Enable dynamic IP addressing for the LSF cluster. See [Configuring dynamic IP addressing](#) on page 38.

## Requirements

To use DHCP with LSF, your system must include all of the following:

- ◆ DHCP server
- ◆ WINS server
- ◆ DNS server
- ◆ LSF hosts acting as DHCP clients

Install Microsoft DNS server and WINS server on the same machine.

## Configuring WINS

The Cache Timeout Value for the WINS Lookup of the DNS should be as short as possible (the 10-minute default may be acceptable, but this should not be increased).

When an LSF client host is assigned a new IP address, you should wait this long before you use the client host to run LSF. This prevents possible conflicts by allowing time for the DNS to be updated.

## Configuring dynamic IP addressing

By default, dynamic IP addressing is not enabled. To enable dynamic IP addressing for the cluster:

- 1 Configure the following parameter in `lsf.conf`: `LSF_DHCP_ENV=Y`

2 Reconfigure the cluster:

```
lsadmin reconfig  
badmin reconfig
```

LSF checks for any configuration errors. If no fatal errors are found, you are asked to confirm reconfiguration. If fatal errors are found, reconfiguration is aborted.

## Configuring dynamic IP addressing

# Displaying GUIs in LSF with Microsoft Terminal Services

This chapter describes how to configure Microsoft Terminal Services and LSF to work together and how to submit and monitor jobs that display GUIs.

## Contents

- ◆ [How LSF Works with Terminal Services](#) on page 42
- ◆ [Requirements](#) on page 43
- ◆ [Configuring Terminal Services for LSF](#) on page 44
- ◆ [Configuring LSF to run Terminal Services jobs](#) on page 45
- ◆ [Submitting LSF jobs to Terminal Services hosts \(tssub\)](#) on page 46
- ◆ [Limiting the number of Terminal Services jobs on a host](#) on page 47
- ◆ [Submitting LSF jobs to Terminal Services hosts from UNIX](#) on page 48

## How LSF Works with Terminal Services

It is possible to run jobs to display graphical GUIs on remote hosts in LSF by using Microsoft Terminal Services.

### Environment variables

The following environment variables are available for Terminal Services jobs:

#### **LSF\_LOGON\_DESKTOP**

When `LSF_LOGON_DESKTOP=1`, jobs run in interactive foreground sessions. This allows GUIs to be displayed on the execution host. If this parameter is not defined, jobs run in the background.

**LSB\_TSJOB** When the `LSB_TSJOB` variable is defined to any value, it indicates to LSF that the job is a Terminal Services job.

**LSF\_TS\_LOGON\_TIME** Specifies the time to create a Windows Terminal Service session. You should configure `LSF_TS_LOGON_TIME` according to your network environment. The default, 300000 milliseconds is suitable for most environments. For a congested network, set `LSF_TS_LOGON_TIME=1000000`

### Job submission

- 1 Submit the job with `tssub` instead of `bsub`. `tssub` is a wrapper around the `bsub` command which only submits jobs to hosts that have the `msts` resource.
- 2 `tssub` sets the `LSB_TSJOB` and `LSF_LOGON_DESKTOP` environment variables. These variables are then transferred to the execution host.
  - ❖ If the job is dispatched to a host in which Terminal Services is not installed or properly configured, the job is set to the `PEND` state and a pending reason is written in `sbatchd.log.host_name`.
  - ❖ If `tssub -I` is specified, a terminal display is visible on the submission host after the job has been started.
  - ❖ If the job is not a GUI job, LSF runs a command window and output is displayed in the command window when something is written to `stdout`.
  - ❖ Pre- and post-execution commands are executed within the terminal session. The job does not complete until post-execution commands complete.
- 3 View job output with the command `tspeek`. If the terminal window is closed, the job remains running. You can reconnect to view the job with `tspeek`.

### Limitations

- ◆ A job submitted as a Terminal Services job cannot be modified to become a non-Terminal Services job with `bmod`
- ◆ The `bsub` option `-o out_file` is not supported for `tssub`
- ◆ Only Windows `bsub` options are supported for `tssub`. For example, you cannot use the options `-Ip`, `-Is`, `-L login_shell` of `bsub` with `tssub`.
- ◆ If user mapping is defined, the user who invokes `tspeek` must have the required privileges to access the session
- ◆ MultiCluster is not supported

## Requirements

- ◆ All submission hosts must have the Terminal Services Full Client Windows Installer (MSI) package installed. This package contains the required Microsoft Terminal Services Advanced Client ActiveX Control. Download it from the Microsoft web site.
- ◆ All execution server hosts must have the Terminal Services and Terminal Services Full Client Windows Installer (MSI) packages installed.
- ◆ Terminal Services server hosts can be in a different domain from submission hosts as long as the user can be authenticated by both domains.
- ◆ The service account on Terminal Services server hosts must be a member of the local administrators group.
- ◆ Your LSF cluster must be working properly.
- ◆ All submission (client) hosts need the ActiveX control, but you don't necessarily have to use the specified MSI, it can also be there if Remote Desktop Connection software is installed (part of the OS installation).
- ◆ To run applications from a Terminal Services client, the FEATURE lines in your LSF license must contain the TS\_OK keyword. FLEXlm detects when a node-locked uncounted license is running under Windows Terminal Server. Without TS\_OK, a user running on a Terminal Server client is denied a license. See the *FLEXlm End User Guide* for more information.

## Registering the Microsoft Terminal Services Advanced Client ActiveX Control

For clients running Windows 2000 or earlier with Remote Desktop Connection, the necessary ActiveX control is installed, but it is not registered by default. The LSF `tssub` command attempts to register the control automatically, but that requires administrator privileges. If you are not an administrator, and the control is not yet registered, `tssub` returns the following error:

```
checkTSAC: Failed to initialize Terminal Services Client. Check that
it is installed and registered: Microsoft Terminal Services Advanced
Client ActiveX Control must be installed and registered.
```

You must be a Local Administrator on the host to register the control. Choose either method:

- ◆ Submit any job with `tssub`. If you have administrator privileges, `tssub` will register the ActiveX control automatically, or
- ◆ Register the ActiveX control manually. Type the following Windows command at an MS-DOS prompt:

```
C:\regsvr32.exe mstscax.dll
```

Once the control is registered on the host, the error does not occur.

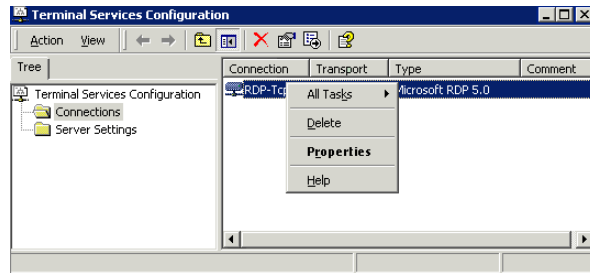
Non-administrator users can run `tssub` with no errors.

The problem does not occur with the Terminal Service Full Client installation, or on newer versions of Windows or Windows XP.

## Configuring Terminal Services for LSF

By default, a Terminal Services connection always prompts for a password. You need to disable this option.

- 1 Start the Microsoft Management Console (MMC) Terminal Services Configuration snap-in (Start | Programs | Administrative Tools | Terminal Services Configuration).



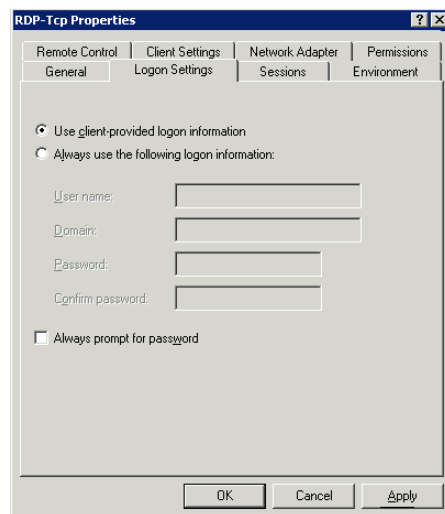
- 2 Right-click the configuration for which you want to disable the default password setting, and select Properties.
- 3 Select the Logon Settings tab.

- a Select Use client provided logon information.

This makes sure you are not using a predefined user to log on to Terminal Services.

- a Clear the Always prompt for password check box.

Future connections will no longer force a password entry.



- 4 Click Apply, then click OK.
- 5 Close the dialog box.

## Configuring LSF to run Terminal Services jobs

### Define the msts static resource

Edit `LSF_CONFDIR\lsf.shared` and define the `msts` static resource. Note that the resource name must be `msts` and values for the resource must be exactly as shown.

```
Begin Resource
RESOURCENAME TYPE INTERVAL INCREASING DESCRIPTION
...
msts Boolean ( ) ( ) (Windows Terminal Server)
...
End Resource
```

### Add the msts resource to hosts

Edit `LSF_CONFDIR\lsf.cluster.cluster_name` and add the `msts` resource to each host on which Terminal Server is installed. For example:

```
Begin Host
HOSTNAME model type server rlm pg tmp RESOURCES
...
hostA ! NTX86 1 - - - (msts)
...
End Host
```

### Create job starters to preserve a user's environment settings (optional)

You may need to create a job starter script to preserve a user's environment settings on the execution host. For details, refer to *Administering Platform LSF*.

## Submitting LSF jobs to Terminal Services hosts (tssub)

Terminal Services Full Client Windows Installer (MSI) package must be installed on submission hosts.

### Submit a job with tssub

```
tssub [bsub_options]
```

The `bsub` option `-o out_file` is not supported for `tssub`.

Only Windows `bsub` options are supported for `tssub`. For example, you cannot use the options `-Ip`, `-Is`, `-L login_shell` of `bsub` with `tssub`.

Submit a job to a host with Terminal Services installed by using the `tssub` command. For example:

```
tssub myjob
```

### View job output with tspeek

You can use `tspeek` from any LSF Windows to view the output of a Terminal Services job:

```
tspeek job_ID
```

You can also use `tspeek` to monitor job output from a Linux host with `rdesktop` installed. You cannot use `tspeek` to monitor job output from UNIX.

For example, if your job ID is 23245:

```
tspeek 23245
```

### Monitor the job with bjobs

If you use `bjobs -l` to monitor the job, you will see a message such as:

```
External Message 2 was posted from LSF\lsfadmin to message box 2
```

The body of the message contains the ID of the terminal session that was created.

## Limiting the number of Terminal Services jobs on a host

The `msts` resource indicates to LSF whether an execution host has Terminal Server installed or not.

To limit the number of Terminal Services jobs that run on a host and keep track of how many jobs are running, define a numeric resource in addition to the `msts` boolean resource. You can alternatively use an `elim` to report how many terminal servers are available for each host.

### Configure a numeric Terminal Server resource

Define the resource in `LSF_CONFDIR\lsf.shared`. For example:

```
Begin Resource
RESOURCENAME TYPE INTERVAL INCREASING DESCRIPTION
...
term_server Numeric 60 N (Terminal Server)
...
End Resource
```

### Submit a job with rusage

When submitting a job, use the `rusage` resource requirement string:

```
% tssub -R"rusage[term_server=1]" myjob
```

## Submitting LSF jobs to Terminal Services hosts from UNIX

In mixed cluster environments, it is possible to submit a Terminal Services job with `bsub` from a UNIX host. You can use `tspeek` to monitor job output from a Linux host with `rdesktop` installed. You cannot use `tspeek` to monitor job output from UNIX.

### Submit a Terminal Services job from UNIX

- 1 On the UNIX submission host, define the environment variables `LSF_LOGON_DESKTOP=1` and `LSB_TSJOB=1`.
  - ❖ When `LSF_LOGON_DESKTOP=1`, it allows GUIs to be displayed on the execution host.
  - ❖ When the `LSB_TSJOB` variable is defined to any value, it indicates the job is a Terminal Services job.
- 2 Submit the job with `bsub` and indicate the `msts` resource requirement. For example:

```
% bsub -R"msts" myjob
```

# Installing LSF in a Mixed Cluster

This chapter describes how to set up LSF in a mixed environment.

## Contents

- ◆ [Setting up a Linux cluster with Windows compute nodes](#) on page 49.
- ◆ [Setting up a Windows cluster with Linux compute nodes](#) on page 53.

## Setting up a Linux cluster with Windows compute nodes

Complete the following steps to set up a Linux cluster with Windows compute nodes.

- 1 [Install the Linux cluster on a shared file system](#) on page 49.
- 2 [Configure the Linux cluster](#) on page 50.
- 3 [Restart the LSF cluster](#) on page 51.
- 4 [Install the Windows compute node](#) on page 52.
- 5 [Start the Windows compute node](#) on page 52.

## Install the Linux cluster on a shared file system

- 1 Install the Linux cluster as described in the Linux install guide, with exceptions to allow for Windows compute nodes.

Edit `install.conf` and specify the following:

- a Enable dynamic hosts.

Enable or add the following line:

```
ENABLE_DYNAMIC_HOSTS=Y
```

- b Optional. Allow EGO to control the LSF daemons.

Enable or add the following line:

```
EGO_DAEMON_CONTROL=Y
```

- c Specify the cluster administrator.

`LSF_ADMIN=user_account`

For example:

`LSF_ADMIN=lsfadmin`

- d Specify the installation directory.

`LSF_TOP=directory`

For example:

`LSF_TOP=$SHARE/LSF_7.0`

## Configure the Linux cluster

- 1 If you allowed EGO to control the LSF daemons, add Windows compute node information to the LSF service configuration files.

- a Edit `LSF_TOP/conf/ego/cluster_name/eservice/esc/conf/services/res.xml`

- b Navigate to the section with the correct Windows host type.

For 64-bit compute hosts, navigate to NTX64. For IA hosts, navigate to NTIA64. For other Windows compute hosts, navigate to NTX86.

- c Add the proper Windows compute node information for the `Command` and `ExecutionUser` tags.

For example:

```
<sc:ActivityDescription>
  <ego:Attribute name="hostType" type="xsd:string">NTX86</ego:Attribute>
  <ego:ActivitySpecification>
    <ego:Command>C:\LSF_7.0\7.0\etc\res.exe -3</ego:Command>
    <ego:ExecutionUser>LSF\lsfadmin</ego:ExecutionUser>
    <ego:EnvironmentVariable name="LSF_ENVDIR">c:\LSF_7.0\
conf</ego:EnvironmentVariable>
  ...
```

- d Edit

`LSF_TOP/conf/ego/cluster_name/eservice/esc/conf/services/sbatchd.xml`

- e Navigate to the section with the correct Windows host type.

For 64-bit compute hosts, navigate to NTX64. For IA hosts, navigate to NTIA64. For other Windows compute hosts, navigate to NTX86.

- f Add the proper Windows compute node information for the `Command` and `ExecutionUser` tags.

For example:

```
<sc:ActivityDescription>
  <ego:Attribute name="hostType" type="xsd:string">NTX86</ego:Attribute>
  <ego:ActivitySpecification>
    <ego:Command>C:\LSF_7.0\7.0\etc\sbatchd.exe -3</ego:Command>
```

```
<ego:ExecutionUser>LSF\lsfadmin</ego:ExecutionUser>
```

...

- 2 Add the Windows cluster administrator account to the your cluster file.
  - a Edit `LSF_CONFDIR/lsf.cluster.cluster_name`
  - b In the `ClusterAdmins` section, add `LSF\lsfadmin` to the Administrators list.

For example:

```
Begin ClusterAdmins
Administrators = lsfadmin LSF\lsfadmin
End ClusterAdmins
```

- 3 Add the LSF user domain to the `lsf.conf` file.

- a Edit `LSF_TOP/LSF7.0/conf/lsf.conf`
- b Add the LSF user domain.

```
LSF_USER_DOMAIN=lsf_user_domain
```

For example:

```
LSF_USER_DOMAIN=LSF
```

- 4 Register the Windows execution password to your Linux cluster.

- a Log on to any host in the cluster as `egoadmin`.
- b Log on to EGO as the cluster administrator.

For example:

```
egosh user logon -u Admin -x mypasswd
```

- c Register the password of the Windows user account.

```
egosh ego execpasswd -u domain\admin -x password
```

For example:

```
egosh ego execpasswd -u LSF\lsfadmin -x lsfpasswd
```

The password must be 31 characters or less.

---

**IMPORTANT:** If you run `egosh ego execpasswd` from a UNIX or Linux host, you must run with the `-noverify` option. Only a Windows host can verify the password for a Windows user. For example:

```
egosh ego execpasswd -u LSF\lsfadmin -x password -noverify
```

---

## Restart the LSF cluster

- 1 Restart the LSF cluster.

```
lsfstartup
```

---

## Install the Windows compute node

- 1 Install the Windows compute node as described in the Windows install guide, with exceptions to be part of a Linux cluster.

Specify the following options during installation:

- a Specify the Linux master host as your master host name.

```
Master_Name=linux_master_name
```

- b If you allowed EGO to control the LSF daemons in your cluster master host, allow EGO to control the LSF daemons in your compute node.

```
EGO_DAEMON_CONTROL=Y
```

- c Specify the same port number as that of the Linux master host.

```
Port_Number=base_port_number
```

- d Specify the cluster administrator to be the same as on your master host.

```
LSF_ADMIN=domain\user_account
```

For example:

```
LSF_ADMIN=LSF\lsfadmin
```

- e Specify the installation directory.

```
LSF_TOP=directory
```

For example:

```
LSF_TOP=C:\LSF_7.0
```

- 2 Register the Windows execution user password to your cluster password file.

```
lspasswd -u domain\admin -p password
```

For example:

```
lspasswd -u LSF\lsfadmin -p lsfpasswd
```

The password must be 31 characters or less.

## Start the Windows compute node

- 1 Start the Windows compute node

```
lsadmin limstartup
```

- 2 If you did not allow EGO to control the LSF daemons, manually start the LSF services.

```
lsfadmin resstartup
```

```
badmin hstartup
```

## Setting up a Windows cluster with Linux compute nodes

Complete the following steps to set up a Windows cluster with Linux compute nodes.

- 1 Install the Windows cluster master host as described in the Windows installation guide and allow EGO to control the LSF daemons.

Specify the following option:

```
EGO_DAEMON_CONTROL=Y
```

- 2 Install the Linux compute node as described in the Linux installation guide and allow EGO to control the LSF daemons.

Edit `install.conf` and add the following line:

```
EGO_DAEMON_CONTROL=Y
```

- 3 Add the LSF user domain to the `lsf.conf` file.

- a Edit `C:\LSF7.0\conf\lsf.conf`

- b Add the LSF user domain.

```
LSF_USER_DOMAIN=lsf_user_domain
```

For example:

```
LSF_USER_DOMAIN=LSF
```

- 4 Restart the Windows cluster.

```
egosh ego restart
```

Setting up a Windows cluster with Linux compute nodes

---

# Index

## A

augmentstarter job starter 32

## C

cluster

checking 7

starting 8

verifying configuration 7

COMPSPEC environment variable 32

COMPUTERNAME environment variable 32

cross-platform user mapping 25

## D

default user mapping 25

lspasswd command 28

wgpasswd command 28

DHCP and LSF 37

DNS (Domain Name System) 37

Domain Name System 37

domain user accounts, multiple 26

drive letter format (UNC) 32

Dynamic Host Configuration Protocol 37

## E

environment variables 31

LSF\_LOGON\_DESKTOP 42

LSF\_TS\_LOGON\_TIME 42

Terminal Services jobs 42

environment, job execution 31

external load indices, configuring in LSF Monitor 36

## F

files, license.dat, location 19

FLEXlm

installs.exe 22

license manager service 22

## H

host names, displaying master 8

hosts

displaying

bhosts command 13

lshosts command 9

displaying configuration 9

dynamic IP addressing 37

## I

installation

LSF Monitor 34

testing 5

installs.exe 22

intervals, configuring in LSF Monitor 36

## J

job execution environment 31

job starters

augmentstarter 32

preservestarter 32

jobs

displaying 14

displaying GUIs with Terminal Services 41

submitting a test job 14

## L

license file (license.dat), TS\_OK keyword 43

license manager service 22

license server

checking 6

checking status (lmstat) 23

displaying status (lmstat) 6, 23

license.dat file

description 19

location 19

TS\_OK keyword 43

licenses

license.dat file 19

Terminal Services 43

TS\_OK keyword 43

LIM (Load Information Manager), checking 8

lmstat command 6

displaying server status 6

using 23

load information, charting with Windows Performance Monitor 33

Load Information Manager (LIM), checking 8

lsadmin command 7

lsclusters command 10

LSF configuration, checking 7

LSF hosts, displaying 9

LSF license, FLEXlm license manager service 22

LSF license file (license.dat)

description 19

location 19

## Index

- LSF Monitor
  - configuring 35
    - external load indices 36
    - sample intervals 36
  - installing 34
  - installing and uninstalling 36
  - starting and stopping 36
  - statistics 36
- LSF products, adding 21
- LSF services, starting 8
- lsf.conf file, LSF\_BINDIR 31
- lsf.conf, LSF\_DHCP\_ENV 38
- LSF\_BINDIR 31
- LSF\_DHCP\_ENV in lsf.conf 38
- LSF\_LOGON\_DESKTOP, variable 42
- LSF\_TS\_LOGON\_TIME, variable 42
- lsfmon 36
- lsgrun command 10
- lshosts command 9
- lsid command 8
- lsinfo command 8
- lsload command 10
- lspasswd command
  - checking RES 10
  - default user mapping 28
  - mixed cluster configuration 52
  - registering user names and passwords 28
  - syntax for user names 29
- lsrun command 10
- M**
- master host, displaying 8
- mixed cluster
  - monitoring Windows Terminal Services jobs (tspeek) 48
  - starting 8
  - submitting Windows Terminal Services jobs 48
  - testing 5
  - user mapping 25
- mixed cluster configuration, lspasswd command 52
- multiple domains 26
- N**
- NTRESKIT environment variable 32
- O**
- OS2LIBPATH environment variable 32
- P**
- passwords, registering with lspasswd 28
- PATH environment variable 31
- permanent license
  - checking 6
  - displaying server status (lmstat) 6
- permanent LSF license, displaying server status (lmstat) 23
- preservestarter job starter 32
- PROCESSOR\_ARCHITECTURE environment variable 32
- PROCESSOR\_LEVEL environment variable 32
- Q**
- queues, displaying 14
- R**
- Remote Execution Server (RES), checking 10
- RES (Remote Execution Server), checking 10
- S**
- sample intervals, configuring in LSF Monitor 36
- servers, WINS, DNS, DHCP 38
- services, starting LSF 8
- statistics in LSF Monitor 36
- syntax of user names 27
- SYSTEMDRIVE environment variable 32
- SYSTEMROOT environment variable 32
- T**
- Terminal Services client, TS\_OK keyword 43
- Terminal Services jobs, environment variables 42
- Terminal Services, displaying GUIs with 41
- test job, submitting 14
- TS\_OK keyword 43
- tspeek command
  - description 46
  - on Linux 48
- U**
- UNC file path format 32
- UNIX compatibility 31
- user mapping 25
- user name syntax
  - LSF commands 29
  - LSF files 29
- user names, registering with lspasswd 28
- users
  - specifying 27
  - viewing 27
- W**
- wgpaswd command
  - default user mapping 28
  - registering user names and passwords 28
- WINDIR environment variable 32
- Windows Internet Naming Service (WINS) 37
- Windows Performance Monitor
  - charting LSF load information 33
  - configuring LSF Monitor 35
  - installing and uninstalling LSF Monitor 36
  - installing LSF Monitor 34
  - lsfmon 36
- WINS (Windows Internet Naming Service) 37