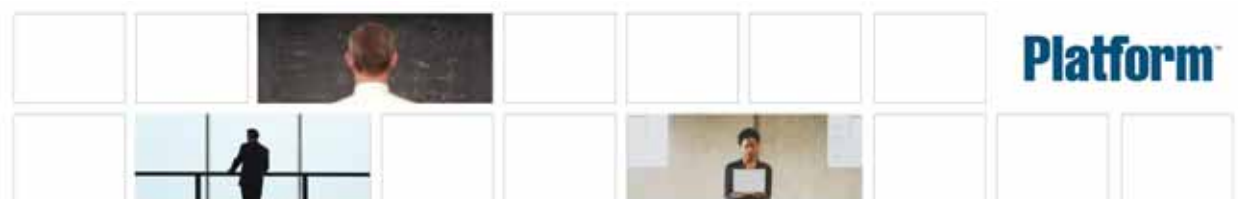

Release Notes for Platform™ LSF™ Version 7

Release date: December 2006
Last modified: January 31, 2008
Comments to: doc@platform.com
Support: support@platform.com

Contents

- ◆ [What's New in Platform LSF Version 7](#) on page 2
- ◆ [Upgrade and Compatibility Notes](#) on page 5
- ◆ [What's Changed in Platform LSF Version 7](#) on page 9
- ◆ [Known Issues](#) on page 33
- ◆ [Download the Platform LSF Version 7 Distribution Packages](#) on page 36
- ◆ [Install Platform LSF Version 7](#) on page 38
- ◆ [Learn About Platform LSF Version 7](#) on page 40
- ◆ [Get Technical Support](#) on page 41



What's New in Platform LSF Version 7

- ◆ Performance, scalability, reliability, usability enhancements
- ◆ LSF on Platform EGO
- ◆ Scheduling enhancements
- ◆ Windows enhancements
- ◆ LSF reports built on EGO
- ◆ Miscellaneous features

For more information

For more details about what's new in Platform LSF Version 7, visit the Platform Computing Web site to see *Features, Benefits & What's New*.

Performance, scalability, reliability, usability enhancements

Support for high job submission rates—LSF now supports higher job submission rates for clusters that include 5000 dedicated hosts (10K dedicated processors/Slots, 20K cores for dual-core processors):

- ◆ Sustainable submission/query rate of 20 jobs per second, with a peak submission rate of 100 jobs per second when multiple users issue the `bsub` command concurrently
- ◆ Five jobs per second for one user, provided that `mbatchd` is not switching an event file at the same time and no external submission script (`esub`) is running
- ◆ Minimum of 90% utilization given an average job runtime of 15 minutes
- ◆ 500K concurrent jobs in the system at any given time
- ◆ 10 million completed jobs per day
- ◆ Support for 8,192-way parallel jobs

Faster time for reconfiguration and failover—You can now configure LSF to obtain host status more quickly, which allows LSF to reschedule jobs within a shorter time.

- ◆ Faster detection of failed and hung execution nodes
- ◆ Reconfiguration and failover within five minutes

Optional EGO management of LSF daemons—including parallel/asynchronous startup/shutdown.

Support for IPv6 address formats—IP addresses can have either a dotted quad notation (IPv4) or IP Next Generation (IPv6) format. LSF supports both formats in mixed IPv4/IPv6 clusters.

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.

- ◆ *Scalability*—EGO enhances LSF scalability. Currently, the LSF scheduler has to deal with a large number of jobs. EGO provides management functionality for multiple schedulers that co-exist in one EGO environment. In LSF Version 7,

although only a single instance of LSF is available on EGO, the foundation is established for greater scalability in follow-on releases that will allow multiple instances of LSF on EGO.

- ◆ *Robustness*—Today, LSF functions as both scheduler and resource manager. EGO decouples these functions, making the entire system more robust. EGO reduces/eliminates the downtime for LSF users while resources are added or removed.
- ◆ *Reliability*—In situations where service is degraded due to noncritical failures such as `sbatchd` or `RES`, by default, LSF does not automatically restart the daemons. The EGO Service Controller can monitor all LSF daemons and automatically restart them if they fail. Similarly, the EGO Service Controller can also monitor and restart other critical processes such as `FlexLM` and `lmgd`.
- ◆ *Additional scheduling functionality*—EGO provides the foundation for EGO-enabled SLA scheduling, which provides LSF with additional and important scheduling functionality.
- ◆ *Centralized management and administration framework.*
- ◆ *Single reporting framework*—across various application heads built around EGO.

Scheduling enhancements

- ◆ *Support for multiple resource requirement strings (-R) options*—Enables administrators to more easily change and add resource requirements and to simplify the use of scripts for job submission.
- ◆ *Multiple first execution host candidates*—You can specify a list of mandatory first host candidates during job submission and at the queue level.
- ◆ *Job submission using JSDL files*—The Job Submission Description Language (JSDL) provides a convenient format for storing descriptions of job requirements. You can save a set of job requirements in a JSDL XML file, and then reuse that file as needed to submit jobs to LSF. The JSDL file can define
 - ❖ Applications that will execute
 - ❖ Sets of resources to be used
 - ❖ Input and output files
 - ❖ Associations between applications, resources, and data sets
- ◆ *EGO-enabled SLA scheduling*—LSF service classes configured in `lsb.serviceclasses` can now be attached to job groups and EGO consumers. Attaching service classes to EGO consumers provides
 - ❖ Multi-level fairshare and greater control of resource allocation, including maximum/minimum goals
 - ❖ Lending and borrowing of cluster resources between consumers, including immediate preemption for resource owners
- ◆ *Application encapsulation*—Centralized definitions for application-specific attributes, including
 - ❖ Pre/post exec and job starter
 - ❖ Job controls

- ❖ Process and processor limits
- ❖ Support for runtime estimates, in addition to runtime limits, for more accurate scheduling. You can now define a runtime estimate that LSF uses for scheduling purposes only. LSF does not kill jobs that exceed the estimate, unless the jobs also exceed a defined run limit.
- ❖ Indication of whether a job is rerunnable
- ❖ Requeue exit values
- ❖ Default resource requirements
- ❖ Control of job chunking based on the application profile

Windows enhancements

- ◆ *Windows installer enhancements*—The new, MSI-based installer does not require a shared file system between Windows and Linux/UNIX.
- ◆ *Decoupling of Windows Services from the LSF cluster definition*—Simplifies movement of hosts between clusters.
- ◆ *Revalidation of Windows credentials during job execution*
- ◆ *Multiple domains on a mixed UNIX-Windows cluster*—LSF now supports jobs submitted from a UNIX host in a multiple-domain Windows environment.
- ◆ *LSF integration with Microsoft CCS*

LSF reports built on EGO

- ◆ *Central database for event and account data*—All `lsb.events` and `lsb.acct` data is captured to a central database using an open published schema, allowing users to utilize their own reporting frameworks for data presentation.
- ◆ *Reporting includes basic SQL query and report samples*
- ◆ *Basic data collection capability*

Miscellaneous features

- ◆ *More robust command execution*—Commands will execute successfully even when the slave LIM is down or just restarting.
- ◆ *Multiple first execution host candidates*—You can now define more than one first execution host candidate for parallel jobs. This improves performance and reduces the number of pending jobs. The scheduler selects a first execution host from among the eligible candidates, and no longer depends on the availability of a single first execution host.
- ◆ *Estimated job runtime*—New application encapsulation feature allows administrators to define a runtime for scheduling purposes.
- ◆ *LSF Make refresh*—Upgraded LSF Make based on GNU Make 3.81 build.
- ◆ *Support for UTMP on Linux, HP-UX, and Solaris*
- ◆ *Improved mbatchd performance during event switching*—For large clusters, you can configure `mbatchd` to fork a child process that handles event switching, thereby reducing the load on `mbatchd`.

Upgrade and Compatibility Notes

- ◆ [Server host compatibility Platform LSF](#)
- ◆ [Upgrade LSF on UNIX and Linux](#)
- ◆ [Migrate LSF on Windows](#)
- ◆ [Maintenance pack and enhancement update availability](#)
- ◆ [System requirements](#)
- ◆ [API compatibility](#)
- ◆ [Automatic parameter migration during upgrade](#)
- ◆ [Multiple cluster configuration](#)

Server host compatibility Platform LSF

IMPORTANT: To use new features introduced in Platform LSF Version 7, you *must* upgrade all hosts in your cluster to LSF 7.

LSF 6.x and 5.x servers are compatible with LSF Version 7 master hosts. All LSF 6.x and 5.x features are supported by Version 7 master hosts.

LSF system support for IPv6

Platform LSF Version 7 is now built with IPv6 support. The following operating systems support IPv4 only:

- ◆ Sun Solaris 2
- ◆ Sun Solaris 7, 8, 9
- ◆ HP-UX 11.0
- ◆ Microsoft Windows 2000 without Service Pack 1 or later

Upgrading HP-UX 11.11 hosts

Platform LSF Version 7 is now built with IPv6 support and requires the following patch files with IPv6 support for HP-UX 11i v1.0 (11.11 - via TOUR) before upgrading to LSF 7:

- ◆ libc patch - PHCO_24400
- ◆ libc header file patch - PHCO_24402
- ◆ libnss_files patch - PHCO_24401
- ◆ libnss_dns patch - PHNE_24129

Upgrade LSF on UNIX and Linux

Run `lsfinstall` to upgrade to LSF Version 7 from an earlier version of LSF on UNIX and Linux. Follow the steps in *Upgrading Platform LSF on UNIX and Linux*.

Migrate LSF on Windows

To migrate an LSF on Windows to LSF Version 7 from an earlier version of LSF on Windows, follow the steps in “Migrate Your Windows Cluster to Platform LSF Version 7” (`lsf_migrate_windows.pdf`).

Maintenance pack and enhancement update availability

At release, Platform LSF Version 7 includes all bug fixes and solutions up to and including the all the bug fixes before February 5, 2007. Fixes after February 2007 will be included in the next LSF enhancement update.

Fixes in the November 2006 Maintenance Pack are included in the March 2007 enhancement update.

As of February 2007, monthly maintenance packs are no longer distributed for LSF Version 7.

System requirements

See the Platform Computing Web site for information about supported operating systems and system requirements for the Platform LSF family of products:

- ◆ Platform LSF
- ◆ Platform LSF License Scheduler

API compatibility

Full backward compatibility: your applications will run under LSF Version 7 without changing any code.

The Platform LSF Version 7 API is fully compatible with the LSF Version 6.x. and 5.x APIs. An application linked with the LSF Version 6.x or 5.x libraries will run under LSF Version 7 without relinking.

To take full advantage of new Platform LSF Version 7 features, including job submission using JSDL and IPv6 address formats, you should recompile your existing LSF applications with LSF Version 7.

New and changed LSF APIs

See the *LSF API Reference* for more information.

The following new APIs have been added for LSF Version 7:

- ◆ `ls_getmyhostname2()`

Automatic parameter migration during upgrade

Since LIM now belongs to EGO, some existing LSF parameters have corresponding EGO parameters name in `ego.conf` (LSF_CONFDIR/`lsf.conf` is a separate file from EGO_CONFDIR/`ego.conf`).

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

If any of the following LSF parameters are already defined in `lsf.conf`, they are automatically copied during upgrade to the corresponding EGO parameters in `ego.conf`. The original LSF settings are maintained for backward compatibility:

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

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

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`. You can keep your existing LSF parameters in `lsf.conf`.

You cannot set LSF parameters (parameter names beginning with `LSF_` or `LSB_`) in `ego.conf`, and you cannot set EGO parameters (parameter names beginning with `EGO_`) in `lsf.conf`.

NOTE: A parameter in `lsf.conf` does not necessarily have exactly the same behavior, 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 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 `ego.conf` default takes effect.

If a parameter is not yet set in `lsf.conf` and there is a corresponding parameter in `ego.conf`, you should set the corresponding EGO parameter in `ego.conf` instead setting the LSF parameter in `lsf.conf`.

LSF 6.2 hosts in your cluster can *only* read `lsf.conf`, so you *must* set LSF parameters only in `lsf.conf`, or make sure that the values are the same in both `lsf.conf` and `ego.conf`.

Multiple cluster configuration

In Platform LSF Version 7, multiple independent clusters can no longer share the same configuration directory. You must install each LSF cluster in a unique location.

What's Changed in Platform LSF Version 7

- ◆ Changed behavior
- ◆ LSF daemon management
- ◆ Directory structure changes
- ◆ New and changed configuration parameters and environment variables
- ◆ New and changed commands, options, and output
- ◆ New and changed files
- ◆ New and changed accounting and job event fields
- ◆ Bugs fixed since December 2006

Changed behavior

Batch command messages

LSF displays new error messages when a batch command cannot communicate with `mbatchd`. You can customize three of these messages in order to provide LSF users with more detailed information and instructions. `bhosts` and `bjobs` might display different messages when `mbatchd` is down and the `LSB_QUERY_PORT` is busy.

The following table lists the parameters in `lsf.conf` you can use to customize messages when a batch command does not receive a response from `mbatchd`. For backwards compatibility, you can use these parameters to set the message to the batch daemon not responding...still trying. message text used in previous versions of LSF.

Reason for no response from <code>mbatchd</code>	Default message	Parameter used to customize the message
<code>mbatchd</code> is too busy to accept new connections or respond to client requests	LSF is processing your request. Please wait...	<code>LSB_MBD_BUSY_MSG</code>
internal system connections to <code>mbatchd</code> fail	Cannot connect to LSF. Please wait...	<code>LSB_MBD_CONNECT_FAIL_MSG</code>
<code>mbatchd</code> is down or there is no process listening at either the <code>LSB_MBD_PORT</code> or the <code>LSB_QUERY_PORT</code>	LSF is down. Please wait...	<code>LSB_MBD_DOWN_MSG</code>

Dynamic host management

Dynamic hosts remain in the cluster unless you manually remove them from `$EGO_TOP/kernel/work/lim/hostcache`.

Only the cluster administrator can modify the `hostcache` file.

LSF License Scheduler

- ◆ `LM_REMOVE_INTERVAL=seconds` is now supported in the Features section of `lsf.licensescheduler` to specify the minimum time a job must have a license checked out before `lmremove` can remove the license. `lmremove` causes `lmgrd` and vendor daemons to close the TCP connection with the application. They will then retry the license checkout. Each feature definition can specify a different value for this parameter. The value specified for a feature overrides the global value defined in the Parameters section.

- ◆ Use `BLC_HEARTBEAT_FACTOR` in the Parameters section of `lsf.licensescheduler` to enable `bld` to detect `blcollect` failure. Define the number of times that `bld` receives no response from a license collector daemon (`blcollect`) before `bld` resets the values for that collector to zero. Each license usage reported to `bld` by the collector is treated as a heartbeat. The default is 3.
- ◆ Use `LM_STAT_INTERVAL=seconds` in the ServiceDomain section of `lsf.licensescheduler` to define a time interval between calls that License Scheduler makes to collect license usage information from FLEXlm license management. The value specified for a service domain overrides the global value defined in the Parameters section. Each service domain definition can specify a different value for this parameter.
- ◆ Use `ENABLE_MINJOB_PREEMPTION=Y` in the Feature section of `lsf.licensescheduler` to minimize the overall number of jobs that License Scheduler preempts. When `ENABLE_MINJOB_PREEMPTION` is set, License Scheduler preempts the minimum number of jobs needed to obtain the required licenses. For example, for a job that requires 10 licenses, License Scheduler preempts a single parallel job that uses 10 or more licenses rather than 10 jobs that use one license.
- ◆ New and changed License Scheduler commands:
 - ❖ `bladmin chkconfig`—Checks LSF License Scheduler configuration in `lsf.licensescheduler` and `lsf.conf`
 - ❖ `blparams`—Displays information about configurable LSF License Scheduler parameters defined in `lsf.conf` and `lsf.licensescheduler`
 - ❖ `blplugins`—Displays plugin activity and the check-in, check-out, and deny counters as seen by the License Scheduler for each feature and service domain

Directory format for Windows directories

Windows does not support a mapped drive as input during the installation. When you must specify a directory, use UNC path.

Post-execution on SGI cpusets

Post-execution processing on SGI cpusets behave differently from previous releases. If `JOB_INCLUDE_POSTPROC=Y` is specified in `lsb.applications`, post-execution processing is not attached to the job cpuset, and Platform LSF does not release the cpuset until post-execution processing has finished.

Banded licensing

The memory limit for S-Class licenses on X86/AMD64/EM64T processors has increased from 8 GB to 16 GB. The other classes of licenses have not changed.

You can use permanent licenses with restrictions in operating system and hardware configurations. These banded licenses have three classes, with the E-class licenses having no restrictions.

Banded licenses now support the following operating systems and hardware configurations:

License type	Supported operating systems	Processor	Physical memory	Physical processors/sockets
B-Class	Linux, Windows, MacOS	Intel X86/AMD64/EM64T	Up to and including 4 GB physical memory on a node	Up to and including 2 processors
S-Class	Linux, Windows, MacOS	Intel X86/AMD64/EM64T	Up to and including 16 GB physical memory on a node	Up to and including 4 processors
E-Class	Linux, Windows, MacOS	Intel X86/AMD64/EM64T	More than 16 GB physical memory on a node	More than 4 processors
	All other LSF-supported operating systems	Intel X86/AMD64/EM64T	N/A	N/A
	N/A	All other supported processors	N/A	N/A

LSF daemon management

Manage LSF daemons two ways:

- ◆ System management through `rc`, `inittab`, etc.
- ◆ Through Platform EGO Service Controller. If LSF daemons exit unexpectedly, EGO Service Controller automatically restarts and monitors `res` and `sbatchd`.

IMPORTANT: LSF `res` and `sbatchd` do not restart automatically if you run `lsadmin resshutdown` and `badmin hshutdown` to manually shut them down. You must run `lsadmin resstartup` and `badmin hstartup` to restart the daemons after host shutdown.

All LSF commands and tools, including `lsadmin` and `badmin` are available under both management models.

Directory structure changes

The installation directory structure has changed. See *Installing Platform LSF on UNIX and Linux* for the details of the new structure. Depending on which products you have installed and platforms you have selected, your directory structure may vary.

New and changed configuration parameters and environment variables

The following new configuration parameters and environment variables are new or changed for LSF Version 7:

ego.cluster

- ◆ EGO_HOST_ADDR_RANGE supports the use of IPv6 addresses in addition to IPv4 addresses.

ego.conf

- ◆ EGO_DHCP_ENV—If defined, enables dynamic IP addressing for all hosts in the cluster.
- ◆ EGO_DUALSTACK_PREFER_IPV6—Define this parameter when you want to ensure that clients and servers on dual-stack hosts use IPv6 addresses only. Setting this parameter configures Platform EGO to sort the dynamically created address lookup list in order of AF_INET6 (IPv6) elements first, followed by AF_INET (IPv4) elements, and then others.

RESTRICTION: IPv4-only and IPv6-only hosts cannot belong to the same cluster.

- ◆ EGO_ENABLE_SUPPORT_IPV6—If set, enables the use of IPv6 addresses in addition to IPv4.
- ◆ EGO_HOST_CACHE_DISABLE=Y—The LSF and EGO hosts file will not be processed at startup.
- ◆ EGO_HOST_CACHE_NTTL—Negative-time-to-live value in seconds. Specifies the length of time the system caches a failed DNS lookup result. If you set this value to zero (0), LSF does not cache the result.
- ◆ EGO_HOST_CACHE_PTTL—Positive-time-to-live value in seconds. Specifies the length of time the system caches a successful DNS lookup result. If you set this value to zero (0), LSF does not cache the result.
- ◆ EGO_STRIP_DOMAIN—If all of the hosts in the cluster can be reached using short host names, you can configure Platform EGO to use the short host names by specifying the portion of the domain name to remove. If your hosts are in more than one domain or have more than one domain name, you can specify more than one domain suffix to remove, separated by a colon (:).

hosts file

- ◆ Supports the use of IPv6 addresses in addition to IPv4.
- ◆ Must be located in EGO_CONFDIR. For backwards compatibility, you should create a symbolic link from LSF_CONFDIR/hosts pointing to EGO_CONFDIR/hosts. If `lsfinstall` detects an LSF_CONFDIR/hosts file during upgrade, it copies it to EGO_CONFDIR/hosts and creates a symbolic link back to LSF_CONFDIR/hosts.

install.config

- ◆ DERBY_DB_HOST="*host_name*"—Platform LSF reporting database host. This parameter takes effect when you install the Platform Management Console (PMC) package for the first time, and is ignored for all other cases.
- ◆ EGO_TOP="*/path*"—Full path to the top-level installation directory. The path to EGO_TOP must be shared and accessible to all hosts in the cluster. It cannot be the root directory (/). The file system containing EGO_TOP must have enough disk space for all host types (approximately 200 MB per host type).

- ◆ `EGO_DAEMON_CONTROL="Y" | "N"`—Enables EGO to control LSF `res` and `sbatchd`. Set the value to "Y" if you want EGO Service Controller to start `res` and `sbatchd`, and restart if they fail. The default is `EGO_DAEMON_CONTROL="N"` (`res` and `sbatchd` are started manually or through operating system `rc`, `inittab`, etc.)
- ◆ `ENABLE_DYNAMIC_HOSTS="Y" | "N"`—Enables dynamically adding and removing hosts. Set the value to "Y" if you want to allow dynamically added hosts,
- ◆ Installation parameters for backwards compatibility—not recommended for new installations: `UNIFORM_DIRECTORY_PATH="path"` and `UNIFORM_DIRECTORY_PATH_EGO="path"`, where *path* is a local directory for the root of the path to the machine-dependent LSF and EGO files. This option is ignored during upgrade. Uniform directory path is maintained for backwards compatibility with LSF. It is not recommended for a new installation. The path must be an absolute path to a local directory and not shared. It cannot be the root directory (/). `UNIFORM_DIRECTORY_PATH` and `UNIFORM_DIRECTORY_PATH_EGO` both MUST be enabled or disabled together. Enabling one parameter while leaving the other disabled results in an error. By default, uniform directory path is not used.
- ◆ If `UNIFORM_DIRECTORY_PATH` and `UNIFORM_DIRECTORY_PATH_EGO` are enabled, you must manually configure the `EGO_CONFDIR` environment variable, or copy `ego.conf` to `/etc` to use EGO commands.

slave.config

- ◆ `EGO_TOP="/path"`—Full path to the top-level installation directory. The path to `EGO_TOP` must be shared and accessible to all hosts in the cluster. It cannot be the root directory (/). The file system containing `EGO_TOP` must have enough disk space for all host types (approximately 200 MB per host type). The default is `$LSF_TOP/ego`
- ◆ `EGO_DAEMON_CONTROL="Y" | "N"`—Enables Platform EGO to control LSF `res` and `sbatchd`. Set the value to "Y" if you want EGO Service Controller to start `res` and `sbatchd`, and restart if they fail. The default is `EGO_DAEMON_CONTROL="N"` (`res` and `sbatchd` are started manually or through operating system `rc`, `inittab`, etc.)

lsb.hosts

- ◆ When EGO-enabled SLA scheduling is configured, all hosts that the SLA will use must be dynamically allocated by EGO. The `lsb.hosts` file must contain a default host line. For example:

```
Begin Host
HOST_NAME MXJ    r1m    pg    ls    tmp    DISPATCH_WINDOW # Keywords
default      !    ( )    ( )    ( )    ( )    ( )    # Example
End Host
```

- ◆ `EXIT_RATE` for a specific host overrides a default `GLOBAL_EXIT_RATE` specified in `lsb.params`.

lsb.modules

- ◆ `schmod_ps`—When configured in `lsb.modules`, `schmod_ps` enables scheduling of EGO-enabled SLA service classes configured in `lsb.serviceclasses`. It enforces the ownership of hosts allocated to an SLA by EGO. Jobs from other service classes cannot run on hosts allocated by EGO.

lsb.params

- ◆ `DEFAULT_SLA_VELOCITY`—The number of slots that the SLA should request for parallel jobs running in the SLA. By default, an EGO-enabled SLA requests slots from EGO based on the number of jobs the SLA needs to run. If the jobs themselves require more than one slot, they will remain pending. To avoid this for parallel jobs, set `DEFAULT_SLA_VELOCITY` to the total number of slots that are expected to be used by parallel jobs.
- ◆ `DEFAULT_APPLICATION`—The name of the default application profile. The application profile must already be defined in `lsb.applications`. When you submit a job to LSF without explicitly specifying an application profile, LSF associates the job with the specified application profile.
- ◆ `ENABLE_DEFAULT_EGO_SLA`—The name of the default service class for EGO-enabled SLA scheduling. If the specified service class does not exist in `lsb.serviceclasses`, LSF creates one with the specified name, velocity of 1, and a time window that is always open. It must be the name of a valid EGO consumer. `ENABLE_DEFAULT_EGO_SLA` is required to turn on EGO-enabled SLA scheduling. All LSF resource management is delegated to Platform EGO, and all LSF hosts are under EGO control. When all jobs running in the default SLA finish, all allocated hosts are released to EGO after the default idle timeout of 120 seconds (configurable by `MAX_HOST_IDLE_TIME` in `lsb.serviceclasses`).
- ◆ `ENABLE_EVENT_STREAM=y | Y`—Used only with event streaming for system performance analysis tools, such as Platform LSF reporting. For new installations the stream feature is enabled by default.
- ◆ `ENABLE_EXIT_RATE_PER_SLOT=Y | N`—Scales the actual exit rate thresholds on a host according to the number of slots on the host. For example, if `EXIT_RATE=2` in `lsb.hosts` or `GLOBAL_EXIT_RATE=2` in `lsb.params`, and the host has 2 job slots, the job exit rate threshold will be 4.
- ◆ `EVENT_STREAM_FILE=directory`—Specifies the directory containing the event data stream file used by system performance analysis tools such as Platform LSF reporting. The default directory is `LSF_TOP/work/cluster_name/logdir/stream`.
- ◆ `EXIT_RATE_TYPE=[JOBEXIT | JOBEXIT_NONLSF] [JOBINIT] [HPCINIT]`—When host exception handling is configured (`EXIT_RATE` in `lsb.hosts` or `GLOBAL_EXIT_RATE` in `lsb.params`), specifies the type of job exit to be handled:
 - ❖ `JOBEXIT`—Job exited after it was dispatched and started running.
 - ❖ `JOBEXIT_NONLSF`—Job exited with exit reasons related to LSF and not related to a host problem (for example, user action or LSF policy). These jobs are not counted in the exit rate calculation for the host.
 - ❖ `JOBINIT`—Job exited during initialization because of an execution environment problem. The job did not actually start running.

- ❖ HPCINIT—Job exited during initialization of a Platform LSF HPC job because of an execution environment problem. The job did not actually start running.
- ◆ GLOBAL_EXIT_RATE=*number*
Specifies a cluster-wide threshold for exited jobs. If EXIT_RATE is not specified for the host in `lsb.hosts`, GLOBAL_EXIT_RATE defines a default exit rate for all hosts in the cluster. Host-level EXIT_RATE overrides the GLOBAL_EXIT_RATE value. If the global job exit rate is exceeded for 5 minutes or the period specified by JOB_EXIT_RATE_DURATION, LSF invokes LSF_SERVERDIR/eadmin to trigger a host exception. For example, GLOBAL_EXIT_RATE=10 defines a job exit rate of 10 jobs for all hosts.
- ◆ LSB_SYNC_HOST_STAT_LIM—Improves the speed with which `mbatchd` obtains host status, and therefore the speed with which LSF reschedules rerunnable jobs: the sooner LSF knows that a host has become unavailable, the sooner LSF reschedules any rerunnable jobs executing on that host. Useful for a large cluster.
- ◆ MAX_EVENT_STREAM_SIZE=*integer*—Determines the maximum size in MB of the `lsb.stream` file used by system performance analysis tools such as Platform LSF reporting. When the MAX_EVENT_STREAM_SIZE size is reached, LSF logs a special event EVENT_END_OF_STREAM, closes the stream and moves it to `lsb.stream.0` and a new stream is opened. The default event stream size is 100 MB.
- ◆ MBD_EGO_CONNECT_TIMEOUT—For EGO-enabled SLA scheduling, timeout parameter for network I/O connection with EGO `vemkd`. The default is 3 seconds.
- ◆ MBD_EGO_READ_TIMEOUT—For EGO-enabled SLA scheduling, timeout parameter for network I/O read from EGO `vemkd` after connection with EGO. The default is 3 seconds.
- ◆ MBD_EGO_TIME2LIVE—For EGO-enabled SLA scheduling, specifies how long EGO should keep information about host allocations in case `mbatchd` restarts. The default is 1440 minutes (24 hours).
- ◆ MBD_QUERY_CPUS—*cpu_list* defines the list of master host CPUS on which the `mbatchd` child query processes can run. Format the list as a white-space delimited list of CPU numbers. For example, if you specify `MBD_QUERY_CPUS=1 2 3`, the `mbatchd` child query processes will run only on CPU numbers 1, 2, and 3 on the master host.
- ◆ MBD_USE_EGO_MXJ—By default, when EGO-enabled SLA scheduling is configured, EGO allocates an entire host to LSF, which uses its own MXJ definition to determine how many slots are available on the host. LSF gets its host allocation from EGO, and runs as many jobs as the LSF configured MXJ for that host dictates. MBD_USE_EGO_MXJ forces LSF to use the job slot maximum configured in the EGO consumer. This allows partial sharing of hosts (for example, a large SMP computer) among different consumers or workload managers.

- ◆ **MIN_SWITCH_PERIOD**—To significantly improve the performance of `mbatchd` for large clusters, set this parameter to a value equal to or greater than 600. This causes `mbatchd` to fork a child process that handles event switching, thereby reducing the load on `mbatchd`. `mbatchd` terminates the child process after the **MIN_SWITCH_PERIOD** has elapsed. The default is 0 seconds—no minimum period. Log switch frequency is not restricted. See “Achieving Performance and Scalability” in *Administering Platform LSF* for details.
- ◆ **SLA_TIMER**—For EGO-enabled SLA scheduling. Controls how often each service class is evaluated and a network message is sent to EGO communicating host demand. The default is 10 seconds.
- ◆ The entire path including **JOB_SPOOL_DIR** can up to 4094 characters on UNIX and Linux or up to 255 characters for Windows. This maximum path length includes:
 - ❖ All directory and file paths attached to the **JOB_SPOOL_DIR** path
 - ❖ Temporary directories and files that the LSF system creates as jobs run.The path you specify for **JOB_SPOOL_DIR** should be as short as possible to avoid exceeding this limit.

lsb.queues

- ◆ By default, the limit for **CORELIMIT**, **MEMLIMIT**, **STACKLIMIT** and **SWAPLIMIT** is specified in KB. Use **LSF_UNIT_FOR_LIMITS** in `lsf.conf` to specify a larger unit for the limit (MB, GB, TB, PB, or EB).
- ◆ By default, memory (`mem`) and swap (`swp`) limits in `select[]` and `rusage[]` sections of **RES_REQ** are specified in MB. Use **LSF_UNIT_FOR_LIMITS** in `lsf.conf` to specify a larger unit for these limits (GB, TB, PB, or EB).

lsb.serviceclasses

- ◆ **CONSUMER=ego_consumer_name**—For EGO-enabled SLA service classes, the name of the EGO consumer from which hosts are allocated to the SLA. This parameter is mandatory for EGO-enabled SLA scheduling in order for LSF to receive hosts from EGO.
- ◆ **EGO_RES_REQ=res_req**—For EGO-enabled SLA service classes, the EGO resource requirement that specifies the characteristics of the hosts that EGO will assign to the SLA. Must be a valid EGO resource requirement. The EGO resource requirement string supports `select` sections, but the format is different from LSF resource requirements. For example:
`EGO_RES_REQ=select(linux && maxmem > 100)`
- ◆ **MAX_HOST_IDLE_TIME=seconds**—For EGO-enabled SLA service classes, number of seconds that the SLA will hold its idle hosts before LSF releases them to EGO. Each SLA can configure a different idle time. The default is 120 seconds. Do not set this parameter to a small value, or LSF may release hosts too quickly.

lsf.cluster

- ◆ On UNIX and Linux, `LSF_CONFDIR/lsf.cluster.cluster_name` is now symbolically linked to `EGO_CONFDIR/ego.cluster.cluster_name`. Before making the link, the installer checks if `LSF_CONFDIR/lsf.cluster.cluster_name` is already a symbolic link to another file, and if so, it maintains the link.

lsf.conf

- ◆ On Windows, `lsf.cluster.cluster_name` and `ego.cluster.cluster_name` are separate files. You must make configuration changes to both files. External resources must be mapped in `ego.cluster.cluster_name` only.
- ◆ `FLOAT_CLIENTS_ADDR_RANGE` supports the use of IPv6 addresses in addition to IPv4 addresses.
- ◆ `LSF_HOST_ADDR_RANGE` is now `EGO_HOST_ADDR_RANGE`.
- ◆ `LSB_LOAD_TO_SERVER_HOSTS`—Highly recommended for large clusters to decrease the load on the master LIM. Forces the client `sbatchd` to contact the local LIM for host status and load information. The client `sbatchd` only contacts the master LIM or a LIM on one of the `LSF_SERVER_HOSTS` if `sbatchd` cannot find the information locally.
- ◆ `LSB_MBD_BUSY_MSG`—Specifies the message displayed when `mbatchd` is too busy to accept new connections or respond to client requests. Define this parameter if you want to customize the message.
- ◆ `LSB_MBD_CONNECT_FAIL_MSG`—Specifies the message displayed when internal system connections to `mbatchd` fail. Define this parameter if you want to customize the message.
- ◆ `LSB_MBD_DOWN_MSG`—Specifies the message displayed by the `bhosts` command when `mbatchd` is down or there is no process listening at either the `LSB_MBD_PORT` or the `LSB_QUERY_PORT`. Define this parameter if you want to customize the message.
- ◆ `LSF_DUALSTACK_PREFER_IPV6`—Define this parameter when you want to ensure that clients and servers on dual-stack hosts use IPv6 addresses only. Setting this parameter configures LSF to sort the dynamically created address lookup list in order of `AF_INET6` (IPv6) elements first, followed by `AF_INET` (IPv4) elements, and then others.

RESTRICTION: IPv4-only and IPv6-only hosts cannot belong to the same cluster. In a MultiCluster environment, you cannot mix IPv4-only and IPv6-only clusters.

- ◆ `LSF_EGO_DAEMON_CONTROL`—Optional. Enables EGO Service Controller to control LSF `res` and `sbatchd` startup. Set the value to "Y" if you want EGO Service Controller to start `res` and `sbatchd`, and restart them if they fail. `sbatchd` and `res` will be controlled by EGO service controller. The default is "N", which means `sbatchd` and `res` will not be controlled by EGO. To configure this parameter at installation, set `EGO_DAEMON_CONTROL` in `install.config` so that `res` and `sbatchd` start automatically as EGO services.

If you manually set `EGO_DAEMON_CONTROL=Y` after installation, you must configure LSF `res` and `sbatchd` startup to AUTOMATIC in the EGO configuration files `res.xml` and `sbatchd.xml` under `EGO_ESRVDIR/esc/conf/services`.

To avoid conflicts with existing LSF startup scripts, leave this parameter undefined if you use a script (for example in `/etc/rc` or `/etc/inittab`) to start LSF daemons.

- ◆ `LSF_ENABLE_SUPPORT_IPV6`—If set, enables the use of IPv6 addresses in addition to IPv4 addresses.
- ◆ `LSF_EGO_ENVDIR`—Directory where all Platform EGO configuration files are installed. These files are shared throughout the system and should be readable from any host. The default is `EGO_TOP/conf`.
- ◆ `LSF_HOST_CACHE_NTTL`—Negative-time-to-live value in seconds. Specifies the length of time the system caches a failed DNS lookup result. If you set this value to zero (0), LSF does not cache the result.
- ◆ `LSF_HOST_CACHE_PTTL`—Positive-time-to-live value in seconds. Specifies the length of time the system caches a successful DNS lookup result. If you set this value to zero (0), LSF does not cache the result.
- ◆ For backwards compatibility, you must ensure that the following parameters in `lsf.conf` have the same as the value as the corresponding parameter in `ego.conf`. To ensure that your cluster works properly, you must define these parameters in both `lsf.conf` and `ego.conf`:
 - ❖ `LSF_LIM_PORT` and `EGO_LIM_PORT`
 - ❖ `LSF_LICENSE_FILE` and `EGO_LICENSE_FILE`
 - ❖ `LSF_PIM_INFODIR` and `EGO_PIM_INFODIR`
- ◆ The default value for `LSF_LIM_PORT` is now 7869 (this default has changed in LSF Version 7 from the pre-LSF 7 value of 6879)
- ◆ `LSF_LOG_MASK` no longer specifies LIM logging level. For LIM, you must use `EGO_LOG_MASK` in `ego.conf` to control message logging for LIM. The default value for `EGO_LOG_MASK` is `LOG_WARNING`.
- ◆ `LSF_UNIT_FOR_LIMITS=unit`—Enables scaling of large units in resource usage limits. When set, `LSF_UNIT_FOR_LIMITS` applies cluster-wide to limits at the job-level (`bsub`), queue-level (`lsb.queues`), and application level (`lsb.applications`). The limit unit specified by `LSF_UNIT_FOR_LIMITS` also applies to limits modified with `bmod`, and the display of resource usage limits in query commands (`bacct`, `bapp`, `bhist`, `bhosts`, `bjobs`, `bqueues`, `lsload`, and `lshosts`).

IMPORTANT: Before changing the units of your resource usage limits, you should completely drain the cluster of all workload. There should be no running, pending, or finished jobs in the system.

In a MultiCluster environment, you should configure the same unit for all clusters.

lsf.licensescheduler

- ◆ *Parameters section:*
 - ❖ `BLC_HEARTBEAT_FACTOR`—defines the number of times that `bld` receives no response from a license collector daemon (`blcollect`) before `bld` resets the values for that collector to zero.

- ◆ *ServiceDomain section*—`LM_STAT_INTERVAL=seconds` defines a time interval between calls that License Scheduler makes to collect license usage information from FLEXlm license management. The value specified for a service domain overrides the global value defined in the Parameters section. Each service domain definition can specify a different value for this parameter.
- ◆ *Feature section:*
 - ❖ `ENABLE_MINJOB_PREEMPTION=Y`—minimizes the overall number of preempted jobs by enabling job list optimization. For example, for a job that requires 10 licenses, License Scheduler preempts one job that uses 10 or more licenses rather than 10 jobs that each use one license.
 - ❖ `LM_REMOVE_INTERVAL=seconds`—specifies the minimum time a job must have a license checked out before `lmremove` can remove the license. `lmremove` causes `lmgrd` and vendor daemons to close the TCP connection with the application. They will then retry the license checkout. The value specified for a feature overrides the global value defined in the Parameters section. Each feature definition can specify a different value for this parameter.

lsf.shared

- ◆ On UNIX and Linux, `LSF_CONFDIR/lsf.shared` is now symbolically linked to `EGO_CONFDIR/ego.shared`. Before making the link, the installer checks if `LSF_CONFDIR/lsf.shared` is already a symbolic link to another file, and if so, it maintains the link.
- ◆ On Windows, `lsf.shared` and `ego.shared` are separate files. You must make configuration changes to both files. External resources must be defined in `ego.shared` only.
- ◆ A resource name cannot be any of the following reserved names:
`cpu cpuf io logins ls idle maxmem maxswp maxtmp type model status`
`it`
`mem ncpus define_ncpus_cores define_ncpus_procs`
`define_ncpus_threads ndisks pg r15m r15s r1m swap swp tmp ut`

lsf.sudoers

When LSF daemon control through EGO Service Controller is configured users must have EGO credentials for EGO to start `res` and `sbatchd` services. By default, `lsadmin` and `badmin` invoke the `egosh user logon` command to prompt for the user name and password of the EGO administrator to get EGO credentials.

Use the following parameters to bypass EGO logon to start `res` and `sbatchd` automatically:

- ◆ `LSF_EGO_ADMIN_USER`—User name of the EGO administrator. The default administrator name is `Admin`.
- ◆ `LSF_EGO_ADMIN_PASSWD`—Password of the EGO administrator.

To configure LSF daemon control through EGO at installation, set `EGO_DAEMON_CONTROL="Y"` in `install.config`.

Environment variables

Environment variables related to file names and job spooling directories support paths that contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.

Environment variables related to command names and job names can contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.

The following environment variables are new in LSF Version 7:

- ◆ LSF_EXECUTE_DOMAIN
- ◆ LSB_SUB_APP_NAME

The following environment variables have changed in LSF Version 7:

- ◆ LS_SUBCWD—The current working directory can be up to 4094 characters long for UNIX and Linux or up to 255 characters for Windows
- ◆ LSB_JOBNAME—The job name can be up to 4094 characters long for UNIX and Linux or up to 255 characters for Windows.
- ◆ LSB_SUB_COMMAND_LINE—The job command line can be up to 4094 characters long for UNIX and Linux or up to 255 characters for Windows.

New and changed commands, options, and output

The following command options and output are new or changed for LSF Version 7:

bapp (new)

Displays information about application profiles configured in `lsb.applications`.

By default, returns the following information about all application profiles: application name, job slot statistics, and job state statistics.

In MultiCluster, returns the information about all application profiles in the local cluster.

Application profile names and attributes are set up by the LSF administrator.

By default, the limit for CORELIMIT, MEMLIMIT, STACKLIMIT, and SWAPLIMIT display is shown in KB. Use LSF_UNIT_FOR_LIMITS in `lsf.conf` to specify a larger unit for display (MB, GB, TB, PB, or EB).

bacct

- ◆ `-app application_profile_name`—Displays accounting information about jobs submitted to the specified application profile. You must specify an existing application profile
- ◆ Options related to file names support paths up to 4094 characters long
- ◆ The job name assigned by the user, or the command string assigned by default at job submission with `bsub`. If the job name is too long to fit in this field, then only the latter part of the job name is displayed. The displayed job name or job command can contain up to 4094 characters for UNIX, or up to 255 characters for Windows.
- ◆ MEM and SWP display—By default, memory usage is shown in MB. Use LSF_UNIT_FOR_LIMITS in `lsf.conf` to specify a larger unit for display (MB, GB, TB, PB, or EB)

bclusters

`-app` displays available application profiles in remote clusters. Application profile configuration information is displayed under the heading `Remote Cluster Application Information`. Application profile information is only displayed for the job forwarding model. `bclusters` does not show local cluster application profile information.

bhist

- ◆ `bhist -l` displays the name of the application profile used by the job

- ◆ Options related to file names support paths up to 4094 characters long
 - ◆ The job name specified with `-J` can contain up to 4094 characters for UNIX, or up to 255 characters for Windows.
- bhosts**
- ◆ `bhosts -l` and `bhosts -w` display new status `closed_EGO`. For EGO-enabled SLA scheduling, `closed_EGO` indicates that the host is closed because it has not been allocated by EGO to run LSF jobs. Hosts allocated from EGO display status `ok`.
- bjgroup**
- `bjgroup` displays the name of the service class that the job group is attached to with `bgadd -sla service_class_name`.
- bjobs**
- ◆ `-app application_profile_name`—Displays information about jobs submitted to the specified application profile. You must specify an existing application profile.
 - ◆ `-sla service_class_name`—Displays information about jobs assigned to a default system service class configured with `ENABLE_DEFAULT_EGO_SLA` in `lsb.params`.
 - ◆ Use `-sla` with `-g` to display job groups attached to a service class. Once a job group is attached to a service class, all jobs submitted to that group are subject to the SLA.
 - ◆ The job name specified with `-J` can contain up to 4094 characters for UNIX, or up to 255 characters for Windows.
 - ◆ The displayed job command can contain up to 4094 characters for UNIX, or up to 255 characters for Windows.
- bkill**
- ◆ `-app application_profile_name`—Operates only on jobs associated with the specified application profile. You must specify an existing application profile.
 - ◆ Use `-g` with `-sla` to kill jobs in job groups attached to a service class.
- bladmin**
- `ckconfig [-v]`—Checks LSF License Scheduler configuration in `lsf.licensescheduler` and `lsf.conf`. By default, `bladmin ckconfig` displays only the result of the configuration file check. If warning errors are found, `bladmin` prompts you to display detailed messages. The `-v` (verbose mode) option displays detailed messages about configuration file checking to `stderr`.
- blparams (new)**
- Displays information about configurable LSF License Scheduler parameters defined in `lsf.conf` and `lsf.licensescheduler`.
- blplugins (new)**
- Displays plugin activity and the check-in, check-out, and deny counters as seen by the License Scheduler for each feature and service domain.
- bmgroup**
- When hosts are allocated to an EGO-enabled SLA, they are dynamically added to a host group created by the SLA. When the host is released to EGO, the entry is removed from the host group. `bmgroup` displays the hosts allocated by EGO to the host group created by the SLA.

bmod

- ◆ The `-app` option modifies a job by associating it to the specified application profile. The `-appn` option dissociates the specified job from its application profile. If the application profile does not exist, the job is not modified. You can only modify the application profile for pending jobs.
- ◆ Options related to file names and job spooling directories support paths that contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.
- ◆ Options related to command names and job names can contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.

bparams

`bparams -l` displays the values of the following new parameters, if they are defined in `lsb.params`.

- ◆ `DEFAULT_APPLICATION`
- ◆ `ENABLE_DEFAULT_EGO_SLA`
- ◆ `LSB_SYNC_HOST_STAT_LIM`
- ◆ `MBD_EGO_CONNECT_TIMEOUT`
- ◆ `MBD_EGO_READ_TIMEOUT`
- ◆ `MBD_EGO_TIME2LIVE`
- ◆ `MBD_QUERY_CPUS`
- ◆ `MBD_USE_EGO_MXJ`
- ◆ `SLA_TIMER`

bpeek

- ◆ Options related to file names and job spooling directories support paths that contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.
- ◆ The job name specified with `-J` can contain up to 4094 characters for UNIX, or up to 255 characters for Windows.

bqueues

- ◆ By default, the limit for `CORELIMIT`, `MEMLIMIT`, `STACKLIMIT`, and `SWAPLIMIT` display is shown in KB. Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for display (MB, GB, TB, PB, or EB).
- ◆ By default, memory (`mem`) and swap space (`swp`) is shown in MB. Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for display (MB, GB, TB, PB, or EB).

brequeue

When `JOB_INCLUDE_POSTPROC=Y` is set in an application profile in `lsb.applications`, job requeue will happen only after post-execution processing, not when the job finishes.

bsla

The `bsla` command displays the new keywords

- ◆ `CONSUMER`
- ◆ `EGO_RES_REQ`
- ◆ `MAX_HOST_IDLE_TIME`

If the SLA is under reclaim, additional keywords are displayed:

- ◆ `NUM_RECALLED_HOSTS`

bsub

◆ RECALLED_HOSTS_TIMEOUT

- ◆ `-app application_profile_name`—Submits the job to the specified application profile. You must specify an existing application profile. If the application profile does not exist in `lsb.applications`, the job is rejected.
- ◆ Supports input, output, and error file paths up to 4094 characters long. For file names that include %J and %I, the part of the name greater than 4094 will be truncated after expanding.
- ◆ Supports the use of a JSDL file to specify job submission options using the `-jsdl` or `-jsdl_strict` options
- ◆ When `ENABLE_DEFAULT_EGO_SLA` is configured in `lsb.params`, jobs submitted without `-sla` are attached to the default service class.
- ◆ You can now specify more than one first execution host candidate for a parallel job using the `-m` option
- ◆ You can use `-g` with `-sla` to attach all jobs in a job group to a service class and have them scheduled as SLA jobs. It is not possible to have some jobs in a job group not part of the service class. Multiple job groups can be created under the same SLA. You can submit additional jobs to the job group without specifying the service class name again.
- ◆ Options related to file names and job spooling directories support paths that contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.
- ◆ Options related to command names and job names can contain up to 4094 characters for UNIX and Linux, or up to 255 characters for Windows.
- ◆ The job name specified with `-J` can contain up to 4094 characters for UNIX, or up to 255 characters for Windows.
- ◆ If `JOB_SPOOL_DIR` is specified, the `-is` and `-zs` options spools the input file to the specified directory and uses the spooled file as the input file for the job. `JOB_SPOOL_DIR` can be any valid path up to a maximum length up to 4094 characters on UNIX and Linux or up to 255 characters for Windows.
- ◆ The job command can be up to 4094 characters long for UNIX and Linux or up to 255 characters for Windows. If no job name is specified with `-J`, `bjobs`, `bhist` and `bacct` display the command as the job name.
- ◆ By default, options for the following resource usage limits are specified in KB:
 - ❖ Core limit (`-C`)
 - ❖ Memory limit (`-M`)
 - ❖ Stack limit (`-S`)
 - ❖ Swap limit (`-v`)

Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for the these limits (MB, GB, TB, PB, or EB).

lshosts

By default, the amount of `maxmem` and `maxswp` is displayed in KB. The amount may appear in MB depending on the actual system memory or swap space. Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for the limit (GB, TB, PB, or EB).

lsload By default, the amount of `mem` and `swp` is displayed in KB. The amount may appear in MB depending on the actual system memory or swap space. Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for the limit (GB, TB, PB, or EB).

lspasswd You can now 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. `lspasswd` also allows revalidation of credentials.

xlsadmin (obsolete) `xlsadmin` is no longer supported.

New and changed files

The following files have been added or changed in Platform LSF Version 7:

lsb.applications The `lsb.applications` file defines application profiles, and contains many of the same parameters as `lsb.queues`. Use application profiles to define common parameters for the same type of jobs, including the execution requirements of the applications, the resources they require, and how they should be run and managed. This file is optional. Use the `DEFAULT_APPLICATION` parameter in `lsb.params` to specify a default application profile for all jobs. LSF does not automatically assign a default application profile.

This file is installed by default in `LSB_CONFDIR/cluster_name/configdir`.

By default, the limit for `CORELIMIT`, `MEMLIMIT`, `STACKLIMIT` and `SWAPLIMIT` is specified in KB. Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for the limit (MB, GB, TB, PB, or EB).

By default, memory (`mem`) and swap (`swp`) limits in `select[]` and `rusage[]` sections of `RES_REQ` are specified in MB. Use `LSF_UNIT_FOR_LIMITS` in `lsf.conf` to specify a larger unit for these limits (GB, TB, PB, or EB).

EGO configuration files for LSF daemon management (`res.xml` and `sbatchd.xml`)

The following files are located in `EGO_ESRVDIR/esc/conf/services/`:

- ◆ `res.xml`—EGO service configuration file for `res`.
- ◆ `sbatchd.xml`—EGO service configuration file for `sbatchd`.

When LSF daemons control through EGO Service Controller is configured, `lsadmin` uses the reserved EGO service name `res` to control the LSF `res` daemon, and `badmin` uses the reserved EGO service name `sbatchd` to control the LSF `sbatchd` daemon.

win_install.config (obsolete) The `win_install.config` file is no longer used by the Platform LSF for Windows installation.

Symbolic links to LSF files TIP: If your installation uses symbolic links to other files in the directories containing these new files, you must manually create links to these new files.

New and changed accounting and job event fields

lsb.acct

The following fields are new or changed in the JOB_FINISH record:

- options3 (%d)** Bit flags for job processing
- app (%s)** Application profile name
- cwd (%s)** Current working directory (up to 4094 characters for UNIX or 255 characters for Windows)
- inFile (%s)** Input file name (up to 4094 characters for UNIX or 255 characters for Windows)
- outFile (%s)** output file name (up to 4094 characters for UNIX or 255 characters for Windows)
- errFile (%s)** Error output file name (up to 4094 characters for UNIX or 255 characters for Windows)
- jobName (%s)** Job name (up to 4094 characters for UNIX or 255 characters for Windows)
- command (%s)** Complete batch job command specified by the user (up to 4094 characters for UNIX or 255 characters for Windows)
- commandSpool (%s)** Spool command file (up to 4094 characters for UNIX or 255 characters for Windows)
- inFileSpool (%s)** Spool input file (up to 4094 characters for UNIX or 255 characters for Windows)

lsb.events

The following fields are new or changed in the JOB_NEW record:

- options3 (%d)** Bit flags for job processing
- app (%s)** Application profile name
- cwd (%s)** Current working directory (up to 4094 characters for UNIX or 255 characters for Windows)
- inFile (%s)** Input file name (up to 4094 characters for UNIX or 255 characters for Windows)
- outFile (%s)** Output file name (up to 4094 characters for UNIX or 255 characters for Windows)
- errFile (%s)** Error output file name (up to 4094 characters for UNIX or 255 characters for Windows)
- jobName (%s)** Job name (up to 4094 characters for UNIX or 255 characters for Windows)
- command (%s)** Job command (up to 4094 characters for UNIX or 255 characters for Windows)

The following fields are new or changed in the JOB_MODIFY2 record:

- jobName (%s)** Job name (up to 4094 characters for UNIX or 255 characters for Windows)
- options3 (%d)** Bit flags for job processing
- app (%s)** Application profile name
- delOption3 (%d)** Delete options for the `options3` field
- inFile (%s)** Input file name (up to 4094 characters for UNIX or 255 characters for Windows)
- outFile (%s)** Output file name (up to 4094 characters for UNIX or 255 characters for Windows)
- errFile (%s)** Error output file name (up to 4094 characters for UNIX or 255 characters for Windows)
- command (%s)** Job command (up to 4094 characters for UNIX or 255 characters for Windows)

inFileSpool (%s) Spool input file (up to 4094 characters for UNIX or 255 characters for Windows)

commandSpool (%s) Spool command file (up to 4094 characters for UNIX or 255 characters for Windows)

cwd (%s) Current working directory (up to 4094 characters for UNIX or 255 characters for Windows)

The following field is new in the JOB_EXECUTE record:

execCwd (%s) Current working directory job used on execution host (up to 4094 characters for UNIX or 255 characters for Windows)

Bugs fixed since December 2006

The following bugs have been fixed in the March 2007 enhancement update since the November 2006 Maintenance Pack:

79220	Date	2007-02-04
	Description	Number of started jobs in of a share account shown by bqueues -l does not match the actual one after running badmin reconfig
	Component	mbatchd
	Platform	All
	Impact	Some shell scripts may break
82400	Date	2007-02-02
	Description	When logging on remotely, and the remote host does not allow logon locally, the authentication will fail for ProcessManager users
	Component	eauth.userpass.exe eauth_userpass.exe
	Platform	Windows
	Impact	Clients cannot log in to jfd and submit jobs
81923	Date	2007-02-01
	Description	hostsetup exits before completion
	Component	hostsetup
	Platform	All
	Impact	hostsetup does not run correctly
77119	Date	2007-01-30
	Description	mbdrestart changes RUN_TIME in host partition fairshare
	Component	mbatchd
	Platform	All
	Impact	User account information and/or user share priority will be wrong
81303	Date	2007-01-24
	Description	Submitted job does not run because password has become not valid in LSF database
	Component	lsfint.lib package

81303	Date	2007-01-24
	Platform	Windows
	Impact	Job does not run
80616	Date	2007-01-21
	Description	melim does not exit when lim is killed
	Component	melim
	Platform	Linux and UNIX
	Impact	melim stays alive after lim is gone. Multiple melims on the host after start up new lim
81434	Date	2007-01-17
	Description	When working with a huge events file, bhist is killed because it requests too much memory
	Component	bhist
	Platform	AIX
	Impact	bhist cannot work with a huge events file
79644	Date	2007-01-16
	Description	bhist on the remote clusters displays a wrong RUNLIMIT in MultCluster
	Component	bhist
	Platform	All
	Impact	Incorrect bhist output
73615	Date	2007-01-16
	Description	LSF does not shut down properly in between run levels
	Component	LSF script startup mbatchd
	Platform	UNIX
	Impact	LSF will not shut down and restart properly in between run level changes
80169	Date	2007-01-14
	Description	The startTime recorded in the lsb.acct file for each job in a chunked set is not the start time for the job, but rather the time when the chunk of jobs that job was part is sent to the execution host
	Component	mbatchd
	Platform	All
	Impact	lsb.acct file cannot be used to determine the actual wall clock run time of a job.

What's Changed in Platform LSF Version 7

80734	Date	2007-01-11
	Description	The external scheduler option string submitted with bsub -ext is lost after the job is checkpointed and restarted
	Component	sbatchd bchkpnt brestart
	Platform	Linux
	Impact	Without the external scheduler option, LSF cannot work with the external scheduling
61991	Date	2007-01-08
	Description	Reserved resources are not released when RUN_WINDOWS is closed
	Component	sbatchd mbatchd
	Platform	All
	Impact	Resource usage is affected
80232	Date	2006-12-27
	Description	When customer runs lsf_daemons start on a non-LSF host, lim dies but res and sbatchd keep running
	Component	lsf_daemons
	Platform	UNIX and Linux
	Impact	LSF daemons run on non-LSF hosts
79346	Date	2006-12-24
	Description	!U is not replaced in the eadmin.hmail and user cannot receive the notification email
	Component	eadmin.cmd mbatchd.exe
	Platform	Windows
	Impact	Mail cannot be received when customers set LSB_MAILTO=!U
80363	Date	2006-12-22
	Description	If a parallel task is manually killed, the pam job output display of task exit status might be wrong
	Component	pam
	Platform	All
	Impact	Incorrect information about why and how a parallel job task is terminated
79945	Date	2006-12-20
	Description	Some environment variables (CWD, CLEARCASE_ROOT) are not set correctly when submitting Clearcase jobs from Windows to UNIX
	Component	sbatchd
	Platform	All
	Impact	Low

80114	Date	2006-12-20
	Description	mbatchd takes a long time to write lsb.acct file for each job
	Component	mbatchd
	Platform	All
	Impact	Slow client response
75584	Date	2006-12-20
	Description	pam receives SISEGV after pthread_create() fails
	Component	pam
	Platform	Linux
	Impact	Job will fail and pam will core dump if a large stack limit is set
79427	Date	2006-12-18
	Description	Threshold is ignored if resource value is exactly the same as threshold
	Component	mbatchd
	Platform	All
	Impact	Job can be dispatched even if the host is closed_Busy because loadSched is equal to the threshold
80166	Date	2006-12-13
	Description	Without ego_base license, lim cannot start
	Component	lim
	Platform	All
	Impact	Serious
79374	Date	2006-12-13
	Description	New openmpi mpirun options are not supported - job will fail
	Component	openmpi_wrapper
	Platform	Linux
	Impact	Job cannot run due to wrong parsing of mpirun options
79805	Date	2006-12-11
	Description	lim wants to check out dual-core license even though the CPU is not dual-core
	Component	lim
	Platform	UNIX and Linux
	Impact	lim wants to check out dual-core license even though the CPU is not dual-core
78757	Date	2006-12-11
	Description	bjobs must consistently exit with -1 when jobs not found
	Component	bjobs

What's Changed in Platform LSF Version 7

78757	Date	2006-12-11
	Platform	All
	Impact	Scripts calling bjobs may not work
78745	Date	2006-12-08
	Description	Job name with wildcard does not match all matching jobs if JOB_DEP_LAST_SUB is set
	Component	bparams mbatchd
	Platform	All
	Impact	Some scripts depend on the correct behavior
79842	Date	2006-12-08
	Description	If LSF_HPC_EXTENSIONS is defined in lsf.conf, rusage report of a parallel job on first execution node is not accurate - one process is double-counted, some do not show up in bjobs
	Component	pam
	Platform	All
	Impact	rusage report is not correct
79777	Date	2006-12-08
	Description	If LSF_OEM_LICENSE_PATH is not valid, lim cannot get license from license file set in LSF_LICENSE_FILE
	Component	lim
	Platform	All
	Impact	lim cannot work if LSF_OEM_LICENSE_PATH is not valid
78373	Date	2006-12-07
	Description	When many users are configured in a fairshare tree, mbatchd takes long time to replay 200K jobs
	Component	mbatchd
	Platform	All
	Impact	mbatchd does not respond for a long period of time
79539	Date	2006-12-07
	Description	Incorrect LIM warning messages for dual core license even though LIM can recognize the dual core license
	Component	lim
	Platform	All
	Impact	Confusing warning messages
79633	Date	2006-12-06
	Description	sbatchd logs misleading message about unlock hosts
	Component	sbatchd

79633	Date	2006-12-06
	Platform	All
	Impact	User cannot see the real reason of the failure
78527	Date	2006-12-06
	Description	Cluster is unlicensed due to lack of license
	Component	lim.exe
	Platform	All Windows platforms (Intel x86/x64/IA64/Single-Core/Dual-Core, AMD x86/X64/Single-Core/Dual-Core)
	Impact	Cluster unlicensed due to lack of license
78956	Date	2006-12-05
	Description	When LSF dispatches a large number of jobs in one scheduling cycle, master lim receives lots of host information queries from sbatchd and slows down master host performance
	Component	sbatchd
	Platform	All
	Impact	Master host slows down
76846	Date	2006-12-05
	Description	After using bkill -r kill on running array jobs, the jobs still keep running
	Component	mbatchd
	Platform	UNIX
	Impact	Jobs cannot be killed completely
79431	Date	2006-12-04
	Description	lsadmin will core dump when LSF_RSH is defined in lsf.conf
	Component	badmin lsadmin
	Platform	All
	Impact	Medium
79213	Date	2006-12-01
	Description	lsfinstall does not work if the lsf.shared started with Begin Cluster when the LSF is reinstalled
	Component	lsfinstall
	Platform	UNIX and Linux
	Impact	Cannot reinstall successfully
79069	Date	2006-12-01
	Description	API cpuset_create() fails with "Invalid arguments", job goes back to pending state
	Component	rla schmod_cpuset.so

What's Changed in Platform LSF Version 7

79069	Date	2006-12-01
	Platform	SGI cpuset integration
	Impact	In the best case, the job gets rescheduled to a different node and it runs. In the worst case, it gets scheduled to the same node and the allocation fails again.

Known Issues

- ◆ [Platform LSF Version 7](#)
- ◆ [Platform LSF on Windows](#)
- ◆ [Platform LSF License Scheduler](#)
- ◆ [Platform LSF reporting](#)
- ◆ [Platform LSF License Scheduler reporting](#)
- ◆ [EGO-enabled SLA scheduling limitations](#)

Platform LSF Version 7

SGI cpusets and JOB_INCLUDE_POSTPROC

If you specify `JOB_INCLUDE_POSTPROC=Y` in an application profile in `lsb.applications` to enable job post-execution to be included in job finish status reporting, SGI cpusets behave differently from previous releases.

The post-execution processing is not attached to the job cpuset, but Platform LSF does not release the cpuset until post-execution processing has finished.

lsfstartup on Mac OS X When `LSF_EGO_DAEMON_CONTROL="Y"` is specified in `lsf.conf`, running `lsfstartup` displays incorrect error messages, but the cluster can be started correctly.

When you see the following message

```
Error(s) found in previous operation, continue? [y/n]y
```

choose yes (y) to continue startup.

Platform LSF on Windows

cmd.exe permissions For jobs that run on a Windows Server 2003, x64 Edition platform, users must have "Read" and "Execute" privileges for `cmd.exe`.

Post-execution process tracking

`JOB_POSTPROC_TIMEOUT` configured in an application profile in `lsb.applications` has no effect on Windows execution hosts because post-execution processing on Windows tracks only the direct parent command. Child processes of the post-execution command remain running.

Platform LSF License Scheduler

Symptoms With the flexible grid integration plugin enabled, `bladmin reconfig` has the following problems when reconfiguring License Scheduler after configuration change in `lsf.licensescheduler`:

- ◆ The error `extfilter_init(): Error creating external filter listening socket is logged to bld.log`
- ◆ The error `Vendor daemon could not connect to external filter server is logged to the vendor daemon log file`
- ◆ License Scheduler commands display `The License Scheduler server bld is unreachable`

Workaround After changing any configuration in `lsf.licensescheduler`, run `bladmin shutdown` to shut down `bld`. After waiting at least one minute, then run `blstartup` to restart `bld`. Do not run `bladmin reconfig`.

Platform LSF reporting

The default out-of-box configuration for Platform LSF reporting with Oracle database can only support up to 1 million jobs per day. If your data volume is greater than this, contact Platform Support (support@platform.com) for recommended configuration.

In the Service Level Agreement (SLA) report for throughput, the starting point of the Optimal line is inconsistent with the starting point of the time window. In the Service Level Agreement (SLA) report for velocity, the velocity goal line does not cover the last bar in the chart.

Platform LSF License Scheduler reporting

Platform LSF License Scheduler is not supported on Linux IA64 hosts. By default, the reporting data loader for the Platform LSF License Scheduler daemon `bld` is disabled on Linux IA64 hosts.

- ◆ If you install Platform LSF and the Platform Management Console with Platform LSF License Scheduler in a cluster that includes Linux IA64 hosts, you must configure the Platform LSF License Scheduler daemon `bld` as a Platform EGO service, and enable the reporting data loader for `bld`.

Complete the following steps:

- a Edit `EGO_TOP/eservice/esc/conf/services/plc_service.xml` and specify the host name where `bld` is running. For example:

```
<ego:ResourceGroupName>ManagementHosts</ego:ResourceGroupName>
<ego:ResourceRequirement>select('hostA')</ego:ResourceRequirement>
```

- b Edit `EGO_TOP/lsf7.0/ego/perf/conf/plc/plc_lsf.xml` and change `Enable="false"` to `Enable="true"` to enable the `bld` data loader. For example:

```
<DataLoader Name="bldloader" Interval="300" Enable="true"
LoadXML="dataloader/bld.xml" />
```

- ◆ If you install only Platform LSF and the Platform Management Console on a Linux IA64 host without Platform LSF License Scheduler, you do not need to enable the `bld` data loader.

EGO-enabled SLA scheduling limitations

- Parallel jobs** Resource allocation is based on the number of jobs, not the slots required by the job. EGO-enabled SLA requests resource based on velocity and the number of pending jobs. If a parallel job requires multiple processors, the SLA may request fewer processors than the requirement, which causes the job to remain pending. To avoid this, you can configure larger velocity in the SLA.
- MultiCluster** Resource export under the lease model is not guaranteed. With EGO-enabled SLA scheduling, all resources are dynamic, so the exported hosts may not be allocated to LSF.
- Advance reservations** EGO-enabled SLA does not support advance reservations. Advanced reservations need to reserve resources for a specified time window, which is not currently supported in EGO.
- Job-level resource requirements (bsub -R)** LSF takes the resource requirement into consideration for scheduling, but if the resource request does not match the resource requirement specified in the service class, the host allocated by EGO cannot match the specified resource requirement, and the job remains pending. LSF treats the allocated host as idle and returns it to EGO. The pending job causes another request to be sent to EGO, which allocates another host, which may or may not satisfy the resource requirement.
Use `EGO_RES_REQ=res_req` in the service class configuration to specify all job resource requirements.
- Job-level host preference (bsub -m)** Specific job-level host requests are similar to `bsub -R` (essentially the same as `bsub -R "select host_name"`). The specified host is not guaranteed to be allocated by EGO. The job remains pending until the specified host actually allocated.
Use `EGO_RES_REQ=res_req` in the service class configuration to specify all job resource requirements.

Download the Platform LSF Version 7 Distribution Packages

Download the LSF distribution packages through FTP at <ftp.platform.com>.

IMPORTANT: The latest Platform LSF Version 7 release is Update 2. Distribution packages are available only for Platform LSF Version 7 Update 2 and Platform LSF Version 7 Update 1.

Download steps

Prerequisites: Access to the Platform FTP site is controlled by login name and password. If you cannot access the distribution files for download, send email to support@platform.com.

- 1 Log on to the LSF file server.
- 2 Change to the directory where you want to download the LSF distribution files. Make sure that you have write access to the directory. For example:

```
# cd /usr/share/lsf/tarfiles
```
- 3 FTP to the Platform FTP site:

```
# ftp ftp.platform.com
```
- 4 Provide the login user ID and password provided by Platform.
- 5 Change to the directory for the LSF Version 7 release:

```
ftp> cd /distrib/7.0
```
- 6 Set file transfer mode to binary:

```
ftp> binary
```
- 7 For LSF on UNIX and Linux, get the installation distribution file.

TIP: Before installing LSF on your UNIX and Linux hosts, you must uncompress and extract `lsf7.0_lsinstall.tar.Z` to the same directory where you download the LSF product distribution tar files.

- 8 Get the distribution packages for the products you want to install on the supported platforms you need.
- 9 Download the latest Platform LSF Version 7 documentation from
`/distrib/7.0/docs/`.
- 10 Download the latest Platform EGO Version 1.2 documentation from
`/distrib/7.0/docs/`.
- 11 Optional. Download the Platform Management Console (PMC) distribution package.

NOTE: To take advantage of the Platform LSF reporting feature, you *must* download and install the Platform Management Console. The reporting feature is only supported on the same platforms as the Platform Management Console: 32-bit and 64-bit x86 Windows and Linux operating systems.

- 12 Exit FTP.

```
ftp> quit
```

Archive location of previous update releases

Directories containing release notes and distribution files for previous LSF Version 7 update releases are located on the Platform FTP site under `/distrib/7.0/archive`. Archive directories are named relative to the current update release:

- ◆ LSF Version 7 Update 1: `/distrib/7.0/archive/update1`

Install Platform LSF Version 7

Installing Platform LSF involves the following steps:

- 1 Get a DEMO license (`license.dat` file).
- 2 Run the installation programs.

Get a Platform LSF demo license

Before installing Platform LSF Version 7, you must get a demo license key.

Contact `license@platform.com` to get a demo license.

Put the demo license file `license.dat` in the same directory where you downloaded the Platform LSF product distribution tar files.

Run the UNIX and Linux installation

Use the `lsfinstall` installation program to install a new LSF Version 7 cluster, or to upgrade from an earlier LSF version.

See *Installing Platform LSF on UNIX and Linux* for new cluster installation steps.

See the *Platform LSF Reference* for detailed information about `lsfinstall` and its options.

Run the Windows installation

Platform LSF on Windows 2000, Windows 2003, and Windows XP is distributed in the following packages:

- ◆ `lsf7.0_win32.msi`
- ◆ `lsf7.0_win-x64.msi`
- ◆ `lsf7.0_win-ia64.msi`

See *Installing Platform LSF on Windows* for installation steps.

Install Platform LSF License Scheduler

See *Using Platform LSF License Scheduler* for installation and configuration steps.

Install Platform LSF HPC

Use `lsfinstall` to install a new Platform LSF HPC cluster or to upgrade LSF HPC from a previous release.

IMPORTANT: Make sure `ENABLE_HPC_INST=Y` is specified in `install.config` to enable Platform LSF HPC installation.

See *Using Platform LSF HPC* for installation and configuration steps.

Special installation steps for the Platform Management Console on Linux IA64

To install the Platform Management Console on Linux IA64 hosts, you must download and install the Linux IA64 version of BEA Jrockit 5.0 JRE.

1 Download the Linux IA64 version of BEA Jrockit 5.0 JRE.

a Open the BEA download page.

`http://commerce.bea.com/products/weblogicjrockit/5.0/jr_50.jsp`

b Save the download file to your local disk.

For JRockit 5.0 R27.1 JRE Linux (Intel Itanium - 64-bit), save the file named `jrockit-R27.1.0-jre1.5.0_08-linux-ipf.bin`.

c Make sure that the `.bin` file is executable.

`chmod +x jrockit-R27.1.0-jre1.5.0_08-linux-ipf.bin`

2 Install the JRE on the Linux IA64 host.

a Change to a shared directory where you want to install BEA Jrockit.

b Run the installer in console mode.

`jrockit-R27.1.0-jre1.5.0_08-linux-ipf.bin -mode=console`

The installation creates a new directory:

`jrockit-R27.1.0-jre1.5.0_08`

3 Follow the steps in *Installing Platform LSF on UNIX and Linux* to run `lsfinstall` to install Platform LSF and the Platform Management Console.

4 Make a symbolic link to the JRE.

For example, if you installed the JRE under `/opt/jre`:

`cd $EG0_TOP/jre`

`ln -s /opt/jre/jrockit-R27.1.0-jre1.5.0_08-linux-ipf linux-ia64`

5 Check the symbolic link to the JRE.

If the symbolic link is correct, you should see the contents of the `linux-ia64` directory:

`cd $EG0_TOP/jre/linux-ia64`

`ls`

`bin/ lib/ LICENSE license.bea README.TXT`

Learn About Platform LSF Version 7

Information about Platform LSF is available from the following sources:

- ◆ [World Wide Web and FTP](#)
- ◆ [Platform LSF documentation](#)
- ◆ [Platform EGO documentation](#)
- ◆ [Platform training](#)

World Wide Web and FTP

Information about Platform LSF Version 7 is available in the LSF Version 7 area of the Platform FTP site (<ftp.platform.com/>).

The latest information about all supported releases of Platform LSF is available on the Platform Web site at www.platform.com.

If you have problems accessing the Platform web site or the Platform FTP site, send email to support@platform.com.

my.platform.com

my.platform.com—Your one-stop-shop for information, forums, e-support, documentation and release information. my.platform.com provides a single source of information and access to new products and releases from Platform Computing.

On the Platform LSF Family product page of my.platform.com, you can download software, patches, updates and documentation. See what's new in Platform LSF Version 7, check the system requirements for Platform LSF, and browse the latest documentation updates through the Platform LSF Knowledge Center.

Platform LSF documentation

The Platform LSF Knowledge Center is your entry point for LSF documentation. After downloading and extracting the LSF documentation distribution file, browse the file `docs/lsf/7.0/index.html` to access the documentation.

If you have installed the Platform Management Console, access the Platform LSF documentation through link to the Platform Knowledge Center.

Platform EGO documentation

The Platform EGO Knowledge Center is your entry point for Platform EGO documentation. It is installed when you install LSF. To access the EGO documentation, browse the file `EGO_TOP/docs/ego/1.2/index.html`.

If you have installed the Platform Management Console, access the Platform EGO documentation through link to the Platform Knowledge Center.

Platform training

Platform's Professional Services training courses can help you gain the skills necessary to effectively install, configure and manage your Platform products. Courses are available for both new and experienced users and administrators at our corporate headquarters and Platform locations worldwide.

Customized on-site course delivery is also available.

Find out more about Platform Training at

www.platform.com/Services/Training/, or contact Training@platform.com for details.

Get Technical Support

Contact Platform

Contact Platform Computing or your LSF vendor for technical support. Use one of the following to contact Platform technical support:

Email support@platform.com

World Wide Web www.platform.com

Mail Platform Support
Platform Computing Inc.
3760 14th Avenue
Markham, Ontario
Canada L3R 3T7

When contacting Platform, please include the full name of your company.

See the Platform Web site at www.platform.com/Company/Contact.Us.htm for other contact information.

Get patch updates and other notifications

To get periodic patch update information, critical bug notification, and general support notification from Platform Support, contact supportnotice-request@platform.com with the subject line containing the word "subscribe".

To get security related issue notification from Platform Support, contact securenotice-request@platform.com with the subject line containing the word "subscribe".

We'd like to hear from you

If you find an error in any Platform documentation, or you have a suggestion for improving it, please let us know:

Email

`doc@platform.com`

Mail

Information Development
Platform Computing Inc.
3760 14th Avenue
Markham, Ontario
Canada L3R 3T7

Be sure to tell us:

- ◆ The title of the manual you are commenting on
- ◆ The version of the product you are using
- ◆ The format of the manual (HTML or PDF)

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.

Document redistribution policy

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 Corporation in the United States and in other jurisdictions.

POWERING HIGH PERFORMANCE, PLATFORM COMPUTING, PLATFORM SYMPHONY, PLATFORM JOBSCHEDULER, and the PLATFORM and PLATFORM LSF logos are trademarks of Platform Computing Corporation 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.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

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.

Macrovision, Globetrotter, and FLEXlm are registered trademarks or trademarks of Macrovision Corporation in the United States of America and/or other countries.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

Copyright

Intel, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries 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

www.platform.com/legal-notices/third-party-license-agreements