Platform LSF
Version 9 Release 1

Release Notes for IBM Platform License Scheduler

GI13-3414-00

Platform LSF
Version 9 Release 1

Release Notes for IBM Platform License Scheduler

GI13-3414-00

Note
Before using this information and the product it supports, read the information in "Notices" on page 7.

First edition

This edition applies to version 9, release 1 of IBM Platform LSF (product number 5725G82) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Release Notes for IBM Platform License Scheduler

Version: 9.1

Release date: March 2013

Last modified: February 2013

Support: www.ibm.com/support

Upgrade and Compatibility Notes

LSF compatibility

IBM Platform License Scheduler (License Scheduler) 9.1 is fully compatible with LSF 7.0.6, 8.0, 8.0.1, 8.3, and 9.1.

System requirements

Operating system support:

- AIX 6 and AIX 7 on POWER
- Solaris 10 and 11 on Sparc
- Linux on x86-64 Kernel 2.6 and 3.0
- Windows 2003/2008/XP/7 32-bit (client only)

Scalability

Functional limits

- 200 projects associated with each feature, 250 features simultaneously
- 1 **blcollect** can pull 400 features from a single license server

Non-functional limits

- 2000 features per license server, 60 license servers, 20 projects per feature
- 2000 features per license server, 1 license server, 200 projects per feature
- 400 features per collector

  You can use an alternate executable that wraps the **lmstat** command. These wrappers can do several things including:

  – Run **lmstat** with additional switches to isolate output required for License Scheduler management.

  – Run **lmstat** and modify the output format to avoid **blcollect** errors.

  – Run **lmstat** and modify the license count to take some licenses out of the control of License Scheduler.

What's Changed in License Scheduler Version 9.1

Licensing features obsolete

None yet

New platform support

License Scheduler is now available on the following platforms:

None yet

Fast dispatch project mode

Fast dispatch project mode is a new approach to license distribution that increases license utilization for project licenses. Fast dispatch project mode has the scheduling performance of cluster mode with the functionality of project mode.

In fast dispatch project mode, License Scheduler does not have to run the FlexNet command **lmstat** to verify that a license is free before each job dispatch. As soon as a job finishes, the cluster can reuse its licenses for another job of the same project, which keeps gaps between jobs small. However, because License Scheduler does not run **lmstat** to verify that the license is free, there is an increased chance of a license checkout failure for jobs if the license is already in use by a job in another project.

Hierarchical project group paths

By default, hierarchical project groups in fast dispatch project mode are the same as hierarchical project groups in project mode. Fast dispatch project mode also supports the use of hierarchical project group paths, which helps License Scheduler dispatch more jobs in fast dispatch project mode.

Enabling hierarchical project group paths enables the following:

- Features can use hierarchical project groups with project and project group names that are not unique, as long as the projects or project groups do not have the same parent. That is, you can define projects and project groups in more than one hierarchical project group.

  For example, the `p4` project can be defined for project groups `g4` and `g6`, each with its specific resource allocation within the project groups.
- When specifying `-Lp` *license_project*, you can use paths to describe the project hierarchy without specifying the root group.

  For example, if you have `topgrp` as your root group, which has a child project group named `g1` with a child project group named `g3`, which has a project named `p1`, you can use `-Lp /g1/g3/p1` to specify this project.
- Hierarchical project groups have a default project named `others` with a default share value of 0. Any projects that do not match the defined projects in a project group are assigned into the `others` project. If the `others` project has a share value of 0, this project can still use licenses if the defined projects with shares are not using the licenses. Therefore, by default, the `others` project has the lowest priority within a project group.

  If there is already a project named `others`, the preexisting `others` project specification overrides the default project.

Defining hierarchical project groups for fast dispatch project mode is the same as for project mode, allowing for project and project group names that are not unique.

You can also specify different project groups with different root groups. Different features can use different root groups (as defined by the GROUP_DISTRIBUTION parameter), each with its own project group hierarchy and share policies.

Merge multiple user@host keys for parallel jobs

When LSF dispatches a parallel job, License Scheduler attempts to check out `user@host` keys in the parallel job constructed using the user name and all execution host names, and merges the corresponding checkout information on the service domain if found.

For example, in project mode, for feature `F1` with two projects (`P1` and `P2`) in service domain `sd1`, with ten tokens, a parallel job is dispatched to four execution hosts using the following command:

```
bsub -n 4 -Lp P1 -R "rusage[F1=4]" mycmd
```

The job on each execution host checks out one `F1` license from the `sd1` service domain. If the four execution hosts are `hostA`, `hostB`, `hostC`, and `hostD`, there are checkout keys for `user@hostA`, `user@hostB`, `user@hostC`, and `user@hostD`, and each entry contributes corresponds with one token checked out. These tokens all merge into data for the `P1` project in the `F1` feature.

If `MERGE_BY_SERVICE_DOMAIN=Y` is defined, License Scheduler also merges multiple `user@host` data for parallel jobs across different service domains.

If `CHECKOUT_FROM_FIRST_HOST_ONLY=Y` is defined, License Scheduler only considers `user@host` information for the first execution host of a parallel job when merging the license usage data. Setting in individual Feature sections overrides the global setting in the Parameters section. If a feature has multiple Feature sections (using **`LOCAL_TO`**), each section must have the same setting for **`CHECKOUT_FROM_FIRST_HOST_ONLY`**.

Releasing resources for suspended jobs

Enabling the **`LSF_LIC_SCHED_PREEMPT_SLOT_RELEASE`** parameter in `lsf.conf` now allows you to release memory and slot resources of a License Scheduler job that is suspended. These resources are only available to pending License Scheduler jobs that request at least one license that is the same as the suspended job. Previously, enabling this parameter only released slots of suspended License Scheduler jobs, which can then by used by pending jobs even if they do not request License Scheduler resources.

By default, the job slots are the only resources available after a job is suspended. You can also specify that memory resources are available by enabling preemption for memory resources. To enable memory resource preemption, specify `PREEMPTABLE_RESOURCES = mem` in `lsb.params`.

Changed commands

License Scheduler 9.1 features the following changes to the following commands:

blstat In cluster mode, **`blstat`** also displays the ideal number of tokens allocated to the cluster in the `TARGET` column.

In fast dispatch project mode, **`blstat -c`** also displays the actual and ideal number of tokens allocated to the cluster in the `ALLOC` and `TARGET` columns.

blusers

In cluster mode, **`blusers -J`** displays tokens for CLASS-C features, which are tokens that are checked out to features that a job did not explicitly request. These features have an `INUSE` value, but no `RUSAGE` value.

taskman

You can now use the logical OR operator (||) to separate rusage string siblings, in addition to the logical AND operator (:) that is already supported.

For example,

```
taskman -Lp P1 -R "rusage[f1=1:f2=1||f1=5:f3=1||f4=1]" myjob
```

Known Issues and Limitations

Jobs using more than one feature triggering preemption

In project mode, when a job that uses more than one feature triggers a preemption, an over-preemption may occur. For example, only one job needs to be preempted, but **bld** preempts two or more jobs.

To work around this issue, use fast dispatch project mode instead of project mode.

Released license tokens are reserved again after restart

For dynamic license features with duration specified in rusage, after restarting **bld** or **mbatchd** (with **badmin reconfig**, **badmin mbdrestart** or **bladmin reconfig**), released licenses are reserved again for the specified duration.

When single job requires more tokens than the allocation buffer

In cluster mode, you must make sure that you set the allocation buffer for dynamic distribution of licenses greater than the expected maximum number of license tokens required by any single job.

Set file descriptor limit large enough

Make sure the operating system file descriptor limit is large enough to support all **taskman** tasks, License Scheduler (**bl***) commands, and connections between License Scheduler and LSF. Use LS_MAX_TASKMAN_SESSIONS in `lsf.licensescheduler` to define the maximum number of taskman jobs that can run simultaneously.

blusers -J shows only tokens and features requested by the job

In project mode, **blusers -J** does not display additional tokens checked out by the job or features not originally requested by the job.

Installation

When installing License Scheduler standalone, the installer removes EGO environment variables from `cshrc.lsf` and `profile.lsf`. Specify a different LSF_TOP from the LSF installation to install standalone License Scheduler.

Preemption

If the LSF administrator has defined `JOB_CONTROLS` in `lsb.queues` so that job controls (such as the signal SIGTSTP) take effect when License Scheduler preemption occurs, **LIC_SCHED_PREEMPT_STOP=Y** in `lsf.conf` must also be defined for License Scheduler preemption to work.

Theoretical limit for license utilization

License Scheduler is often held up as a license utilization optimization engine. Unfortunately application behavior and interaction with a license server can limit the maximum theoretical utilization a business can meet.

Managing licenses has complex interdependencies and behaviors. When a job launches, it does NOT immediately check out a license and hold that license for the duration of the job execution. Applications frequently check out a license after the application is launched and do not keep it until the job terminates. Some applications even do multiple license checkout/in during a single run.

If you are not using the **DURATION** and **DYNAMIC** parameters in License Scheduler, the time an application is running without a license checked out is lost license utilization because LSF/License Scheduler holds the licenses in a **RESERVED** state. When checking the license server state using **lmstat**, it appears as though there are unallocated keys but no additional jobs being dispatched.

Due to unpredictably and license model complexity, loss of license utilization is a fact of managing licenses using License Scheduler when license checkout time is not identical to application execution time.

feature.servicedomain.dat file grows too big with large scale configuration

If, for example, you configure License Scheduler with 500 features, 100 projects, and 50 service domains, License Scheduler records information into data files every minute, causing potential performance issues.

License Scheduler preempted job not redispatched before pending jobs

Fairshare policy is based in part on accumulated in use tokens. Since preempted jobs may already have accumulated in use time, new pending jobs may be dispatched first.

brun job is preempted by License Scheduler and resumed by mbschd

License Scheduler sends a preemption command to the **mbatchd** to preempt a **brun** job. The job is preempted but then runs again. This repeats until the job is done.

Incorrect output for hierarchical fairshare among Project Groups

When License Scheduler hierarchical fairshare is configured, running **blinfo** without the **-G** flag displays incorrect share information. The same error occurs in output from **blparams** and **blinfo -p**.

Same class-C features cannot be merged to the tasks

In project mode, when there is more than one job in the system submitted by the same user and on the same host, License Scheduler distributes them to each job in a round robin fashion giving each job at most one class-C license.

The job suspended unnecessarily on the specific condition involving hierarchical fairshare

This bug only impacts on batch jobs in project mode. Assume **LM_STAT_INTERVAL** is set to a low value (for example, 10) in `lsf.licensescheduler`. The **mbatchd** updates license request every 15 seconds. This combination means that a running job can be preempted in the next scheduling session if there is ownership defined.

Freed licenses are reserved after bladmin reconfig

After taskman reconnects to **bld** (after running **bladmin reconfig**), a taskman job start time is the current time not the previous taskman started time. The **bld** reserves token for the duration.

Install License Scheduler 9.1

Installing License Scheduler involves the following steps:

1. Get an entitlement file.
2. Run the installation program.

Run the License Scheduler installation

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

IBM Platform Application Center 9.1

IBM Platform Application Center provides a web-based user interface for job submission, job and LSF host monitoring, and reporting. License Scheduler monitoring and reporting is now included in IBM Platform Application Center.

You must install LSF before installing IBM Platform Application Center. See *Installing IBM Platform Application Center* for installation and configuration steps.

Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web

sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Intellectual Property Law
Mail Station P300
2455 South Road,
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application

programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

LSF®, Platform, and Platform Computing are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Printed in USA

GI13-3414-00