



UNIVA, AN ALTAIR COMPANY

GRID ENGINE DOCUMENTATION

---

# Grid Engine Release Notes

---

*Author:*  
Altair Engineering

*Version:*  
8.6.17

November 4, 2020

© 2020 ALTAIR ENGINEERING, INC. ALL RIGHTS RESERVED.

WE ARE CURRENTLY LISTED ON NASDAQ AS ALTR. UNIVA IS AN ALTAIR COMPANY

# Contents

<b>1 License</b>	<b>1</b>
<b>2 Supported Operating Systems, Versions and Architectures</b>	<b>7</b>
<b>3 Supported and Tested Docker Versions</b>	<b>9</b>
3.1 Known Docker Issues That Affect Univa Grid Engine . . . . .	14
<b>4 Fixes and Enhancements</b>	<b>15</b>
4.1 Major Enhancements . . . . .	15
4.1.1 (8.6.14) Upgrade Notes . . . . .	15
4.1.2 (IMPORTANT) UGE Product Parts and Features Declared as Deprecated	15
4.1.3 (UGE 8.6.11) Resolving additional group id's and definable ranges . .	15
4.1.4 (UGE 8.6.8) Possibility to disable qinstance access checks during sub- mission . . . . .	16
4.1.5 (UGE 8.6.8) New Scheduler Parameters for Tuning Wildcard PE Scheduling	16
4.1.6 (UGE 8.6.8) New PE_SORT_ORDER to Improve Scheduling Time . . . . .	16
4.1.7 (UGE 8.6.8) Support for Linux Mount Namespace . . . . .	17
4.1.8 (UGE 8.6.8) Rework of profiling for all threads and execution daemon .	17
4.1.9 (UGE 8.6.8) GPU Usage of Jobs Reported by DCGM . . . . .	17
4.1.10 (UGE 8.6.8) Automatically Adding All Job Owners Supplementary Group IDs to a Docker Job . . . . .	19
4.1.11 (UGE 8.6.7) DRMAA2 Related Changes and Enhancements . . . . .	19
4.1.12 (UGE 8.6.7) Official Red Hat 8 Support . . . . .	20
4.1.13 (UGE 8.6.5) Support for Linux Mount Namespace (obsoleted, see above under UGE 8.6.8) . . . . .	20
4.1.14 (UGE 8.6.3) Bulk Changes for Exec Hosts and Projects . . . . .	21
4.1.15 New Core Binding Strategies and PE Behavior . . . . .	21
4.1.16 Affinity Based Job Placement . . . . .	22
4.1.17 Managing Access to Devices with RSMAPs . . . . .	22
4.1.18 Integration with Nvidia DCGM . . . . .	23
4.1.19 Performance Improvements . . . . .	24
4.1.20 Reworked Dispatch Job Information . . . . .	24
4.1.21 Data Compression at Communication Layer . . . . .	25

---

4.1.22	Using RSMAPs with Topology Masks and XOR Operator . . . . .	26
4.2	Other . . . . .	27
4.2.1	JSV Improvements . . . . .	27
4.2.2	Reduce qghost Data Request Sizes at sge_qmaster . . . . .	28
4.2.3	Upgrade Advance and Standing Reservations . . . . .	28
4.2.4	per_pe_task_prolog and per_pe_task_epilog . . . . .	28
4.2.5	Support for nvidia-docker 2.0 . . . . .	28
4.2.6	The Master Task of a Parallel Docker Job in a Container . . . . .	29
4.2.7	Run the Container as root, Allow to Run Prolog etc. as a Different User . . . . .	29
4.2.8	Automatically Map User ID and Group ID of a User Into the Container . . . . .	29
4.2.9	Create a container_pe_hostfile with all Container Hostnames . . . . .	29
4.2.10	Docker Daemon Response Timeout . . . . .	30
4.2.11	Cgroups and Containers . . . . .	30
4.2.12	Specify Arguments to Autostart Docker Jobs . . . . .	30
4.2.13	New Client Command <b>qalter</b> . . . . .	31
4.2.14	Changes to the <b>loadcheck</b> Command . . . . .	31
4.2.15	Changed startup behaviour of the execution daemon . . . . .	31
4.2.16	Forwarding the environment for a qrsh (without command) and qlogin job is enabled . . . . .	31
4.2.17	Suppress the user switch inside Docker jobs . . . . .	32
4.2.18	Allow to skip Docker image check in execution daemon . . . . .	32
4.2.19	New values for initial_state of a queue . . . . .	32
4.2.20	Removed reporting of m_mem_free_nX load values . . . . .	32
4.2.21	GPU Affinity as Soft Request . . . . .	32
4.2.22	Trigger command to force re-read of host_aliases file . . . . .	33
4.2.23	Automatically set jobs to hold state if they trigger errors . . . . .	33
4.2.24	Switch off submit host verification . . . . .	33
4.2.25	Log daemon startup errors to syslog . . . . .	33
4.2.26	Enabled monitoring and deadlock detection feature for all platforms . . . . .	33
4.2.27	sub-cgroup feature . . . . .	34
4.2.28	Removing stale "context" entries in queue . . . . .	34
4.2.29	Cleanup of output file path handling . . . . .	34
4.2.30	New logging option for hostname or ip lookup calls . . . . .	34

---

4.2.31	Changes in handling of job arguments for (t)csh users . . . . .	35
4.3	Full List of Fixes and Enhancements . . . . .	36
4.3.1	8.6.0 (and Also Fixed for a 8.5.*, 8.4.*, 8.3.* Patch Release) . . . . .	36
4.3.2	8.6.0 (and Also Fixed for a 8.5.* and 8.4.* Patch Release) . . . . .	36
4.3.3	8.6.0 (and Also Fixed for a 8.5.* Patch Release) . . . . .	37
4.3.4	8.6.0 . . . . .	41
4.3.5	8.6.1 . . . . .	47
4.3.6	8.6.2 . . . . .	47
4.3.7	8.6.3 . . . . .	47
4.3.8	8.6.4 . . . . .	48
4.3.9	8.6.5prealpha_devbuild_1 . . . . .	49
4.3.10	8.6.5prealpha_devbuild_2 . . . . .	51
4.3.11	8.6.5prealpha_devbuild_3 . . . . .	52
4.3.12	8.6.5alpha1 . . . . .	52
4.3.13	8.6.5alpha2 . . . . .	53
4.3.14	8.6.5 . . . . .	53
4.3.15	8.6.6 . . . . .	53
4.3.16	8.6.7alpha1 . . . . .	54
4.3.17	8.6.7 . . . . .	55
4.3.18	8.6.8prealpha_devbuild_1 . . . . .	55
4.3.19	8.6.8 . . . . .	56
4.3.20	8.6.9 . . . . .	58
4.3.21	8.6.10 . . . . .	58
4.3.22	8.6.11 . . . . .	58
4.3.23	8.6.12 . . . . .	60
4.3.24	8.6.13 . . . . .	60
4.3.25	8.6.14 . . . . .	60
4.3.26	8.6.15 . . . . .	61
4.3.27	8.6.16 . . . . .	61
4.3.28	8.6.17 . . . . .	61
<b>5</b>	<b>Upgrade Notes</b>	<b>62</b>
5.1	Upgrade Requirements . . . . .	62

<b>6</b>	<b>Compatibility Notes</b>	<b>63</b>
6.1	Changes in Windows Execution Host sgepasswd File . . . . .	63
6.2	Scheduler Log File . . . . .	63
6.3	Removed Scheduler-Parameter <code>queue_sort_method</code> . . . . .	63
6.4	Changes for <b>qconf</b> Exit States . . . . .	64
6.5	Changes for Scheduler Profiling . . . . .	64
6.6	Changed Limit Calculations . . . . .	64
6.7	New Default for Job Verification of DRMAA Submitted Jobs . . . . .	65
6.8	Default for Integer Complexes . . . . .	65
6.9	Deprecated Functionality . . . . .	66
6.10	Removed Functionality . . . . .	66
6.11	Changed UGERest configuration location . . . . .	66
6.12	Changed UGERest restlet framework . . . . .	66
6.13	Changed drmaa library path . . . . .	66
<b>7</b>	<b>Known Issues and Limitations</b>	<b>66</b>

# 1 License

## TERM SOFTWARE LICENSE AGREEMENT

This Agreement (defined below) is between the individual or entity agreeing to this Agreement and Univa Corporation, a Delaware corporation (Univa) with its registered office at 2300 N Barrington Road, Suite 400, Hoffman Estates, IL 60195.

BY ACCEPTING THIS AGREEMENT, EITHER BY CLICKING A BOX INDICATING YOUR ACCEPTANCE OR BY EXECUTING A QUOTATION THAT REFERENCES THIS AGREEMENT, YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU ARE ENTERING INTO THIS AGREEMENT ON BEHALF OF A COMPANY OR OTHER LEGAL ENTITY, YOU REPRESENT THAT YOU HAVE THE AUTHORITY TO BIND SUCH ENTITY TO THESE TERMS AND CONDITIONS, IN WHICH CASE "CUSTOMER" SHALL REFER TO SUCH ENTITY. IF YOU DO NOT HAVE SUCH AUTHORITY, OR IF YOU DO NOT AGREE WITH THESE TERMS AND CONDITIONS, THEN YOU MUST NOT ACCEPT THIS AGREEMENT AND MAY NOT USE THE UNIVA SOFTWARE.

1. SCOPE. This Agreement governs the licensing of the Univa Software and Support provided to Customer.
  - Univa Software means the software described in the quotation referencing this Agreement ("Quotation"), all updates and enhancements provided by Univa, Univa's user documentation for such software, and license keys (Univa Software). This Software is licensed and is not sold to Customer.
  - Third-Party Software/Open Source Software licensing terms are addressed at the bottom of this Agreement.
  - Capacity means the amount of computing resources on which Customer is licensed to use the Univa Software, as described on the applicable Quotation. Attached as Appendix A are definitions of the terms that may be used in a Quotation to define Capacity.
2. LICENSE. Subject to the terms of this Agreement, Univa grants Customer a non-exclusive, non-transferable, non-sub licensable, license, solely during the term of this Agreement and in accordance with (and not to exceed) the Capacity, to install and use the Univa Software on Customer-controlled systems (including Customer-controlled cloud computing environments) for Customer's internal business purposes. If Customer is installing the Univa Software, Univa further grants Customer the right to make a reasonable number of copies of the Univa Software for archival and backup purposes. Customer's contractors and majority owned affiliates are allowed to use the Univa Software under the terms of this Agreement. Customer is responsible for compliance with the terms of this Agreement.
3. RESTRICTIONS. Univa reserves all rights not expressly granted in Section 2 above. Without limiting the preceding sentence, Customer may not, and covenants that it will not, directly or indirectly, and may not authorize any third party to:
  - (a) Sell, transfer, assign, sublicense, or rent the Univa Software or use the Univa Software for the benefit of third parties (including through as a software service provider or in an outsourcing environment);

- (b) reverse engineer (except to the extent expressly permitted by applicable law despite this limitation), decompile, disassemble, modify, translate, or otherwise attempt to discover the source code of, the Univa Software;
- (c) except for a reasonable number of backup or archival copies, copy the Univa Software;
- (d) create derivative works from the Univa Software; or
- (e) circumvent or otherwise interfere with any authentication or security measures of the Univa Software.

#### 4. PROPRIETARY RIGHTS AND CONFIDENTIALITY.

- (a) **Proprietary Rights.** The Univa Software, workflow processes, designs, know-how and other technologies provided by Univa as part of the Univa Software are the proprietary property of Univa and its licensors, and all right, title and interest in and to such items, including all associated intellectual property rights, remain only with Univa. The Univa Software is protected by applicable copyright, trade secret, and other intellectual property laws. Customer may not remove any product identification, copyright, trademark or other notice from the Univa Software.
- (b) **Confidentiality.** Recipient may not disclose Confidential Information of Discloser to any third party or use the Confidential Information in violation of this Agreement.
- (c) **Confidential Information** means all proprietary or confidential information that is disclosed to the recipient (Recipient) by the discloser (Discloser), and includes, among other things:
  - any and all information relating to Univa Software or Support provided by a Discloser, its financial information, software code, flow charts, techniques, specifications, development and marketing plans, strategies, and forecasts;
  - as to Univa, the Univa Software and the terms of this Agreement (including without limitation, pricing information).
- (ii) **Confidential Information** excludes information that:
  - was rightfully in Recipient's possession without any obligation of confidentiality before receipt from the Discloser;
  - is or becomes a matter of public knowledge through no fault of Recipient;
  - is rightfully received by Recipient from a third party without violation of a duty of confidentiality;
  - is independently developed by or for Recipient without use or access to the Confidential Information; or
  - is licensed under an open source license.

Customer acknowledges that any misuse or threatened misuse of the Univa Software may cause immediately irreparable harm to Univa for which there is no adequate remedy at law. Univa may seek immediate injunctive relief in such event.

- 5. **PAYMENT.** Customer will pay all fees due under a Quotation within 30 days of the invoice date, plus applicable sales, use and other similar taxes.

6. WARRANTY DISCLAIMER. UNIVA DISCLAIMS ALL EXPRESS AND IMPLIED WARRANTIES, INCLUDING WITHOUT LIMITATION ANY WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE UNIVA SOFTWARE MAY NOT BE ERROR FREE, AND USE MAY BE INTERRUPTED.
7. TERM AND TERMINATION. This Agreement commences on the date it is accepted by or on behalf of Customer and expires as set forth on the applicable Quotation. This Agreement will be automatically renewed on its expiry for an additional one year period unless a written notification of termination has been received 60 days prior to term expiry. Either party may terminate this Agreement upon a material breach of the other party after a 30 day notice/cure period, if the breach is not cured during such time period. Upon termination of this Agreement or expiration of an order, Customer must discontinue using the Univa Software, de-install it and destroy or return the Univa Software and all copies, within 5 days. Upon Univa' request, Customer will provide written certification of such compliance.

SUPPORT INCLUDED. Univa's technical support and maintenance services (Support) is included with the fees paid under an order. Univa may change its Support terms, but Support will not materially degrade during any paid term. More details on Support are located at <http://www.univa.com/resources/files/support.pdf>

8. LIMITATION OF LIABILITY AND DISCLAIMER OF DAMAGES. IN NO EVENT WILL UNIVA BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, EXEMPLARY, PUNITIVE, TREBLE, OR CONSEQUENTIAL DAMAGES (INCLUDING, WITHOUT LIMITATION, LOSS OF BUSINESS, REVENUE, PROFITS, GOODWILL, DATA, OR ECONOMIC ADVANTAGE, AND COSTS OF SUBSTITUTE GOODS OR SERVICES) ARISING OUT OF OR RELATING TO THIS AGREEMENT, HOWEVER CAUSED, AND BASED ON ANY THEORY OF LIABILITY, WHETHER FOR BREACH OF CONTRACT, BREACH OF WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY, OR OTHERWISE, EVEN IF CUSTOMER IS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCEPT FOR CUSTOMER'S PAYMENT OBLIGATIONS, UNIVA'S TOTAL LIABILITY (INCLUDING ATTORNEYS' FEES) ARISING OUT OF OR RELATED TO THIS AGREEMENT WILL NOT EXCEED THE AMOUNT PAID BY CUSTOMER UNDER THE QUOTATION GIVING RISE TO THE CLAIM DURING THE 12-MONTH PERIOD PRIOR TO THE DATE THE CLAIM AROSE. THESE LIMITATIONS WILL APPLY NOTWITHSTANDING ANY FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED OR EXCLUSIVE REMEDY.
9. INTELLECTUAL PROPERTY INDEMNITY. If a third party claims that the Univa Software infringes that party's patent, copyright or other intellectual property right, then Univa will defend Customer against that claim at Univa' expense and pay all costs, damages, and attorney's fees, that a court finally awards or that are included in a settlement approved by Univa, provided that Customer:
  - (a) promptly notifies Univa in writing of the claim; and
  - (b) allows Univa to control, and cooperates with Univa in, the defense and any related settlement.

If Univa receives information about an infringement claim related to the Univa Software, then Univa will at its discretion and at no cost to Customer obtain a license to enable Customer to continue to use the Univa Software or modify the Univa Software so that it is no

longer infringing. If Univa determines that these alternatives are not reasonably available, Univa may terminate the license to the Univa Software and refund any unused fees.

Univa' obligations above do not apply if the infringement claim is based on Customer's breach of this Agreement, the use of the Univa Software in combination with products not supplied or approved by Univa in writing, or Customer's failure to use any updates made available by Univa.

This section contains Customer's exclusive remedies and Univa' sole liability for infringement claims.

10. GOVERNING LAW AND EXCLUSIVE FORUM. This Agreement is governed by the laws of the State of Illinois, without regard to conflict of law principles. Any dispute arising out of or related to this Agreement may only be brought in the state of Illinois. Customer consents to the personal jurisdiction of such courts and waives any claim that it is an inconvenient forum. The prevailing party in litigation is entitled to recover its attorneys' fees and costs from the other party.
11. MISCELLANEOUS.
  - (a) Inspection. Upon request by Univa, Customer must provide a usage report at least 60 days before the renewal anniversary. Univa, or its representative, may audit Customer's usage of the Univa Software at any Customer facility. Customer will cooperate with such audit. Customer agrees to pay within 30 days of written notification any fees applicable to Customer's use of the Univa Software in excess of the license.
  - (b) Entire Agreement. This agreement, which incorporates by reference Appendix A, and all Quotations referencing this agreement (collectively, the "Agreement"), constitutes the entire agreement between the parties, and supersedes all prior or contemporaneous negotiations, representations or agreements, whether oral or written, related to this subject matter.
  - (c) Modification Only in Writing. No modification or waiver of any term of this Agreement is effective unless signed by both parties.
  - (d) Non-Assignment. Neither party may assign or transfer this Agreement to a third party, except that the Agreement may be assigned upon notice as part of a merger, or sale of all or substantially all of the business or assets, of a party.
  - (e) Export Compliance. Customer must comply with all applicable export control laws of the United States, foreign jurisdictions and other applicable laws and regulations.
  - (f) US Government Restricted Rights. Univa provides the Univa Software, including related data and technology, for ultimate federal government end use solely in accordance with the following: The government hereby agrees that the software qualifies as "commercial" computer software. Government technical data and software rights related to the software include only those rights customarily provided to the public as defined in this Agreement. This customary commercial license is provided in accordance with FAR 12.211 (Technical Data) and FAR 12.212 (Software) and, for Department of Defense transactions, DFAR 252.227-7015 (Technical Data – Commercial Items) and DFAR 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation). If a government agency has a need for rights not granted under these terms, it must negotiate with Univa to determine if there are acceptable terms

for granting those rights, and a mutually acceptable written addendum specifically granting those rights must be included in any applicable agreement.

- (g) Independent Contractors. The parties are independent contractors with respect to each other.
- (h) Enforceability. If any term of this Agreement is invalid or unenforceable, the other terms remain in effect.
- (i) No PO Terms. Univa rejects additional or conflicting terms of a Customer's form-purchasing document.
- (j) No CISG. The United Nations Convention on Contracts for the International Sale of Goods does not apply.
- (k) Survival. All terms that by their nature survive termination or expiration of this Agreement, will survive.

Additional software specific licensing terms:

Grid Engine and Univa Software incorporate certain third-party software listed at the URL below. These licenses are accepted by use of the software and may represent license grants with restrictions which Univa is bound to provide. We are hereby notifying you of these licenses.

- Third Party Software means certain third-party software which is provided along with the Univa Software, and such software is licensed under the license terms located at: <http://www.univa.com/resources/licenses/>
- Open Source Software means certain opens source software which is provided along with the Univa Software, and such software is licensed under the license terms located at: <http://www.univa.com/resources/licenses/>

## APPENDIX A

### CAPACITY METRICS

The Univa Software is licensed against numbers of cores, vCPUs, Cloud-Instances or GPUs. The following are definitions of the terms that may be used in the Quotation to define "Capacity" for the purposes of the license grant of Univa Software.

**Cores** The number of "compute cores" as clearly advertised by the CPU vendor. Cores are often referred to as "physical cores" as they are clearly visible as repeated sections on the CPU die to differentiate them from any virtual processing units. **CPUs:** CPUs or Central Processing Units are the main computational chips for processing by bare-metal servers. Typical examples for CPUs are Intel x86 CPUs or AMD and ARM CPUs. **Bare-metal server:** A bare metal server is defined as a computer that runs a single copy of an operating system (e.g. Linux or Windows) which has only that single operating system between the user's applications and the computer hardware.

**vCPU** Virtual computational processing units (vCPUs) represents a share (or thread) of a physical CPU seen as a Core by the operating system running on a VM (Virtual Machine) which constitutes a Cloud Server. For the purpose of licensing, 1 Core is equivalent to 1 vCPU **Virtual Machines:** Each Virtual Machine that runs its own copy of an operating system and provides a view to application and end users that is almost undistinguishable from a bare-metal server with an operating system. Multiple copies of Virtual Machines can run on one bare-metal server.

**Cloud-Instance** A cloud-instance is the computational unit with a single copy of an operating system which customers rent from cloud computing providers. This can either be a virtual machine or a bare-metal server.

**GPU** Graphical Processing Units (GPUs) are add-on boards to bare-metal servers that enable acceleration of some class of applications by off-loading all or parts of the computation to a single or multiple GPUs being attached to the server.

Rev: Mar 2019

## 2 Supported Operating Systems, Versions and Architectures

Univa Grid Engine supports various platforms, hardware architectures and versions of operating systems. Find the full list in following table:

Operating System	Version	Architecture
SLES	11, 12	x86, x86-64
	15	x86-64
RHEL	6	x86, x86-64
	7	x86, x86-64, Power8/9
	8	x86-64
CentOS	6	x86, x86-64
	7	x86, x86-64, Power8/9
	8	x86-64
openSUSE Leap	42	x86-64, ARM64
Oracle Linux	6 or higher, 7	x86, x86-64
Ubuntu	10.04LTS - 18.04LTS	x86, x86-64
Oracle Solaris	10, 11	x86-64, SPARC 64bit
HP-UX	11.0 or higher	64bit
IBM AIX	7.1 or later	64bit
macOS	10.11 or higher	x86, x86-64
Microsoft Windows	XP Professional (SP3)	32 bit
Microsoft Windows	Server 2003 / 2003 R2	32 bit
Microsoft Windows	Vista Enterprise / Ultimate	32 and 64bit
Microsoft Windows	Server 2008 / 2008 R2	32 and 64bit
Microsoft Windows	7 Professional / Enterprise / Ultimate	32 and 64bit
Microsoft Windows	Server 2012 / 2012 R2	32 and 64bit
Microsoft Windows	8 / 8.1 Pro / Enterprise	32 and 64bit
Microsoft Windows	10 Pro / Enterprise	32 and 64bit

Table 1: Supported Operating Systems, Versions and Architectures

*PLEASE NOTE:* Hosts running the Microsoft Windows operating system cannot be used as master or shadow hosts.

*PLEASE NOTE:* Univa Grid Engine qmaster is fully supported on Linux and Solaris. We provide binaries in Univa Grid Engine for running the qmaster on other operating systems but they are not supported and delivered as a courtesy. If you require qmaster support on other architectures please contact us at [support@univa.com](mailto:support@univa.com).

*PLEASE NOTE:* if you require Univa Grid Engine support for older versions of the above operating systems please contact our sales or support team.

### 3 Supported and Tested Docker Versions

In principle, Univa Grid Engine supports these Docker versions on these Linux distributions:

Linux Distribution	Docker versions
RedHat, CentOS, Debian and Ubuntu	1.12.0 to 1.13.0 and 17.03.0-ce to 19.03.1-ce
SLES and openSUSE	1.12.0 to 1.13.0 and 17.03.0-ce to 19.03.1-ce

But in the past some Docker versions did not work properly and were withdrawn later. There were different Docker builds provided under the same version number showing a slightly different behavior, so it is not possible to ensure Univa Grid Engine does work with all Docker versions between 1.12.0 and 19.03.1. The following table shows which Docker versions were tested on which Linux distribution. The table includes the "Git commit" ID of the Docker source code, the Go language version which was used to build Docker, the build date and - for completeness - the version of the Docker RemoteAPI. Other Docker versions were not tested or are too broken to test successfully.

- Red Hat Enterprise Linux 7.4 (Maipo), Kernel 3.10.0-693.11.1.el7.x86\_64

Docker	Git commit	Go	Build date	API	containerd
1.12.0	8eab29e	go1.6.3		1.24	
1.12.1	23cf638	go1.6.3		1.24	
1.12.2	bb80604	go1.6.3		1.24	
1.12.3	6b644ec	go1.6.3		1.24	
1.12.4	1564f02	go1.6.4	Mon Dec 12 23:41:49 2016	1.24	
1.12.5	7392c3b	go1.6.4	Fri Dec 16 02:23:59 2016	1.24	
1.12.6	78d1802	go1.6.4	Tue Jan 10 20:20:01 2017	1.24	
1.13.0	49bf474	go1.7.3	Tue Jan 17 09:55:28 2017	1.25	
17.03.0-ce	3a232c8	go1.7.5	Tue Feb 28 08:10:07 2017	1.26	
17.03.1-ce	c6d412e	go1.7.5	Mon Mar 27 17:05:44 2017	1.27	
17.03.2-ce	f5ec1e2	go1.7.5	Tue Jun 27 02:21:36 2017	1.27	
17.06.2-ce	cec0b72	go1.8.3	Tue Sep 5 20:00:25 2017	1.30	
17.07.0-ce	8784753	go1.8.3	Tue Aug 29 17:43:23 2017	1.31	
17.09.0-ce	afdb6d4	go1.8.3	Tue Sep 26 22:42:49 2017	1.32	
17.09.1-ce	19e2cf6	go1.8.3	Thu Dec 7 22:25:03 2017	1.32	

Docker	Git commit	Go	Build date	API	containerd
17.12.0-ce	c97c6d6	go1.9.2	Wed Dec 27 20:12:46 2017	1.35	
18.03.0-ce	0520e24	go1.9.4	Wed Mar 21 23:13:03 2018	1.37	
18.03.1-ce	9ee9f40	go1.9.5	Thu Apr 26 07:23:58 2018	1.37	
18.06.0-ce	0ffa825	go1.10.3	Wed Jul 18 19:10:42 2018	1.38	
18.06.1-ce	e68fc7a	go1.10.3	Tue Aug 21 17:25:29 2018	1.38	
18.06.2-ce	6d37f41	go1.10.3	Sun Feb 10 03:48:29 2019	1.38	
18.06.3-ce	d7080c1	go1.10.3	Wed Feb 20 02:28:17 2019	1.38	
18.09.0	4d60db4	go1.10.4	Wed Nov 7 00:19:08 2018	1.39	1.2.0
18.09.1	4c52b90	go1.10.6	Wed Jan 9 19:06:30 2019	1.39	1.2.0
18.09.2	6247962	go1.10.6	Sun Feb 10 03:47:25 2019	1.39	1.2.0
18.09.3	774a1f4	go1.10.8	Thu Feb 28 06:02:24 2019	1.39	1.2.0
19.03.0	aeac9490dc	go1.12.5	Wed Jul 17 18:14:16 2019	1.40	1.2.2
19.03.1	74b1e89	go1.12.5	Thu Jul 25 21:19:36 2019	1.40	1.2.2

- CentOS 7, Kernel 3.10.0-693.2.2.el7\_x86\_64

Docker	Git commit	Go	Build date	API	containerd
1.12.0	8eab29e	go1.6.3		1.24	
1.12.1	23cf638	go1.6.3		1.24	
1.12.2	bb80604	go1.6.3		1.24	
1.12.3	6b644ec	go1.6.3		1.24	
1.12.4	1564f02	go1.6.4	Mon Dec 12 23:41:49 2016	1.24	
1.12.5	7392c3b	go1.6.4	Fri Dec 16 02:23:59 2016	1.24	
1.12.6	78d1802	go1.6.4	Tue Jan 10 20:20:01 2017	1.24	
1.13.0	49bf474	go1.7.3	Tue Jan 17 09:55:28 2017	1.25	
1.13.1	092cba3	go1.7.5	Web Feb 8 06:38:28 2017	1.26	
17.03.0-ce	3a232c8	go1.7.5	Tue Feb 28 08:10:07 2017	1.26	
17.03.1-ce	c6d412e	go1.7.5	Mon Mar 27 17:05:44 2017	1.27	
17.03.2-ce	f5ec1e2	go1.7.5	Tue Jun 27 02:21:36 2017	1.27	
17.06.2-ce	cec0b72	go1.8.3	Tue Sep 5 20:00:25 2017	1.30	
17.07.0-ce	8784753	go1.8.3	Tue Aug 29 17:43:23 2017	1.31	
17.09.0-ce	afdb6d4	go1.8.3	Tue Sep 26 22:42:49 2017	1.32	
17.09.1-ce	19e2cf6	go1.8.3	Thu Dec 7 22:25:03 2017	1.32	

Docker	Git commit	Go	Build date	API	containerd
17.12.0-ce	c97c6d6	go1.9.2	Wed Dec 27 20:12:46 2017	1.35	
18.03.0-ce	0520e24	go1.9.4	Wed Mar 21 23:13:03 2018	1.37	
18.03.1-ce	9ee9f40	go1.9.5	Thu Apr 26 07:23:58 2018	1.37	
18.06.0-ce	0ffa825	go1.10.3	Wed Jul 18 19:10:42 2018	1.38	
18.06.1-ce	e68fc7a	go1.10.3	Tue Aug 21 17:25:29 2018	1.38	
18.06.2-ce	6d37f41	go1.10.3	Sun Feb 10 03:48:29 2019	1.38	
18.06.3-ce	d7080c1	go1.10.3	Wed Feb 20 02:28:17 2019	1.38	
18.09.0	4d60db4	go1.10.4	Wed Nov 7 00:19:08 2018	1.39	1.2.0
18.09.1	4c52b90	go1.10.6	Wed Jan 9 19:06:30 2019	1.39	1.2.0
18.09.2	6247962	go1.10.6	Sun Feb 10 03:47:25 2019	1.39	1.2.0
18.09.3	774a1f4	go1.10.8	Thu Feb 28 06:02:24 2019	1.39	1.2.0
19.03.0	aeac9490dc	go1.12.5	Wed Jul 17 18:14:16 2019	1.40	1.2.2
19.03.1	74b1e89	go1.12.5	Thu Jul 25 21:19:36 2019	1.40	1.2.2

Note: The version 1.13.1 is the one provided by Docker. The version 1.13.1-63 provided by CentOS is broken and NOT supported by Univa Grid Engine!

- Ubuntu 16.04.3 LTS, Kernel 4.4.0-103-generic x86\_64

Docker	Git commit	Go	Build date	API	containerd
1.12.0	8eab29e	go1.6.3	Thu Jul 28 22:11:10 2016	1.24	
1.12.1	23cf638	go1.6.3	Thu Aug 18 05:33:38 2016	1.24	
1.12.2	bb80604	go1.6.3	Tue Oct 11 18:29:41 2016	1.24	
1.12.3	6b644ec	go1.6.3	Wed Oct 26 22:01:48 2016	1.24	
1.12.4	1564f02	go1.6.4	Tue Dec 13 00:08:34 2016	1.24	
1.12.5	7392c3b	go1.6.4	Fri Dec 16 02:42:17 2016	1.24	
1.12.6	78d1802	go1.6.4	Tue Jan 10 20:38:45 2017	1.24	
1.13.0	49bf474	go1.7.3	Tue Jan 17 09:58:26 2017	1.25	
17.03.0-ce	3a232c8	go1.7.5	Tue Feb 28 08:01:32 2017	1.26	
17.03.1-ce	c6d412e	go1.7.5	Mon Mar 27 17:14:09 2017	1.27	
17.03.2-ce	f5ec1e2	go1.7.5	Tue Jun 27 03:35:14 2017	1.27	
17.06.2-ce	cec0b72	go1.8.3	Tue Sep 5 19:59:11 2017	1.30	
17.09.0-ce	afdb6d4	go1.8.3	Tue Sep 26 22:40:56 2017	1.32	

Docker	Git commit	Go	Build date	API	containerd
17.09.1-ce	19e2cf6	go1.8.3	Thu Dec 7 22:23:00 2017	1.32	
17.12.0-ce	c97c6d6	go1.9.2	Wed Dec 27 20:09:53 2017	1.35	
18.03.0-ce	0520e24	go1.9.4	Wed Mar 21 23:08:31 2018	1.37	
18.03.1-ce	9ee9f40	go1.9.5	Thu Apr 26 07:15:30 2018	1.37	
18.06.0-ce	0ffa825	go1.10.3	Wed Jul 18 19:09:05 2018	1.38	
18.06.1-ce	e68fc7a	go1.10.3	Tue Aug 21 17:23:21 2018	1.38	
18.06.2-ce	6d37f41	go1.10.3	Sun Feb 10 03:46:30 2019	1.38	
18.06.3-ce	d7080c1	go1.10.3	Wed Feb 20 02:26:20 2019	1.38	
18.09.0	4d60db4	go1.10.4	Wed Nov 7 00:16:44 2018	1.39	1.2.0
18.09.1	4c52b90	go1.10.6	Wed Jan 9 19:02:44 2019	1.39	1.2.0
18.09.2	6247962	go1.10.6	Sun Feb 10 03:42:13 2019	1.39	1.2.2
18.09.3	774a1f4	go1.10.8	Thu Feb 28 05:59:55 2019	1.39	1.2.2
19.03.0	aeac949	go1.12.5	Wed Jul 17 18:14:42 2019	1.40	1.2.2
19.03.1	74b1e89e8a	go1.12.5	Thu Jul 25 21:20:09 2019	1.40	1.2.2

- Ubuntu 16.10, Kernel 4.8.0-59-generic x86\_64

Docker	Git commit	Go	Build date	API	containerd
1.13.0	49bf474	go1.7.3	Tue Jan 17 10:05:19 2017	1.25	
17.03.0-ce	3a232c8	go1.7.5	Tue Feb 28 08:05:01 2017	1.26	
17.03.1-ce	c6d412e	go1.7.5	Mon Mar 27 17:17:43 2017	1.27	
17.03.2-ce	f5ec1e2	go1.7.5	Tue Jun 27 03:59:22 2017	1.27	

- Ubuntu 17.04, Kernel 4.10.0-42-generic x86\_64

Docker	Git commit	Go	Build date	API	containerd
17.09.0-ce	afdb6d4	go1.8.3	Tue Sep 26 22:41:24 2017	1.32	
17.12.0-ce	c97c6d6	go1.9.2	Wed Dec 27 20:09:19 2017	1.35	

- Ubuntu 17.10, Kernel 4.13.0-19-generic x86\_64

Docker	Git commit	Go	Build date	API	containerd
17.06.2-ce	cec0b72	go1.8.3	Tue Sep 5 19:57:44 2017	1.30	
17.09.0-ce	afdb6d4	go1.8.3	Tue Sep 26 22:41:24 2017	1.32	
17.09.1-ce	19e2cf6	go1.8.3	Thu Dec 7 22:23:07 2017	1.32	
17.12.0-ce	c97c6d6	go1.9.2	Wed Dec 27 20:09:47 2017	1.35	

- Ubuntu 18.04, Kernel 4.15.0-24-generic x86\_64

Docker	Git commit	Go	Build date	API	containerd
17.12.0-ce	c97c6d6	go1.9.2	Wed Dec 27 20:09:47 2017	1.35	
18.03.0-ce	0520e24	go1.9.4	Wed Mar 21 23:08:36 2018	1.37	
18.03.1-ce	9ee9f40	go1.9.5	Thu Apr 26 07:15:45 2018	1.37	

- Ubuntu 18.10, Kernel 4.18.0-15-generic x86\_64

Docker	Git commit	Go	Build date	API	containerd
18.06.0-ce	Offa825	go1.10.3	Wed Jul 18 19:07:56 2018	1.38	
18.06.1-ce	e68fc7a	go1.10.3	Tue Aug 21 17:23:15 2018	1.38	
18.06.2-ce	6d37f41	go1.10.3	Sun Feb 10 03:46:20 2019	1.38	
18.06.3-ce	d7080c1	go1.10.3	Wed Feb 20 02:26:34 2019	1.38	
18.09.0	4d60db4	go1.10.4	Wed Nov 7 00:16:44 2018	1.39	1.2.0
18.09.1	4c52b90	go1.10.6	Wed Jan 9 19:02:44 2019	1.39	1.2.2
18.09.2	6247962	go1.10.6	Sun Feb 10 03:42:13 2019	1.39	1.2.2
18.09.3	774a1f4	go1.10.8	Thu Feb 28 05:59:55 2019	1.39	1.2.2
19.03.0	aeac9490dc	go1.12.5	Wed Jul 17 18:13:27 2019	1.40	1.2.2
19.03.1	74b1e89	go1.12.5	Thu Jul 25 21:19:53 2019	1.40	1.2.2

- openSUSE Leap 42.3, Kernel 4.4.92-31-default x86\_64

Docker	Git commit	Go	Build date	API	containerd
1.12.0	8eab29e	go1.6.3		1.24	
1.12.1	23cf638	go1.6.3		1.24	
1.12.2	bb80604	go1.6.3		1.24	

---

Docker	Git commit	Go	Build date	API	containerd
1.12.3	6b644ec	go1.6.3		1.24	
1.12.4	1564f02	go1.6.4	Mon Dec 12 23:41:28 2016	1.24	
1.12.5	7392c3b	go1.6.4	Fri Dec 16 02:24:38 2016	1.24	
1.12.6	78d1802	go1.6.4	Tue Jan 10 20:20:13 2017	1.24	
1.13.0	49bf474	go1.7.3	Tue Jan 17 10:00:08 2017	1.25	
17.03.0-ce	60ccb22	go1.7.5	Thu Feb 23 10:55:03 2017	1.26	
17.03.1-ce	c6d412e	go1.7.5	Fri Mar 24 00:53:12 2017	1.27	
17.09.1-ce	f4ffd25	go1.8.7	Tue Jun 12 12:05:08 2018	1.32	

---

### 3.1 Known Docker Issues That Affect Univa Grid Engine

- With Docker 17.09.0-ce and Docker 17.12.0-ce, specifying the **-oom-kill-disable** switch has no effect, when using the **docker** command line client nor the **-xd "-oom-kill-disable"** switch of the Univa Grid Engine submit clients.

## 4 Fixes and Enhancements

### 4.1 Major Enhancements

#### 4.1.1 (8.6.14) Upgrade Notes

This release prevents persistency corruption issues and improves the performance for high scalability and high throughput configurations. It is highly recommended to upgrade to this release as soon as possible.

The upgrade from 8.6.x clusters can be done with an operational cluster (“hot upgrade”). Details concerning the hot upgrade can be found in the support portal. Please note that an upgrade procedure needs to be executed after shutdown of the old cluster and before the new daemons are started. A reference to the update script as well as instructions how to use it can be found in the: [Hot Upgrade Procedure](#).

Customers upgrading from Univa Grid Engine 8.5.x or older need to follow the instructions in chapter 4 of the installation manual.

#### 4.1.2 (IMPORTANT) UGE Product Parts and Features Declared as Deprecated

- Support for Java and TCL Job Submission Verifiers will be dropped and corresponding scripts will be removed from the product packages beginning with 8.7. Bourne Shell, Python and Perl JSVs will be available and fully supported. Examples for corresponding JSV scripts can be found in \$SGE\_ROOT/util/resources/jsv.
- The Hadoop package will not be part of the UGE distribution beginning with 8.7.

Please note that all mentioned product parts and features are still available with this patch release and will also be available for future patch releases of 8.6.

#### 4.1.3 (UGE 8.6.11) Resolving additional group id's and definable ranges

Univa Grid Engine 8.6.11 introduces the following new features to optimize performance that allow to reduce the load on configured user/group resolving mechanisms like NIS/LDAP/Active Directory/.. :

UGE clients which do not require supplementary group ids, will never resolve them and never send them to qmaster. Following is the list of clients which do not require them: qstat, qrstat, qhost, qacct, qquota, qevent, sge\_passwd, lostat, loacct, and loquota.

A new environment variable, SGE\_SUP\_GRP\_EVAL=0 is introduced which suppresses sending to qmaster, and thereby not resolving of supplementary groups for all UGE clients.

A new `execd_param`, SUP\_GRP\_EVAL=0 results in writing environment variable SGE\_SUP\_GRP\_EVAL=0 for the shepherd. When set to -1 (default), `qmaster_params ENABLE_SUP_GRP_EVAL=FALSE` takes precedence of writing SGE\_SUP\_GRP\_EVAL=0 to the environment file. No environment variable will be written if `ENABLE_SUP_GRP_EVAL=TRUE`.

A new environment variable, `SGE_DISABLE_SUP_GRP_EVAL_RANGE=<range>` is introduced, where `<range>` can be any normal or open-ended gid range. These gids won't be sent to the qmaster. To reduce resolving overhead of unresolvable addgrpid for UGE commands issued withing UGE job, `execd` always writes addgrpid to shepherds environment file. For open-ended ranges, the max value considered is 9,999,999.

A new `execd_param`, `DISABLE_SUB_GRP_EVAL=<range>` is introduced which combines addgrpid and the provided range, writing them to the shepherds environment file as `SGE_DISABLE_SUP_GRP_EVAL_RANGE`.

#### 4.1.4 (UGE 8.6.8) Possibility to disable qinstance access checks during submission

If a user tries to submit a job into a cluster that either has no qinstances defined (cluster queue exists but host list or referenced host groups are empty) or where the submitting user has no access to any of the existing queue instances then that job is automatically rejected. To disable this behaviour the `qmaster_param` named `DISABLE_QINSTANCE_CHECK` can be set to `true` or `1`.

#### 4.1.5 (UGE 8.6.8) New Scheduler Parameters for Tuning Wildcard PE Scheduling

Three new scheduler parameters have been added for tuning the scheduling of jobs with wildcard PE requests. Without special tuning the scheduler will always try to find the best matching resource assignment. For example, the PE providing most slots of a PE range request, the assignment with most soft requests being granted or the reservation with the earliest job start time. Therefore the scheduling algorithm will go over all possible matching PEs and try to find the best one which can result in high scheduling times.

The following parameters can be used to define that possible but not optimal scheduling results will be accepted which can significantly reduce scheduling times (see also `sched_conf.5`):

- `PE_ACCEPT_FIRST_ASSIGNMENT`: The first valid assignment or reservation will be used. This parameter is best combined with `PE_SORT_ORDER=CAPACITY`, see below.
- `PE_ACCEPT_ASSIGNMENT_TIME`: When an assignment has been found the search for better resources will be stopped after the time specified by `PE_ACCEPT_ASSIGNMENT_TIME`.
- `PE_ACCEPT_SOFT_VIOLATIONS`: The scheduling algorithm will stop searching for better resources when the assignment found so far has not more than the specified number of soft violations (number of soft requests which cannot be granted).

#### 4.1.6 (UGE 8.6.8) New PE\_SORT\_ORDER to Improve Scheduling Time

The new value `CAPACITY` is available for the scheduler parameter `PE_SORT_ORDER`.

When it is set, the parallel environment list is sorted before dispatching/reservation scheduling of every parallel job:

- for dispatching the parallel environment list is sorted by free slots descending.

- for reservation scheduling the parallel environment list is sorted by the time of the last entry in the resource diagram ascending. This is the time where reservation scheduling can be guaranteed.

PE\_SORT\_ORDER=CAPACITY is best combined with PE\_ACCEPT\_FIRST\_ASSIGNMENT=TRUE and can have significant performance impact for specific scenarios.

#### 4.1.7 (UGE 8.6.8) Support for Linux Mount Namespace

Starting with Univa Grid Engine 8.6.8, it is possible to make use of Linux mount namespaces in the following way:

- create a base tmpdir
- create a hostgroup containing the Linux hosts that support Linux mount namespaces (unshare() system call)
- create a queue that defines a common base tmpdir using the Linux mount namespace syntax extension (see queue\_conf.5)
- run a corresponding test job

A possible test job script can be found under `$SGE_ROOT/util/resources/lns/mytestscript.sh`. In the job output file you can verify the corresponding bind mount and mount namespace settings of the job.

#### 4.1.8 (UGE 8.6.8) Rework of profiling for all threads and execution daemon

The profiling functionality for threads in `sge\_qmaster(8)` and `sge\_execd(8)` has been revised. It is now possible to set an output time by using the newly introduced `PROF_TIME` configuration parameter. It is also possible to enable profiling for the reader threads with `PROF_READER` configuration parameter. More information can be found in `sge_conf(5)` man page. The profiling output now also contains the proportion of the level specific wall clock time on the total wall clock time (see also `sge_diagnostics(5)` man page).

#### 4.1.9 (UGE 8.6.8) GPU Usage of Jobs Reported by DCGM

With UGE 8.6.8 GPU usage reporting can be enabled for Nvidia GPUs.

Prerequisite is that the `nv-hostengine` is running on the execution host and communication between `sge_execd` and the `nv-hostengine` is enabled by configuring the `execd_param` `UGE_DCGM_PORT` in the global or local configuration, see man page `sge_conf(5)`. Once `UGE_DCGM_PORT` is set GPU specific load values are reported by `sge_execd`, see [Integration with Nvidia DCGM](#).

Per Job usage can be enabled by adding the `execd_param` `ENABLE_DCGM_JOB_USAGE=TRUE`. By default it is disabled.

The following usage values can be reported per GPU and be part of the usage output of `qstat -j job_id`:

usage value	description
cuda.<gpu>.boardLimitViolationTime	Amount of seconds we were at reduced clocks due to being at the board's max voltage
cuda.<gpu>.eccDoubleBit	Count of ECC double bit errors that occurred
cuda.<gpu>.eccSingleBit	Count of ECC single bit errors that occurred
cuda.<gpu>.energyConsumed	Energy consumed by the gpu in Joules
cuda.<gpu>.lowUtilizationTime	Amount of seconds we were at reduced clocks due to low utilization
cuda.<gpu>.maxGpuMemoryUsed	Maximum amount of GPU memory that was used in MB
cuda.<gpu>.memoryClock_min	minimum Memory clock in MHz
cuda.<gpu>.memoryClock_max	maximum Memory clock in MHz
cuda.<gpu>.memoryClock_avg	average Memory clock in MHz
cuda.<gpu>.memoryUtilization_min	minimum GPU Memory Utilization in percent
cuda.<gpu>.memoryUtilization_max	maximum GPU Memory Utilization in percent
cuda.<gpu>.memoryUtilization_avg	average GPU Memory Utilization in percent
cuda.<gpu>.numOtherComputePids	Count of otherComputePids entries that are valid
cuda.<gpu>.numOtherGraphicsPids	Count of otherGraphicsPids entries that are valid
cuda.<gpu>.numXidCriticalErrors	Number of valid entries in xidCriticalErrorsTs
cuda.<gpu>.overallHealth	The overall health of the system
cuda.<gpu>.pcieReplays	Count of PCI-E replays that occurred
cuda.<gpu>.pcieRxBandwidth_min	minimum PCI-E MB read from the GPU
cuda.<gpu>.pcieRxBandwidth_max	maximum PCI-E MB read from the GPU
cuda.<gpu>.pcieRxBandwidth_avg	average PCI-E MB read from the GPU
cuda.<gpu>.pcieTxBandwidth_min	minimum PCI-E MB written to the GPU
cuda.<gpu>.pcieTxBandwidth_max	maximum PCI-E MB written to the GPU
cuda.<gpu>.pcieTxBandwidth_avg	average PCI-E MB written to the GPU
cuda.<gpu>.powerViolationTime	Number of seconds we were at reduced clocks due to power violation
cuda.<gpu>.processUtilization_sm	Process SM and Memory Utilization (in percent)
cuda.<gpu>.processUtilization_mem	Process SM and Memory Utilization (in percent)
cuda.<gpu>.reliabilityViolationTime	Amount of seconds we were at reduced clocks due to the reliability limit
cuda.<gpu>.smClock_min	minimum SM clock in MHz
cuda.<gpu>.smClock_max	maximum SM clock in MHz

usage value	description
cuda.<gpu>.smClock_avg	average SM clock in MHz
cuda.<gpu>.smUtilization_min	minimum GPU SM Utilization in percent
cuda.<gpu>.smUtilization_max	maximum GPU SM Utilization in percent
cuda.<gpu>.smUtilization_avg	average GPU SM Utilization in percent
cuda.<gpu>.syncBoostTime	Amount of seconds we were at reduced clocks due to sync boost
cuda.<gpu>.thermalViolationTime	Number of seconds we were at reduced clocks due to thermal violation

Which values to report is defined via the global/local configuration attribute *gpu\_job\_usage*. The following values can be configured, default is *none*.

setting	description
<i>none</i>	no gpu usage values will be reported
<i>all</i>	all implemented usage values will be reported, see the table above
<variable list>	comma or space separated list of variable names, without <i>cuda.&lt;gpu&gt;</i> e.g. <i>eccDoubleBit, maxGpuMemoryUsed, memoryClock_max, ...</i>

#### 4.1.10 (UGE 8.6.8) Automatically Adding All Job Owners Supplementary Group IDs to a Docker Job

Using the special keyword **SGE\_SUP\_GRP\_EVAL** with the **-group-add** option of the **-xd** switch during job submission allows to automatically add all supplementary groups to the group list of the job user inside of the Docker container. Additional group IDs can be specified by using the **-group-add** option several times.

E.g.:

```
# qsub -l docker,docker_images="*some_image*", -xd "-group-add SGE_SUP_GRP_EVAL,-group-add 789" ...
```

makes Univa Grid Engine add all additional group IDs of the job owner **on the execution host** as well as the group id 789.

#### 4.1.11 (UGE 8.6.7) DRMAA2 Related Changes and Enhancements

With UGE 8.6.7 DRMAA2 reflects the failed reason field that is also visible in qacct. To get this information the implementation specific attribute *uge\_ji\_failed* has to be requested via JobInfo.

#### 4.1.12 (UGE 8.6.7) Official Red Hat 8 Support

Univa Grid Engine supports Red Hat Enterprise Linux 8. However, Red Hat Enterprise Linux 8 does not officially support Docker, but Docker CE can be installed using the official repository for Cent OS 7:

Install `yum-config-manager` and add the `docker-ce` repository:

```
$ sudo yum install yum-utils
$ sudo yum-config-manager --add-repo
    https://download.docker.com/linux/centos/docker-ce.repo
```

Once the repository is added, `yum list docker-ce` lists the available Docker CE versions:

```
$ yum list docker-ce --showduplicates | sort -r
docker-ce.x86_64          3:18.09.2-3.el7          docker-ce-stable
docker-ce.x86_64          3:18.09.1-3.el7          docker-ce-stable
docker-ce.x86_64          3:18.09.0-3.el7          docker-ce-stable
...
```

Now Docker CE can be installed and enabled:

```
$ sudo yum install docker-ce
$ sudo systemctl enable --now docker
```

#### 4.1.13 (UGE 8.6.5) Support for Linux Mount Namespace (obsoleted, see above under UGE 8.6.8)

Starting with Univa Grid Engine 8.6.5 it is possible to make use of Linux mount namespaces in prolog and epilog scripts in the following way:

- create a shared base `tmpdir`
- enable `LINUX_MOUNT_NAMESPACE` `execd` parameter.
- create a hostgroup containing the linux hosts that support Linux mount namespaces (`unshare()` system call)
- create a queue that defines a common base `tmpdir`, prolog and epilog and use the hostgroup defined above
- run a corresponding test job

Further setup information can be found in `$SGE_ROOT/util/resources/lns/README` and examples for prolog, epilog and testing scripts are located there as well.

#### 4.1.14 (UGE 8.6.3) Bulk Changes for Exec Hosts and Projects

Beginning with UGE 8.6.3 `qconf` as well as the config API will support bulk operations for the exec host and project object. Bulk operations allow to get, add, modify or delete multiple configuration objects with one UGE operation. Internal communication overhead for those operations will be reduced compared to multiple individual operations.

`qconf` now supports following additional switches:

- `qconf -seld [execlist]` shows a detailed list of all exec hosts or hosts in `execlist`
- `qconf -sprjld [projectlist]` show a detailed list of all projects or projects in `projectlist`
- `qconf -Ae dirname` adds exec hosts from every file in directory `dirname`
- `qconf -Me dirname` modifies exec hosts from every file in directory `dirname`
- `qconf -Aprj dirname` adds projects form every file in directory `dirname`
- `qconf -Mprj dirname` modifies projects from every file in directory `dirname`

#### 4.1.15 New Core Binding Strategies and PE Behavior

For PE-jobs it is hard or even impossible to know in advance how many tasks are going to be scheduled and on how many hosts. Therefore, with Univa Grid Engine version 8.6.0, the core-binding request behavior has changed to accommodate this fact. The binding-request is now changed to a “per PE-task” request where previously it was a “per host” request. This means that the requested amount of cores for a PE-job are assigned “per PE-task”.

For example, if a job with

```
qsub -pe mype 5-7 -binding linear:2 -b y sleep 60
```

is submitted, it means that each PE-task will get 2 cores, no matter on which host or on how many hosts the tasks are scheduled.

There are now different binding-strategies, most of them exist in two versions: “host aware” and “host unaware” strategies. For example, there are two versions of linear binding strategies: `linear` and `linear_per_task`. Host unaware strategies have the suffix “`_per_task`”.

With “host aware” strategies, all tasks that end up on a host have to adhere to the given strategy together. For “host unaware” strategies, each task has to adhere to the strategy on its own. This is less strict and usually more tasks can fit on a host.

All available core binding strategies are:

```
linear:<amount>[:<socket>,<core>]
linear_per_task:<amount>
striding:<amount>:<n>[:<socket>,<core>]
striding_per_task:<amount>:<n>
explicit:[<socket>,<core>;...]<socket>,<core>
explicit_per_task:[<socket>,<core>;...]<socket>,<core>
```

```

balance_sockets:<amount>
pack_sockets:<amount>
one_socket_balanced:<amount>
one_socket_per_task:<amount>

```

See man page `submit(5)` for more details and examples.

#### 4.1.16 Affinity Based Job Placement

Univa Grid Engine 8.6.0 adds an affinity job placement policy that allows assignment to each host or queue an affinity value for each consumed resource of jobs that are running on the host or queue.

Affinity can be positive or negative. Positive affinity will attract other pending jobs, negative affinity will reject other pending jobs. Attraction/rejection will work on host and/or queue level if this is enabled by setting the weighting parameters `weight_host_affinity` and/or `weight_queue_affinity`.

Sorting based on this affinity value will cause

- affinity (so that jobs are packed on clusters of hosts or queues),
- anti-affinity (so that jobs are distributed on hosts in the cluster or queues residing on hosts)
- or best fit (if a mixture of positive and negative affinity values is defined for different resources)

Find more information concerning job placement according to affinity values in the section 3.7.1 (Host/Queue Sorting) of the Admin Guide.

#### 4.1.17 Managing Access to Devices with RSMAPs

Univa Grid Engine 8.6.0 allows to manage access to host devices via RSMAPs. Each id of a RSMAP complex can be configured to represent a device on the host by setting the new parameter "device". Each device can be represented by more than one RSMAP id. In the example below a RSMAP complex **gpu** is initialized with two ids and each id is mapped to a Nvidia GPU:

```
complex_values    gpu=2(gpu0[device=/dev/nvidia0] gpu1[device=/dev/nvidia1])
```

The assigned devices are shown in the `qstat` output of a job:

```
granted devices  host: /dev/nvidia0
```

In a default environment the configuration and assignment of devices has no effect on the scheduling, but if `cgroups` are available the `cgroups` parameter "devices" can be set to a list of devices that should be managed by Univa Grid Engine. Read/write access to all devices in the list will be blocked via `cgroups` and jobs will only be able to access devices that were assigned to them via RSMAPs. With the following configuration Univa Grid Engine will manage access to all Nvidia GPUs (i.e. all devices from `/dev/nvidia0` to `/dev/nvidia254`):

```
cgroups_params cgroup_path=/sys/fs/cgroups devices=/dev/nvidia[0-254]
```

#### 4.1.18 Integration with Nvidia DCGM

Univa Grid Engine 8.6.0 is integrated with NVIDIA's Data Center GPU Manager (DCGM) that provides detailed information about GPU resources.

##### Supported Versions

Univa Grid Engine supports the following DCGM versions:

Univa Grid Engine version	Supported DCGM version
8.6.0 to 8.6.14	1.3.3 to 1.7.2
8.6.15 and newer	1.5.6 to 2.0.10

Newer DCGM versions may work with Univa Grid Engine without any problems, but they were not tested and are not officially supported.

##### DCGM Load Values

Support for DCGM can be enabled on host level by setting the `execd` parameter **UGE\_DCGM\_PORT** to the port DCGM uses to communicate on the specific host. If DCGM is running Univa Grid Engine will automatically retrieve load values for the installed and supported GPU devices from DCGM. For each available device the load values are reported in the format **cuda.<cuda\_id>.<attribute>=<value>** and are visible via `qconf -se`:

```
cuda.0.affinity=SCTTCTTCTTCTTCTTCTTCTTCTTCTTcttcttcttcttcttcttctt,
cuda.0.gpu_temp=36,
cuda.0.mem_free=16280.000000M,
cuda.0.mem_total=16280.000000M,
cuda.0.mem_used=0.000000M,
cuda.0.name=Tesla P100-PCIE-16GB,
cuda.0.power_usage=28.527000,
cuda.0.verstr=390.46,
cuda.1.affinity=SccttcttcttcttcttcttcttcttSCTTCTTCTTCTTCTTCTTCTT,
cuda.1.gpu_temp=40,
cuda.1.mem_free=16160.000000M,
cuda.1.mem_total=16160.000000M,
cuda.1.mem_used=0.000000M,
cuda.1.name=Tesla V100-PCIE-16GB,
cuda.1.power_usage=27.298000,
cuda.1.verstr=390.46,
cuda.devices=2
```

If RSMAP complexes are used to manage GPU devices, each RSMAP id can be mapped to a CUDA device with the new parameter **cuda\_id**. The reported load values can then be used by UGE during the scheduling of GPU devices:

```
complex_values gpu=2(gpu0[device=/dev/nvidia0,cuda_id=0] \
                    gpu1[device=/dev/nvidia1,cuda_id=1])
```

Please note that up to Univa Grid Engine 8.6.7 DCGM load values contained the host name in the format **<hostname>.cuda.<cuda\_id>.<attribute>=<value>**.

### GPU-CPU-Affinity

If DCGM is enabled, Univa Grid Engine 8.6.0 allows requesting the special load value **affinity**. If a job requests a GPU and **affinity**, it will automatically be bound to the cores that have a good affinity to the assigned GPU. This ensures that the data between the CPU and GPU is transferred in the fastest way possible. **affinity** can be requested as hard request with `-l gpu=1[affinity=1]` or `-l gpu=1[affinity=true]`, or as soft request with `-l gpu=1[affinity=2]`. If it is requested as hard request, the job will not be scheduled if Univa Grid Engine cannot find enough CPU cores needed for a GPU device. If it is requested as a soft request, Univa Grid Engine will try to schedule the job with a GPU/CPU combination with good affinity and schedule the job anyway without binding any CPU cores, if it cannot find a suitable GPU/CPU combination. If less cores are needed the request can be combined with the **-binding** switch.

The following requests a GPU device and binds the cores that have a good affinity to the assigned GPU:

```
% qsub -l gpu=1[affinity=true] -b y sleep 1000
```

### 4.1.19 Performance Improvements

Performance and Scalability of Univa Grid Engine has been improved in the following areas:

- general improvements affecting all components by optimizations of the Univa Grid Engine data store and updated memory allocator (jemalloc library)
- scheduler optimizations for various job profiles, esp. for parallel jobs with and without resource reservation
- higher throughput of the communication library by introducing an additional thread pool for handling incoming/outgoing data
- compression of communication data can reduce network load and avoid network saturation on slow network connections
- the impact of running many `qsub -sync`, `drmaa` or `drmaa2` clients on `sge_qmaster` has been reduced

### 4.1.20 Reworked Dispatch Job Information

The mechanisms that provide Dispatch Job information have been reworked and many issues have been fixed.

**qalter -w p** is deprecated. Instead administrators should enable scheduler job information by setting `schedd_job_info` to `if_requested` in the scheduler configuration so that users can request dispatch job information for individual jobs via **-rdi** submit switch and make the dispatch job information visible with **qstat -j jid**.

Administrators have additional possibilities to define limits that reduce memory requirements for dispatch job information.

#### 4.1.21 Data Compression at Communication Layer

The Univa Grid Engine communication library was enhanced to support data compression before transferring data over the network.

Large Univa Grid Engine clusters may produce high network data traffic. The network load depends mainly on cluster size and average job run times. In order to reduce the amount of data sent over the network Univa Grid Engine 8.6.0 supports data compression.

The additional effort to compress the data before sending is handled by introducing a commlib work thread pool which is also used for uncompressed data. On multi core architectures, uncompressed data transfer should show a performance improvement.

Setting up the new compression mode and configure thread pool settings in Univa Grid Engine requires new configuration parameters that are specified in the following Univa Grid Engine configuration areas:

man page	Description
bootstrap(5)	New parameter <code>communication_params</code> in bootstrap file
sgc_diagnostics(5)	Verify and Adjust compression and thread pool setup
sgc_conf(5)	New <code>qmaster_params</code> and <code>execd_params</code> parameter <code>CL_WP_THREADS</code> for setting up work threads (overwrite bootstrap settings)
qping(1)	New compression specific output and options (see also qping enhancements section below)

Summary of changes and enhancements:

- Compression cannot be configured at Installation time. It must be enabled manually after installation or update of Univa Grid Engine.
- Univa Grid Engine 8.6.0 uses zlib for data compression - additional compression methods are planned to be supported in follow-up Univa Grid Engine releases.
- Univa Grid Engine 8.6.0 zlib compression is supported for following architectures: darwin-x64, lx-amd64, lx-arm64, lx-arm7, lx-x86, sol-amd64, sol-sparc64, sol-x86
- The **qping** binary can be used to verify compression settings.
- It is possible to enable compression only for single clients like (**qconf**, **qstat**) by setting up an environment variable (see bootstrap(5) man page).
- Compression setting changes in bootstrap configuration require a restart of Univa Grid Engine daemons.
- Commlib work pool setting changes at **sgc\_qmaster** and **sgc\_execd** can be done during runtime of the affected daemon.

- Univa Grid Engine 8.6.0 commlib modifications show a higher performance compared to older Univa Grid Engine versions if compression is not enabled; however this depends on availability of cpus on sge\_qmaster host.
- Turning on compression will cause cpu overhead but dramatically reduces network traffic. Depending on the network speed individual compression settings may need to be adjusted (see sge\_diagnostics(5) man page).

Also **qping** is enhanced for data compression and commlib work pool settings

- **qping -dump** shows new data compression specific columns (compressed and uncompressed message length, compression ratio and method)
- **qping -info** shows active commlib work threads and min/max settings for the requested daemon
- The sge\_conf(5) qmaster\_params PROF\_COMMLIB\_TIME now shows also the nr of active work threads used by commlib in the **sge\_qmaster** messages log file.
- qping has new command line switches for basic output filtering (**-from**, **-to** and **-format**)

All these additions are described in the qping(1) man page.

#### 4.1.22 Using RSMAPs with Topology Masks and XOR Operator

RSMAP complex attributes with topology masks and the XOR operator (^) can be combined to achieve flexible PE task placement and best application performance.

Assume the following definition of the complex attributes *gpu* and *hgpu*:

```
$ qconf -sc | egrep "#name|gpu"
#name shortcut  type    relop  requestable consumable default  urgency aapre
#-----
gpu   gpu          RSMAP  <=    YES      YES      0       0       NO
hgpu  hgpu         RSMAP  <=    YES      HOST     0       0       NO
```

Sample setting on a node on host level for *complex\_values*:

2 sockets, 14 cores per socket, 28 sockets in total

```
complex_values gpu=4(0:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC \
1:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC \
2:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC \
3:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC) \
hgpu=4(0:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC \
1:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC \
2:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC \
3:SCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC)
```

Assume the *allocation\_rule* of the *mpi* PE is 28.

The first example reserves 4 machines with 28 slots on each machine for one hour immediately. Host level resources are only reserved in the second example (4 *hgpus* per host, 16 in total):

```
% qrsub -pe mpi 112 -d 1:0:0
% qrsub -pe mpi 112 -l hgpu=4 -d 1:0:0
```

Submit a job with 4 PE tasks on one host and request 4 *hgpus*:

```
% qsub [-ar <id>] -pe mpi 4 -par 4 -l hgpu=4
```

While only 4 queue slots on the host will be used, the topology mask of the *hgpu* resource will mark all 28 cores to be in use. If the cluster is configured that all jobs are submitted with a binding request no other jobs will be dispatched to this machine.

Schedule a job on 4 machines and get any two *gpus* on each host.

```
% qsub [-ar <id>] -pe mpi 4 -par 1 -l hgpu=2
```

A parallel job which should get the same *gpu* for each PE task on each host would be submitted as follows:

```
% qsub [-ar <id>] -pe mpi 4 -par 1 -l 'hgpu=1(^)'
```

This is a shortcut for

```
% qsub [-ar <id>] -pe mpi 4 -par 1 -l 'hgpu=1(1^^2^^3^^4)'
```

If the job should get the two PE tasks with one *gpu* for each task on the same socket on each host, the XOR operator can be used:

```
% qsub [-ar <id>] -pe mpi 8 -par 2 -l 'hgpu=2([0-1]^^[2-3])'
```

## 4.2 Other

### 4.2.1 JSV Improvements

The Univa Grid Engine packages now contain a Python implementation for JSVs. New example scripts for JSV and core-binding were also added.

Scripts and examples can be found at “*\$SGE\_ROOT/util/resources/jsv*”.

### 4.2.2 Reduce qhost Data Request Sizes at sge\_qmaster

By setting the environment variable `SGE_GDI_REQUEST_REDUCE_LEVEL` it is possible to reduce the amount of data transferred from **sge\_qmaster** to qhost clients. A detailed description can be found in the `qhost(1)` man page (ENVIRONMENTAL VARIABLES).

### 4.2.3 Upgrade Advance and Standing Reservations

When upgrading from Univa Grid Engine versions 8.5.4 and higher to Univa Grid Engine 8.6.0 it is now possible to upgrade advance and standing reservations as well by calling

```
# ./inst_sge -upd-ars
```

after the qmaster and execd upgrade are completed.

Prerequisite is that the host on which the upgrade is done (usually the master host) is a submit host and can resolve the user names of the AR owners and in AR `acl_list/xacl_list`.

See also the chapter about upgrading in the Installation Guide.

### 4.2.4 per\_pe\_task\_prolog and per\_pe\_task\_epilog

It is now possible to define a special prolog and epilog script that is started for each slave task of a tightly integrated parallel job that is started by **qrsh -inherit**. The prolog is started before the slave task command or script, the epilog is started after it. See man page `sge_pe(5)` for details.

### 4.2.5 Support for nvidia-docker 2.0

NVIDIA provides the version 2.0 of their Docker Container Runtime which allows to access GPUs from within Docker containers. Univa Grid Engine now supports using this Container Runtime.

In order to make use of it, first the execution host has to be installed properly, see <https://github.com/NVIDIA/nvidia-docker>. The job that wants to use the GPU must tell Docker to use the NVIDIA runtime by specifying the `-xd "--runtime=nvidia"` switch on the **qsub** or **qrsh** command line. In order to select a specific GPU, the environment variable `NVIDIA_VISIBLE_DEVICES` must be set to for the whole container by specifying it with the `-xd "--env NVIDIA_VISIBLE_DEVICES=0"` switch.

Both the `-xd "--runtime=..."` and the `-xd "--env ..."` switch are supported since Univa Grid Engine version 8.6.3.

Univa Grid Engine 8.6.16 also supports the Docker run option `gpus` to select GPUs for the container. The switch accepts either `all` to select all GPUs on the host, any integer `> 0` to select a specific amount of GPUs or the parameter `device` followed by a list of device ids to select specific GPUs, e.g. `-xd "--gpus=device=\"0,1\""`. Please note that `-xd "--gpus=..."` requires Docker API version 1.40 or newer.

#### 4.2.6 The Master Task of a Parallel Docker Job in a Container

The master task of a parallel Docker job can now be started in a container. Previously, the master task was started on the host itself but not in a container, while all slave tasks were started in containers. This new behavior requires a properly setup network, container network and DNS. See the Administrators Guide for details.

#### 4.2.7 Run the Container as root, Allow to Run Prolog etc. as a Different User

With the `execd_params START_CONTAINER_AS_ROOT` it is now possible to start all Docker containers as root and allow the `prolog`, `pe_start`, `per_pe_task_prolog`, `per_pe_task_epilog`, `pe_stop` and `epilog` scripts to be started as a different user than the job owner. This change does not apply to “autostart Docker jobs”, i.e. jobs that specify **-b y NONE** as job script in order to use the entrypoint that is defined in the Docker image instead of using the `sgc_container_shepherd` as the container entrypoint.

#### 4.2.8 Automatically Map User ID and Group ID of a User Into the Container

If the `START_CONTAINER_AS_ROOT` parameter is set to “true”, it is now necessary that the Univa Grid Engine admin user, the job user and all pre and post script users are defined inside the container. Because this is usually not the case, by setting the `AUTOMAP_CONTAINER_USERS` parameter to “TEMPORARY”, Univa Grid Engine transfers the user ID and group ID of any of these users from the host to the container. But only Univa Grid Engine itself can use this user ID information there, it is not available for the job or any of the scripts started by Univa Grid Engine!

If `AUTOMAP_CONTAINER_USERS` is set to `PERSISTENT`, Univa Grid Engine writes an entry to the “/etc/passwd” file inside the Docker container for all these users. This allows lookup of the user information in a script, but it does not allow to switch to this user!

**Caution!** If `AUTOMAP_CONTAINER_USERS=PERSISTENT` is specified and if a user maps the “/etc/passwd” and “/etc/group” file into the container, the host files are modified!

#### 4.2.9 Create a `container_pe_hostfile` with all Container Hostnames

If a parallel Docker job is started where the container hostnames are selected from RSMAPs, the execution daemon of the master task writes a `container_pe_hostfile` with all the container hostnames in the `pe_hostfile` format if the `execd_params CONTAINER_PE_HOSTFILE_COMPLEX` is set to the name of the RSMAP complex that defines the hostnames.

E.g.: If there is a RSMAP “cont\_hosts” declared and on each execution host it defines values like:

```
cont_host=4(host1_cont1 host1_cont2 host1_cont3 host1_cont4)
```

and a job is submitted using

```
# qsub -pe mype 4 -l docker,docker_images="*image:latest*",\
      cont_host=1 job_script.sh
```

and the scheduler decides to schedule the master task to host1, two slave tasks to host2 and one slave task to host3, the “container\_pe\_hostfile” might contain:

```
host1_cont3 1 <NULL> <NULL>
host2_cont1 1 <NULL> <NULL>
host2_cont4 1 <NULL> <NULL>
host3_cont2 1 <NULL> <NULL>
```

This allows to read this information in a `per_pe_task_prolog` and set the hostnames of the containers inside of the containers accordingly.

#### 4.2.10 Docker Daemon Response Timeout

A new `execd_params DOCKER_RESPONSE_TIMEOUT` is introduced. This defines the time Univa Grid Engine waits for a response from the Docker daemon to a request Univa Grid Engine sent to the Docker daemon. This does not mean the full response must be received within the timeout; the timeout counter is reset after each character Univa Grid Engine receives from the Docker daemon in response to a specific request.

If this parameter is not specified, the default value of 60 s is used. The minimum timeout is 10 s, the maximum timeout is 86400 s. If `DOCKER_RESPONSE_TIMEOUT` is not within this range, the default value is used.

#### 4.2.11 Cgroups and Containers

The cgroups generated by Univa Grid Engine are now used in Docker containers. There is no special configuration value, when Univa Grid Engine is configured to use cgroups, these cgroups are also used within Docker containers.

There is a known issue with cgroups and Docker that is not yet understood:

On some Linux distributions some Docker versions expects the cgroups path to contain a “.slice” postfix, which makes Univa Grid Engine unable to start Docker jobs in cgroups - the jobs will fail then. This was observed on CentOS 7 with Docker 1.12.6 build c4618fb, while it was not observed with Docker 1.12.6 build 78d1802 on the same host. It was also not observed on different distributions than CentOS 7.

There are indications that it could be related to the Docker SELinux packages that are provided for CentOS 7, if they do not match the Docker version exactly, this error seems to be more likely.

#### 4.2.12 Specify Arguments to Autostart Docker Jobs

The autostart Docker jobs, i.e. Docker jobs that were submitted as binary jobs with the keyword `NONE` as job script, now accept arguments. These arguments are provided to

the binary or script defined in the `ENTRYPOINT` of the Docker image and are appended to arguments that are defined in the `ENTRYPOINT`. This works only with suitable Docker images. Whether a Docker image is suitable can be tested by manually starting

```
$ docker run -it image:latest arg1 arg2 arg3
```

on the execution host. If the script or binary defined in the `ENTRYPOINT` gets these arguments, it should also work with an Univa Grid Engine job.

#### 4.2.13 New Client Command `qralter`

A new `qralter` client command has been added to Univa Grid Engine allowing to modify existing advance reservations, see [User Guide -> Reservations](#).

#### 4.2.14 Changes to the `loadcheck` Command

`loadcheck` will now optionally output information about GPUs available on a host like the GPU name, available memory and cpu affinity.

`loadcheck` outputs a name and a value column, with Univa Grid Engine 8.6.0 the width of the name column has been increased.

#### 4.2.15 Changed startup behaviour of the execution daemon

The execution daemon startup behaviour can be changed with Univa Grid Engine 8.6.4. Normally, if an execution daemon successfully connects to the `qmaster`, but does not get its configuration sent, the execution daemon quits. Usually this is the case if the execution daemon runs on a host that is not configured as execution host (yet).

By setting the environment variable `"SGE_EXECD_KEEP_TRYING_TO_GET_CONFIG"` to `"1"` before the execution daemon is started, the execution daemon stays running and periodically tries to get its configuration from the `qmaster`. This allows e.g. to first start the execution daemon and later configure it at the `qmaster`.

If `"SGE_EXECD_KEEP_TRYING_TO_GET_CONFIG"` is set to `"0"` or if it is not set, the old behaviour is in effect, i.e. the execution daemon quits if it does not get its configuration from the `qmaster`.

See issue GE-6235 "Provide a way to change `execd` startup behaviour -> infinite connection retries"

#### 4.2.16 Forwarding the environment for a `qrsh` (without command) and `qlogin` job is enabled

With Univa Grid Engine 8.6.4 and the "builtin" interactive job mechanism configured, it is now possible to forward the environment from the submit host to the job using the `"-V"` switch, like it is possible for `"qsub"` jobs all along.

#### 4.2.17 Suppress the user switch inside Docker jobs

With Univa Grid Engine 8.6.4 it is now possible to start an autostart Docker job as the user that is defined in the Docker image the container the job uses is created from. In order to allow this, the Univa Grid Engine administrator must configure the "qmaster\_params" "ENABLE\_XD\_RUN\_AS\_IMAGE\_USER=1" (see `sge_conf(5)`) to allow the users to submit a job with the option "-xd\_run\_as\_image\_user y[es]|n[o]" (see `submit(1)`).

With this option, an autostart Docker job (i.e. a Docker job that has no job script argument, but the keyword "NONE" instead) is started as the user that is defined in the Docker image specified in the job submit command line.

#### 4.2.18 Allow to skip Docker image check in execution daemon

With Univa Grid Engine 8.6.4, a boolean "execd\_params" "DOCKER\_SKIP\_IMAGE\_CHECK" was added. If it is set to 1 or TRUE the check if the image for a hard requested docker\_images request actually exists is skipped.

See `sge_conf(5)` for details.

#### 4.2.19 New values for initial\_state of a queue

With Univa Grid Engine 8.6.4, the new possible values "execd\_enabled" and "execd\_disabled" have been added for the initial\_state parameter of a queue. If set to any of these values, the queue will only be enabled/disabled when the execd is (re-)started, not when qmaster is restarted. The current behaviour if the parameter is set to "disabled" and "enabled" was not changed.

#### 4.2.20 Removed reporting of m\_mem\_free\_nX load values

With Univa Grid Engine 8.6.5, reporting of `m_mem_free_nX` (i.e. `m_mem_free_n0`, `m_mem_free_n1`, ..) as well as `m_mem_used_nX` as load values has been removed. The reason for this change is the lack of information provided by the Linux kernel about Buffers and Caches on NUMA nodes. This can lead to drastically lower values being reported than what is actually available, thus prohibiting jobs from being scheduled onto such a host. Since there is currently no way to report memory values that are at least approximately close to reality, none will be reported.

`m_mem_total_nX` are still going to be reported.

#### 4.2.21 GPU Affinity as Soft Request

With Univa Grid Engine 8.6.5, GPU affinity for jobs that request GPUs via RSMAPs can be requested as a soft request with `-l GPU=1[affinity=2]`. If a job requests GPUs and affinity as soft request, Univa Grid Engine will try to find and assign a CPU/GPU combination with good affinity (as reported by DCGM) and if there is no suitable CPU/GPU combination available, Univa Grid Engine will schedule the job without binding any CPU cores. A job with `-l GPU=1[affinity=true]` or `-l GPU=1[affinity=1]` will not be scheduled if no GPU/CPU combination with good affinity is available.

#### 4.2.22 Trigger command to force re-read of host\_aliases file

With Univa Grid Engine 8.6.7 the qconf client got a new command line switch (-uha) which triggers qmaster to re-read the current host\_aliases file in order to adopt changes immediately. More information can be found in the *qconf(1)* man page. In addition to that *sge\_conf(5)* describes a new qmaster\_params setting to disable the default host\_aliases check which is done every 60 seconds.

This enhancement can be used to create automated scripts for adding/removing/adjusting the host\_aliases file which then will trigger qmaster process to adopt the modifications as fast as possible.

#### 4.2.23 Automatically set jobs to hold state if they trigger errors

With Univa Grid Engine 8.6.7 and 8.6.8 new qmaster\_params parameters *MAX\_AJ\_QFAIL\_HOLD* and *MAX\_AJ\_TFAIL\_HOLD* were introduced. They can be used to define a threshold to set jobs into hold state if the job is responsible for setting queues into error state or many tasks are failing with a job error. Jobs that produce queue errors are usually rescheduled to another queue. This can trigger the situation that a job is responsible for setting all or many queues into error state. Another use case is to stop job task execution when some tasks reporting a job error and it makes no sense to continue further tasks of the job. The new parameters can be used to prevent such situations. More information can be found in the *sge\_conf(5)* man page.

#### 4.2.24 Switch off submit host verification

With Univa Grid Engine 8.6.7 a new qmaster\_params parameter *ALLOW\_ANY\_SUBMITHOSTS* was introduced. It can be used to disable submit host verification. More information can be found in the *sge\_conf(5)* man page.

#### 4.2.25 Log daemon startup errors to syslog

If the variable *SGE\_USE\_SYSLOG\_AT\_STARTUP=1* is set in the environment of the *sge\_qmaster(8)*, *sge\_execd(8)* or *sge\_shadowd(8)*, during the startup of the daemon before it is able to log to its messages file, all errors and warning which would be written to a temporary messages file in */tmp* are also written to the syslog. If these messages really appear in the syslog depends also on the configuration of the syslog daemon.

#### 4.2.26 Enabled monitoring and deadlock detection feature for all platforms

The deadlock detection (see "DEADLOCK DETECTION" section in manpage for *sge\_diagnostics(5)* ) and message file monitoring (see "MESSAGE FILE MONITORING" section in manpage for *sge\_diagnostics(5)* ) is now supported for all architectures.

#### 4.2.27 sub-cgroup feature

With Univa Grid Engine 8.6.7, it is possible to have the job user get ownership either over all the per-job cgroups created under all subsystems or just under the memory subsystem. This can be achieved with a new `execd` param, `CGROUPS_USER_CONTROL`. If set to `ALL`, the job user gets ownership over all job cgroups under all the subsystems. If set to `MEMORY`, the job user gets ownership only over the job cgroups under memory subsystem. If set to `NONE` (default), the ownership is not transferred at all to the job user.

#### 4.2.28 Removing stale "context" entries in queue

Univa Grid Engine stores queue annotation messages ("`qmod -msg`") in a hidden "context" queue attribute. Those messages were not always deleted when a host was removed from a queue. This could lead to unnecessary hostname resolving in `qmaster` and have a performance impact when hostnames became unresolvable.

Univa Grid Engine 8.6.9 fixes this issue with GE-7726 and GE-6078. With any queue or hostgroup changes context entries for removed hosts will now be removed.

After upgrading to UGE 8.6.9 or higher it is recommended to make a (pseudo) queue change for all queues. Run as Univa Grid Engine administrator the command

```
# qconf -mq <qname>
```

And enter e.g. a space at the end of the first line and save the file. This will remove all stale annotation messages in the context field.

#### 4.2.29 Cleanup of output file path handling

Univa Grid Engine 8.6.10 is fixing the issue GE-8194 (UGE pseudo environment variables replacement not working for `qsub -e/-o`). The fix also changed following behaviour:

Using a pseudo variable in output files without one additional separation character (e.g. `qsub -o "JOB_IDTASK_ID"`) did result in not parsing the pseudo variable following a previous one.

The pseudo variable "`$TASK_ID`" is only supported for jobs with array tasks. For jobs without array tasks definition (e.g. `qsub` without `-t` option) the pseudo variable will be replaced with the string "undefined".

More information can be found in the `submit(1)` man page.

#### 4.2.30 New logging option for hostname or ip lookup calls

Univa Grid Engine 8.6.11 supports a new messages file logging parameter for `qmaster` and execution daemons. The new parameter "`LOG_RESOLVING_TIME`" is supported for "`qmaster_params`" and "`execd_params`" in the global or local cluster configuration (see `sgc_conf(5)`). It can be used to figure out performance problems when a Univa Grid Engine daemon needs to resolve hostnames or IP addresses via lookup services (e.g. NIS, DNS, ...).

### 4.2.31 Changes in handling of job arguments for (t)csh users

Univa Grid Engine 8.6.17 is fixing the issue GE-8356 (Quoting of command lines causes job start errors in (t)csh). This fix adds a new execd param, `ENABLE_BACKSLASH_ESCAPE`. Depending on the value of this param, the following behavior is expected:

If set to no, backslashes inside job arguments are never escaped.

If set to yes (default), backslashes inside job arguments are properly escaped.

If set to auto, the behavior will be same as when set to no for the following scenarios: When `shell_start_mode` is set to `posix_compliant` and queue's default shell (or the shell requested with `-S`) is (t)csh). When `shell_start_mode` is set to `unix_behavior` or `script_from_stdin` and if the job script is a (t)csh script.

Setting environment variable, `SGE_BACKSLASH_ESCAPE={no|yes|auto}` will take precedence and override the above param value.

Note: (t)csh users might want to consider setting environment variable (preferably in users `.cshrc`), `backslash_quote` to avoid issues with escaping double-quotes in the arguments.

### 4.3 Full List of Fixes and Enhancements

#### 4.3.1 8.6.0 (and Also Fixed for a 8.5.\*, 8.4.\*, 8.3.\* Patch Release)

GE-3721 qstat -j "\*" -u "user1" is not working  
 GE-5290 qstat (-xml) does not accept filter switches when -j "\*" is specified  
 GE-5831 UGE REST Server Crashing when POSTing new complex  
 GE-5941 renewing certificates makes sgepasswd file unreadable  
 GE-6575 stree-edit utility broken  
 GE-6697 qstat -j "\*" does not support "\$user" placeholder set in sge\_qstat request file  
 GE-6734 wrong scheduler info messages shown for jobs  
 GE-6764 very long dispatching time due to RQS rule result in scheduler timeout  
 UWSA-77 Basic authentication fails under solaris frequently  
 UWSA-81 Add support for listening only on localhost  
 UWSA-169 REST Service: 'none' auth method doesn't work  
 UWSA-186 requested jobEnvironment is not shown in jobs  
 UWSA-188 jobEnvironment ugerestsdk contains error in json converter  
 UWSA-189 ugerest is showing same scheduler conf twice  
 UWSA-190 upgrade of restlet-jse-2.3.6 to restlet-jse-2.3.10  
 UWSA-199 UGERest and other jar files do not show their build version  
 UWSA-200 Add version info route to ugerest

#### 4.3.2 8.6.0 (and Also Fixed for a 8.5.\* and 8.4.\* Patch Release)

GE-4389 enhance qsub man page with JSV modification examples for core binding  
 GE-5835 long scheduling times with wildcard PEs and resource reservation  
 GE-6018 Setting limits based on percentage.  
 GE-6103 jobs are bound to cores even if no binding is requested  
 GE-6356 drmaa2\_j\_get\_info does not provide full job information  
 -> not all fields are filled  
 GE-6402 Scheduler might not respect RQS limits during the time RQS rules are changed  
 GE-6432 qdel -u "\*" is only allowed to managers, not to operators  
 GE-6462 on native Windows (win-x86), environment variable values containing an equal sign are truncated  
 GE-6478 Very long load value of loadsensor causes segfault of execd  
 GE-6510 Core binding: striding-strategy counts needed cores wrong and rejects viable hosts  
 GE-6511 gdi\_request\_limits does not behave as documented  
 GE-6534 sge\_execd crash with core dump with GPGPU jobs  
 GE-6568 Allow for forced job deletion through UGE REST  
 GE-6595 Docker interactive job can't be deleted by qdel  
 GE-6671 sudo requests for same user as ugerest service user are rejected  
 GE-6682 exec host cannot startup if no admin or submit host

GE-6709 huge virtual memory requirements with test DRMAA1 application  
 GE-6728 sched\_conf.5 man page is inaccurate regarding the  
 PREFER\_SOFT\_REQUESTS scheduler param  
 GE-6771 qsub -sync stops immediately on native Windows (win-x86)  
 GE-6787 qmaster and execd logging "invalid value (33026) for ar->op"  
 GE-6855 sge\_qmaster abort() with critical logging "got NULL element  
 for RUE\_utilized\_now"  
 GE-7237 UGE qmaster daemon may crash in 'lo\_thread' thread  
 GE-7240 UGE qmaster does not handle LO delete cluster request  
 UWSA-193 additional job usage values for execd\_params  
 ENABLE\_MEM\_DETAILS=1 are missing

### 4.3.3 8.6.0 (and Also Fixed for a 8.5.\* Patch Release)

GE-285 allow parallel job allocation scheme be specified at submission time  
 GE-3146 backfilling a resource reservation before a calendar or an AR is broken  
 GE-4305 Better documentation for basic share tree use case  
 GE-4726 adding a way that multiple jobs can be bound to the same socket  
 using all cores  
 GE-5436 qmon should not reset value "-1" for sharetree usage\_weight\_list list  
 GE-5569 native Windows (win-x86) sge\_execd exits if it cannot access the  
 act\_qmaster file  
 GE-5636 qmaster error logging "rc\_add\_job\_utilization 0 slot amount" indicates  
 wrong job reservation  
 GE-5650 with allocation rule \$fill\_up, slave resource requests are not obeyed  
 when master resource and queue requests are provided  
 GE-5806 parallel jobs might not startup due to wrong RQS calculations  
 GE-5830 Scheduler fails to handle a job submitted with option "-mbind cores"  
 GE-5848 allow to use UGE cgroups in Docker jobs  
 GE-6020 allow native Windows (win-x86) functions to retry to logon users  
 several times  
 GE-6105 RSMAP attributes should support per slot resource allocation for PE jobs  
 GE-6183 sge\_shepherd sets limits too high for master task  
 GE-6236 accounting file broken on Solaris if accounting line has 1023 characters  
 GE-6237 jobs are being restarted even if the '-r no' was specified  
 during submission  
 GE-6270 allow to run prolog etc. as root inside a Docker container  
 GE-6271 sge\_container\_shepherd fails if the prolog is started as root  
 GE-6407 add documentation for config-api  
 GE-6413 shepherd does not handle all error responses to a pull Docker image request  
 GE-6434 incomplete binding requests shown in qstat for long lists of binding requests  
 GE-6454 improve documentation of Docker integration  
 GE-6480 wrong/missing error messages and wrong exit status when initializing  
 invalid RSMAP ranges  
 GE-6483 document placeholders in Docker requests in the UserGuide and man pages  
 GE-6489 use of external loadsensors might deadlock execution daemon  
 GE-6490 load sensor specific errors are not logged into execd messages file  
 GE-6493 RSMAP map entry selection request does not work

- GE-6497 support halftime -1 setting in scheduler config to disable past usage for sharetree
- GE-6506 qstat does not show granted PE
- GE-6508 Accounting shows wrong fail and exit states for 'qdel -f jobID'
- GE-6521 Qmaster Crashes with LO Enabled and Job Dependencies
- GE-6524 AR shouldn't be scheduled to unknown queue instances at time of submission
- GE-6527 document qsub -xd --help
- GE-6528 allow to specify arguments to autostart Docker jobs
- GE-6536 No error message when a job cannot get scheduled due to RSMAP-ranges
- GE-6537 Submitting RSMAP-range jobs via -adds is broken
- GE-6549 qstat/qhost outgoing packet size is much bigger than the data which is finally displayed
- GE-6553 RQS limits incorrectly applied when PE job submitted with "-l h=<host>"
- GE-6566 qsub rejects valid RSMAP resource requests
- GE-6569 Allow for Deletion of non jobsession jobs in DRMAA2
- GE-6573 reduce resulting qhost client requested data sizes transferred via network
- GE-6583 scheduler is wrongly skipping hosts or queue instances for parallel jobs that are using master task specific requests
- GE-6597 execd uses 100% CPU if it can't delete a finished Docker container
- GE-6615 enhance error logging if load sensor cannot be started
- GE-6617 provide workaround in native Windows (win-x86) sgeexecd.bat script for start /b bug on Windows 10, version 10.0.15063
- GE-6618 native Windows (win-x86) execution daemon crashes if qloadsensor does not work
- GE-6620 error messages from qloadsensor.exe (win-x86) are not forwarded to the execd messages file
- GE-6623 add the UGE admin user to the Performance Monitor Users group on native Windows 10 (win-x86)
- GE-6629 qhost NSOC and NCOR incorrect on lx-arm64
- GE-6637 using hostgroups in rqs limit definition can trigger short qmaster hang at startup or rqs modify request
- GE-6640 qloadsensor.exe report "no error" if an error occurs while initializing the PDH service
- GE-6644 any epilog SIGSEGVs and sets queue in error state with `execed_params INHERIT_ENV=false`
- GE-6664 When host aliases are configured `qsub -sync y` reports "commlib info: successfully updated host aliases (add: 0, del: 0)"
- GE-6665 while a non allocated reservation is "active" in a standing reservation jobs submitted into the SR get rejected
- GE-6670 The global configuration parameter "gdi\_request\_limits" not working for aliased hostnames
- GE-6678 Improve `accept()` handling in `commlib`
- GE-6687 RSMAP-topology-masks jobs and -binding jobs result in wrong scheduling decisions
- GE-6695 `qrstat` does not output `cal_depth` and `cal_jump` information for standing reservations
- GE-6696 `save_sge_config.sh` needs to dump advance/standing reservations for upgrade to newer versions
- GE-6706 unexpected logging and possibly incorrect accounting if multiple

- array tasks of a job are running on a host
- GE-6713 shadowd on Solaris cannot start sge\_qmaster
  - GE-6737 queue calendars closing queues not considered when max\_reservation > 0
  - GE-6739 Parallel job requesting pe range not scheduled even if resources available
  - GE-6741 jobs submitted into AR with RSMAP resources are not scheduled
  - GE-6744 qrstat does not output queue request (-q) and immediate request (-now y)
  - GE-6746 support a "pack socket" core binding strategy
  - GE-6754 Adding new session (qconf -asi) not working on admin only host
  - GE-6755 save\_sge\_config.sh does not dump all advance/standing reservations
  - GE-6774 Jobs do not get the correct binding when a specific RSMAP id with topology mask is requested
  - GE-6775 RSMAP ids are granted even if the requested cores are already in use
  - GE-6777 huge erroneous reader thread logging at calendar state transition
  - GE-6782 qstat -njd is not working as documented in the man page
  - GE-6785 Supplementary groups in manager and operator lists are ignored during access validation for job deletions.
  - GE-6786 false logging for 'qmaster\_params': 'gdi\_timeout', 'gdi\_retries' and 'gdi\_ping'
  - GE-6788 qmaster logging about receiving older load report
  - GE-6796 Calendar modification/state transition might cause repeated timed calendar events for up to one second.
  - GE-6799 unexpected logging messages in sge\_qmaster messages file
  - GE-6800 implement a per task prolog and epilog for tightly integrated parallel jobs
  - GE-6812 started docker job at execd may result in sge\_shepherd process eating up all memory
  - GE-6814 removing non "lo\_\*" complex entry triggers error logging of lothread
  - GE-6818 error logging: getgrgid(...) failed: Numerical result out of range
  - GE-6826 Spaces within a comma separated list of switch arguments are not allowed
  - GE-6829 disallow mapping /etc/passwd and /etc/group into the container if AUTOMAP\_CONTAINER\_USERS=PERSISTENT is specified
  - GE-6835 update host configurations for changed hostnames not working if a changed hostname is matching as substring of another host
  - GE-6836 improve execd reconnect behavior after qmaster shutdown and restart
  - GE-6841 allow to configure the Docker daemon communication timeout
  - GE-6843 Integration with latest Docker CE version(17.09)
  - GE-6852 interactive Docker jobs take very long to finish if the qrsh client is killed and an epilog is configured
  - GE-6853 Sometimes not able to delete pending job if docker device-mapping is used
  - GE-6862 Release notes contain 'current version' also for features that were introduced with FCS
  - GE-6869 arch script does not support Mac OS X High Sierra
  - GE-6874 normal docker job required no\_root\_quash if execd spooling directory shared via nfs
  - GE-6884 qconf help menu lists ss switch twice
  - GE-6901 "messages" file can have the name ".<digit>" instead of "messages" in case of file creation issues
  - GE-6907 Allow to specify allocation rule (-par switch of qsub) via JC's
  - GE-6913 expose scheduler thread ID in the qping output
  - GE-6916 sge\_ca script does not show output for skip install question if CA

- directories are already existing
- GE-6924 PE jobs that request a combination of consumable HOST/JOB and YES RSMAP complexes might not get the correct amount of ids assigned
  - GE-6931 the main shepherd of an interactive Docker job uses 100% CPU and doesn't end if output file is deleted
  - GE-6932 child shepherd waits infinite for mutex in shepherd\_trace()
  - GE-6934 jobs that are submitted into AR and request RSMAP ids block ids that were not assigned
  - GE-6939 very long ticket calculation times for parallel array tasks
  - GE-6945 cannot create AR for queues with jc\_list "ANY\_JC"
  - GE-6946 Delete an array task will break the -tc <number> condition
  - GE-6948 parse error message from Docker response and provide it as job error reason
  - GE-6959 sge\_execd calls sched\_setaffinity function repeatedly
  - GE-6972 Problem with gridengine/kernel
  - GE-6973 Problem with gridengine/kernel
  - GE-6974 Problem with gridengine/kernel
  - GE-6983 if job is killed by execd enforced limit the job failed state in accounting file should reflect this
  - GE-6997 the native Windows (win-x86) execd should log that it cannot find or start the qloadsensor.exe
  - GE-7002 test Docker integration with Docker versions up to 17.12
  - GE-7008 Jobs requesting RSMAP complexes that are defined without Ids are not scheduled
  - GE-7009 no job dispatching with max\_reservation > 0 and consumable+PE+binding combination request
  - GE-7027 In AFS/KRB Grid Engine installations coshepherd/set\_token\_cmd is not cleaned up
  - GE-7039 AR slot oversubscription
  - GE-7042 Resource reservation due to start time prevents dispatching the next job of the same category
  - GE-7056 Failed/deleted docker interactive job's shepherd remains and increase trace file forever
  - GE-7060 Possible crash of qmaster clients
  - GE-7062 reservation for PE jobs with RSMAP requests is too far in the future
  - GE-7063 submitting an AR to a PE can violate the PE's fixed allocation rule
  - GE-7067 jobs with reservation and RSMAP complex requests cause error logging in qmaster messages file
  - GE-7068 PE jobs that request binding and RSMAP complexes defined as consumable HOST might not be scheduled
  - GE-7069 running jobs requesting host level consumable prevent ARs to be scheduled
  - GE-7078 ARs not backfilled with backfilling=h\_rt
  - GE-7085 Logging of PDC create execd message owned by root user
  - GE-7108 CRITICAL "error: lGetElemStrFirst(ST\_name): run time type error" when submitting advance reservation with project
  - GE-7111 CSP effective secret key algorithm is not used correctly
  - GE-7112 qmaster error logging "rc\_add\_job\_utilization 0 slot amount (job <job\_id> obj global type RUNNING)"

GE-7144 UGE 8.5.5 grid master continuously segfaulting  
 GE-7184 provide host level configurable job umask setting  
 GE-7187 epilog scripts are not always executed when running docker job  
 GE-7263 unset closed file pointers after call to sge\_peclose()  
 UWSA-194 JobClass name of submitted job is missing  
 UWSA-195 fix the OpenSSL security warning

#### 4.3.4 8.6.0

GE-168 enhance qacct [-A account\_string] to qacct [-A [account\_string]]  
 GE-639 job umasks should be configurable per job  
 GE-1969 qconf -me reports success on incorrect file  
 GE-2186 schedd job info shows old and outdated message although  
 job is meanwhile running.  
 GE-2242 Inadequate job\_info messages for resource quotas with  
 parallel jobs  
 GE-2464 schedd\_job\_info can cause immense memory consumption  
 GE-2739 No way to remove a -notify flag, once it was set  
 GE-2748 Jobs with no suitable queues at all should be easier detect  
 GE-3279 Description of 'job\_is\_first\_task' in `man sge\_pe` should be rephrased  
 GE-3614 add a way to distribute a parallel job on different  
 sockets without knowing the exact architecture on  
 submission time  
 GE-3621 add a placeholder for the -binding parameter which  
 aligns the amount of slots with the amount of requested cores  
 GE-3754 write documentation for sge\_share\_mon  
 GE-3787 string load values are reported only up to 1024 characters  
 GE-3803 qacct fails with accounting files bigger than 4GB on  
 native Windows (win-x86)  
 GE-3864 qsub -w e/-w v do not consider attributes which are load  
 values only  
 GE-3965 qalter -w p of a job in user hold prints 'verification:  
 job is already running'  
 GE-4028 DOS CR-LF in submitted scripts causes shell search errors  
 GE-4059 qalter -clearp silently exits  
 GE-4275 qalter -w e|w|v|p <job\_id> identifies jobs as running when  
 they are not eligible for scheduling  
 GE-4358 misleading diagnose message for qalter -w p that refers to  
 queue 'global' instead of host  
 GE-4429 Improve qalter -w option documentation  
 GE-4430 Qalter -w p on a held job says 'job is already running'  
 GE-4433 qalter returns nothing when run without a job  
 GE-4498 man lacks detailed information about the new pss, rss,  
 smem, vmem values  
 GE-4678 job class parameter V does not work like described in  
 sge\_job\_class man page  
 GE-4733 finding rqcs excluded queues is printing a useless  
 error message

- GE-4734 reduce impact of qalter -w p on sge\_qmaster operation
- GE-4785 qmod -rq can trigger a qmaster crash when the queue is in o state
- GE-4793 qalter -tc messages are not displayed, each time of execution
- GE-4794 Information about the setting of tc switch can not be retrieved
- GE-4817 Add automatic corebinding magnitude selection when used with PE's
- GE-4820 Document the '-w' Options to qsub as a way to validate job submissions
- GE-4825 provide Python binding for JSVs
- GE-4839 rescheduling might produce unexpected error messages
- GE-5069 new binding strategy "linear\_socket" instead of requesting "linear" plus -l sched\_binding\_per\_socket=1
- GE-5084 qmaster\_params "OLD\_RESCCHEDULE\_BEHAVIOR" should not be declared deprecated
- GE-5100 forwarding of job error messages from native Windows (win-x86) exec hosts to qmaster
- GE-5248 remove deprecated gdi\_multi\_read\_req setting
- GE-5281 Add support for devices subsystem in cgroup integration
- GE-5304 bug in mirror interface causes segfault in drmaa2 application
- GE-5395 adding a way for listing the state of all global resources
- GE-5403 Job should run on free core first
- GE-5471 Need tool to provide overview of used and requested cluster resources
- GE-5535 enable schedd\_job\_info for specific jobs only
- GE-5537 standing reservation, which enhances the AR
- GE-5542 request for a rerun limit for jobs
- GE-5571 qalter -w e|v|p does not take RQS limit rules into account that contain job class filters
- GE-5573 scheduler triggers job delivery before complete execd cleanup of rescheduled job
- GE-5606 pe\_hostfile documentation is not exact enough
- GE-5686 prolog/epilog race conditions when jobs are rescheduled
- GE-5691 qrsh -cwd is broken
- GE-5706 gdi\_request\_limits parameter is missing in global config
- GE-5725 upgrade procedure fails if LO\_ROOT is set
- GE-5733 job in hold state gets tickets and is reported as running by "qalter -w p"
- GE-5773 Ability to change the cgroup name 'UGE' to another name
- GE-5799 Deliver Univa Grid Engine software as RPMs
- GE-5857 handle delete requests of event clients preferred
- GE-5872 gdi\_request\_limits should support to limit event client registrations
- GE-5888 resources defined in the global host are not available for -masterl requests
- GE-5910 make DRMAA2 compatible with the AR object changes
- GE-5911 make WebService API compatible with the AR object changes
- GE-5919 correct sge\_diagnostics man page header and footer
- GE-5924 Standing Reservation XML output
- GE-5926 Standing Reservation must go in Error state when no more allocations are possible

GE-5927 support to upgrade standing reservations  
GE-5928 support Python configuration API for standing reservations  
GE-5939 reimplement manual rescheduling done by qmod -r  
GE-5947 drmaa2 complete reservation session features  
GE-5956 re-connect request for an event client even on qmaster shutdown  
GE-5963 qalter -w p doesn't report correctly when -masterl switch is used  
GE-5987 examine fopen() system call problem  
GE-6014 Enhance sge\_diagnostics man page with info about scheduler profiling  
GE-6038 schedd\_job\_info true/false is not case insensitive  
GE-6071 Support for Affinity/Anti-Affinity in UGE  
GE-6086 Fix Java code (JGDI, UGEREST etc.) to support standing reservations  
GE-6125 for demo binaries the version string printed by -help command line option and in messages files should contain "demo"  
GE-6129 update BerkeleyDB to version 6.2  
GE-6141 DRAMA2 still does not use GDI sessions which will have an impact on performance  
GE-6143 qalter -w p requests are executed by worker threads  
GE-6147 category string should be created and normalized in qmaster thread when jobs are added or modified  
GE-6159 add qalter to modify end time of Advance Reservations  
GE-6249 remove qtcsh from distribution  
GE-6255 execd job reports get lost due to a race condition in sge\_qmaster  
GE-6256 Parallelize sorting in CULL to improve scheduler performance  
GE-6260 Compression/uncompression of data passed/received from commlib layer  
GE-6261 event\_master thread performance might be improved by processing events for different event clients in parallel  
GE-6275 Determine number of unused cores (threads) in qmaster and execd  
GE-6277 Improve performance of basic CULL functionality  
GE-6287 Move scheduler category to master  
GE-6328 out-of-the-box functionality to show utilization of global variables  
GE-6334 hostgroup changes via -\*attr do not update queue instance settings  
GE-6339 QERROR message should include task ID for array jobs  
GE-6364 check if username needs to be part of the scheduler category string  
GE-6412 deny job submission with empty non boolean request  
GE-6415 core binding jobs fail to be scheduled on free cores.  
GE-6418 Implement host\_sort\_formula in schedd configuration object  
GE-6424 Adapt config API to reflect object changes  
GE-6439 possible performance degradation in scheduler when debiting dispatched jobs  
GE-6440 create and maintain affinity cache in worker / scheduler thread  
GE-6441 output affinity information with qstat -F and qhost -F

- GE-6450 change naming scheme of fallback messages files
- GE-6455 Look at all open, fdopen, close, etc. calls and map them to a global function
- GE-6458 qconf -srqsl and other "show list" options report an error if no data is configured
- GE-6468 Introduce and document new parameter that influences host resolving timeout handling
- GE-6476 Enhancing sge-diagnostic man page
- GE-6482 scheduler profiling does not cover full scheduler code
- GE-6484 Wrong/missing error-logging for loadsensors
- GE-6491 affinity shall only be reduced for preempted jobs for complex variables which are preemptable
- GE-6492 change host and queue sorting in scheduler thread to reflect affinity
- GE-6494 Wrong binding-parameters are logged in qmaster-messages file instead of sent to client
- GE-6499 Add entry for "used\_slots" to sge\_pe man page & admin guide
- GE-6500 fix compensation\_factor description in sched\_conf(5)
- GE-6526 Excessive Memory Usage with large RSMAP Ranges
- GE-6530 Core binding: Binding request for PE-jobs should be "per task" instead of "per host"
- GE-6531 Core binding: execd applies granted core-binding for PE-jobs in a wrongish way
- GE-6532 Core binding: Create new binding strategies with better names
- GE-6533 Core binding: Number of granted cores should be part of accounting
- GE-6538 Add RSMAP-ranges as <name>:<amount> to a host does not work
- GE-6540 remove queue\_sort\_method from scheduler config (replaced by weight\_queue\_seqno)
- GE-6544 Core-binding: qsub checking of binding-strategy is not strict enough (linearasdf:4 is accepted)
- GE-6545 adapt Docker version check functions to new Docker version scheme
- GE-6561 Core binding: Remove scheduler parameter "sched\_binding\_per\_socket"
- GE-6567 ranges in combination with RSMAP id requests do not work
- GE-6578 Duplicate calendar entries associated with host\_aliases
- GE-6600 Core-binding: cgroups with PE-jobs not correct
- GE-6611 build unsupported platforms with classic spooling support only
- GE-6622 Order of fields in the accounting man page does not match qacct output
- GE-6638 accounting(5) man page field description is broken for "ioops"
- GE-6642 operators cannot delete a (standing) reservation
- GE-6646 Add support for NVIDIA DataCenter GPU Manager
- GE-6647 Automatically use affinity for GPU and CPU where possible
- GE-6649 Use environment variable CUDA\_VISIBLE\_DEVICES to hide disabled GPUs
- GE-6655 Document how to tag CPU cores as 'already in use' so that they will not be considered for core and memory binding
- GE-6666 qhost man page is lacking description for resources that are free-but-still-bound
- GE-6677 DRMAA jobs submit always with -w e which has a negative performance impact on qmaster

- GE-6691 support halftime -1 setting in scheduler config to disable past usage for sharetree
- GE-6692 cleanup of CULL for halftime -1 that requires minor release change
- GE-6693 allow to set halftime to -1 in qmon/config API and UGE Rest
- GE-6707 qalter -w e|w does not return a validation message.
- GE-6708 qping and other uge binaries dump core if local hostname is not resolveable
- GE-6720 calender off times not considered by resource reservation
- GE-6736 Remove Solaris dtrace support and related things from UGE and LO
- GE-6738 drmaa2\_open\_jsession does not recreate htable of job, jtemplate and jinfo
- GE-6752 update jemalloc to current version (5.1.0)
- GE-6753 update hwloc lib to current version (1.11.7)
- GE-6758 switch to a newer OS (e.g. CentOS 6) for the lx-\* build
- GE-6761 qrstat -ar <ar\_id> prints the PE request as granted\_parallel\_environment
- GE-6766 xml schemas need to be verified, updated and made UGE version specific
- GE-6772 update man page for queue initial\_state to include qmaster migration scenario
- GE-6781 qstat -j reports job as running but shows also schedd job information why the job cannot be started
- GE-6783 shepherd trace file grows too large for interactive Docker jobs
- GE-6831 Alternative solution to limit the number of event clients in use by abusive users
- GE-6834 unexpected logging on rescheduling of tight pe jobs
- GE-6837 config-api test in master branch fails due to missing field per\_pe\_task\_prolog
- GE-6839 sge\_execd is not properly handling short outages of sge\_qmaster resulting in delayed reporting of load values
- GE-6840 Generate all nroff man pages from markdown source files
- GE-6854 pe\_n, pe\_min\_X, pe\_max\_X missing in jsv(1) man page
- GE-6857 requesting RSMAP complexes as soft request with -soft and/or l\_soft does not work
- GE-6859 qacct -A does not show column name with account string
- GE-6860 Implement a way to disable the chaining of RSMAP ids
- GE-6887 reduce the number of copy operations done in event master thread when distributing events to clients
- GE-6891 qalter -clearp does not provide a proper response message
- GE-6902 enable immediate load report sending
- GE-6903 improve communication connection shutdown and gdi timeout handling
- GE-6917 chaining of multiple RSMAP complexes with XOR parameter does not work
- GE-6918 it is possible to specify invalid PE allocation rules
- GE-6919 config-api missing field
- GE-6920 update openssl library to most current version
- GE-6923 Add job category field to 'qstat -j' output
- GE-6947 show user name trying exceed max\_u\_jobs on qmaster message log
- GE-6955 cgroups definition of subdir\_name only working if beginning with "/" on some docker versions
- GE-6982 accounting should contain the pe task id for tasks of tightly

- integrated pe jobs
- GE-6991 extend monitoring to track time waiting in mutex\_lock
- GE-6993 environment of Docker pe task shows wrong RSMAP ids if complex is consumable YES
- GE-7001 wrong behaviour if RSMAP topology mask does not fit on actual host topology
- GE-7017 support XOR RSMAP request per PE task only
- GE-7020 Enhance current commlib message protocol to support compression
- GE-7026 build process for markdown man pages is broken
- GE-7031 display requested RSMAP selection in qstat -j/qstat -r
- GE-7038 enhance qping to show compressed data size if applicable
- GE-7041 qalter might cause error message and reject the modification if combined with -w
- GE-7045 backup might fail with "mv: cannot move ... File exists"
- GE-7054 Integrate zlib into build process
- GE-7057 Rescheduling due to preemption does not respect force-switch
- GE-7058 make compression configurable
- GE-7064 SGE\_HGR\_ variables are set to wrong values when consumables are requested via -masterl and -l
- GE-7065 debiting of PE jobs is wrong if they contain a "-masterl" request
- GE-7099 correct the message displayed during upgrade procedure
- GE-7122 write -xd parameters into job config for non docker job
- GE-7134 update berkeley db libs to current version (6.2.x)
- GE-7135 update postgresql libs to current version 10.x
- GE-7146 mbind nlocal not functioning properly (even in 8.6.0)
- GE-7159 Increase performance of worker threads for job delivery
- GE-7177 integrate performance tools support for UGE/LO build process and TS
- GE-7179 optimize RQS code that consumes much cpu time in parsing for lWhere
- GE-7180 optimize scheduler runtime by optimizing CULL functions and other low level UGE functions
- GE-7185 AR verification of jclass parameter is either missing or does not handle special keywords
- GE-7190 update nvidia cudatoolkit used for cuda\_load\_sensor to current version
- GE-7197 interactive Docker jobs do not end if the docker container is killed and no input is provided to the qrsh client
- GE-7202 category string in accounting file is wrong for RSMAP requests
- GE-7205 greedy RR: job is backfilled although it should not
- GE-7206 greedy RR: qstat -rr shows negative job ID
- GE-7225 remove GUI installer from distribution packages
- GE-7234 cgroups support does not work with Docker 1.13.1-63 provided by CentOS
- GE-7245 Can't update from 8.4.5 to 8.5.5 with CSP configured
- GE-7249 The qacct -A switch is showing wrong values
- GE-7250 restarting sge\_qmaster with bdb spooling invalidates sharetree node ids in qconf -sstree output
- GE-7252 No config-update for execution daemons if reader threads are disabled
- GE-7275 wrong handling of thread local storage can trigger crash on qmaster shutdown
- GE-7281 qalter -V exit code is incorrect when environment variables like LD\_LIBRARY\_PATH are set

GE-7282 performance degradation in scheduling of parallel jobs  
UWSA-86 authentication issues with RESTapi  
UWSA-149 get jobs call hangs for ugerest  
UWSA-192 REST should handle multiple simultaneous requests of addJob  
UWSA-201 Resource usage not reported for pe jobs via ugeREST

#### 4.3.5 8.6.1

GE-6950 A Docker job requesting duplicate mount points sets the host in error state  
GE-7285 Make python-JSV work with version 2 and 3  
GE-7290 qconf -sconf requires manager privileges  
GE-7328 qstat -r is showing wrong values for requested resources  
UWSA-206 ugerest api is failing with TLS memory allocation error

#### 4.3.6 8.6.2

GE-7350 jobs are no longer started after having been modified via qalter by job name  
GE-7354 do not use jemalloc on lx-arm64

#### 4.3.7 8.6.3

GE-5257 Customer would like back the -binding option for qlogin  
GE-6053 for docker jobs need to create additional binds  
GE-6207 qstat -j<multiple\_job\_ids> prints all reservation times of all jobs for every job  
GE-6542 A more detailed 'Best-Fit' example should be added to the admin guide  
GE-6813 removed unresolvable admin and submit hosts re-appear after qmaster restart with BDB spooling  
GE-6848 qconf -sds should also show cluster queues that have no queue instances  
GE-6890 for Docker jobs and execd spooling on root\_squash too many files have write permissions for others  
GE-7055 Support docker run --env option by qsub -xd for docker container job  
GE-7089 support bulk changes for exec host and project objects  
GE-7096 support for nvidia-docker 2.0  
GE-7142 qstat -j shows env=NONE for jobs submitted with variable without value (-v var=)  
GE-7335 execd crash due to expand\_path ~ instead of ~/  
GE-7345 lothread should wait for lodail to be ready before sending state changes  
GE-7349 description of DISABLE\_NAME\_SERVICE\_LOOKUP\_CACHE in sge\_conf is incorrect  
GE-7351 sge\_shepherd stays running if Docker job is deleted

GE-7365 RSMAP-request with XOR does not work when RSMAP is defined with a range  
 GE-7367 gdi\_request\_limits doesn't work as documented for qmod requests  
 GE-7370 support Docker up to version 18.06  
 GE-7373 formatting issues with some man pages makes option search impossible  
 GE-7382 cgroup subdir\_name cannot always handle subdir-structures  
 GE-7390 qselect with RSMAPs returns no queues or strtok errors  
 GE-7395 SGE\_HGR\_TASK variable should also be exported for non PE jobs  
 GE-7400 (Re)opening BDB database might fail without corresponding logging in the message file  
 GE-7407 timer for job cleanup of tightly integrated job not setup on qmaster restart  
 GE-7408 timeout handler for cleanup of tightly integrated job must request global lock  
 GE-7409 no setup of queued signal handler for jobs at qmaster restart  
 GE-7410 no error message if qselect is used with -l without a value  
 GE-7412 qmaster is not accepting new RW requests  
 GE-7414 SGE\_HGR\_TASK\_ variable is exported for jobs that request RSMAPs as soft request but don't get any  
 GE-7418 added more logging information for job report handling  
 GE-7423 reaching reschedule limit for job crashes qmaster  
 GE-7427 inst\_sge -upd-ars fails on Ubuntu Linux  
 GE-7429 qmaster crash in commlib

#### 4.3.8 8.6.4

GE-4340 enable -V switch for qrsh (without command) and qlogin command  
 GE-5805 Provide limit to 'exit 99' re-scheduling attempts  
 GE-6235 Provide a way to change execd startup behaviour -> infinite connection retries  
 GE-6451 sge\_do\_log() function doesn't try to write a panic file if logging to the messages file is not possible  
 GE-6473 sge\_share\_mon issues  
 GE-6512 implement a way to suppress the user switch inside docker images for docker jobs  
 GE-6650 Make GPU affinity manually configurable  
 GE-7036 more info on qmaster message log when -tcon yes for non array job  
 GE-7088 add a CLIENT\_COMMAND parameter to JSV  
 GE-7209 named pipe execd-shepherd file descriptor leak  
 GE-7271 support to set "loglevel" in local cluster configuration  
 GE-7310 ensure UGE compatibility with DCGM versions up to 1.4.6  
 GE-7337 -xd --shm-size doesn't work correctly  
 GE-7359 More information in qmaster message log when job request invalid host  
 GE-7388 "commlib returns can't find connection" errors for "qsub -sync" jobs  
 GE-7438 frequent logging of "deletion of one or more tasks skipped for job..." in qmaster messages file  
 GE-7440 add support for -umask switch to qrsh

GE-7442 jobs get RSMAP id assigned if -l rsmmap=0 is requested  
 GE-7446 cgroups: Some slave-tasks might not get started in cgroup on a host  
 GE-7447 rpm package ugerest openjdk dependency fails  
 GE-7450 qmon does not show qinstances anymore  
 GE-7454 Allow a job to identify how often the job was executed in the past  
 GE-7455 rescheduling: rerun\_limit does not work with "exit 99"  
 GE-7457 add execd\_param to disable docker image check in sge\_execd  
 GE-7458 Docker jobs are not rescheduled on exit 99  
 GE-7462 qconf -Ae <directory> failed with internal error messages  
 GE-7468 RQS report value double than what actual usage  
 GE-7478 generation of debug information when modifying host groups might impact performance  
 GE-7481 Combine processor reports and exec host load reports  
 GE-7485 optimize performance of modification of host groups (qconf -Mhgrp)  
 GE-7486 optimize performance of exec host deletion  
 GE-7497 improve scheduling time for clusters having slots defined on exec host level  
 GE-7509 queue initial\_state=disabled should not disable queue after qmaster restart/migration  
 UWSA-208 ugerest configuration is partially destroyed when running distinst  
 UWSA-210 libjgdi.so reports undefined symbol: rsmmap\_request\_to\_string  
 UWSA-211 load\_values do not show up in uge rest api call execheosts/global

#### 4.3.9 8.6.5prealpha\_devbuild\_1

GE-4575 deleting array tasks can block qmaster for a significant time  
 GE-4638 showq does not take d\_rt into account  
 GE-4823 Missing header in sge\_conf(5) for cgroups killing parameter  
 GE-5730 support UGE on IBM Power 9 architecture on Linux (lx-ppc64le)  
 GE-5912 comma-separated userset-lists are not accepted  
 GE-6584 mem\_free should be allowed to increase even for running jobs  
 GE-6700 low m\_mem\_free load values prevent job dispatching  
 GE-6749 "qstat -s r" is broken for array jobs with suspended tasks  
 GE-6750 "qdel -s r" is broken for array jobs with suspended tasks  
 GE-6801 accounting man page field description is wrong for "wallclock"  
 GE-6805 incorrect booking of consumables results in scheduling errors and massive error logging  
 GE-6830 enable RSMAP placeholders to be resolved for -masterl requests, too.  
 GE-6832 Add more information from man page to userguide/admin guide  
 GE-6863 job with -pty yes failed if output was directed to folder with permission 754  
 GE-6958 execd should also cleanup dockers cgroups dir inside cgroups uge jobs dir if not done by docker  
 GE-6978 UGE/cgroup leaves orphan processes after qdel if the job is submitted within an tightly integrated openmpi  
 GE-6990 DRMAA2 Python binding to support customer workflow  
 GE-7244 ensure that all generated man page (not only NROFF) are part of the packages

- GE-7268 projects & xprojects fields in execution host configuration, not allowing comma separated values
- GE-7332 pending job with "-tcon y" and "-t" prevents other jobs from running
- GE-7337 "-xd --shm-size" doesn't work correctly
- GE-7343 add a commlib parameter not to do hostname resolving but parse the ip address out of the hostname
- GE-7383 put all queue instances on host in error state on Cgroups related errors
- GE-7391 PE slave tasks should be handled in own cgroups
- GE-7434 Use OOM Notification API in UGE Cgroups implementation to notify when jobs exceed memory
- GE-7443 'sgeexecd stop' fails if corresponding host is no admin host
- GE-7469 Unable to edit queue slots configuration using qconf -datr
- GE-7475 enhance monitoring information of reader thread that does event processing
- GE-7476 sge\_get\_config\_version\_for\_host() does hostname resolving which can be avoided
- GE-7482 handle permission check GDI requests in listener thread.
- GE-7483 add a possibility to give manager requests higher priority
- GE-7487 use sge\_mutex\_lock instead of pthread\_mutex\_lock in all modules used by sge\_qmaster
- GE-7489 improve performance for qmod -rq
- GE-7490 new added execution hosts not handling reschedule\_unknown setting
- GE-7492 bad dispatching rate with many big array jobs in a huge cluster
- GE-7495 rescheduling of sequential jobs unnecessarily creates timer events
- GE-7496 enable cgroups based killing for PE tasks
- GE-7499 shepherd should not be part of the memory Cgroup
- GE-7502 add scheduler param to suppress sending of running job tickets
- GE-7503 improve general rescheduling performance
- GE-7504 improve reschedule\_unknown triggered rescheduling behavior on massive execution host lost
- GE-7505 add additional worker pool that allows to handle priority requests
- GE-7506 improve performance of adding large number of execution hosts
- GE-7511 newline in jsv\_add\_env function let qsub crash
- GE-7513 unexpected logging: invalid task number 0 for job ... in "ORT\_ptickets" order
- GE-7519 execd might not always send a load report within configured load\_report\_time
- GE-7520 unexpected connection renewal of execution daemons
- GE-7528 qconf -si sid is ignored if not used as first switch
- GE-7540 jobs do not get the correct binding if affinity is requested and the affinity mask is overwritten with a topology mask
- GE-7546 job cleanup problem for PE jobs at execution daemon
- GE-7550 do not report m\_mem\_free\_nX/m\_mem\_used\_nX as load values
- GE-7556 implement deadlock detection for sge\_qmaster daemon
- GE-7574 concurrent array jobs (-tcon yes) get partially started when the number of running tasks exceeds maxujobs
- GE-7586 qmaster becomes unresponsive and needs 100% CPU after qalter

- GE-7592 ensure UGE compatibility with DCGM versions up to 1.5.6
- GE-7594 resources held by a standing reservations are not properly freed when the standing reservation is modified or is deleted after sge\_qmaster restart
- GE-7595 standing reservation id restarts at 0 after modifying resource requests with qralter
- GE-7597 resources held by a standing reservations are not properly freed when the complex\_values of the queue are modified
- GE-7654 massive logging of "GDI session <string>: created=<nr>, processed=<nr>, required=<nr>"

#### 4.3.10 8.6.5prealpha\_devbuild\_2

- GE-3942 qmon man page refers to schedd\_conf(5)
- GE-3946 Explain priority normalization and fix issues with sge\_priority man page
- GE-4020 getservbyname segfaults when called with wrong amount of arguments
- GE-4381 Enhance error message logging for flat file spooling
- GE-4402 SGE\_DRMAA\_ENABLE\_ERROR\_STATE variable is not documented in wiki pages
- GE-4469 Need clear description how to handle certificates within UGE CSP installations
- GE-5125 Add addition information in man page for qsub
- GE-5305 jsv does not recognize -now switch
- GE-5346 qdel <jobid> -t <taskid> deletes complete job
- GE-6751 qalter -clearp and -clears doesn't delete "hold\_jid" attributes
- GE-7517 queue is set in error state if Docker daemon is overloaded
- GE-7553 SR in error state because past SR instance is in "E" state
- GE-7572 Docker background process doesn't allow to exit the interactive job
- GE-7580 cgroup directory of docker job remains although job finished
- GE-7581 PE job with consumable HOST request doesn't get correct resources if max\_reservation is greater than 0
- GE-7587 Fix spelling errors in AdminGuide
- GE-7596 docker -xd command can't export multiple values
- GE-7598 Docker jobs set queue instance in error because permission to write pe\_hostfile is denied
- GE-7626 consumable records in the reporting file end with a delimiter (comma)
- GE-7645 job killed by cgroups limit results in zero values for "ru\_\*" usage fields
- GE-7646 job killed by cgroup OOM notification is pending in "E" state
- GE-7656 need parameter to enable qmaster abort() when a deadlock was detected
- GE-7657 improve deadlock detection reliability
- UWSA-230 Restarting Qmaster Breaks Synchronization With UGEREST

**4.3.11 8.6.5prealpha\_devbuild\_3**

- GE-4378 Respect better parsing for admin user in bootstrap file
- GE-5513 "The filename or extension is too long" error when creating job process on native Windows (win-x86)
- GE-5801 Windows (win-x86) qloadsensor.exe doesn't recognize if execd is killed or crashed
- GE-5826 port RESTRING matching to native Windows (win-x86)
- GE-6019 Improve error logging of native Windows (win-x86) RedirectStdHandles() function
- GE-6222 UGE Starter Service of native Windows (win-x86) fails to start second execd if the first cannot be started
- GE-6223 add check to native Windows (win-x86) installer if an execd is already installed to the same \$SGE\_CELL or uses the same port
- GE-6226 the native Windows (win-x86) execution daemon doesn't report m\_cores, m\_sockets and m\_threads
- GE-6809 Could not edit non reserved complex when advance reservation with pe is running
- GE-6896 qconf -me parser allows to define a RSMAP twice which cause sequential errors
- GE-7059 qsub job verification "-w e" is executed before server side JSV run
- GE-7323 sgepasswd tool enhancement or change of behaviour
- GE-7623 wrong debiting of consumables with consumable type HOST
- GE-7624 support Docker up to version 18.09
- GE-7641 newlines in qmaster messages file after hold modifications breaks file format
- GE-7697 it is possible to remove a host exclusive consumable from a host which is referenced in an AR
- GE-7698 rqs: wildcard projects with at least two different limits crashed qmaster
- GE-7704 make sure libxml2.dll exists on native Windows (win-x86) exec host
- GE-7705 port work binary to native Windows (win-x86)
- UWSA-223 ugerest should not truncate /tmp/UGERestService0.0.log at startup

**4.3.12 8.6.5alpha1**

- GE-3938 Man page sge\_priority needs prio/pprio clarity
- GE-4391 add more information about the importance of the gid range to the install guide
- GE-4454 The sge\_priority man page has a typo regarding prior calculation
- GE-5299 A way to remove hold\_jid switch
- GE-6457 log full communication between shepherd and Docker daemon to the shepherd trace file
- GE-6802 job are handled as still running, even if all processes are already stopped
- GE-7028 Completed job remains in dr state and is not cleaned up at execd
- GE-7243 man page qsub -P description incorrect

- GE-7253 errors with consumable capacity management after "qalter -when now" or after reduction of capacity
- GE-7336 typos in AdminsGuideGE, chapter 3.9.5, Tuning the sge\_shadowd
- GE-7404 cgroups based devices blocking mechanism does not block all devices
- GE-7606 ./inst\_sge -copycerts <hostlist> command line parsing not correct
- GE-7631 Docker -xd doesn't work for --sysctl option
- GE-7639 qmaster crashes in scheduler thread in scheduler\_method()
- GE-7664 qloadsensor exe reports wrong core counts
- GE-7674 placeholder '\$' character can't work with double quote("") unlike manual described
- GE-7693 re-schedule standing reservations without allocation or being in error state
- GE-7702 even if it is already installed, native Windows (win-x86) installer always asks for SGE admin user password to install UGE\_Starter\_Service.exe
- GE-7710 handle leak in UGE Job Starter Service causes start failures of some jobs
- GE-7720 active SR stays in "E" state after qmaster restart even after all execds reconnect
- GE-7724 not all markdown man pages are part of the packages
- GE-7737 qalter -when now might break EXCLUSIVE consumable functionality
- GE-7750 synchronize execd signaler thread with execd main thread
- UWSA-157 Job State can be stale until full job refresh
- UWSA-176 clusterqueuesummary contains null entry when hostlist is NONE in queue
- UWSA-216 Cannot view tasks in Rerun Queue (Rq) State
- UWSA-222 ugerest connection and shutdown issues
- UWSA-224 ugerest does not log its version information at startup

#### 4.3.13 8.6.5alpha2

- GE-6648 Make GPU/CPU affinity a possible hard and soft request
- GE-6652 Implement DCGM callbacks in case of GPU failures
- GE-7362 Document RSMAPs in man-pages and Usersguide, including their request-syntax
- GE-7673 RQS computes double usage for RSMAPs of type HOST for PE jobs
- GE-7748 formatting issue makes subordinate example impossible to read
- GE-7762 ensure UGE compatibility with DCGM versions up to 1.6.3

#### 4.3.14 8.6.5

No additional fixes compared to 8.6.5alpha2

#### 4.3.15 8.6.6

- GE-7767 wrong accounting exit\_status=52 if failed=52 due to

- cgroups killing API support
- GE-7778 start interactive jobs in a way that exported \$PATH gets not overwritten
- GE-7783 remote automatic execd uninstallation starts itself infinitely on remote hosts
- GE-7790 drmaa2 python crash qmaster
- GE-7791 \$PATH set twice in environment prohibits qrsh jobs with (t)osh as login shell to find commands
- GE-7793 pattern matching for hostnames in qsub is broken

#### 4.3.16 8.6.7alpha1

- GE-3271 use syslog during the time where components would normally log into log files located in /tmp
- GE-3687 documentation for resource reservation needs to be more detailed
- GE-4531 Explain in man qsub "-R" what the option does
- GE-5301 mention in documentation that client side JSV is not supported on native Windows (win-x86)
- GE-6160 qrstat should show ARs where user has access
- GE-6179 add a means to switch on and off debug logging (DPRINTF) of sge\_execd during runtime
- GE-6525 document exit status (of intermediate accounting records) and failed field in reporting(5) and accounting(5)
- GE-6740 the configuration parser cannot handle white spaces before and after configuration values
- GE-6809 Could not edit non reserved complex when advance reservation with pe is running
- GE-6962 Put all remaining array job tasks into hold state instead of queue into E state
- GE-7046 qstat -xml has xml syntax issue with qrsh in some specific case
- GE-7421 prevent win-x86 job processes from running away by working around the Windows Job Object race condition
- GE-7484 RSMAP on global host breaks all RSMAPs
- GE-7619 qhost, qstat,... -xml show wrong path to schemaLocation file
- GE-7640 queue instance stays in orphaned state forever after non-admin user forced job deletion
- GE-7649 collect information about CPU/GPU licensing
- GE-7655 "qconf -sep" -cb not ready for counts >= 100k
- GE-7661 Query GPU count on every execd
- GE-7662 Report GPU count on execd to qmaster
- GE-7741 Add host\_provider=<string> exed\_param that allows Launch to tag cloud based execution nodes
- GE-7759 drmaa2 error: smallest event number n is greater than number 1 i am waiting for
- GE-7769 provide transactional trigger to update host\_aliases file
- GE-7770 ensure RHEL 8 support for UGE
- GE-7801 update info about cuda loadsensor in UGE documentation
- GE-7814 Scan pci bus for Nvidia 3D controller if CUDA is not available.

- GE-7815 Change UGE pdf documents to standard Univa font and some typos found during review.
- GE-7820 add information about multiplier/shortcut to RSMAP man pages and documentation
- GE-7828 remove usermappings.5 man page from distribution
- GE-7831 remove qtcsh.1 and qtask.5 man page from distribution
- GE-7832 UGE installation fails when service names are used for qmaster and execd ports
- GE-7833 builtin CUDA topology detection creates wrong affinity masks
- GE-7840 document syntax of RSMAP XOR in sge\_resource\_map(5) and sge\_types(1)
- GE-7848 qconf accepts invalid values for boolean parameters
- GE-7853 allow native Windows (win-x86) work.exe binary to fork itself
- GE-7859 Docker interactive job creates infinite shepherd logging and huge trace file
- GE-7860 ALLOW\_ANY\_SUBMITHOSTS=TRUE as qmaster\_param should allow submission from any host independent of of the defined submit hosts
- GE-7865 drmaa v1 example code showing wrong include path
- GE-7868 sge\_qmaster crashes when qalter -w p is used with a list of job\_ids
- UWSA-231 UGERest does not show job stat of PE job which run in multiple nodes
- UWSA-233 UGERest seems to crash regularly in a Navops Launch demo installation

#### 4.3.17 8.6.7

- GE-5839 Provide a method for REGEX matching on strings longer than 2048 characters
- GE-6740 The configuration parser cannot handle white spaces before and after configuration values
- GE-7862 callback function for event "EVENT ADD LICENSE USAGE RECORD" failed
- GE-7874 m\_mem\_free memory limit might not work if hostname resolving differs on execd hosts
- GE-7878 dbwriter installation fails on PostgreSQL 10.9
- GE-7648 provide job "failed" reason in DRMAAv1 and DMRAAv2
- GE-7837 Job Category causes scheduler to ignore scheduling an eligible job
- GE-7558 sub-cgroup feature

#### 4.3.18 8.6.8prealpha\_devbuild\_1

- GE-3894 cleaning up job runtime files, using linux namespaces
- GE-6151 job error mail not helpful because error message is cut off
- GE-6593 sge\_execd hangs, jobs stay in t state
- GE-6653 Use DCGM to provide additional information about GPU usage of jobs
- GE-6672 execd\_params S\_DESCRIPTOR cannot be increased to values > 65535
- GE-6961 JSV does not work if hold\_jid list is too long
- GE-6963 disallow built-in share-tree usage values as complex attributes
- GE-7087 quota keeps the limit of complex although qalter release resources
- GE-7137 qstat -q <queue\_name> filter seems to be broken, when job requests RSMAP
- GE-7325 output of memory information is broken on lx-amd64
- GE-7493 LOG\_REQUEST\_PROCESSING\_TIME messages for scheduler orders should contain the order type
- GE-7758 sub-cgroup feature

GE-7593 Enable monitoring and deadlock detection for all platforms  
 GE-7707 SGE\_BINDING env variable can be overwritten by shell  
 GE-7797 drmaaV2-python JobSession example job submission crashes qmaster  
 GE-7816 XOR RSMAP request issues cause jobs staying pending  
 GE-7822 sge\_shepherd is burning 100% cpu when running an IJS job  
 GE-7854 the names of cuda load values delivered via dcfgm shouldn't contain the host name  
 GE-7866 commandline and build instructions for drmaa v1 c code is wrong  
 GE-7867 man page shows default value for finished\_jobs setting of 100  
 GE-7899 provide profiling about jemalloc memory allocation and heap size  
 GE-7905 qmaster got stalled four minutes after reinitialization of scheduler  
 GE-7911 qstat for ulx-x86 dumps core if used gcc is < 4.4.7  
 GE-7917 timeouts due to small buffer causes getgrgid\_r() errors with failing clients and job starts  
 GE-7921 execd for arch=sol-x86 cannot connect in csp mode  
 GE-7924 qconf -Ace/-ace might fail if the attributes are in the wrong order  
 GE-7926 Config API does not support setting 'affinity' on complex resources  
 GE-7927 PE AR does not reserve the correct amount of HOST consumables  
 GE-7933 make config-api compatible with python3  
 GE-7935 loadcheck truncates m\_topology after 100 characters  
 GE-7941 loadcheck binary does not show the same DCGM load values as qconf -se  
 GE-7943 update jemalloc to current version 5.2.1  
 GE-7958 add support for Docker 19.03  
 GE-7960 qrsh\_starter might crash because it does not init thread local monitoring setup  
 UWSA-237 hostsummary doesn't show gc, hc values

#### 4.3.19 8.6.8

GE-5943 reliable drmaa2 library testsuite testing  
 GE-5944 jdrmaa2 junit testing  
 GE-6651 Add support for DCGM configuration changes  
 GE-6723 qstat -explain m -xml does not show Administrator message  
 GE-6730 pe scheduling: accept first possible assignment  
 GE-6731 pe scheduling: accept up to n soft violations  
 GE-6732 pe scheduling: accept suboptimal assignment if dispatching already took a configurable amount of time  
 GE-6733 pe scheduling: additional algorithm for PE\_SORT\_ORDER  
 GE-6769 qstat -flt project <projectID> also shows job in qw without filtering the projectID  
 GE-6789 qstat -q filter broken for project jobs if queue has projects list set to NONE  
 GE-6961 JSV does not work if hold\_jid list is too long  
 GE-6984 config-api package is unusable and its doc is not clear about install/build  
 GE-6986 Jobs which request binding that can be fulfilled by fewer RSMAP ids do not get any Ids assigned  
 GE-7007 support HOST consumable for INT and DOUBLE

- GE-7393 Create a specification for an enhancement that allows PE task specific resource requests.
- GE-7444 Create specification for OR request
- GE-7541 Docker: automatically add user's supplementary groups to `-xd --group-add`
- GE-7575 add support for CPU/GPU reporting/licensing
- GE-7621 qmod message is cleared when any change is made to queue
- GE-7667 starting Windows (win-x86) GUI jobs without letting them show the GUI on the visible desktop does not work
- GE-7766 jobs with topology/affinity mask and `-binding` request may not be scheduled correctly
- GE-7824 drmaa2-python pypi package support
- GE-7827 Disable job submission when grace period is expired
- GE-7837 Job Category causes scheduler to ignore scheduling an eligible job
- GE-7855 useless profiling output in qmaster messages file
- GE-7870 support SLES 15
- GE-7896 Greedy Resource Reservation is reserving nodes that are not available
- GE-7912 `'-e ~/foobar'` results in `getpwnam_r()` errors and sets job in error state
- GE-7915 Create UGE packages that contain the used open source code
- GE-7920 no or misleading error message when `SGE_ROOT` is not set
- GE-7922 Legacy queue Sequence Number is not working using the Admin doc suggested config
- GE-7929 issues with automatic GPU detection
- GE-7930 qalter / qconf segmentation fault at client setup errors
- GE-7948 update documentation about consumables in `usage_weight_list`
- GE-7951 ensure UGE compatibility with DCGM versions up to 1.7.1
- GE-7969 sol-amd64 support on AMD Opteron Sun hardware broken
- GE-7970 DCGM affinity/binding does not work if the qmaster is running on a host that isn't lx-amd64
- GE-7971 Investigate error with `"qconf -Mlur"` when using a host with `execd_param "host_provider"` set
- GE-7974 show in monitoring how long the global lock was held
- GE-7975 improve profiling of worker threads
- GE-7979 Put all remaining array job tasks into hold state on job error
- GE-7989 deletion of jobs running on native Windows (win-x86) can take minutes
- GE-7997 soft requests are ignored for jobs requesting wildcard pes
- GE-8006 have a means to limit the size of the `submit_cmd` written to `accounting/reporting`
- GE-8008 segv of qconf when `SGE_ROOT` is not set
- GE-8018 Allow to disable qinstance access checks when jobs are submitted
- GE-8031 `qstat -flt` accepts a list of projects/pes, but applies filter only with the first name in that list
- GE-8043 `"qconf -*attr"` should deny to make operations on "context" attribute
- GE-8045 improve profiling of qmaster threads and `execd`
- GE-8051 qsub nested argument behavior changes
- GE-8056 RSMAPs on global host are not assignment correctly if a job requests binding
- GE-8061 `SGE_HGR_TASK_` is not set correctly for RSMAPs on global host
- GE-8066 SHELL environment variable not set to login shell of user for

- qrsh without command
- GE-8096 qacct -j <jobid> crashes on native Windows (win-x86)
  - UWSA-209 add version functionality in the help menu
  - UWSA-234 Restart issues when ugerest runs on non master host
  - UWSA-235 UGE REST does not reliably start
  - UWSA-236 UGERest installation of Scenario 2 documentation should be checked fully if this works
  - UWSA-237 hostsummary doesn't show gc, hc values
  - UWSA-238 Documentation shows wrong URI path for exehosts endpoint
  - UWSA-239 Jobs in Eqw state are not returned in jobsummary rest call
  - UWSA-240 install\_ugerest reports success even ugerest cannot start due to binding error

#### 4.3.20 8.6.9

- GE-6041 When adding a message (-msg "") disabling all queue instances on a host only one instance has message. not all, as should be.
- GE-6078 Messages stored for queue instances (qmod -msg) are not deleted when host is removed from queue
- GE-7726 "context" in queue instance remains when queue message gets deleted or host removed from queue
- GE-7957 Removing queue entries with "qconf -purge" not working for not resolveable hosts
- GE-7991 Message applied to a group of qinstances during a state change is only assigned to qinstances of one cqueue
- GE-8078 Optimize limit rule handling using ACLs
- GE-8162 resource debiting issues cause poor cluster utilization
- GE-8167 "qconf -purge" not working on "context" attribute
- GE-8171 resource diagram should also contain the current state of all RSMAP ids
- GE-8187 remove info logging about CUDA runtime library

#### 4.3.21 8.6.10

- GE-8194 UGE pseudo environment variables replacement not working for qsub -e/-o

#### 4.3.22 8.6.11

- GE-2218 Add new option -name to uidgid binary to return username
- GE-3838 add "how to configure the Windows host" to the "Preparing Windows Hosts" section of the installation guide
- GE-4257 In man page for sge\_conf and sge\_execd, correct/add load\_report\_time references
- GE-4258 define the units of variables in sge\_conf man page
- GE-4458 allow specification of terabytes and petabytes for memory values
- GE-5253 qmod -cj job exit with wrong status

- GE-5809 wrong error messages if RSMAP complex values are defined on queue level
- GE-6041 When adding a message (-msg "") and disabling all queue instances on a host only one instance shows the message
- GE-6078 Messages stored for queue instances (qmod -msg) are not deleted when host is removed from queue
- GE-6844 analyze logging "context" of "all.q" is empty - Adding new element(s).'
- GE-7030 XOR RSMAP request syntax rejected in JCs
- GE-7274 memory usage values are negative on windows 7 and 10 hosts
- GE-7287 event handler callback functions fail after QMaster restart
- GE-7352 qalter -rdi y prints wrong message if schedd\_job\_info is turned off
- GE-7726 "context" in queue instance remains when queue message gets deleted or host removed from queue
- GE-7763 allow UGE to work with unsupported new Docker versions
- GE-7872 qalter and qmod job changes are not logged in reporting file
- GE-7932 ability to suppress resolving/sending of additional group ids by client commands
- GE-7950 qresub prints additional message to stdout
- GE-7957 Removing queue entries with "qconf -purge" not working for not resolveable hosts
- GE-7976 verify and improve the spooling and spooling-io profiling for classic spooling
- GE-7977 make warning time for host name resolving configurable
- GE-7978 LOG\_SPOOLING\_TIME/LOG\_REQUEST\_PROCESSING time should always log as WARNING, document content
- GE-7991 Message applied to a group of qinstances during a state change is only assigned to qinstances of one cqueue
- GE-8010 wrong scheduling decisions with -masterl switch and max\_reservation > 0
- GE-8015 openmpi.template has wrong defaults
- GE-8036 RQS "hosts" quotas not updated when hostgroups in a hostgroup are changed
- GE-8037 enable schedd\_runlog and debug tracing in Scheduler for a specific job only
- GE-8044 do not resolve in clients UGE's additional group id's and defineable ranges
- GE-8058 disable topology masks for RSMAPs on global host
- GE-8060 sgemaster segfault on Ubuntu
- GE-8070 pending array job with -hold\_jid\_ad dependency cannot be deleted
- GE-8071 wrong or no reservation if all hosts/queues are blocked by an EXCL complex
- GE-8073 enable schedd\_runlog and debug tracing in Scheduler for a specific job only
- GE-8078 Optimize limit rule handling using ACLs
- GE-8089 sge\_shadowd is restarted by SMF service on term signal
- GE-8091 cleanup SMF module
- GE-8103 requested resources are not freed after deleting a suspended job
- GE-8162 resource debiting issues cause poor cluster utilization
- GE-8165 event master does not handle lWhere filterting without lWhat setting
- GE-8167 "qconf -purge" not working on "context" attribute
- GE-8170 ensure UGE compatibility with DCGM versions up to 1.7.2

GE-8171 resource diagram should also contain the current state of all RSMAP ids  
 GE-8187 remove info logging about CUDA runtime library  
 GE-8191 job deletion in qmaster ignores MAX\_JOB\_DELETION\_TIME for  
 not enrolled array task jobs  
 GE-8192 improve job array task deletion performance  
 GE-8194 UGE pseudo environment variables replacement not working  
 for qsub -e/-o  
 GE-8199 install script problems on solaris 10 hosts  
 GE-8200 RSMAP requests in job classes are not correctly displayed/spooled  
 GE-8203 Windows jobs show wrong error behaviour when sgepasswd entry is wrong  
 GE-8206 used\_slots in PE may go up quickly and prevent job dispatching  
 GE-8209 unexpected logging of "commlib info: got read error (closing qmaster)"  
 in qmaster messages file  
 GE-8214 qalter -w p keeps worker thread 100% busy without ever ending  
 GE-8216 formatting and semantics issue with schedd\_job\_info messages  
 GE-8218 wrong logging of job states in messages files  
 GE-8222 lothread restart or uge qmaster restart can result in UGE jobs  
 hanging in "l" state  
 UWSA-243 Add Python 3 REST API endpoints  
 USWA-244 show additional category\_id of job

#### 4.3.23 8.6.12

GE-7983 user share tree gives no tickets to jobs submitted in a project  
 GE-8087 incorrect scheduling/debiting of JOB consumables on global host  
 GE-8235 incorrect booking of resources in SRs  
 GE-8243 (implicit) -cal\_jmp 0 ignored for SR creations  
 GE-8245 qrstat output issues  
 GE-8246 wrong timed\_event thread error logging on AR rescheduling  
 GE-8247 qacct -q without -l request might report unexpected error logging  
 GE-8253 modifying the end time of a standing reservation can lead to  
 unavailable resources

#### 4.3.24 8.6.13

GE-8248 shepherd error on hosts with no suitable devices if cgroups  
 parameter "device" is defined  
 GE-8269 per JOB consumable RSMAPs may cause dispatching and job execution  
 issues  
 GE-8280 wrong behaviour for pending jobs if the consumable attribute of a  
 RSMAP is changed  
 UWSA-249 Job Array Concurrency Value Not Available  
 UWSA-250 Provide the amount of pending tasks of an array job

#### 4.3.25 8.6.14

GE-6377 Move libdrmaa.so to own separate lib directory

GE-7535 Job submission by 'qrsh -verbose' prompts message "JSV has been started/stopped"  
 GE-7856 -tc doesn't limit the number of tasks with MIN\_PENDING\_ENROLLED\_TASKS  
 GE-8302 parsing issues with floating point numbers as RSMAP ids  
 GE-8073 missing usage and start/end time for docker jobs with h\_rt limit  
 GE-8267 drmaa2-python library\_manager.py popen  
 GE-8273 Replace EULA in installer with most current version  
 GE-8288 Add support for LMDB in UGE and LO  
 GE-8289 DB transaction errors with large read operations  
 GE-8290 incorrect spooling of qinstance on signalling of queue  
 UWSA-251 cannot get information about available PE using ugerestsdk.js

**4.3.26 8.6.15**

GE-8292 qstat data transfer and qmaster size increasing  
 GE-8308 add migrate\_spooling.sh to common package  
 GE-8313 potential deadlock in LMDB  
 UWSA-254 Automatic display of standard complex resource types in UGERest

**4.3.27 8.6.16**

GE-8298 add support for the Docker --gpus parameter  
 GE-8353 ensure UGE compatibility with DCGM versions up to 2.0.10  
 GE-8369 use cuda\_id parameter to resolve GPU RSMAPs for Docker jobs  
 GE-8375 qmaster deadlock with LMDB spooling  
 GE-8382 add Docker version as requestable load value  
 UWSA-257 Add support for Execution Time to Job Info

**4.3.28 8.6.17**

GE-8242 resource diagram improvements  
 GE-8356 Quoting of command lines causes job start errors in (t)csh  
 GE-8362 resource diagram doesn't show full queues  
 GE-8379 modifying hostgroup objects used in queue configuration object can create unexpected queue instances  
 GE-8385 wrong RSMAP booking prevents job dispatching to available resources  
 GE-8402 update manuals with new logo and copyright text  
 GE-8404 execds need 5 minutes to reconnect after being suspended/clock changed  
 GE-8405 gpu job usage does not work with DCGM versions lower than 2.0.10  
 GE-8407 qrsh job is not scheduled immediately  
 GE-8409 errors in qmaster messages file with server-side JSV rejecting all jobs

## 5 Upgrade Notes

### 5.1 Upgrade Requirements

This is a summary of the Upgrade Matrix that describes how you can carry out the transition from Sun or Oracle Grid Engine 6.2uX, Univa Grid Engine 8.X.X to Univa Grid Engine 8.6 when you are currently using classic or database spooling (PostgreSQL, BDB). If the current version of Grid Engine you are using is missing in the overview, then please look at the full Upgrade Matrix located in the section Updating Univa Grid Engine in the Installation Guide.

Version	Upgrade Method
Univa Grid Engine 8.X.X	Backup/Restore
Oracle Grid Engine 6.2u6-6.2u8	Backup/Restore
Sun Grid Engine 6.2u5	Backup/Restore
Sun Grid Engine 6.2u1-6.2u4	Upgrade to SGE 6.2u5 and then Backup/Restore
Sun Grid Engine 6.2 FCS	Upgrade to SGE 6.2u5 and then Backup/Restore

Table 16: Upgrading from SGE, OGE, UGE 8.X.X to Univa Grid Engine 8.6.X

## 6 Compatibility Notes

### 6.1 Changes in Windows Execution Host sgepasswd File

The encryption algorithm for the “\$SGE\_ROOT/\$SGE\_CELL/common/sgepasswd” file passwords has been changed from RC4 to AES-256-CBC:

If you upgrade to the current version of Univa Grid Engine you need to convert your existing sgepasswd file during the upgrade procedure.

Become root and execute the following command on the master machine:

```
# sgepasswd -c
```

This will create a backup of your original ‘sgepasswd’ file as ‘sgepasswd.old\_algorithm\_backup’ and create the new compatible sgepasswd file. Otherwise encryption related error messages may show up.

If you create a new sgepasswd file from scratch no additional steps compared to previous versions are required.

### 6.2 Scheduler Log File

In previous versions of Univa Grid Engine if both the master and some slave tasks of a parallel job were scheduled to the same queue instance, one line was written to the scheduler log file containing the number of slots that was granted to this job on the queue instance of the master task. Such a line looked like this: 249:1:STARTING:1522328357:2772638938:Q:B@host1:slots:3.00000

From Univa Grid Engine 8.6.0 on, two lines are written, one for the master task, one for the slave tasks: 249:1:STARTING:1522328357:2772638938:Q:B@host1:slots:1.00000  
249:1:STARTING:1522328357:2772638938:Q:B@host1:slots:2.00000

This is because of changed handling of the master task which was necessary to fix GE-5888, see the “Fixes and Enhancements” section for details.

### 6.3 Removed Scheduler-Parameter `queue_sort_method`

With the newly implemented affinity-feature, the scheduler-parameter `queue_sort_method` has been replaced with the new parameters `weight_host_sort`, `weight_queue_host_sort`, `weight_queue_seqno`. With these it is not only possible to configure a queue-sorting solely based on `seq_no`, or on the `host_load_formula`, as it was with `queue_sort_method`. But it is now also possible to configure a mixed sorting strategy, where the amount of the weight-parameters decide the ratio with which they influence the sorting.

For configuring what used to be `queue_sort_method = load`, one now has to set `weight_queue_seqno` to a significantly smaller number than `weight_queue_host_sort`. For `queue_sort_method = seq_no`, simply set `weight_queue_seqno` a significantly higher value than `weight_queue_host_sort`.

## 6.4 Changes for qconf Exit States

The qconf will no longer report an error when showing the content of empty configuration objects. If a configuration object contains no elements the exit status of qconf will be 0 and there will be no error message. The change affects following qconf command line options:

- qconf -scall show a list of all calendar names
- qconf -sckptl show all ckpt interface definitions
- qconf -sconfl show a list of all local configurations
- qconf -sel show a list of all exec servers
- qconf -sh show a list of all administrative hosts
- qconf -shgrpl show host group list
- qconf -sjcl show job class list
- qconf -sm show a list of all managers
- qconf -so show a list of all operators
- qconf -spl show all parallel environments
- qconf -sprjl show a list of all projects
- qconf -sql show a list of all queues
- qconf -srqsl show resource quota set list
- qconf -ss show a list of all submit hosts
- qconf -sul show a list of all userset lists
- qconf -suserl show a list of all users

## 6.5 Changes for Scheduler Profiling

The scheduler profiling option is revised and cleaned up. The most important changes are:

- Enhanced sge\_diagnostics man page with info about scheduler profiling
- The scheduler profiling table shows new profiling levels: "ticket calculation", "scheduler thread", "ssos init", "config update", "wait for order completion", "mirror events" and "set event client params".
- The scheduler profiling level "pending ticket calculatio" was renamed to "priority calculation".
- The scheduler profiling level "copy lists" was renamed to "data preparation".
- The scheduler profiling level "scheduler" was renamed to "scheduling".
- The scheduler profiling level "wait" was renamed to "waiting for events".

## 6.6 Changed Limit Calculations

The resulting limit calculation is revised and fixed. The most important changes are:

- The configured consumable type (NO, YES, JOB, HOST) will have no influence on any resulting limit for tight integrated parallel jobs.
- Previous version calculated the resulting limits by far to high (depending on pe and consumable settings)

- This also affects the cgroups `h_vmem` observation

Univa Grid Engine versions prior to 8.5.1 showed, by far, a too high limit value setting for limits like e.g. `h_vmem`. The limit adjustments are now adapted. A detailed overview of the limit observation and how it works is described in the `sge_diagnostics(1)` man page (JOB LIMITS).

ATTENTION: If you're updating to 8.6.0 from a previous version it is recommended to verify the used limit requests of your jobs. It might be necessary to change the request value. If the limit is set too low or adjusted to fit the old limit adjustment jobs which were running fine might fail after installing this version.

## 6.7 New Default for Job Verification of DRMAA Submitted Jobs

In the past a job verification (`-w e`) was triggered for all jobs submitted via DRMAA. This had a negative performance impact on `qmaster` for all DRMAA submitted jobs which might have had also a negative impact on the cluster throughput. The verification default for DRMAA jobs has now been disabled (`-w n`) for Univa Grid Engine 8.6.0. Users that nevertheless want to get jobs verified can still enable this through the DRMAA native specification.

## 6.8 Default for Integer Complexes

Older Univa Grid Engine installations allowed to request an integer complex resource during job submission without the need to specify the amount. Implicitly 1 was used when the amount of requested resources was omitted. This is not possible anymore beginning with 8.6.0.

## 6.9 Deprecated Functionality

'qalter -w p' is deprecated. Instead scheduler job information has to be used.

Starting with 8.6.5 qmaster\_param ENABLE\_REDUCE\_MEM\_FREE is deprecated and will be removed with 8.7.0. Decreasing and increasing of mem\_free is now possible without any configuration.

## 6.10 Removed Functionality

Following components/features were removed with version 8.6.0 of Univa Grid Engine:

- Graphical Installer
- qtcsch

## 6.11 Changed UGERest configuration location

Starting with UGERest 8.6.4 the configuration files are installed into \$SGE\_ROOT/\$SGE\_CELL/common/ugeresst\_conf.

This is done automatically when using ./install\_ugeresst script for installation. The reason for this change was accidental overwriting of configuration files when unpacking a patch into an existing installation. If you still need your old configuration files for further reference to existing settings, backup the \$SGE\_ROOT/ugeresst/conf directory before installing a new ugerest package file.

## 6.12 Changed UGERest restlet framework

With upgrade 8.6.11 the restlet framework version has changed from 2.3.12 to 2.4.2. This requires to run ugerest with JDK 1.8 or higher.

## 6.13 Changed drmaa library path

With upgrade 8.6.14 and above, drmaa and drmaa2 libraries and jar files are installed under \$SGE\_ROOT/drmaa and \$SGE\_ROOT/drmaa/lib/\$SGE\_ARCH. Please update your LD\_LIBRARY\_PATH accordingly.

# 7 Known Issues and Limitations

None.