N1 Grid Engine 6 用户指南  
Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A  
文件号码 817–7681–11  
2005 年 7 月  
版权所有 2005 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 保留所有权利。  
本文档及其相关产品的使用、复制、分发和反编译均受许可证限制。未经 Sun 及其许可方（如果有）的事先书面许可，不得以任何形式、任何手段复制  
本产品或文档的任何部分。第三方软件，包括字体技术，均已从 Sun 供应商处获得版权和使用许可。  
本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。 UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地  
区独家许可的注册商标。  
Sun、 N1、 Sun Microsystems、 Sun 徽标、 docs.sun.com、 AnswerBook、 AnswerBook2、 N1 和 Solaris 是 Sun Microsystems, Inc. 在美国和其他国  
家/地区的商标或注册商标。所有 SPARC 商标的使用均已获得许可，它们是 SPARC International, Inc. 在美国和其他国家/地区的商标或注册商标。 标  
有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。  
OPEN LOOK 和 Sun™ 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。 Sun 感谢 Xerox 在研究和开发可视或图形用户界面的  
概念方面为计算机行业所做的开拓性贡献。 Sun 已从 Xerox 获得了对 Xerox 图形用户界面的非独占性许可证，该许可证还适用于实现 OPEN LOOK GUI  
和在其他方面遵守 Sun 书面许可协议的 Sun 许可证持有者。  
美国政府权利－商业用途。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（ Federal Acquisition Regulations，即“联邦政府采购  
法规”）的适用条款及其补充条款。  
本文档按“原样”提供，对所有明示或默示的条件、陈述和担保，包括对适销性、适用性或非侵权性的默示保证，均不承担任何责任，除非此免责声明  
的适用范围在法律上无效。  
050728@12762  
目录  
前言 **15**  
**1 N1 Grid Engine 6** 软件简介 **19**  
什么是网格计算技术？ 19  
通过管理资源和策略来管理工作负荷 21  
系统的运行方式 21  
使资源与请求相匹配 22  
作业和队列 22  
使用策略 23  
Grid Engine 系统组件 25  
主机 25  
守护进程 26  
队列 26  
客户端命令 27  
QMON， Grid Engine 系统的图形用户界面 28  
**2 Grid Engine** 系统导航 **29**  
“QMON Main Control” 窗口 29  
启动 “QMON Main Control” 窗口 29  
定制 QMON 30  
用户和用户类别 31  
用户访问权限 32  
管理人员、操作员和拥有者 33  
显示队列和队列属性 34  
显示队列列表 34  
显示队列属性 34  
**3**  
▼ 用 QMON 显示队列属性 34  
解释队列属性信息 36  
主机和主机功能 36  
查找主控主机的名称 37  
显示执行主机列表 37  
显示管理主机列表 37  
显示提交主机列表 37  
可请求的属性 37  
显示可请求属性列表 38  
**3** 提交作业 **41**  
提交一个简单的作业 41  
▼ 如何从命令行提交简单的作业 42  
▼ 如何用 QMON 提交简单的作业 43  
提交批处理作业 47  
关于 shell 脚本 47  
shell 脚本示例 48  
常规 shell 脚本扩展 48  
提交扩展作业和高级作业 52  
用 QMON 提交扩展作业 52  
从命令行提交扩展作业 56  
用 QMON 提交高级作业 57  
从命令行提交高级作业 60  
定义资源需求 61  
作业从属性 64  
提交阵列作业 64  
提交交互式作业 65  
用 QMON 提交交互式作业 66  
用 qsh 提交交互式作业 68  
用 qlogin 提交交互式作业 68  
透明的远程执行 69  
用 qrsh 远程执行 69  
使用 qtcsh 透明地分布作业 70  
用 qmake 处理并行 makefile 72  
如何调度作业 74  
作业的优先级 74  
票券策略 74  
队列选择 75  
**4** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
**4** 监视和控制作业与队列 **77**  
监视和控制作业 77  
用 QMON 监视和控制作业 77  
从命令行监视和控制作业 86  
通过电子邮件监视作业 89  
监视和控制队列 89  
用 QMON 监视和控制队列 89  
用 qmod 控制队列 96  
使用点校验作业 97  
用户级别点校验 97  
内核级别点校验 97  
迁移点校验作业 98  
编写点校验作业脚本 98  
点校验的文件系统要求 100  
**5** 统计和报告 **101**  
启动 Accounting and Reporting Console 101  
▼ 如何启动 Accounting and Reporting Console 101  
创建和运行简单查询 104  
▼ 如何创建简单查询 104  
▼ 如何创建视图配置 108  
▼ 定义图数据源的方法 111  
▼ 如何运行简单查询 115  
▼ 如何编辑简单查询 115  
创建和运行高级查询 116  
▼ 如何创建高级查询 116  
▼ 如何运行高级查询 117  
▼ 如何编辑高级查询 117  
高级查询的后期绑定 118  
**6** 错误消息和错误诊断 **119**  
软件如何检索错误报告 119  
不同的错误或退出代码所表示的结果 120  
在调试模式下运行 Grid Engine 系统程序 123  
诊断问题 125  
未分派的暂挂作业 125  
报告为错误状态 E 的作业或队列 125  
**5**  
常见问题的错误诊断 126  
典型的 Accounting and Reporting Console 错误 130  
**A** 数据库模式 **133**  
模式表格 133  
sge\_job 133  
sge\_job\_usage 134  
sge\_job\_request 136  
sge\_job\_log 136  
sge\_share\_log 137  
sge\_host 138  
sge\_host\_values 139  
sge\_queue 139  
sge\_queue\_values 139  
sge\_department 140  
sge\_department\_values 140  
sge\_project 141  
sge\_project\_values 141  
sge\_user 141  
sge\_user\_values 142  
sge\_group 142  
sge\_group\_values 142  
预定义视图列表 143  
view\_accounting 143  
view\_job\_times 144  
view\_jobs\_completed 144  
view\_job\_log 144  
view\_department\_values 145  
view\_group\_values 146  
view\_host\_values 146  
view\_project\_values 147  
view\_queue\_values 147  
view\_user\_values 148  
派生值列表 148  
**6** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
词汇表 **149**  
索引 **153**  
**7**  
**8** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
表 表  
**2–1** 用户类别和相关的命令功能 31  
表 **6–1** 作业相关的错误或退出代码 120  
表 **6–2** 并行环境相关的错误或退出代码 121  
表 **6–3** 队列相关的错误或退出代码 121  
表 **6–4** 点校验相关的错误或退出代码 121  
表 **6–5** qacct -j failed 字段代码 122  
**9**  
**10** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
图 图**1–1**  
三种网格类型 20  
图 **1–2** Grid Engine 系统中策略之间的相互关系 24  
图 **1–3** “QMON Main Control” 窗口，已定义 28  
图 **3–1** “QMON Main Control” 窗口 44  
图 **3–2** “Submit Job” 对话框 45  
图 **3–3** “Job Control” 对话框 46  
图 **3–4** “Select a File” 对话框 47  
图 **3–5** 扩展作业提交示例 56  
图 **3–6** 高级作业提交示例 59  
图 **3–7** “Requested Resources” 对话框 62  
图 **3–8** “Interactive Submit Job” 对话框， “General“ 选项卡 67  
图 **3–9** “Interactive Submit Job” 对话框， “Advanced” 选项卡 68  
**11**  
**12** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
示例  
示例 **2–1** 队列属性信息 34  
示例 **2–2** 显示的属性定义集属性 39  
示例 **3–1** 简单的 shell 脚本 48  
示例 **3–2** 使用嵌入脚本的命令行选项 50  
示例 **4–1** qstat -f 输出示例 87  
示例 **4–2** qstat 输出示例 88  
示例 **4–3** 点校验作业脚本示例 98  
示例 **5–1** “Accounting per Department” 查询的结果分三列显示在表格中，分别为  
： time、 department 和 cpu。 111  
示例 **5–2** 显示结果 - 选择配置 112  
示例 **5–3** 显示结果 - 饼图 112  
示例 **5–4** 所有部门的 Cpu、 Io 和 Mem 用量 113  
示例 **5–5** 显示结果 - 选择配置 113  
示例 **5–6** 显示结果 - 条形图 114  
示例 **5–7** 后期绑定示例 118  
**13**  
**14** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
前言  
《 N1 Grid Engine 6 用户指南》包含以下内容：  
■ 介绍 N1 Grid Engine 6 软件在复杂的计算环境中的主要作用。  
■ 该产品主要组件的介绍和功能定义。  
■ 对于了解 N1 Grid Engine 6 软件环境很重要的术语词汇表。  
此版本的用户指南专门适用于 Grid Engine 6 Update 4。  
本书的读者  
本手册适合需要使用 N1 Grid Engine 6 软件的工程师和技术人员阅读。而且，如果您负  
责管理运行 N1 Grid Engine 6 软件的网络计算机主机系统，则需要理解本书中的概念。  
本书结构  
第 1 章介绍 N1 Grid Engine 6 软件的概念和主要组件。本章还包括用户命令的概述以及  
QMON 图形用户界面的介绍。  
第 2 章介绍如何显示有关运行 N1 Grid Engine 6 软件的联网的计算机主机系统的组件信  
息，例如用户、队列、主机和作业属性。  
第 3 章提供有关如何提交作业以进行处理的信息。  
第 4 章提供关于如何监视和控制作业及队列的信息。本章还包含点校验作业的信息。  
第 5 章介绍 Accounting and Reporting Console 的使用方法。  
**15**  
第 6 章包含常见问题的描述以及解决方案。  
附录 A 详细介绍报告数据库的数据模型  
词汇表含有产品专用的单词和短语以及对它们的定义。  
相关书籍  
N1 Grid Engine 6 软件文档集所包含的其他书目有：  
■ 《 N1 Grid Engine 6 安装指南》  
■ 《 N1 Grid Engine 6 管理指南》  
■ 《 N1 Grid Engine 6 发行说明》  
联机访问 Sun 文档  
可以通过 docs.sun.comSM Web 站点联机访问 Sun 技术文档。您可以浏览 docs.sun.com  
文档库或查找某个特定的书名或主题。 URL 为 http://docs.sun.com。  
订购 Sun 文档  
Sun Microsystems 提供了一些印刷的产品文档。有关文档的列表以及如何订购这些文  
档，请参阅位于 http://docs.sun.com 的 “Buy printed documentation”。  
印刷约定  
下表说明了本书中使用的印刷约定。  
**16** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
表 **P–1** 印刷约定  
字体 含义 示例  
AaBbCc123 命令、文件和目录的名称；计算机屏幕输  
出  
编辑 .login 文件。  
使用 ls -a 列出所有文件。  
machine\_name% you have  
mail.  
**AaBbCc123** 用户键入的内容，与计算机屏幕输出的显  
示不同  
machine\_name% **su**  
Password:  
*AaBbCc123* 命令行占位符：要使用实名或值替换 删除文件的命令为 rm  
filename。  
*AaBbCc123* 书名、新词或术语以及要强调的词 阅读《用户指南》的第 6 章。  
请勿保存文件。  
命令示例中的 shell 提示符  
下表列出了 C shell、 Bourne shell 和 Korn shell 的默认系统提示符和超级用户提示符。  
表 **P–2** Shell 提示符  
**Shell** 提示符  
C shell machine\_name%  
C shell 超级用户 machine\_name#  
Bourne shell 和 Korn shell $  
Bourne shell 和 Korn shell 超级用户 #  
**17**  
**18** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
第 **1** 章  
N1™ Grid Engine 6 软件简介  
本章提供了运行 N1™ Grid Engine 6 软件（ *Grid Engine* 系统）的联网的计算机主机系  
统的背景信息。 本章包括以下主题：  
■ 网格计算技术简介  
■ 产品各重要组件的说明  
■ 用户和管理人员可以使用的客户端命令的详细列表  
■ QMON（ Grid Engine 系统图形用户界面）的概述  
什么是网格计算技术？  
网格是执行任务的计算资源集。 对于用户来说，最简单的网格形式就是一个大型系统，  
该系统中，用户可从单个点访问强大的分布式资源。在本节的后面内容中将介绍网格的  
复杂形式， 网格可以为用户提供多个访问点。 在每种情况下，用户都把网格视为一个  
单独的可计算资源。资源管理软件（例如， N1 Grid Engine 6 软件（ *Grid Engine* 软  
件））接受用户提交的作业。该软件使用资源管理策略安排作业在网格内适当的系统上  
运行。 用户可以同时提交数百万的作业而不必担心作业在何处运行。  
不存在完全相同的两个网格。一种规模不能适用于所有情形。网格当前具有以下三种主  
要类型，规模从单个系统到使用数千个处理器的超级计算环境。  
■ 群集网格是最为简单的网格。群集网格由同时工作的一组计算机主机组成。 对于单  
独的项目或单独的部门，群集网格为用户提供单点访问。  
■ 校园网格可使同一个组织内的多个项目或部门共享计算资源。 组织可以使用校园网  
格来处理各种任务，从周期的商务处理到图形渲染、数据采集等等。  
■ 全局网格是校园网格的集合，它可以跨越组织界限创造巨大的虚拟系统。 用户可访  
问的计算能力远远超过其组织自有的计算资源。  
图 1–1 显示了这三种网格类型。在群集网格中，用户作业仅由群集中的一个系统处理。  
但是，用户的群集网格可能是更为复杂的校园网格的一部分。同样，校园网格可能是最  
庞大的全局网格的一部分。在这种情况下，用户的作业可以由世界各地的任何一台执行  
主机处理。  
**19**  
图 **1–1** 三种网格类型  
N1 Grid Engine 6 软件是 Sun 最新的资源管理解决方案，可以提供校园网格需要的能力  
以及灵活性。本产品对现有的群集网格很有用，它可以使其顺利过渡到创建校园网格。  
Grid Engine 系统通过将所有现有的群集网格安置在校园网格内，从而实现其过渡。另  
外， Grid Engine 系统对于首次尝试网格计算模型的企业校园是一个良好的开端。  
Grid Engine 软件在企业技术和管理人员设定的企业资源策略基础上组织计算资源的分  
发。 Grid Engine 系统使用这些策略检查校园网格内可用的计算资源。系统自动收集、  
分配和分发这些资源，使校园网格资源的使用实现最优化。  
为在校园网格内进行合作, 使用网格的项目拥有者必须执行以下操作：  
■ 协商策略  
■ 对特殊项目要求手动越权时具有策略上的灵活性  
■ 使策略能够自动得到监视和加强  
**20** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
Grid Engine 软件可以对诸多竞争使用计算资源的项目和部门的权益进行协调。  
通过管理资源和策略来管理工作负荷  
Grid Engine 系统是一种先进的资源管理工具，用于管理不同种类的分布式计算环境。  
工作负荷管理意味着对共享资源的使用进行控制，以最大限度地实现企业的目标，例如  
生产率、时间限制、服务水平等等。工作负荷管理通过管理资源、 执行策略完成。站点  
配置确保系统实现使用率和生产量的最大化，同时使系统支持不同的时间限制和重要级  
别。 作业的限期是时间限制的实例。作业的优先级和用户份额是重要程度的实例。  
Grid Engine 软件为由多种共享资源组成的 UNIX 环境提供了高级资源管理和策略管理  
功能。在以下的主要性能方面， Grid Engine 系统优于标准的负荷管理工具：  
■ 创新的动态调度和资源管理允许 Grid Engine 软件加强特定站点的管理策略。  
■ 动态搜集性能数据向调度程序提供即时的作业级的资源使用情况和系统负荷信息。  
■ 通过基于证书安全协议 (CSP) 加密的方法增强安全性。不同于以纯文本格式传输消  
息，这种更安全的系统使用密钥对消息进行了加密。  
■ 高级策略管理，可定义并实施企业目标，如生产量、时间限制以及服务级别。  
Grid Engine 软件为用户提供了向网格提交可计算的请求任务的方式，以透明地分布相  
关工作负荷。 用户可以向网格提交批处理作业、交互式作业以及并行作业 。  
该产品还支持点校验程序。点校验作业在工作站间迁移，无需用户对负荷要求进行干  
预。  
对于管理人员来说，该软件提供了全面的监视和控制作业的工具。  
系统的运行方式  
Grid Engine 系统执行以下全部任务：  
■ 接受外来作业。作业是指用户对计算机资源的请求。  
■ 将工作放入留置区域直到可以运行该作业。  
■ 将作业从留置区域发送到执行设备。  
■ 管理正在运行的作业。  
■ 作业完成时，将作业执行记录写入日志。  
第 1 章 • N1™ Grid Engine 6 软件简介 **21**  
使资源与请求相匹配  
比如，假设有一家位于世界某一首都城市的巨大的“金融中心”银行。在银行大厅中有  
很多等待服务的顾客。每个顾客都有不同的需求。一个顾客想从他的帐户支取一小部分  
现金。 在其后到来的顾客与银行的一位投资专家有约。 她希望在进行复杂的投资之前  
先得到专家的建议。 在前两个顾客前面的一位顾客想申请一笔大的贷款，和她前面的八  
个顾客一样。  
不同需求的顾客需要银行的不同类型的服务和不同的服务级别。可能这一天银行恰好有  
很多职员可以为一个简单地从其帐户支取现金的顾客服务。但同时银行可能只有一个或  
两个高级职员来为许多贷款申请人员服务。 在其他时候，情况可能恰恰相反。  
结果，顾客必须进行不必要的等待才能获得所需的服务。只有当很多顾客的需求得以迅  
速确认并与可使用的资源迅速匹配时，顾客才能获得迅速的服务。  
如果 Grid Engine 系统是银行的管理人员，则会以不同的方式组织服务。  
■ 一进入银行大厅，顾客就会被问及他们的姓名、联系方式和服务需要。  
■ 还将记录每个顾客到达的时间。  
■ 根据顾客在大厅提供的信息，银行将为以下顾客提供服务：  
■ 需要与适当的且可立即获得的资源相匹配的顾客  
■ 具有最高优先级需求的顾客  
■ 在大厅中等待时间最长的顾客  
■ 在“ Grid Engine 系统银行”， 一个银行职员可能同时帮助多个顾客。 Grid Engine  
系统会尽量将新来的顾客分配到负荷最少、最适当的银行职员处。  
■ 作为银行的管理人员， Grid Engine 系统允许银行定义服务策略。 典型的服务策略如  
下：  
■ 因为商业用户可以创造更多的利润，所以优先向商业顾客提供服务  
■ 因为某些顾客到目前为止曾得到过较差的服务，所以要确保这些特定的顾客接受  
到良好的服务  
■ 确保预约的的顾客接受到及时的响应  
■ 针对银行主管人员直接提出的要求，优先提供服务  
■ Grid Engine 系统管理人员可以自动实现、监视和调整这些策略。拥有优先访问权限  
的顾客会更快地得到服务。这些顾客会引起银行职员的更多注意，但是银行职员提供  
给这些顾客的帮助应在具有相同访问权限的顾客间共享。 Grid Engine 管理人员能够  
认识到顾客是否没有进展。管理人员会立即做出响应，方法是调整服务级别以符合银  
行的服务策略。  
作业和队列  
在 Grid Engine 系统中， 作业相当于银行顾客。 不同的是，作业在计算机留置区域而不  
是大厅等待。 队列相当于银行职员，它为作业提供服务。与银行顾客一样，每个作业对  
可用内存、执行速度、可用的软件许可证以及类似需求都是不同的。只有特定的队列才  
能提供相应的服务。  
**22** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
继续上面的类比， Grid Engine 软件按以下方式仲裁可使用的资源和作业要求：  
■ 通过 Grid Engine 系统 提交作业的用户宣布作业的要求配置。同时，系统检索用户  
的身份。系统还检索用户与项目或用户组的从属关系。同样还存储用户提交作业的时  
间。  
■ 当队列可运行新的作业时， Grid Engine 系统为队列确定适当的作业。 系统会立即分  
派具有最高优先级别的或等待时间最长的作业 。  
■ 队列允许并发执行多个作业。 Grid Engine 系统尽量在负荷最小和最适合的队列中启  
动新作业。  
使用策略  
群集管理员可以根据站点的具体情况为站点定制适当的高级使用策略。有四种使用策略  
可用：  
■ 紧急程度。 使用该策略时，每项作业的优先级基于其紧急值。该紧急值得自作业的资  
源需求、作业的时间限期，以及该作业在运行前的等待时间。  
■ 职能。 使用该策略时，管理人员可以根据用户或作业与特定用户组、项目等的从属关  
系而提供特别的处理。  
■ 基于份额。 使用该策略时，服务的级别取决于分配的份额、其他用户和用户组的相应  
的份额、所有用户过去资源的使用以及系统中当前存在的用户。  
■ 越权。 该策略要求群集管理员手动介入修改自动策略的执行。  
策略管理自动控制群集中共享资源的使用，从而最好地达到管理的目标。高优先级别的  
作业得到优先分派。同其他作业争用资源时，这种作业可以得到更高的 CPU 权益。  
Grid Engine 软件根据策略定义的目标监视所有作业的进程并相应地调整它们的相对优  
先级。  
使用票券管理策略  
功能、基于份额的以及越权策略通过 Grid Engine 系统 中的名为票券的概念进行定义。  
票券就像上市公司的股票。拥有的股份越多，您对于公司就越重要。如果股票持有人 A  
持有的股票份额是股票持有人 B 的两倍，则股票持有人 A 就有两倍于股票持有人 B 的投  
票权。所以股票持有人 A 对于公司的重要程度就两倍于股票持有人 B。作业拥有的票券  
越多就越重要。如果作业 A 拥有两倍于作业 B 的票券，则作业 A 可使用两倍于 B 的资  
源。  
作业还可以从功能、基于份额的以及越权策略得到票券。总的票券数以及从每个票券策  
略再次得到的票券数经常随着时间而改变。  
管理员控制分配给各项票券策略的票券总数。就像作业的票券分配，该分配决定了票券  
策略之间的相对重要程度。通过分配给特定票券的票券池，管理人员可以按不同的方式  
运行 Grid Engine 系统。例如，系统可以运行在只有基于份额的策略模式中。或者系统  
可以运行两种策略模式的组合，例如 90% 基于份额的策略， 10% 职能策略。  
第 1 章 • N1™ Grid Engine 6 软件简介 **23**  
使用紧急程度策略分配作业的优先级。  
紧急程度策略可以和其他两种的作业优先级规范结合使用：  
■ 由职能、基于份额的和越权策略指定的票券数  
■ 由 qsub –p 命令指定的优先级值  
可以根据以下三方面的信息，分配作业的紧急程度值：  
■ 作业的资源需求  
■ 作业在运行之前必须等待的时间  
■ 作业完成所必须的时间，即作业限期  
管理员可以权衡每方面信息的重要性然后得出作业的整体紧急程度值。有关更多信息，  
请参阅《 N1 Grid Engine 6 管理指南》中的第 5 章 “管理策略和调度程序”。  
图 1–2 显示各策略之间的相互关系。  
图 **1–2** Grid Engine 系统中策略之间的相互关系  
**24** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
Grid Engine 系统组件  
以下几节介绍了最重要的 Grid Engine 系统 组件的功能。  
主机  
Grid Engine 系统有四种基本的主机类型：  
■ 主机  
■ 执行主机  
■ 管理主机  
■ 提交主机  
主控主机  
主控主机是所有群集活动的中心。主控主机运行主守护进程 sge\_qmaster 和调度程序  
守护进程 sge\_schedd。两个守护进程都控制 Grid Engine 系统组件，例如，队列和作  
业。守护进程维护着关于组件状态的表格、关于用户访问权限的表格以及其他任务的表  
格。  
默认情况下，主控主机也是管理主机和提交主机。请参阅介绍这些主机的各节。  
执行主机  
执行主机是有权限执行作业的系统。因此，执行主机具有附加到它的队列实例。执行主  
机运行执行守护进程 sge\_execd。  
管理主机  
管理主机有权限运行 Grid Engine 系统的任何类型的管理操作。  
提交主机  
提交主机只允许用户提交和控制批处理作业。具体地说，用户登录提交主机后，可以使  
用 qsub 命令提交作业、使用 qstat 命令监视作业状态，还可以使用 Grid Engine 系统  
OSF/1 Motif 图形用户界面 QMON，该界面的说明参见第 28 页中的 “ QMON， Grid  
Engine 系统的图形用户界面” 。  
第 1 章 • N1™ Grid Engine 6 软件简介 **25**  
注 **–** 一个系统可以作为多种主机类型进行操作。  
守护进程  
三个守护进程提供了 Grid Engine 系统的功能。  
sge\_qmaster – 主守护进程  
群集管理和调度活动中心 sge\_qmaster 维护与主机、队列、作业、系统负荷以及用户  
权限相关的表格。 sge\_qmaster 接收来自 sge\_schedd 的调度决定和相应执行主机上  
sge\_execd 的操作请求。  
sge\_schedd – 调度程序守护进程  
调度程序守护进程在 sge\_qmaster 的帮助下维护群集状态的最新视图。调度程序守护  
进程做出以下调度决定：  
■ 将哪个作业分派到哪个队列  
■ 如何重新排序和重新指定作业的优先级，以维护份额、优先级或限期  
守护进程随后将这些决定转发给 sge\_qmaster（启动所需的操作）。  
sge\_execd – 执行守护进程  
执行守护进程对其主机上的队列实例和队列实例中的作业的运行负责。执行主机定期向  
sge\_qmaster 转发其主机上的作业状态或负荷信息。  
队列  
队列是允许同时在一台或多台主机上运行一种作业类的容器。队列确定某些作业属性，  
例如作业是否可迁移。作业在运行的整个过程中，始终与队列相关联。 与队列的关联性  
可以影响针对作业发生的一些事件。例如，如果暂停队列，则与该队列相关联的作业均  
被暂停。  
作业不需要直接提交到队列。只需指定作业的需求配置文件即可。配置文件可能包括诸  
如内存、操作系统、可使用的软件等要求。 Grid Engine 软件自动将作业分派给适当的  
队列和执行负荷较轻的主机。如果将作业提交到指定的队列，则该作业将附加到该队  
列。结果是， Grid Engine 系统守护进程无法选择负荷较轻或更加适合的设备。  
队列可以驻留在一个单独的主机上，也可以扩展到多个主机。因此， Grid Engine 系统  
队列也称作群集队列。群集队列可使用户和管理员通过单个队列配置在多个执行主机组  
成的群集中工作。挂接到群集队列的每个主机都从群集队列接收到自己的队列实例 。  
**26** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
客户端命令  
命令行用户界面是一组辅助程序（命令），通过这组辅助程序，您可执行以下任务：  
■ 管理队列  
■ 提交和删除作业  
■ 检查作业状态  
■ 暂停或启用队列和作业  
Grid Engine 系统提供了以下一组辅助程序：  
■ qacct – 从群集日志文件抽取仲裁记录信息。  
■ qalter – 更改已经提交但暂挂的作业的属性。  
■ qconf – 提供群集配置和队列配置的用户界面。  
■ qdel – 为用户、操作员或管理人员提供向作业或其子集发送信号的方式。  
■ qhold – 阻止已提交作业的执行。  
■ qhost – 显示执行主机的状态信息。  
■ qlogin – 启用 telnet 或类似的带有自动选择的低负荷、适当主机的登录会话。  
■ qmake – 标准 UNIX make 工具的替代工具。 qmake 延伸了 make 的功能，使其能够  
在一组合适的计算机之间分布一系列的 make 步骤。  
■ qmod – 使拥有者暂停或启用队列。所有与该队列相关的活动的进程也都将得到信  
号。  
■ qmon – 提供 X Windows Motif 命令界面和监视工具。  
■ qresub – 通过复制运行的或暂挂的作业创建新作业。  
■ qrls – 将作业从之前分配到的留置区域（例如，通过 qhold）释放。  
■ qrsh – 可用于多种用途，比如以下的情形。  
■ 通过 Grid Engine 系统提供交互式应用程序的远程执行。 qrsh 是与标准的 UNIX  
工具 rsh 相比较而言。  
■ 允许提交批处理作业，执行时支持终端 I/O 和终端控制。终端 I/O 包括标准输  
出、标准错误和标准输入。  
■ 提供一个在批处理作业结束之前保持活动状态的提交客户机。  
■ 允许在 Grid Engine 软件控制下远程执行并行作业任务。  
■ qselect – 打印与指定的选择标准对应的队列名称列表。 qselect 的输出通常输入  
到其他的 Grid Engine 系统命令，以便对选定的队列组进行操作。  
■ qsh – 在较轻负荷主机的 xterm 中打开交互式 shell。在该 shell 中，任何类型的交  
互式作业都可以运行。  
■ qstat – 提供与群集相关的所有作业和队列的状态列表。  
■ qsub – 将批处理作业提交到 Grid Engine 系统的用户界面。  
■ qtcsh – 完全兼容并可替代从普遍认同并广泛使用的 UNIX C shell (csh) 衍生出来  
的 tcsh。 qtcsh 提供的命令 shell 可以通过 Grid Engine 软件透明地将指定应用程  
序的执行分布到适当且负荷较低的主机上。  
第 1 章 • N1™ Grid Engine 6 软件简介 **27**  
QMON， Grid Engine 系统的图形用户界面  
使用图形用户界面 (GUI) 工具 QMON，您可以完成大多数 Grid Engine 系统任务。 图 1–3  
显示了 “QMON Main Control” 窗口，用户和管理员操作时通常从该窗口开始。主控制窗  
口中的每个图标都是一个 GUI 按钮，您可以点击开始执行多样化的任务。要查看按钮的  
名称（其中也介绍了它的功能），请把指针移到按钮上面。  
图 **1–3** “QMON Main Control” 窗口，已定义  
**28** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
第 **2** 章  
Grid Engine 系统导航  
本章介绍如何显示关于 Grid Engine 系统组件（例如，用户、队列、主机和作业属性）  
的信息。本章还介绍了一些可以帮助您使用该软件的基本概念和术语。有关该产品的完  
全背景信息，请参阅第 1 章。  
本章还包含用于完成下列任务的指导：  
■ 第 29 页中的 “启动 “QMON Main Control” 窗口”  
■ 第 30 页中的 “定制 QMON”  
■ 第 34 页中的 “显示队列列表”  
■ 第 34 页中的 “用 QMON 显示队列属性”  
■ 第 35 页中的 “从命令行显示队列属性”  
■ 第 37 页中的 “查找主控主机的名称”  
■ 第 37 页中的 “显示执行主机列表”  
■ 第 37 页中的 “显示管理主机列表”  
■ 第 37 页中的 “显示提交主机列表”  
■ 第 38 页中的 “显示可请求属性列表”  
“QMON Main Control” 窗口  
Grid Engine 系统具有图形用户界面 (GUI) 命令工具，即 “QMON Main Control” 窗口。  
“QMON Main Control” 窗口可使用户执行包括提交作业、控制作业以及搜集重要信息在  
内的大多数 Grid Engine 系统功能。  
启动 “QMON Main Control” 窗口  
要启动 “QMON Main Control” 窗口，请从命令行键入以下命令：  
% qmon  
**29**  
显示消息窗口后，会显示 “QMON Main Control” 窗口。  
请参阅图 1–3 以了解这些图标的含义。将指针移动到图标按钮上时，屏幕上会显示图标  
按钮的名称。按钮的名称描述了按钮的功能。  
本指南中的许多指导均要求使用 “QMON Main Control” 窗口。  
定制 QMON  
QMON 的外观主要通过专门指定的资源文件进行定义。 *sge-root*/qmon/Qmon 编译时已包  
括合理的默认值，其中还包括示例资源文件。  
群集管理可以完成以下任一任务：  
■ 在标准位置（如 /usr/lib/X11/app-defaults/Qmon）安装站点特定的默认值  
■ 标准 .Xdefaults 或 .Xresources 文件中包括 QMON– 特定资源定义  
■ 将站点专用的 Qmon 文件放在标准搜索路径所引用的位置，例如， XAPPLRESDIR  
如果出现任一相关情形，请向管理员咨询。  
此外，用户可以配置个人优先权。用户可以修改 Qmon 文件。 Qmon 文件可以移到主目  
录或专用的 XAPPLRESDIR 搜索路径指向的其他位置。用户还可以在其专用  
.Xdefaults 文件或 .Xresources 文件中包括必要的资源定义。您也可以使用 xrdb  
命令安装专用的 Qmon 资源文件。可以在操作过程中使用 xrdb 命令，也可以在启动  
X11 环境时使用 xrdb 命令，例如在 .xinitrc 资源文件中。  
有关可能的定制设置的详细信息，请参阅 Qmon 文件示例中的注释行。  
定制 qmon 的另一个方法是使用 “Job Customize and Queue Customize” 对话框。这些  
对话框的外观请参见第 82 页中的 “定制 “Job Control” 显示” 和第 95 页中的 “筛选  
群集队列和队列实例” 。在这两个对话框中，用户可以使用 “Save” 按钮来存储筛选内  
容，并在其主目录中显示 .qmon\_preferences 文件的定义。 QMON 重新启动时，将读  
取该文件，随后 QMON 重新激活之前定义的操作。  
**30** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
用户和用户类别  
Grid Engine 系统用户分为四个类别。每个类别中的用户都有权访问各自的 Grid Engine  
系统命令集。  
■ 管理人员 – 管理人员拥有完全操 Grid Engine 系统的能力。默认情况下，管理主机的  
超级用户都具有管理人员权限。  
■ 操作员 – 操作员可以执行很多管理人员命令，但是进行配置更改（例如，添加、删除  
和更改队列）除外。  
■ 拥有者 – 队列拥有者可以暂停或启用其拥有的队列。队列拥有者还可以暂停或启用其  
拥有的队列中的作业。队列拥有者没有其他的管理权限。  
■ 用户 – 用户有一定的访问权限，这些权限在第 32 页中的 “用户访问权限” 中进行  
了介绍。用户没有群集管理或队列管理能力。  
表 2–1 显示了不同类别的用户可使用的命令。  
表 **2–1** 用户类别和相关的命令功能  
命令 管理器 操作员 拥有者 用户  
qacct 全部 全部 仅限自己的作业 仅限自己的作业  
qalter 全部 全部 仅限自己的作业 仅限自己的作业  
qconf 全部 不能对系统设置  
进行更改  
仅显示配置和访  
问权限  
仅显示配置和访  
问权限  
qdel 全部 全部 仅限自己的作业 仅限自己的作业  
qhold 全部 全部 仅限自己的作业 仅限自己的作业  
qhost 全部 全部 全部 全部  
qlogin 全部 全部 全部 全部  
qmod 全部 全部 仅限于作业和拥  
有的队列  
仅限自己的作业  
qmon 全部 不能对系统设置  
进行更改  
不能进行配置  
更改  
不能进行配置更  
改  
qrexec 全部 全部 全部 全部  
qselect 全部 全部 全部 全部  
qsh 全部 全部 全部 全部  
qstat 全部 全部 全部 全部  
第 2 章 • Grid Engine 系统导航 **31**  
表 **2–1** 用户类别和相关的命令功能 （续）  
命令 管理器 操作员 拥有者 用户  
qsub 全部 全部 全部 全部  
用户访问权限  
管理员可以限制队列或其他工具（例如，并行环境接口）的访问权限。访问权限可以限  
制到特定的用户或用户组。  
注 **–** Grid Engine 软件自动考虑通过群集管理操作配置的访问权限。仅当您希望查询个  
人的访问权限时，以下的各节内容才变得重要。  
出于限制访问权限的目的，管理员应创建并维护一份存取控制表 (ACL)。 ACL 包含用户  
名称和 UNIX 组名称。 ACL 随后添加到队列或并行环境接口配置中的允许访问或禁止访  
问列表中。有关详细信息，请参阅 queue\_conf(5) 或 sge\_pe(5) 手册页。  
属于允许访问控制列表中列出的 ACL 中的用户具有访问队列或并行环境接口的权限。  
禁止访问控制列表中的 ACL 中的用户则无权访问相关的资源。  
ACL 还用来定义项目，与项目相对应的用户具有项目访问权限，即用户可以将其作业包  
含在项目中。管理员还可以在每个项目的基础上限制群集资源的访问权限。  
单击 “QMON Main Control” 窗口的 “User Configuration” 按钮将打开 “User  
Configuration” 对话框。可在该对话框中查询您有权访问的 ACL。有关详细信息，请参  
阅《 N1 Grid Engine 6 管理指南》中的第 4 章 “管理用户访问权限”。  
单击 “QMON Main Control” 窗口中的 “Project Configuration” 图标可以显示项目访问权  
限。有关详细信息，请参阅《 N1 Grid Engine 6 管理指南》中的“定义项目”。  
从命令行使用以下命令，您可以得到一份当前已配置的 ACL 列表：  
% qconf -sul  
通过以下命令可以列出一个或多个访问权限列表中的条目：  
% qconf -su *acl-name*[,*...*]  
ACL 由用户帐户名称和 UNIX 组名称组成，其中 UNIX 组名称以前缀 @ 标识。通过这  
种方式，可以确定您的帐户属于哪个 ACL。  
注 **–** 如果您有权使用 newgrp 命令切换主 UNIX 组，您的访问权限可能会有所不同。  
对于您有权访问或访问遭到拒绝的队列或并行环境接口，您现在可以对其执行检查。查  
询队列或并行环境接口配置信息，如第 34 页中的 “显示队列和队列属性” 和《 N1  
Grid Engine 6 管理指南》中的“用 QMON 配置并行环境”所述。  
**32** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
允许访问的列表名为 user\_lists。禁止访问的列表名为 xuser\_lists。如果您的用  
户帐户或主 UNIX 组与允许访问的列表相关联，则您具有访问相关资源的权限。如果您  
与禁止访问的列表相关联，则您无权访问队列和并行环境接口。如果这两个列表均为  
空，则具有有效帐户的每个用户都可以访问相关资源。  
可以使用以下命令从命令行控制项目配置：  
% qconf -sprjl  
% qconf -sprj *project-name*  
该命令分别显示已定义的项目列表和具体的项目配置列表。项目通过 ACL 定义。如前  
段所述，必须查询 ACL 配置。  
如果对项目具有访问权限，则可以提交从属于该项目的作业。可以在命令行使用以下命  
令提交这些作业：  
% qsub -P *project-name options*  
群集配置、主机配置和队列配置使用与 ACL 相同的方式定义项目访问权限。为达到此  
目的，这些配置使用 project\_lists 和 xproject\_lists 参数。  
管理人员、操作员和拥有者  
使用以下命令显示 Grid Engine 系统管理人员的列表：  
% qconf -sm  
使用以下命令显示操作人员列表：  
% qconf -so  
注 **–** 在默认情况下，管理主机的超级用户被认为是管理人员。  
队列配置中包含作为特定队列拥有者的用户，如第 34 页中的 “显示队列和队列属性”  
中所述。可以键入以下命令显示队列配置：  
% qconf -sq {*cluster-queue* | *queue-instance* | *queue-domain*}  
相关的队列配置条目称为 owner\_list。  
第 2 章 • Grid Engine 系统导航 **33**  
显示队列和队列属性  
为优化使用站点的 Grid Engine 系统，应当熟悉队列的结构。还应当熟悉为 Grid  
Engine 系统配置的队列属性。  
显示队列列表  
“QMON Queue Control” 对话框在第 89 页中的 “用 QMON 监视和控制队列” 中显示并  
进行了介绍。该对话框提供了已安装队列及其属性的简单概述。  
从命令行键入以下命令显示队列列表：  
% qconf -sql  
显示队列属性  
可使用 QMON 或命令行显示队列属性。  
▼ 用 QMON 显示队列属性  
**1.** 启动 **“QMON Main Control”** 窗口。  
**2.** 单击 **“Queue Control”** 按钮。  
将显示 “Cluster Queue” 对话框。  
**3.** 选择 一个队列，然后单击 **“Show Detached Settings”**。  
将显示 “Browser” 对话框 。  
**4.** 在 **“Browser”** 对话框中，单击 **“Queue”**。  
**5.** 在 **“Cluster Queue”** 对话框中，单击 **“ Queue Instances”**。  
**6.** 选择一个队列实例。  
“Browser” 对话框列出了选定的队列实例的队列属性。 队列属性信息  
下图中的示例显示了部分队列属性信息。  
步骤  
示例 **2–1**  
**34** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
从命令行显示队列属性  
要使用命令行显示队列属性，请键入以下命令：  
% qconf -sq {*queue* | *queue-instance* | *queue-domain*}  
将显示与上图相似的信息。  
第 2 章 • Grid Engine 系统导航 **35**  
解释队列属性信息  
您可以在 queue\_conf(5) 手册页中找到每种队列属性的详细说明。  
以下是一些重要参数的列表：  
■ qname – 请求的队列名称。  
■ hostlist – 与队列相关联的主机和主机组的列表。  
■ processors – 该队列可以访问的多处理器系统的处理器。  
注意 **–** 除非必要，否则请不要修改这些参数值。  
■ qtype – 队列中可以运行的作业类型。当前，作业类型可为批处理或交互式。  
■ slots – 可以和该队列同时执行的作业数。  
■ owner\_list – 队列拥有者，在第 33 页中的 “管理人员、操作员和拥有者” 中有介  
绍。  
■ user\_lists – 用户访问权限列表中列在该参数下的用户或组标识符可以访问该队  
列。有关更多信息，请参阅第 32 页中的 “用户访问权限” 。  
■ xuser\_lists – 用户访问权限列表中列在该参数下的用户或组标识符不能访问该队  
列。有关更多信息，请参阅第 32 页中的 “用户访问权限” 。  
■ project\_lists – 列在该参数下的带有项目标识符的提交作业有权访问该队列。有  
关更多信息，请参阅《 N1 Grid Engine 6 管理指南》中的“定义项目”。  
■ xproject\_lists – 列在该参数下的带有项目标识符的提交作业不能访问该队列。  
有关更多信息，请参阅《 N1 Grid Engine 6 管理指南》中的“定义项目”。  
■ complex\_values – 根据提供给队列的特定属性定义集资源属性指定性能。有关更  
多信息，请参阅 第 37 页中的 “可请求的属性” 。  
主机和主机功能  
单击 “QMON Main Control” 窗口中的 “Host Configuration” 按钮，显示与您的群集中的  
主机相关联的功能的概述。 但是，只有具有管理人员权限才能更改配置。  
《 N1 Grid Engine 6 管理指南》中的第 1 章 “配置主机和群集”介绍了主机配置对话框。  
以下各节介绍了从命令行检索这类信息的命令。  
**36** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
查找主控主机的名称  
主控主机的位置在任何时间都可以在当前的主控主机和一个隐藏主控主机间进行迁移。  
因此，主控主机的位置对用户来说是透明的。  
使用文本编辑器打开 *sge-root*/*cell*/common/act\_qmaster 文件。  
当前主控主机的名称在该文件中。  
显示执行主机列表  
要显示群集中被配置为执行主机的主机列表，请使用以下命令：  
% qconf -sel  
% qconf -se *hostname*  
% qhost  
qconf -sel 命令显示当前配置为执行主机的所有主机的名称列表。 qconf -se 命令  
显示指定的执行主机的详细信息。 qhost 命令显示有关执行主机的状态和负荷信息。  
有关使用 qconf 命令时所显示信息的详细说明，请参阅 host\_conf(5) 手册页。请参  
阅 qhost(1) 手册页以了解有关其输出和其他选项的详细信息。  
显示管理主机列表  
使用以下命令可以显示具有管理权限的主机列表：  
% qconf -sh  
显示提交主机列表  
使用以下命令可以显示提交主机列表：  
% qconf -ss  
可请求的属性  
用户提交一项作业时，可以为作业指定一个要求配置文件。为顺利运行，用户可以指定  
作业需要的主机或队列的属性或特征。 Grid Engine 软件将这些作业要求与群集的主机  
或队列配置进行匹配，然后为作业查找适当的主机。  
第 2 章 • Grid Engine 系统导航 **37**  
可用于指定作业要求的属性与以下情况之一有关：  
■ 群集，例如所需的网络共享磁盘空间  
■ 单独的主机，例如操作环境结构  
■ 队列，例如允许的 CPU 时间  
属性也可能从站点策略衍生而来，例如安装的软件仅在特定的主机上可用。  
可用的属性包含以下属性：  
■ 队列属性列表 – 请参阅第 34 页中的 “显示队列和队列属性”  
■ 全局属性和与主机相关的属性列表 – 请参阅《 N1 Grid Engine 6 管理指南》中的“将  
资源属性分配到队列、主机和全局群集”  
■ 管理员定义的属性  
不过为了方便，管理员通常只定义所有可请求的属性的一个子集。  
“Requested Resources” 对话框显示当前可请求的属性，如下图所示。  
使用 “QMON Submit Job” 对话框访问 “ Requested Resources” 对话框。可请求的属性在  
“Available Resources” 列出。  
显示可请求属性列表  
要显示所配置的资源属性列表，请在命令行中键入以下命令：  
% qconf -sc  
**38** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
Grid Engine 系统 属性定义集包含所有资源属性的定义。有关资源属性的更多信息，请  
参阅《 N1 Grid Engine 6 管理指南》中的第 3 章 “配置属性定义集资源属性”。另请参阅  
complex(5) 手册页有关属性定义集格式的说明。  
从 qconf -sc 命令输出的结果与示例 2–2 相似。  
示例 **2–2** 显示的属性定义集属性  
gimli% qconf -sc  
#name shortcut type relop requestable consumable default urgency  
#----------------------------------------------------------------------------------------  
arch a RESTRING == YES NO NONE 0  
calendar c STRING == YES NO NONE 0  
cpu cpu DOUBLE >= YES NO 0 0  
h\_core h\_core MEMORY <= YES NO 0 0  
h\_cpu h\_cpu TIME <= YES NO 0:0:0 0  
h\_data h\_data MEMORY <= YES NO 0 0  
h\_fsize h\_fsize MEMORY <= YES NO 0 0  
h\_rss h\_rss MEMORY <= YES NO 0 0  
h\_rt h\_rt TIME <= YES NO 0:0:0 0  
h\_stack h\_stack MEMORY <= YES NO 0 0  
h\_vmem h\_vmem MEMORY <= YES NO 0 0  
hostname h HOST == YES NO NONE 0  
load\_avg la DOUBLE >= NO NO 0 0  
load\_long ll DOUBLE >= NO NO 0 0  
load\_medium lm DOUBLE >= NO NO 0 0  
load\_short ls DOUBLE >= NO NO 0 0  
mem\_free mf MEMORY <= YES NO 0 0  
mem\_total mt MEMORY <= YES NO 0 0  
mem\_used mu MEMORY >= YES NO 0 0  
min\_cpu\_interval mci TIME <= NO NO 0:0:0 0  
np\_load\_avg nla DOUBLE >= NO NO 0 0  
np\_load\_long nll DOUBLE >= NO NO 0 0  
np\_load\_medium nlm DOUBLE >= NO NO 0 0  
np\_load\_short nls DOUBLE >= NO NO 0 0  
num\_proc p INT == YES NO 0 0  
qname q STRING == YES NO NONE 0  
rerun re BOOL == NO NO 0 0  
s\_core s\_core MEMORY <= YES NO 0 0  
s\_cpu s\_cpu TIME <= YES NO 0:0:0 0  
s\_data s\_data MEMORY <= YES NO 0 0  
s\_fsize s\_fsize MEMORY <= YES NO 0 0  
s\_rss s\_rss MEMORY <= YES NO 0 0  
s\_rt s\_rt TIME <= YES NO 0:0:0 0  
s\_stack s\_stack MEMORY <= YES NO 0 0  
s\_vmem s\_vmem MEMORY <= YES NO 0 0  
seq\_no seq INT == NO NO 0 0  
slots s INT <= YES YES 1 1000  
swap\_free sf MEMORY <= YES NO 0 0  
swap\_rate sr MEMORY >= YES NO 0 0  
swap\_rsvd srsv MEMORY >= YES NO 0 0  
swap\_total st MEMORY <= YES NO 0 0  
swap\_used su MEMORY >= YES NO 0 0  
tmpdir tmp STRING == NO NO NONE 0  
virtual\_free vf MEMORY <= YES NO 0 0  
第 2 章 • Grid Engine 系统导航 **39**  
示例 **2–2** 显示的属性定义集属性 （续）  
virtual\_total vt MEMORY <= YES NO 0 0  
virtual\_used vu MEMORY >= YES NO 0 0  
# >#< starts a comment but comments are not saved across edits --------  
name 列与 qconf -sq 命令显示的第一列相同。 shortcut 列包含第一列中的名称的  
缩写。该缩写由管理员定义。在 qsub 命令的请求选项中，用户既可以提供全称，也可  
提供缩写。  
requestable 列表明资源属性是否可以在 qsub 命令中使用。管理员因此便可以禁止  
群集用户为他们的作业直接请求特定的计算机或队列。通过将 qname、 hostname 条  
目，或这两个条目均设置为禁止请求，管理员可禁止直接的请求。 将队列或主机设置为  
不可请求， 意味着可能的用户请求通常由多个队列解决，这就加强了 Grid Engine 系统  
的均衡能力。  
relop 列定义用来计算队列或主机是否满足用户请求的关系操作员。 执行的比较如下  
：  
User\_Request relop Queue/Host/... -Property  
如果比较的结果为 “false”，则用户的作业不能在队列或主机中运行。例如，可为队列  
q1 配置 100 秒的软性 (soft) CPU 时间限制。将队列 q2 配置为可提供 1000 秒的软性  
CPU 时间限制。有关用户进程限制的介绍，请参阅 queue\_conf(5) 和 setrlimit(2)  
手册页。  
consumable 和 default 列用于管理员声明可使用资源。请参阅《 N1 Grid Engine 6  
管理指南》中的“可使用资源”。  
用户可请求可使用属性，如请求其他属性一样。但是 Grid Engine 系统内部资源薄记是  
不同的。  
假定用户提交了以下请求：  
% qsub -l s\_cpu=0:5:0 nastran.sh  
s\_cpu=0:5:0 请求至少具有 5 分钟软性 CPU 时间限制的队列。因此，只有至少可提供  
5 分钟软性 CPU 运行时限制的队列才能正确设置以运行该作业。有关语法的详细信息，  
请参阅 qsub(1) 手册页。  
注 **–** Grid Engine 软件只有在多个队列或主机可运行同一作业时，才考虑调度进程中的负  
荷信息。  
**40** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
第 **3** 章  
提交作业  
本章提供了关于提交作业的背景信息以及如何提交作业以进行处理的指导。本章从如何  
运行一个简单的作业示例开始。本章的后面部分将介绍如何运行更加复杂的作业。  
本章包含用于完成以下任务的指导。  
■ 第 42 页中的 “如何从命令行提交简单的作业”  
■ 第 43 页中的 “如何用 QMON 提交简单的作业”  
■ 第 52 页中的 “用 QMON 提交扩展作业”  
■ 第 56 页中的 “从命令行提交扩展作业”  
■ 第 57 页中的 “用 QMON 提交高级作业”  
■ 第 60 页中的 “从命令行提交高级作业”  
■ 第 64 页中的 “用 QMON 提交阵列作业”  
■ 第 65 页中的 “从命令行提交阵列作业”  
■ 第 66 页中的 “用 QMON 提交交互式作业”  
■ 第 68 页中的 “用 qsh 提交交互式作业”  
■ 第 68 页中的 “用 qlogin 提交交互式作业”  
提交一个简单的作业  
请使用本节的信息和指导以熟悉提交作业时涉及的基本过程。  
注 **–** 如果使用未经授权的用户帐户安装了N1 Grid Engine 6 软件，则必须以该用户登录  
才能运行作业。有关详细信息，请参阅《 N1 Grid Engine 6 安装指南》中的“安装帐  
户”。  
**41**  
▼ 如何从命令行提交简单的作业  
运行任何 Grid Engine 系统命令之前，必须正确设置执行搜索路径和其他环境条件。  
**1.** 从命令行，键入下面的命令：  
■ 如果使用 csh 或 tcsh 作为命令解释器，请键入以下内容：  
% source *sge-root*/*cell*/common/settings.csh  
*sge-root* 指定 Grid Engine 系统根目录的位置。该目录在安装过程开始时进行指  
定。  
■ 如果使用 sh、 ksh 或 bash 作为命令解释器，请键入以下内容：  
# . *sge-root*/*cell*/common/settings.sh  
注 **–** 可以将这些命令添加到适当的 .login、 .cshrc 或 .profile 文件。通过  
添加这些命令，确保后面启动的交互式对话的正确设置。  
**2.** 将以下简单作业脚本添加到群集。  
您可以在 /*sge-root*/examples/jobs/simple.sh 文件中找到以下作业。  
#!/bin/sh  
##  
# (c) 2004 Sun Microsystems, Inc. Use is subject to license terms.  
# This is a simple example of a SGE batch script  
# request Bourne shell as shell for job  
#$ -S /bin/sh  
#  
# print date and time  
date  
# Sleep for 20 seconds  
sleep 20  
# print date and time again  
date  
键入以下命令：该命令假定 simple.sh 是脚本文件的名称，该文件位于当前工作目  
录下。  
% qsub simple.sh  
如果显示以下消息，则 qsub 认为已成功提交作业：  
your job 1 (“simple.sh”)  
has been submitted  
**3.** 键入下面的命令检索作业的状态信息。  
% qstat  
步骤  
**42** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
收到的状态报告应能提供 Grid Engine 系统当前可识别到的所有作业的信息。对于每  
项作业，状态报告列出以下各项：  
■ 作业 ID， 包含在提交信息中的唯一编号  
■ 作业脚本的名称  
■ 作业的拥有者  
■ 状态指示符，例如， r 表示正在运行  
■ 提交或启动时间  
■ 运行作业的队列名称  
如果 qstat 没有产生输出，则表示系统没有识别到作业。例如，作业已完成。  
通过检查 stdout 和 stderr 重定向文件可以控制已完成作业的输出。默认情况  
下，这些文件在运行作业的主机上作业拥有者的根目录下生成。文件名称由作业脚本  
文件名称、扩展名 .o （对于 stdout 文件）或扩展名 .e （对于 stderr 文件）和  
唯一的作业 ID 组成。因此，可以分别在 simple.sh.o1 和 simple.sh.e1 下找  
到作业的 stdout 和 stderr 文件。如果首次在新安装的 Grid Engine 系统中执行  
作业，则需要这些名称。  
▼ 如何用 QMON 提交简单的作业  
另一种更加简便的提交和控制作业以及获取 Grid Engine 系统概要信息的方法是使用图  
形用户界面 QMON。在其他设备中， QMON 为提交和控制任务提供了 “Submit Job” 对话  
框和 “Job Control” 对话框。  
**1.** 在命令行中键入以下命令：  
% qmon  
启动过程中，首先显示消息窗口，然后显示 “QMON Main Control” 窗口。  
步骤  
第 3 章 • 提交作业 **43**  
图 **3–1** “QMON Main Control” 窗口  
**2.** 单击 **“Job Control”** 按钮，然后单击 **“Submit Jobs”** 按钮。  
将鼠标指针移动到按钮上时，会显示按钮名称（如 “Job Control”）。  
显示 “Submit Job” 和 “Job Control” 对话框，如下图所示：  
**44** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
图 **3–2** “Submit Job” 对话框  
第 3 章 • 提交作业 **45**  
图 **3–3** “Job Control” 对话框  
**3.** 在 **“Submit Job”** 对话框中，单击 **“Job Script”** 字段右边的图标。  
显示 “Select a File” 对话框。  
**46** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
图 **3–4** “Select a File” 对话框  
**4.** 选择脚本文件。  
例如，从命令行示例中选择 simple.sh 文件。  
**5.** 单击 **“OK”** 关闭 **“Select a File”** 对话框。  
**6.** 在 **“Submit Job”** 对话框中，单击 **“ Submit”**。  
几秒钟后，便可以从 “Job Control” 对话框内监视作业。首先会在 “Pending Jobs” 选  
项卡下看到作业。一旦开始运行，作业便会迅速移到 “Running Jobs” 选项卡下。  
提交批处理作业  
以下各节介绍如何通过 Grid Engine 系统提交更加复杂的作业。  
关于 shell 脚本  
Shell 脚本（又称批处理作业）是汇编在文件中的命令行指令序列。通过 chmod 命令可  
以执行脚本文件。调用脚本时会启动一个命令解释器。对每条指令的解释方式与运行脚  
本的用户手动键入指令时的解释方式相同。 csh、 tcsh、 sh 和 ksh 是典型的命令解释  
器。可以在一个 shell 脚本中调用仲裁命令、应用程序和其他 shell 脚本。  
命令解释器可以作为登录 shell 调用。为此，命令解释器必须包含在 Grid Engine 系统  
配置的 login\_shells 列表中，该配置对运行作业的特定的主机或队列有效。  
第 3 章 • 提交作业 **47**  
注 **–** 对于群集中配置的各个主机和队列， Grid Engine 系统 的配置可能会有所不同。使  
用 qconf 命令的 -sconf 和 -sq 选项，可以显示生效的配置。有关详细信息，请参阅  
qconf(1) 手册页。  
如果像 login shell 那样调用命令解释器，则作业环境与登录并运行脚本的环境相同。例  
如，使用 csh 时， 将执行 .login 和 .cshrc，同时还执行系统默认的启动资源文  
件，如 /etc/login。但是，如果 csh 没有作为 login-shelll 调用，则仅执行  
.cshrc。有关是否作为 login-shell 调用之间的差异的说明，请参阅命令解释器的  
手册页。  
shell 脚本示例  
示例 3–1 是一个简单的 shell 脚本。脚本首先从 Fortran77 源文件中编译应用程序  
flow，然后运行该应用程序。  
示例 **3–1** 简单的 shell 脚本  
#!/bin/csh  
# This is a sample script file for compiling and  
# running a sample FORTRAN program under N1 Grid Engine 6  
cd TEST  
# Now we need to compile the program "flow.f" and  
# name the executable "flow".  
f77 flow.f -o flow  
本地系统用户指南提供了关于构建和定制 shell 脚本的详细信息。您或许还希望参阅  
sh、 ksh、 csh 或 tcsh 手册页。以下各节重点介绍为 Grid Engine 系统准备批处理脚  
本需要考虑的特别事项。  
通常情况下，可以将任何可从命令提示符下手动运行的 shell 脚本提交到 Grid Engine  
系统 。这样的 shell 脚本不需要终端连接和交互式用户介入。标准错误和标准输出设备  
例外，它们是自动重定向的 。因此，可以将示例 3–1 提交给 Grid Engine 系统，且脚本  
将执行预期的操作。  
常规 shell 脚本扩展  
一些常规 Shell 脚本的扩展会影响在 Grid Engine 系统 的控制下运行的脚本的操作。以  
下各节将介绍这些扩展。  
**48** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
如何选择命令解释器  
提交作业时，您可以指定用来处理作业脚本文件的命令解释器。示例请参阅图 3–5。不  
过，如果未进行指定，配置变量 shell\_start\_mode 确定如何选择命令解释器：  
■ 如果 shell\_start\_mode 设置为 unix\_behavior，则脚本文件的第一行将指定  
命令解释器。脚本文件的第一行必须以 #! 开头。如果第一行不是以 #! 开头，  
Bourne Shell sh 将做为默认的方式使用。  
■ 对于 shell\_start\_mode 的所有其他设置，默认的命令解释器由作业启动时所在  
的队列的 shell 参数决定。请参阅第 34 页中的 “显示队列和队列属性” 和  
queue\_conf(5) 手册页。  
输出重定向  
由于批处理作业没有终端连接，因此必须将它们的标准输出和标准错误重定向到文件。  
Grid Engine 系统可使用户定义将输出重定向到的文件的位置。如果未指定输出文件，  
则使用默认设置。  
该文件的标准位置是运行作业的当前工作目录。默认的标准输出文件名是  
*job-name*.o*job-id*，默认的标准错误输出会重定向到 *job-name>*.e*job-id*。可由脚本文件名  
生成 *job-name*，或由用户定义作业名。请参阅 submit(1) 手册页中的 -N 选项示例。  
*job-id* 是 Grid Engine 系统分配给作业的专有标识符。  
在阵列作业任务中，任务标识符通过一个点号分隔添加到文件名称中。因此，最终的标  
准重定向路径是 *job-name*.o*job-id*.*task-id>* 和 *job-name*.e*job-id*.*task-id*。有关更多信息，  
请参阅第 64 页中的 “提交阵列作业” 。  
如果标准位置不合适，用户可以用 QMON 指定输出方向，如图 3–6 所示。或者用户可以  
使用 qsub 命令的 -e 和 -o 选项指定输出位置。标准输出和标准错误输出可以合并到一  
个文件。重定向可以在每个执行主机的基础上指定。也就是说，执行作业的主机不同，  
输出重定向文件的位置也因之不同。构建定制的专有重定向文件路径时，可将虚设备环  
境变量与 qsub -e 和 -o 选项一起使用。下面是这些变量的列表。  
■ HOME – 执行计算机的根目录  
■ USER – 作业拥有者的用户 ID  
■ JOB\_ID – 当前的作业 ID  
■ JOB\_NAME – 当前作业名称，请参阅 -N 选项  
■ HOSTNAME – 执行主机的名称  
■ TASK\_ID – 阵列作业任务索引编号  
作业运行时，这些变量将使用实际的值，重定向路径也由这些实际的值组成。  
有关更多详细信息，请参阅 qsub(1) 手册页。  
第 3 章 • 提交作业 **49**  
有效的注释  
带有前导符号 # 的行将被视为 shell 脚本中的注释。 但是， Grid Engine 系统可以识别  
特殊的注释行并以特殊的方式使用。这种脚本行的其余部分作为 qsub 命令的命令行变  
量列表的一部分。这些特殊的注释行中提供的 qsub 选项在 “QMON Submit Job” 对话框  
中也可进行解释。相应的参数在选择脚本文件时预设。  
默认状态下，特殊的注释行以 #$ 前缀字符串标识。可以使用 qsub -C 命令重新定义前  
缀字符串。  
这种使用特殊注释的方法称为提交自变量的脚本嵌入。以下是使用嵌入脚本命令行选项  
的脚本示例。  
示例 **3–2** 使用嵌入脚本的命令行选项  
#!/bin/csh  
#Force csh if not Grid Engine default  
#shell  
#$ -S /bin/csh  
# This is a sample script file for compiling and  
# running a sample FORTRAN program under N1 Grid Engine 6  
# We want Grid Engine to send mail  
# when the job begins  
# and when it ends.  
#$ -M EmailAddress  
#$ -m b,e  
# We want to name the file for the standard output  
# and standard error.  
#$ -o flow.out -j y  
# Change to the directory where the files are located.  
cd TEST  
# Now we need to compile the program "flow.f" and  
# name the executable "flow".  
f77 flow.f -o flow  
# Once it is compiled, we can run the program.  
flow  
**50** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
环境变量  
运行作业时，会有一系列变量预设到作业环境中。以下是这些变量的列表：  
■ ARC – 运行作业的节点的结构名称。该名称编译到 sge\_execd 二进制文件中。  
■ SGE\_ROOT – Grid Engine 系统的根目录，可在启动前将该目录设置到 sge\_execd  
文件中，也可使用默认的 /usr/SGE。  
■ SGE\_BINARY\_PATH – 安装 Grid Engine 系统二进制文件的目录。  
■ SGE\_CELL – 运行作业的单元。  
■ SGE\_JOB\_SPOOL\_DIR – 运行作业时， sge\_shepherd 用来存储作业相关数据的目  
录。  
■ SGE\_O\_HOME – 提交作业的主机上作业拥有者的根目录路径。  
■ SGE\_O\_HOST – 提交作业的主机。  
■ SGE\_O\_LOGNAME – 提交作业的主机上作业拥有者的登录名称。  
■ SGE\_O\_MAIL – 作业提交命令上下文中 MAIL 环境变量的内容。  
■ SGE\_O\_PATH – 作业提交命令上下文中 PATH 环境变量的内容。  
■ SGE\_O\_SHELL – 作业提交命令上下文中 SHELL 环境变量的内容。  
■ SGE\_O\_TZ – 作业提交命令上下文中 TZ 环境变量的内容。  
■ SGE\_O\_WORKDIR – 提交作业命令的工作目录。  
■ SGE\_CKPT\_ENV – 点校验作业运行的点校验环境。用 qsub -ckpt 命令选择点校验  
环境。  
■ SGE\_CKPT\_DIR – 点校验接口的 ckpt\_dir 路径 。仅为点校验作业设置。更多信  
息，请参阅 checkpoint(5) 手册页。  
■ SGE\_STDERR\_PATH – 作业的标准错误流转向到的文件路径名称。该文件通常用于加  
强来自前导脚本、收尾脚本、并行环境启动与停止脚本或者点校验脚本错误信息的输  
出。  
■ SGE\_STDOUT\_PATH – 作业标准输出流转向到的文件路径名称。该文件通常用于加强  
来自前导脚本、收尾脚本、并行环境启动与停止脚本或者点校验脚本信息的输出。  
■ SGE\_TASK\_ID – 阵列作业内代表任务的任务标识符。  
■ ENVIRONMENT – 通常设置为 BATCH。该变量表示脚本将以批处理模式运行。  
■ HOME – passwd 文件中的用户主目录路径。  
■ HOSTNAME – 运行作业的节点的主机名称。  
■ JOB\_ID – 提交作业时， sge\_qmaster 分配给作业的专有标识符。作业 ID 是从 1  
到 9,999,999 的十进制整数。  
■ JOB\_NAME – 作业名（由 qsub *script filename* 生成）、句点和作业 ID 数字。您可使  
用 qsub -N 替代此默认设置。  
■ LOGNAME – passwd 文件中的用户登录名称。  
■ NHOSTS – 并行作业所使用的主机数量。  
■ NQUEUES – 分配给作业的队列数量。对于串行作业，该数目通常为 1。  
■ NSLOTS – 并行作业所使用的队列位置数。  
第 3 章 • 提交作业 **51**  
■ PATH – 默认的 shell 搜索路径是  
： /usr/local/bin:/usr/ucb:/bin:/usr/bin。  
■ PE – 作业运行的并行环境。该参数仅限于并行作业。  
■ PE\_HOSTFILE – 包含虚拟并行计算机定义的文件的路径，该虚拟并行计算机由 Grid  
Engine 系统指定给并行作业。 此变量只能用于并行作业。有关该文件格式的详细信  
息，请参阅 sge\_pe 中的 $pe\_hostfile 参数说明。  
■ QUEUE – 运行作业的队列名称。  
■ REQUEST – 作业的请求名称。该名称是作业脚本文件名称或 qsub -N 命令明确指定  
给作业的名称。  
■ RESTARTED – 表明是否重新启动点校验作业。如果该值设定为 1，则表明作业至少  
中断过一次。因此作业重新启动。  
■ shell – passwd 文件中用户的登录 shell。  
注 **–** SHELL 不一定是用于作业的 shell。  
■ TMPDIR – 作业临时工作目录的绝对路径。  
■ TMP – 同 TMPDIR。该变量用于提供与 NQS 的兼容性。  
■ TZ – 从 sge\_execd 导入的时区变量（若已设置）。  
■ USER – passwd 文件中的用户登录名称。  
提交扩展作业和高级作业  
扩展作业和高级作业是更为复杂的作业提交形式。尝试提交这些作业之前，必须了解关  
于该过程的一些重要的背景信息。以下各节提供了这些信息。  
用 QMON 提交扩展作业  
“Submit Job” 对话框的 “General” 选项卡允许为扩展作业配置以下参数。 “General” 选  
项卡如图 3–2 中所示。  
■ Prefix – 脚本嵌入提交选项使用的前缀字符串。有关详细信息，请参阅第 50 页中的  
“有效的注释” 。  
■ Job Script – 使用的作业脚本。单击 “Job Script” 字段右边的图标打开文件选择对话  
框。文件选择框如图 3–4 中所示。  
■ Job Tasks – 提交阵列作业的任务 ID 范围。有关详细信息，请参阅 第 64 页中的 “  
提交阵列作业” 。  
**52** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ Job Name – 作业的名称。选择作业脚本之后会设置一个默认名称。  
■ Job Args – 作业脚本的自变量。  
■ Priority – 用于设置作业初始优先级的计数框。此优先级用于排列某用户多个作业的  
顺序。当用户在系统中同时有多个作业时，优先级会通知调度程序如何在单个用户作  
业中进行选择。  
注 **–** 为使用户能够设置各自作业的优先级，管理员必须使用调度程序配置的  
weight\_priority 参数启用优先级。有关更多信息，请参阅《 N1 Grid Engine 6  
管理指南》中的第 5 章 “管理策略和调度程序”。  
■ Job Share – 定义相对于其他作业的作业票券份额。作业份额只影响份额树策略和职  
能策略。  
■ Start At – 作业符合执行条件的启动时间。单击 “Start At” 字段右边的图标打开对话  
框，然后输入格式正确的时间。  
■ Project – 作业所属的项目。单击 “Project ” 字段右边的图标在可用的项目中进行选择  
：  
■ Current Working Directory – 表明是否在当前工作目录执行作业的标志。仅在提交主  
机和可能的执行主机之间具有同样的目录层次时才使用该标志。  
第 3 章 • 提交作业 **53**  
■ Shell – 用于运行作业脚本的命令解释器。有关详细信息，请参阅第 49 页中的 “如何  
选择命令解释器” 。单击 “Shell” 字段右边的图标打开对话框，然后键入作业的命令  
解释器规范：  
■ Merge Output – 指明是否将作业的标准输出和标准错误输出合并为标准输出流的标  
志。  
■ stdout – 要使用的标准输出重定向位置。有关详细信息，请参阅第 49 页中的 “输出  
重定向” 。如果未进行指定，则使用默认位置。单击 stdout 字段右边的图标打开  
对话框，输入输出重定向到的位置：  
■ stderr – 要使用的标准错误输出重定向位置，类似于标准输出重定向。  
■ stdin – 要使用的标准输入文件，类似于标准输出重定向。  
■ Request Resources – 单击该按钮定义作业的资源需求。如果作业请求了资源，该按  
钮的颜色会更改。  
■ Restart depends on Queue – 单击该按钮定义在系统崩溃或类似事件终止作业后，作  
业是否可以重新启动 。该按钮还控制重新启动行为是取决于队列还是作业的要求。  
■ Notify Job – 表明在暂停或取消作业之前是否通过 SIGUSR1 或 SIGUSR2 信号通知  
用户的标志。  
■ Hold Job – 表明要为作业指定用户留置状态或作业从属性的标志。只要指定作业处于  
留置状态，作业就不符合执行的条件。有关更多信息，请参阅第 77 页中的 “监视  
和控制作业” 。 “Hold Job” 字段将留置限制为阵列作业的特定任务范围。有关阵列  
作业的信息，请参阅第 64 页中的 “提交阵列作业” 。  
■ Start Job Immediately – 强制在可能的情况下立即启动作业或拒绝作业的标志。如果  
选择该标志，则作业不进入队列。  
**54** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ Job Reservation – 指定为作业保留资源的标志。有关详细信息，请参阅《 N1 Grid  
Engine 6 管理指南》中的“资源预留和回填”中的“资源预留和回填”。  
“Submit Job” 对话框右边的按钮可使您启动多种操作：  
■ **Submit** – 提交当前指定的作业。  
■ **Edit** – 使用 vi 或 EDITOR 环境变量定义的编辑程序在 X 终端编辑选中的脚本文件。  
■ **Clear** – 清除 “Submit Job” 对话框中所有的设置，包括任何指定的资源请求。  
■ **Reload** – 重新载入指定的脚本文件、解析所有脚本嵌入的选项 、解析默认设置并丢  
弃这些设置的中间手动更改。有关详细信息，请参阅第 50 页中的 “有效的注释” 和  
第 60 页中的 “默认的请求文件” 。该操作等同于对以前的脚本文件执行具有后续  
规范的 “Clear” 操作。 只有选择脚本文件时该选项才有效。  
■ **Save Settings** – 将当前的设置保存到文件。将打开文件选择对话框选择文件。保存  
的文件可随后加载，也可作为默认请求使用。有关详细信息，请参阅 **Load Settings**  
和第 60 页中的 “默认的请求文件” 。  
■ **Load Settings** –加载先前用 “Save Settings” 按钮保存的设置。加载的设置将覆盖当  
前设置。请参阅保存设置。  
■ **Done** – 关闭 “Submit Job” 对话框。  
扩展作业示例  
图 3–5 显示了 “Submit Job” 对话框，其中包含大多数参数设置：  
第 3 章 • 提交作业 **55**  
图 **3–5** 扩展作业提交示例  
本示例中配置的作业参数表明以下内容：  
■ 作业的脚本文件为 flow.sh，该文件必须驻留在 QMON 的工作目录下。  
■ 作业名为 Flow。  
■ 脚本文件采用单个自变量 big.data。  
■ 作业启动的优先级为 3。  
■ 2004 年 4 月 22 日凌晨 4:30.44 之后的作业符合执行条件。  
■ 项目定义表明作业从属于项目 crash。  
■ 作业在提交工作目录下执行。  
■ 作业使用 tcsh 命令解释器。  
■ 标准输出和标准错误输出合并到文件 flow.out 中，该文件在当前工作目录中创  
建。  
从命令行提交扩展作业  
要从命令行提交图 3–5 所示的扩展作业请求，请键入以下命令：  
% qsub -N Flow -p -111 -P devel -a 200404221630.44 -cwd \  
-S /bin/tcsh -o flow.out -j y flow.sh big.data  
**56** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
用 QMON 提交高级作业  
“Submit Job” 对话框中的 “Advanced” 选项卡允许定义以下附加参数：  
■ Parallel Environment – 要使用的并行环境界面  
■ Environment – 作业运行前给作业设置的环境变量组。单击 “Environment ” 字段右  
边的图标打开对话框，以获得要导出的环境变量的定义：  
可以从 QMON 的运行时环境获得环境变量。或者，也可以定义自己的环境变量。  
■ Context – 名称/值对的列表，可用来存储和传递与作业相关的信息。可从群集的任  
何位置访问此信息。从命令行使用带有 -ac、 -dc 和 -sc 选项的 qsub、 qrsh、  
qsh、 qlogin 和 qalter 命令修改环境变量。可以使用 qstat -j 命令检索环境变  
量。  
■ Checkpoint Object – 希望并适于对作业执行点校验时使用的点校验环境。有关详细  
信息，请参阅第 97 页中的 “使用点校验作业” 。  
■ Account – 与作业相关联的统计字符串。统计字符串将添加到为作业保留的统计记  
录。统计记录可用于以后的统计分析中。  
■ Verify Mode – 验证标志确定作业的一致性检查模式。为检查作业请求的一致性，  
Grid Engine 系统假定了一个空的、尚未加载的群集。系统将为作业至少查找一个可  
以运行作业的队列。可用的检查模式如下：  
■ **Skip** – 根本不执行一致性检查。  
第 3 章 • 提交作业 **57**  
■ **Warning** – 报告不一致性，但是作业仍可接受。如果提交作业后需要改变群集配  
置，可使用警告模式。  
■ **Error** – 报告不一致性。只要存在不一致的情形，就会拒绝作业。  
■ **Just verify** – 作业尚未提交。有关群集中每个主机和队列与作业的匹配情况，会  
生成一个内容广泛的报告。  
■ Mail – 通过电子邮件通知用户的事件。当前定义了作业的启动、结束、中止和暂停  
事件。  
■ Mail To – 将通知发送到的电子邮件地址列表。单击 “Mail To” 字段右边的图标对话  
框，定义邮件列表：  
■ Hard Queue List, Soft Queue List – 执行作业时强制选择的队列名称列表。对于相应  
的资源请求， “Hard Queue List” 和 “Soft Queue List” 以相同的方式进行处理。  
■ Master Queue List – 符合并行作业主队列条件要求的队列名称列表。并行作业在主  
队列中启动。作业派生并行任务的所有其他的队列称为从队列。  
■ Job Dependencies – 在已提交的作业启动之前必须完成的作业 ID 的列表。新创建的  
作业依赖于这些作业的完成。  
■ Deadline – 限期作业的限期启动时间。在限期启动时间定义的时间点处，限期作业必  
须达到最高优先级以在给定的限期之前完成。要确定限期启动时间，请从作业的限期  
时间中减去作为最高优先级的作业的估计运行时间。单击 “Deadline” 字段右边的图  
标打开如下所示的对话框：  
**58** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 并非所有用户都可提交限期作业。询问系统管理员是否允许您提交限期作业。向  
群集管理员咨询以了解可赋予限期作业的最高优先级。  
高级作业示例  
图 3–6 所示是一个高级作业提交的示例。  
图 **3–6** 高级作业提交示例  
第 3 章 • 提交作业 **59**  
第 55 页中的 “扩展作业示例” 中定义的作业与第 52 页中的 “用 QMON 提交扩展作业  
” 中定义的作业相比，具有以下附加特征：  
■ 该作业需要使用并行环境 mpi。该作业需要创建至少 4 个并行处理程序。如果具有  
可用的进程，则该作业可使用多达 16 个进程。  
■ 该作业需要设置和导出两个环境变量。  
■ 设置两个环境变量。  
■ 统计字符串 FLOW 将添加到作业统计记录。  
■ 一旦作业启动和完成，就必须将邮件发送到 me@myhost.org。  
■ 最好在队列 big\_q 中执行作业。  
从命令行提交高级作业  
要从命令行提交图 3–6 所示的高级作业请求，请键入以下命令：  
% qsub -N Flow -p -111 -P devel -a 200012240000.00 -cwd \  
-S /bin/tcsh -o flow.out -j y -pe mpi 4-16 \  
-v SHARED\_MEM=TRUE,MODEL\_SIZE=LARGE \  
-ac JOB\_STEP=preprocessing,PORT=1234 \  
-A FLOW -w w -m s,e -q big\_q\  
-M me@myhost.com,me@other.address \  
flow.sh big.data  
默认的请求文件  
上面的示例表明高级作业复杂且难以处理，尤其是需要频繁提交类似请求时。为简化命  
令输入和防止犯错，用户可以在脚本文件中嵌入 qsub 选项。或者用户也可以使用默认  
的请求文件。有关更多信息，请参阅第 50 页中的 “有效的注释” 。  
注 **–** 使用 y 自变量指定的 -binary yes|no 选项允许使用 qrsh 提交可执行作业，而  
无需脚本包装。请参阅 qsub 手册页。  
群集管理可以为所有 Grid Engine 系统用户设置默认的请求文件。另外，用户也可以在  
自己的主目录下创建专用的默认请求文件。用户还可在其工作目录下创建特定于应用程  
序的默认请求文件。  
默认的请求文件包含 qsub 选项，默认将应用于一行或多行中的作业。全局群集默认请  
求文件位于 *sge-root*/*cell*/common/sge\_request。一般专用默认请求文件位于  
$HOME/.sge\_request 下。特定于应用程序的默认请求文件位于  
$cwd/.sge\_request 下。  
**60** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
如果具有多个这样的文件，这些文件会合并为一个默认请求文件，其优先顺序如下：  
1. 特定于应用程序的默认请求文件  
2. 一般专用默认请求文件  
3. 全局默认请求文件  
脚本嵌入和 qsub 命令行比默认请求文件具有更高的优先级。 所以脚本嵌入设置会替代  
默认请求文件的设置。 qsub 命令行选项可以再次替代这些设置。  
要放弃以前的设置，请在默认请求文件、嵌入的脚本文件或 qsub 命令行中使用 qsub  
-clear 命令。  
以下是一个专用默认请求文件示例：  
-A myproject -cwd -M me@myhost.com -m b,e  
-r y -j y -S /bin/ksh  
除非被覆盖，否则，以下情形适用于所有用户的作业：  
■ 帐户字符串为 myproject  
■ 作业在当前的工作目录下执行  
■ 作业启动和结束时，邮件通知发送到： me@myhost.com  
■ 标准输出和标准错误输出合并  
■ ksh 用作命令解释器  
定义资源需求  
到目前为止的示例中，提交选项没有向要执行作业的主机提出任何资源需求。 Grid  
Engine 系统假定作业可以在任何主机上运行。然而，在实际操作中为使作业顺利完成，  
执行主机需要满足特定的先决条件。先决条件包括足够的可用内存、安装所需的软件或  
特定的操作系统结构。另外，群集管理员通常对群集中计算机的使用施加限制。例如，  
限定作业可以使用的 CPU 时间。  
Grid Engine 系统为用户提供了为作业查找适当主机的方法，即使用户不具备群集设备  
及其使用策略的知识。用户只需指定他们的作业要求， Grid Engine 系统将查找适当  
的、负荷较轻的主机。  
通过第 37 页中的 “可请求的属性” 中介绍的可请求的属性指定资源需求。 QMON 提供了  
指定作业要求的简便方法。 “Requested Resources” 对话框仅显示 “Available Resource”  
列表中当前符合条件的属性。在 “Submit Job” 对话框中单击 “Request Resources” 打开  
“Requested Resources” 对话框。有关示例，请参阅图 3–7。  
第 3 章 • 提交作业 **61**  
图 **3–7** “Requested Resources” 对话框  
双击属性时，该属性将添加到作业的硬性或软性资源列表。将打开一个对话框，指导您  
输入相关的属性值， BOOLEAN 属性除外（设置为 True）。有关更多信息，请参阅第 63  
页中的 “ Grid Engine 系统如何分配资源” 。  
图 3–7 所示是某个作业的资源配置文件，该配置文件要求使用一台至少具有 750 MB 内  
存、拥有 permas 许可证的 solaris64 主机。如果查找到的满足此要求的队列不止一  
个，将考虑定义过的软性资源需求。如果没有查找到同时满足硬性要求和软性要求的队  
列，则认为任何满足硬性要求的队列为适当的队列。  
注 **–** 仅当多个队列满足作业要求时，调度程序配置的 queue\_sort\_method 参数才能  
确定从何处启动作业。有关详细信息，请参阅 sched\_conf(5) 手册页。  
permas 属性（整数）是全局资源属性的管理员扩展属性。 arch 属性（字符串）是一  
项主机资源属性。 h\_vmem 属性（内存）是队列资源属性。  
也可以使用 qsub 命令提交相同的资源需求配置文件：  
% qsub -l arch=solaris64,h\_vmem=750M,permas=1 \  
permas.sh  
第一个 -l 选项之前隐含的 -hard 开关被跳过。  
代表 750 MB 的 750M 符号是 Grid Engine 系统数量语法的一个示例。对于请求内存使  
用情况的属性，可以为其指定整数十进制、浮点十进制、整数八进制和整数十六进制数  
值。 必须为这些数字附加以下乘数：  
■ k – 乘以 1000。  
■ K – 乘以 1024。  
**62** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ m – 乘以 1000 的 1000 倍。  
■ M – 乘以 1024 的 1024 倍。  
八进制常数以零开始，由 0-7 之间的数字指定。要指定十六进制常数，必须在数字前加  
0x 前缀。必须使用 0-9 之间的数字、 a-f 和 A-F。如果没有附加乘数，这些数值将以字节  
为单位。如果使用浮点十进制，结果值将截断为整数值。  
对于含有时间限制的属性，可以用小时、分钟、秒或者它们的组合指定时间值。在十进  
制中，小时、分钟和秒之间以冒号分隔。 时间值 3:5:11 会转换为 11111 秒。如果小  
时、分钟或秒为 0，可以将相应的位置留空并保留冒号。数值 :5: 表示 5 分钟。  
“Requested Resources” 对话框（如图 3–7 所示）中所使用的表格是一个扩展表格，仅  
在 QMON 中有效。  
Grid Engine 系统如何分配资源  
如上面一节所示了解 Grid Engine 软件 如何处理资源请求并分配资源具有重要意义。下  
面示意性地介绍了 Grid Engine 软件的资源分配运算法则。  
1. 读入并解析所有默认的请求文件。有关详细信息，请参阅第 60 页中的 “默认的请求  
文件” 。  
2. 处理作为嵌入选项的脚本。有关详细信息，请参阅第 50 页中的 “有效的注释” 。  
3. 提交作业时将读取所有的嵌入脚本选项，而不考虑脚本文件的位置。  
4. 从命令行读取并解析所有的请求。  
当收集到全部的 qsub 请求后，将分别处理硬性和软性请求，硬性请求优先。对应于下  
面的优先次序，评估请求：  
1. 从左到右评估脚本/默认的请求文件  
2. 从上到下评估脚本/默认的请求文件  
3. 从左到右评估命令行  
换言之，可以用命令行替代嵌入标志。  
分配请求的硬性资源。如果请求无效，则会拒绝作业的提交。如果提交时有一个或多个  
队列请求无法满足，该作业将假脱机并重新调度以便在以后的时间运行。例如，由于所  
请求的队列忙碌而无法满足作业请求。如果满足所有硬性请求，则分配请求并运行作  
业。  
随后检查软性资源请求。即使一个或全部软性请求都不能满足，作业也能运行。如果有  
多个队列满足硬性请求，并可提供软性资源列表的一部分，则 Grid Engine 软件选择可  
满足最多软性请求的队列。  
作业启动并覆盖分配的资源。  
您可能希望搜集自变量列表选项、嵌入选项或硬性和软性请求之间如何相互影响的信  
息。 您可以通过执行含有 UNIX 命令（如 hostname 或 date）的小型测试脚本文件进  
行试验。  
第 3 章 • 提交作业 **63**  
作业从属性  
构建复杂任务的最简便的方法是将一个任务拆分为子任务。在这种情况下，从属子任务  
是否可以启动取决于其他子任务的完成情况。例如，从属任务必须读取并处理由先导任  
务生成的输出文件。  
Grid Engine 系统通过其作业从属性工具支持交叉从属任务。可以对作业进行配置，使  
其依赖一个或多个其他作业的完成情况。 qsub -hold\_jid 命令对该工具进行了加  
强。可以为提交的作业指定其所从属的作业列表。该作业列表也可以包含阵列作业的子  
集。只有当从属列表中所有的作业完成后，提交作业才具有执行条件。  
提交阵列作业  
Grid Engine 系统的阵列作业工具的理想应用情形是，在作业脚本中对同一操作组需要  
执行参数化的、重复的执行。此应用的典型示例为数字内容创作行业中的图形渲染任  
务。对动画的计算将拆分为帧。可对每帧执行相同的渲染计算。  
阵列作业工具提供了一个提交、监视和控制这类应用程序的简便方法。 Grid Engine 系  
统提供了执行阵列作业的有效的方式，可将计算拆分成独立的任务组，然后再组合为一  
个单独的作业。阵列作业任务通过组索引编号引用。所有任务的索引位于整个阵列作业  
的索引范围内。索引范围是在提交阵列作业过程中通过 qsub 命令定义的。  
可以监视和控制阵列作业。例如，可以整体暂停、恢复或取消阵列作业，也可以暂停、  
恢复或取消阵列作业内的单个任务或任务子集。引用任务时，应将相应的索引编号作为  
作业 ID 的后缀。任务与常规作业的执行方式非常相似。任务可以使用环境变量  
SGE\_TASK\_ID 检索其自身的索引编号并访问为此任务标识符指定的输入数据集。  
用 QMON 提交阵列作业  
请遵循第 43 页中的 “如何用 QMON 提交简单的作业” 中的指导，并认真考虑以下信  
息。  
从 QMON 提交阵列作业的方法实际上与第 43 页中的 “如何用 QMON 提交简单的作业”  
中所描述的简单作业的提交方法相同。唯一的区别是必须在图 3–5 中所示的 “Job Tasks”  
输入窗口中指定任务范围。定义任务范围的语法与 qsub -t 命令的语法相同。有关队  
列索引语法的详细信息，请参阅 qsub(1) 手册页。  
有关监视和控制作业概述以及对阵列作业的特别说明，请参阅第 77 页中的 “监视和控  
制作业” 和第 86 页中的 “从命令行监视和控制作业” 。另外，还可参阅 qstat(1)、  
qhold(1)、 qrls(1)、 qmod(1) 和 qdel(1) 的手册页。  
阵列作业具有 Grid Engine 系统工具提供给常规作业的所有访问权限。特别地，阵列作  
业同时还可以是并行作业。或者阵列作业可以与其他的作用具有交叉从属性。  
**64** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 任务组不能与其他作业或任务组交叉从属。  
从命令行提交阵列作业  
要在命令行提交阵列作业，请键入带有适当的自变量的 qsub 命令。  
以下是如何提交阵列作业的示例：  
% qsub -l h\_cpu=0:45:0 -t 2-10:2 render.sh data.in  
-t 选项定义任务索引范围。在 2-10:2 中，指定 2 为最低的索引编号， 10 为最高的索  
引编号。如果索引的间隔为 2，使用该参数的 :2 部分指定。这样，阵列作业由 5 个索  
引编号分别为 2、 4、 6、 8 和 10 的任务组成。每项任务请求都通过 -l 选项，请求硬性  
的 CPU 时间限制为 45 分钟。 一旦 Grid Engine 系统分派并启动任务，每个任务就执  
行作业脚本 render.sh 。这些任务使用 SGE\_TASK\_ID 判断其编号（ 2、 4、 6、 8 和  
10）。 任务使用它们的索引编号在 data.in 数据文件中查找它们的输入数据记录。  
提交交互式作业  
需要用户直接输入以影响作业结果时，您应当提交交互式而不是批处理作业。典型的情  
况如 X Windows 应用程序，或任务要求解释即时结果以进行进一步的处理。  
可使用以下三种方法创建交互式作业：  
■ qlogin – 在 Grid Engine 软件选定的主机上开始的远程会话。  
■ qrsh – 与标准 UNIX rsh 工具相当的工具。在由 Grid Engine 软件选择的主机上远  
程运行命令。如果未指定命令，则在远程主机上启动远程 rlogin 会话。  
■ qsh – 从运行该作业的计算机上显示的 xterm。按照指定或 DISPLAY 环境变量的设  
置显示。如果未设置 DISPLAY 变量，且未定义显示目标 ，则 Grid Engine 系统将  
xterm 导向到作业提交主机的 X 服务器的 0.0 屏幕。  
注 **–** 为正确运行，需要未所有工具配置正确的 Grid Engine 系统群集参数。必须为  
qsh 命令定义正确的 xterm 执行路径。此类型作业必须具有可用的交互式作业。请  
联系您的系统管理员，了解您的群集是否已就绪，可以执行交互式作业。  
默认的交互式作业处理方法与批处理作业的处理方法不同。提交交互式作业时，如果不  
能执行，则不对其进行排队。如果未对交互式作业进行排队，则表明没有足够的可用资  
源来分派交互式作业 。这种情况下，将通知用户群集当前处于忙碌状态。  
第 3 章 • 提交作业 **65**  
可以使用 qsh、 qlogin 和 qrsh 的 -now no 选项更改此默认行为。如果使用该选项，  
交互式作业和批处理作业一样排入队列。使用 -now yes 选项时，通过 qsub 命令提交  
的批处理作业也可以按照交互式作业的方法进行处理。这样的交互式作业或者立即分派  
运行，或者遭到拒绝。  
注 **–** 交互式作业只能在 INTERACTIVE 类型的 队列中运行。有关详细信息，请参  
阅《 N1 Grid Engine 6 管理指南》中的“配置队列”。  
以下章节介绍使用 qlogin 和 qsh 工具的方法。 qrsh 命令将在第 69 页中的 “透明的  
远程执行” 中以更广泛的环境进行介绍，。  
用 QMON 提交交互式作业  
在 Grid Engine 系统选择的主机上具有 xterm 的作业是可从 QMON 提交的交互式作业的  
唯一类型。  
在 “Submit Job” 对话框的右侧，单击 “Submit” 按钮上面的按钮直到显示交互式图标。  
执行这些操作，然后在 “Submit Job” 对话框中提交交互式作业。请参阅图 3–8 和图  
3–9。  
对话框中各选项的含义和用法与第 47 页中的 “提交批处理作业” 中所介绍的批处理作  
业相同。不同之处在于有几处输入字段是灰色的，这是因为这些字段不适用于交互式作  
业。  
**66** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
图 **3–8** “Interactive Submit Job” 对话框， “General“ 选项卡  
第 3 章 • 提交作业 **67**  
图 **3–9** “Interactive Submit Job” 对话框， “Advanced” 选项卡  
用 qsh 提交交互式作业  
qsh 类似于 qsub 命令。 qsh 支持多个 qsub 选项，还支持使用另一个选项 -display  
来定向要调用的 xterm 的显示。有关详细信息，请参阅 qsub(1) 手册页。  
若要使用 qsh 提交交互式作业，请键入以下命令：  
% qsh -l arch=solaris64  
该命令在任何可用的 Sun Solaris 64 位操作系统主机上启动 xterm 。  
用 qlogin 提交交互式作业  
在任何终端或终端模拟程序中使用 qlogin 命令启动 Grid Engine 系统控制下的交互式  
会话。  
若要使用 qlogin 提交交互式作业，请键入以下命令：  
% qlogin -l star-cd=1,h\_cpu=6:0:0  
该命令位于低负荷的主机上。该主机具有一个可用的 Star-CD 许可证。另外，该主机至  
少应具有一个至少可提供 6 小时硬性 CPU 时限的队列。  
**68** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 根据为 Grid Engine 系统配置使用的远程登录工具的不同，可能需要在登录提示符  
下键入用户名、密码或者同时键入。  
透明的远程执行  
Grid Engine 系统提供了一组密切相关的工具，可支持特定计算任务的透明的远程执  
行。该功能的核心工具是 qrsh 命令，该命令在第 69 页中的 “用 qrsh 远程执行” 中  
进行了介绍。 qtcsh 和 qmake 这两个高级工具是在 qrsh 基础上构建的。这两个命令  
可使 Grid Engine 系统透明地分布隐含的计算任务，从而增强了标准的 UNIX 工具  
make 和 csh。 qtcsh 的介绍请参阅第 70 页中的 “使用 qtcsh 透明地分布作业” 。  
qmake 在 第 72 页中的 “用 qmake 处理并行 makefile” 中进行了介绍。  
用 qrsh 远程执行  
qrsh 根据标准的 rsh 工具构建。有关 rsh 所含功能的详细信息，请参阅  
*sge-root*/3rd\_party 中提供的信息。 qrsh 可用于多种用途，其中包括：  
■ 提供使用 Grid Engine 系统的交互式应用程序的远程执行功能，类似于标准的 UNIX  
rsh 工具。在 HP-UX 系统中， rsh 也称为 remsh。  
■ 提供使用 Grid Engine 系统的交互式登录会话功能，类似于标准的 UNIX 工具  
rlogin。 qlogin 在 Grid Engine 系统中类似于 UNIX 的 telnet 工具。  
■ 允许提交支持终端 I/O（标准输出、标准错误和标准输入） 和终端控制的批处理作  
业。  
■ 提供了一种提交未嵌入 shell 脚本中的独立程序的方法。  
注 **–** 还可以使用带 -b n 选项的 qrsh 命令提交脚本。有关详细信息，请参阅 qrsh  
手册页。  
■ 当批处理作业挂起或运行时，提供一台处于活动状态的提交客户机；只有当作业完成  
或取消后，此客户机才能离开。  
■ 允许 Grid Engine 系统控制的作业任务在并行作业分配的分散资源框架下远程运行。  
请参阅《 N1 Grid Engine 6 管理指南》中的“并行环境与 Grid Engine 软件的紧密集  
成”。  
由于这些性能， qrsh 成为执行 qtcsh 和 qmake 工具所需的主要启用框架。 qrsh 也用  
于紧密集成的具有并行环境（如 MPI 或 PVM）的 Grid Engine 系统。  
第 3 章 • 提交作业 **69**  
用 qrsh 调用透明的远程执行  
键入 qrsh命令，并根据下面的语法添加选项和自变量：  
% qrsh [*options*] *program*|*shell-script* [*arguments*] \  
[> *stdout*] [>&2 *stderr*] [< *stdin*]  
qrsh 几乎可接受 qsub 的所有选项。 qrsh 命令提供以下选项：  
■ -now yes|no – -now yes 指定立即调度作业。如果没有可用资源，则拒绝作业。  
-now yes 是默认选项。如果作业无法在提交时启动， -now no 选项可指定将作业  
排入队列，如同批处理作业一样。  
■ -inherit – qrsh 在启动作业任务时不执行调度过程。相反， qrsh 假定作业已嵌  
入到指定远程执行主机上已分配了适当的资源的并行作业中。这种 qrsh 格式通常用  
于 qmake 和紧密集成的并行环境中。默认情况下，不继承使用外部作业资源。  
■ -binary yes|no – 使用 n 选项进行指定时，可以使用 qrsh 提交脚本作业。  
■ -noshell – 使用该选项，您无需在用户登的录 shell 中为 qrsh 启动命令行。相  
反，应执行不带封装 shell 的命令。由于避免了一些系统开销（如启动 shell 和查找  
shell 资源文件），因此使用该选项可以提高执行速度。  
■ -nostdin – 抑制输入流 STDIN。通过该选项， qrsh 将 -n 选项传递给 rsh 命令。  
当使用 qrsh 命令以并行方式执行多个任务时，抑制输入流就显得尤为重要，例如，  
在 make 进程中。未定义可获取输入的进程。  
■ -verbose – 该选项将输出传送到正在调度的进程。 -verbose 主要用于调试目的，  
因此在默认状态下是关闭的。  
使用 qtcsh 透明地分布作业  
qtcsh 是广泛使用的、从 UNIX C shell 衍生而来的tcsh 的完全兼容的替代品。 qtcsh  
是在 tcsh 基础上构建的。有关 tcsh 所包含功能的详细信息，请参阅  
*sge-root*/3rd\_party 中提供的信息。 qtcsh 具有扩展的命令 shell， 可将指定应用程序  
的执行透明分布到使用 Grid Engine 系统的适当的、负荷较轻的主机 。远程执行的应用  
程序以及应用到所选择的执行主机的要求在名为 .qtask 的配置文件中进行定义。  
对用户透明，这样的应用程序通过 qrsh 工具提交到 Grid Engine 系统。 qrsh 提供标  
准输出、错误输出和标准输入的处理，还提供连接到远程执行程序的终端控制。远程运  
行这样的应用程序与在同一台主机上作为 shell 运行该应用程序之间存在三方面显著的  
不同：  
■ 远程主机可能功能更强大、负荷较轻、并安装了所需的硬件和软件资源。因此，这样  
的远程主机比本地主机更适合，因为本地主机可能根本就无法运行这些应用程序。当  
然这种差别是我们所期望的。  
■ 作业的远程启动和通过 Grid Engine 系统处理会导致轻微的延迟。  
■ 通过交互式作业和 (qrsh) 从而作用于 qtcsh，管理员可以限制资源的使用。如果没  
有足够的适用资源可用于通过 qrsh 启动应用程序，或所有适用系统均过载、则隐含  
的 qrsh 提交将失败。将返回相应的错误信息，例如： Not enough resources  
... try later。  
**70** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
除标准用途外， qtcsh 还是集成第三方代码和工具的理想平台。 qtcsh 执行单个应用  
程序的格式为 qtcsh -c *app-name*。 在集成环境下使用这种 qtcsh 命令形式，可提供  
一个持久稳定几乎永远不需要更改的界面。其中所需的应用程序、工具、集成、站点甚  
至特定用户的配置都包含在一个恰当定义的 .qtask 文件中。该界面深层优点就是可以  
在任何类型的 shell 脚本、 C 程序甚至 Java 应用程序中使用。  
qtcsh 用法  
调用 qtcsh 命令的方法与调用 tcsh 的方法完全相同。 qtcsh 扩展 tcsh 之处在于提  
供支持 .qtask 文件和专门化的 shell 模式组。  
.qtask 文件的定义如下。文件中的每一行都具有以下格式：  
% [!]*app-name qrsh-options*  
可选的前置感叹号 (!) 用于定义全局群集 .qtask 文件和 qtcsh 用户的个人 .qtask  
文件之间冲突时的优先级别。如果全局群集文件中没有感叹号，则用户文件中的冲突定  
义将覆盖全局群集文件中的定义。如果全局群集文件中有感叹号，则不会进行相应的覆  
盖。  
*app-name* 指定应用程序名称，当在 qtcsh 中的命令行中键入时，该程序就提交到 Grid  
Engine 系统进行远程执行。  
*qrsh-options* 指定 qrsh 工具要使用的选项。这些选项定义了应用程序的资源需求。  
命令行中的应用程序名称必须和 .qtask 文件中定义的完全一样。如果该应用程序以路  
径名称为前缀，则进行本地二进制寻址。不进行远程执行。  
执行应用程序名称比较之前， csh 别名会进行扩展。远程执行应用程序可以出现在  
qtcsh 命令行的任意位置，尤其是在标准 I/O 重定向之前或之后。  
因此，以下示例使用了有效且正确的语法：  
# .qtask file  
netscape -v DISPLAY=myhost:0  
grep -l h=filesurfer  
对于此 .qtask 文件，以下的 qtcsh 命令行：  
netscape  
~/mybin/netscape  
cat very\_big\_file | grep pattern | sort | uniq  
将产生以下隐含结果：  
qrsh -v DISPLAY=myhost:0 netscape  
~/mybin/netscape  
cat very\_big\_file | qrsh -l h=filesurfer grep pattern | sort | uniq  
第 3 章 • 提交作业 **71**  
受交换机打开或关闭状态的影响， qtcsh 可在不同的模式下操作。  
■ 本地或远程执行命令。默认情况下，远程执行命令。  
■ 立即执行或批处理方式执行远程操作。默认情况下为立即执行。  
■ 详细输出或非详细输出。默认情况下为非详细输出。  
这些模式设置可以在启动时使用 qtcsh 的选项自变量进行更改，也可在运行时使用  
shell 内置命令 qrshmode 进行更改。有关更多信息，请参阅 qtcsh(1) 手册页。  
用 qmake 处理并行 makefile  
qmake 是标准 UNIX make 工具的替代工具。 qmake 对 make 工具进行了扩展，可将相  
互独立的 make 步骤分布到适当计算机的群集中。 qmake 根据流行的 GNU-make 工具  
gmake 构建。有关 gmake 所含功能的详细信息，请参阅 *sge-root*/3rd\_party 中提供  
的信息。  
为确保分布的 make 进程能够持续运行直至结束， qmake 首先以类似于并行作业的方式  
分配所需的资源。 qmake 随后管理该资源集，而不再与调度程序交互操作。当有可用的  
资源时， qmake 使用 qrsh 工具及其 -inherit 选项分布 make 步骤。  
由于 qrsh 为远程执行的 make 步骤提供标准输出、错误输出和标准输入的处理以及终  
端控制，因此本地执行 make 过程与使用 qmake 之间存在三方面显著的不同：  
■ 假定单个 make 步骤有一定的有效期，并且有足够多的独立 make 步骤以待处理，  
make 过程的并行化会明显提高处理速度。当然这种差别是我们所期望的。  
■ 在将要远程启动的 make 步骤中， qrsh 和远程执行会产生一个微小的系统开销 。  
■ 为利用 qmake 可分布 make 步骤的优点，用户必须指定并行化的最低程度。也就是  
说，用户必须指定可同时执行的 make 步骤的数量。另外，用户还可以指定 make 步  
骤要求的资源特性，例如，可用的软件许可证、计算机结构、内存或 CPU 时间要  
求。  
make 命令通常用于编译复杂的软件包。不过， qmake 的主要应用却不是编译。优秀的  
编程技巧应产生很小的程序文件。因此，编译程序文件作为 make 的一个步骤，仅需花  
费数秒钟的时间。另外，编译通常会引用大量的文件访问。嵌套方式包含的文件会导致  
这种问题。如果以并行方式执行多个 make 步骤，则文件的访问速度不会提高，文件服  
务器是该操作的瓶颈。此瓶颈会有效地串行化所有的文件访问。因此，编译进程的加速  
可能无法令人满意。  
qmake 的其他可能的应用则较为适当。示例之一便是通过 makefile 控制交互从属性和  
复杂的分析任务的工作流程。在这种环境中，每个 make 步骤都是典型的模拟或数据分  
析操作，该操作具有硬性的资源和计算时间要求。这种情况下，可以显著提高速度。  
qmake 用法  
qmake 的命令行语法与 qrsh 的语法类似。  
% qmake [-pe *pe-name pe-range*][*options*] \  
-- [*gnu-make-options*][*target*]  
**72** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** qmake 命令也支持 -inherit 选项，后文有相关介绍。  
请特别注意 -pe 选项的用法，以及它与 gmake -j 选项的关系。两个选项都可以用来  
表述要达到的并行化数量。不同之处在于 gmake 的 -j 选项无法指定某些要求，如要使  
用的并行环境。因此， qmake 假定并行 makes 的默认环境为 make。而且， gmake 的  
-j 选项可以不指定范围，而只指定一个数字。 qmake 将 -j 具有的数字解释为 1-*n* 的  
范围。与此相对， -pe 允许详细指定所有参数。因此，以下两个命令行示例内容相同：  
% qmake -- -j 10  
% qmake -pe make 1-10 --  
下面的命令行不能用 -j 选项表述：  
% qmake -pe make 5-10,16 --  
% qmake -pe mpi 1-99999 --  
除语法外， qmake 支持两种调用模式：不使用 -inherit 选项，从命令行交互式进行  
；或使用 -inherit 选项，在批处理作业中进行。这两种模式启动不同的操作顺序：  
■ 交互式 – 当在命令行调用 qmake 时， make 进程通过使用 qrsh 被隐含地提交到  
Grid Engine 系统。此进程将 qmake 命令行指定的资源需求纳入考虑范围。 Grid  
Engine 系统随后选择一个主控主机执行与并行 make 作业相关联的并行作业。 Grid  
Engine 系统启动该处的 make 过程。由于 make 进程和系统的体系结构之间的从属  
性， make 过程必须从该主控主机启动。 qmake 命令行中指定了所需的体系结构。主  
控主机上的 qmake 进程将单个 make 步骤 的执行委托给分配有该作业的其他主机。  
这些步骤通过并行环境主机文件传递到 qmake 。  
■ 批处理 – 在这种情况下， qmake 及其 -inherit 选项位于批处理脚本中。正如前面  
第一种情况中所述，如果未指定 -inherit 选项，则派生一个新作业。这将导致  
qmake 使用已经分配给嵌入了 qmake 的作业的资源。 qmake 使用 qrsh -inherit  
直接开始 make 步骤。当在批处理模式下调用 qmake 时，资源需求的指定，即 -pe  
和 -j 选项将忽略。  
注 **–** 单个 CPU 作业也必须请求并行环境。  
qmake -pe make 1 --  
如果未要求并行执行，则使用不带 Grid Engine 系统选项和 -- 的 gmake 命令行语  
法调用 qmake 命令。此 qmake 命令的操作与 gmake 一样。  
有关详细信息，请参阅 qmake(1) 手册页。  
第 3 章 • 提交作业 **73**  
如何调度作业  
Grid Engine 软件的策略管理自动控制群集中共享资源的使用，从而最好地达到管理的  
目标。首先分派高优先级的作业。这样的作业具有更高的资源访问权限。群集的管理人  
员可定义高级使用策略。可使用下面的策略：  
■ 职能策略 – 根据与特定用户组、项目等的从属关系而给予特殊处理。  
■ 基于份额的策略 – 服务级别取决于指定的份额权益、其他用户和其他用户组的相应份  
额、所有用户以及系统中当前存在的用户在此之前的资源使用情况。  
■ 紧急程度策略 – 优先处理紧急程度较高的作业。作业的紧急程度取决于它的资源需  
求、必须等待的时间以及提交时是否带有限期。  
■ 越权策略 – 群集管理员手动介入修改自动策略的执行。  
可定期设置 Grid Engine 软件使用基于份额的策略或职能策略或两者都使用。这些策略  
可以按任何比例组合使用，从一种策略权重为零从而只使用另一种策略，到给予两种策  
略相同的权重。  
使用常规的策略，作业可以按初始限期提交。有关限期提交参数的介绍，请参阅第  
57 页中的 “用 QMON 提交高级作业” 。限期作业会打乱常规的作业调度。管理人员也可  
以临时越权基于份额和职能的调度。越权既可应用于单个作业，也可应用于与用户、部  
门或项目相关联的所有作业。  
作业的优先级  
除使用四种策略仲裁所有作业外， Grid Engine 软件有时还允许用户在自己的作业中设  
置优先级。例如，提交多项作业的用户可以指定作业 3 的重要性最高，作业 1 和作业 2  
同等重要，但低于作业 3 的重要性。  
可以使用 “QMON Submit Job” 的 “Priority” 参数、或使用带有 -p 选项的 qsub 设置作  
业的优先级。优先级的范围从 -1024（最低）到 1023（最高）。可根据优先级对单个用  
户的作业排队。当系统中同时存在一个用户的多个作业时，调度程序根据这些作业的优  
先级进行选择。为特定作业分配的相对重要性取决于为该用户的作业指定的最高和最低  
优先级以及该特定作业的优先级值。  
票券策略  
职能策略、基于份额的策略以及越权策略均通过票券来执行。每个票券策略都具有一个  
票券池，票券从这里分配给进入多计算机 Grid Engine 系统的作业。每个使用中的常规  
票券策略分配给每个新作业一些票券。在每次调度时间间隔内，票券策略可为执行作业  
重新分配票券 。本节将介绍各票券策略分配票券的标准。  
票券权衡三种策略。例如，如果没有为职能策略分配票券，则不使用职能策略。如果为  
职能策略池和基于份额的策略池分配相同的票券数，这两种策略在确定作业重要性时就  
具有相同的权重。  
**74** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
Grid Engine 管理人员在进行系统配置时为常规策略分配票券。管理人员和操作员可在  
任意时间更改票券的分配。额外的票券会暂时注入系统以表明越权。票券策略通过票券  
分配进行组合：为多个票券策略分配票券后，作业从使用的每个票券策略中得到一部分  
票券。  
Grid Engine 系统赋予进入系统的作业票券数，以表明作业在使用的每个票券策略下的  
重要性。运行中的作业可能获得票券（如从越权策略）；失去票券（如作业得到的票券  
数高于其应得的票券）；或者在每次调度时间间隔内保持同样的票券数。 作业持有的票  
券数表示 Grid Engine 系统在每次调度时间间隔内授予该作业的资源份额。  
可以使用 QMON 或 qstat -ext 显示作业持有的票券数。请参阅第 77 页中的 “用  
QMON 监视和控制作业” 。 qstat 命令还显示（如使用 qsub -p）指定给作业的优先级  
值。有关详细信息，请参阅 qstat(1) 手册页。  
队列选择  
如果作业不能立即启动，则 Grid Engine 系统不分派请求非特定队列的作业。这样的作  
业在 sge\_qmaster 中标记为假脱机，并不时地尝试重新调度。因此，这样的作业将分  
派到下一个可用的队列中。  
与假脱机作业相反，按名称提交给某个队列的作业直接进入具有该名称的队列，无论该  
作业是可以启动还是需要假脱机。因此，将 Grid Engine 系统的队列看作计算机科学批  
处理队列只对以名称方式请求的作业有效。提交非特定请求的作业使用 sge\_qmaster  
的假脱机机制排入队列，因此使用比较抽象和灵活的队列概念。  
如果调度作业时有多个闲置的队列满足其资源请求，通常将该作业分派到负荷最轻的主  
机内的适当队列。通过将调度程序的配置项 queue\_sort\_method 设置为 seq\_no，  
群集管理人员可以将取决于负荷的分配方案更改为固定顺序的运算法则。队列配置项  
seq\_no 定义队列的先后顺序，将最高优先级的作业分配给具有最低序列号的队列。  
第 3 章 • 提交作业 **75**  
**76** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
第 **4** 章  
监视和控制作业与队列  
提交作业后，还需对它们进行监视和控制。本章提供了有关监视和控制作业与队列的背  
景信息以及如何完成这些任务的说明。本章还包含点校验作业的信息。  
本章包含用于以下任务的指导：  
■ 第 77 页中的 “用 QMON 监视和控制作业”  
■ 第 86 页中的 “用 qstat 监视作业”  
■ 第 88 页中的 “用 qdel 和 qmod 控制作业”  
■ 第 89 页中的 “用 QMON 监视和控制队列”  
■ 第 96 页中的 “用 qmod 控制队列”  
■ 第 99 页中的 “从命令行提交、监视或删除点校验作业”  
■ 第 99 页中的 “用 QMON 提交点校验作业”  
监视和控制作业  
可以通过三种方式监视和控制作业：  
■ 通过 QMON  
■ 从命令行使用 qstat、 qdel 和 qmod 命令  
■ 通过电子邮件  
以下各节对每一种方式进行了介绍。  
用 QMON 监视和控制作业  
QMON 提供专门为控制作业设计的 “Job Control” 对话框。  
要监视和控制提交的作业，请在 “QMON Main Control” 窗口中单击 “Job Control” 按  
钮。显示 “Job Control” 对话框。  
**77**  
“Job Control ” 对话框有三个选项卡： “Running Jobs” 选项卡、 “Pending Jobs” 选项卡  
（ 等待分派到适当资源）和 “Finished Jobs” 选项卡（最近完成的作业）。  
“Submit” 按钮链接到 “Submit Job” 对话框。  
“Job Control” 对话框可使您监视所有系统可识别的运行、暂挂和已完成的作业。还可以  
使用该对话框管理作业。可以更改作业的优先级。还可以暂停、恢复或取消作业。  
在默认格式下， “Job Control” 对话框显示每个运行和暂挂作业的以下列：  
■ JobId  
■ Priority  
■ JobName  
■ Owner  
■ Status  
■ Queue  
通过定制格式可以更改默认显示。有关详细信息，请参阅第 82 页中的 “定制 “Job  
Control” 显示” 。  
**78** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
刷新 “Job Control” 显示  
为保持显示最新信息， QMON 使用一种巡回检测方案从 sge\_qmaster 检索作业的状  
态。 单击 “Refresh” 可强制更新 “Job Control” 的显示。  
选择作业  
使用以下鼠标和按键的组合，选择作业：  
■ 要选择多个不连续的作业，请按住 “Control” 键然后单击两个或更多的作业。  
■ 要选择连续范围的作业，请按住 “Shift” 键然后单击范围中的第一项作业，再单击范  
围中最后一项作业。  
■ 要在选择作业和清除选择之间切换，请按住 “Control” 键同时单击作业。  
还可以使用筛选器选择希望显示的作业。有关详细信息，请参阅第 83 页中的 “筛选  
“Job List”” 。  
管理作业  
可以使用对话框右边的按钮按以下的方式管理选择的作业：  
■ 暂停  
■ 恢复（取消暂停）  
■ 删除  
■ 取消  
■ 发行  
■ 重新定义优先级  
■ 重新调度  
■ 用 qalter 进行修改  
只有作业的拥有者或 Grid Engine 管理人员与操作员可以暂停、恢复、删除、取消作  
业，修改作业优先级和修改作业。请参阅第 33 页中的 “管理人员、操作员和拥有者  
” 。只能暂停或恢复正在运行的作业。只能对暂挂的作业进行重新调度、取消和修改优  
先级以及其他属性。  
暂停作业时，会使用 UNIX 的 kill 命令将信号 SIGSTOP 发送到作业的进程组。  
SIGSTOP 停止作业使其不再使用 CPU 时间。恢复作业时会发送信号 SIGCONT，从而  
取消暂停作业。有关信号处理的更多信息，请参阅 kill(1) 手册页。  
注 **–** 可以强制暂停、恢复和删除作业。即可以在不通知控制作业的 sge\_execd 的情况  
下，使用 sge\_qmaster 注册这些操作。相应的 sge\_execd 无法访问（例如，由于网  
络问题）时，可使用强制方法。选择 “Force” 复选框以达到此目的。  
单击 “Reschedule” 重新调度当前运行的作业。  
第 4 章 • 监视和控制作业与队列 **79**  
使作业处于留置状态  
选中一项暂挂作业并单击 “Hold”，将显示 “Set Hold” 对话框。  
“Set Hold” 对话框可以设置和重新设置用户、操作员和系统的留置。用户留置可以由作  
业拥有者或 Grid Engine 管理人员与操作员设置或重新设置。操作员留置由管理人员或  
操作员设置或重新设置。系统留置只能由管理人员设置或重新设置。只要作业指定了留  
置，作业就不具备运行的条件。另外的设置或重新设置留置的方法是使用 qalter、  
qhold 和 qrls命令 。  
留置阵列作业任务  
“Set Hold” 对话框中的 “Tasks” 字段适用于阵列作业。使用此按钮留置阵列作业的具体  
子任务。注意 “Tasks” 字段中的文本格式。在该字段指定的任务 ID 范围可以是一个单  
独的数字、 *n-m* 形式的简单范围或者是带有步骤大小的范围。例如，任务 ID 范围可由  
2-10:2 指定， 这表示任务 ID 索引 2、 4、 6、 8 和 10。该范围通过环境变量  
SGE\_TASK\_ID（包含 5 个索引编号之一）表示这 5 个任务的全部。有关作业留置的详  
细信息，请参阅 qsub(1) 手册页。  
更改作业优先级  
单击 “Job Control” 对话框中的 “Priority”，显示以下的对话框：  
可在该对话框中为选中的暂挂或运行作业键入新的优先级。根据该优先级排列单个用户  
的作业。当一个用户同时具有多个作业时，调度程序根据优先级在该用户的作业中进行  
选择。  
**80** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
选中一个暂挂作业并单击 “Qalter” 时，将显示 “Submit Job” 窗口。对话框中的所有条  
目都根据提交作业时定义的属性进行设置。不可更改的条目是灰色的。其他条目都可以  
编辑。单击 “Submit Job” 对话框中的 “Qalter” 时，更改便注册到 Grid Engine 系统  
中。 “Qalter” 按钮是 “Submit” 按钮的替代按钮。  
验证作业的一致性  
验证标志用于 “Qalter” 模式时， “Submit Job” 对话框中的 “Verify” 标志具有特殊的含  
义。可以检查暂挂作业的一致性，了解作业尚未调度的原因。为 “Verify” 标志选择所期  
望的一致性检查模式，然后单击 “Qalter”。系统根据选择的检查模式显示不一致警告。  
有关更多信息，请参阅第 57 页中的 “用 QMON 提交高级作业” 和 qalter (1) 手册页  
中的 -w 选项。  
使用 “Why?” 按钮获取有关暂挂作业的信息  
另一种方法也可检查作业处于暂挂状态的原因，选中作业然后单击 “Job Control” 对话  
框中的 “Why?”。 该操作将打开 “Object Browser ” 对话框。该对话框显示阻止调度程  
序在最近一次调度中分派该作业的原因列表。下图显示了该信息的 “Browser” 窗口示例  
： 注  
**–** 仅当调度程序配置参数 schedd\_job\_info 设置为 true 时， “Why?”按钮才生成  
有意义的输出。请参阅 sched\_conf (5) 手册页。显示的调度信息与最近的一次调度时  
间间隔有关。调查作业尚未调度的原因时，该信息可能并不准确。  
清除错误状态  
单击 “Clear Error”，从暂挂作业中删除由于从属性原因而未被调度的暂挂作业的错误状  
态。例如，作业可能没有足够的权限写入到指定的作业输出文件。  
第 4 章 • 监视和控制作业与队列 **81**  
错误状态信息用红色字体显示在暂挂作业列表中。只有纠正错误情况之后才可以删除文  
件，例如使用 qalter。如果请求在作业中止时发送电子邮件，该错误情况就会通过电  
子邮件自动报告。例如，作业可能使用 qsub -m a 命令中止。  
定制 “Job Control” 显示  
要定制默认的 “Job Control” 显示，请单击 “Customize”。显示 “Job Customize” 对话  
框。单击 “Select Job Fields” 选项卡。 “Select Job Fields” 选项卡如下图所示：  
使用 “Job Customize” 对话框以配置要显示的信息集。  
使用 “Job Customize” 对话框，可以选择要显示的作业对象的多个条目。可以筛选您感  
兴趣的作业。在前图的示例中，选择了附加的字段 “Projects”、 “Tickets” 和 “Submit  
Time”。  
下图显示了将定制应用到 “Finished Jobs” 列表后的改进的外观。  
**82** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 使用 “Customize Job” 对话框中的 Save 按钮将定制信息存储到  
.qmon\_preferences 文件。该文件位于用户的主目录下。通过保存定制信息，您可以  
重新定义 “Job Control” 对话框的外观。  
筛选 “Job List”  
以下筛选工具示例仅选择 aa114085 所拥有的并且适于在 solaris64 体系结构上运行  
的作业。  
第 4 章 • 监视和控制作业与队列 **83**  
下图显示了 “Job Control” 对话框中的结果 “Running Jobs” 选项卡。  
**84** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
前面图中显示的 “Job Control” 对话框也是 QMON 如何显示阵列作业的示例。  
使用 “QMON Object Browser” 获取关于作业的附加信息。  
在没有定制 “Job Control” 的情况下，可以使用 “QMON Object Browser” 快速检索作业  
的附加信息，这在第 77 页中的 “用 QMON 监视和控制作业” 中进行了介绍。  
可使用有两种方法打开显示作业信息的 “Object Browser”：  
■ 单击 “QMON Main Control” 窗口中的 “Browser” 按钮，然后单击 “Browser” 对话框  
中的 “Job”。  
■ 将指针移到 “Job Control” 对话框中的作业上。  
以下 “Browser” 窗口显示了所显示的作业信息示例：  
第 4 章 • 监视和控制作业与队列 **85**  
从命令行监视和控制作业  
本节介绍了在命令行如何使用 qstat、 qdel 和 qmod 命令监视、删除和修改作业。  
用 qstat 监视作业  
要监视作业，请根据后面各节的内容，键入以下命令之一：  
qstat  
qstat -f  
qstat -ext  
不带选项的 qstat 仅提供已提交作业的概述。 qstat -f 还包含当前配置的队列的信  
息。 qstat -ext 包括分配给作业的最新作业用法和票券的详细情况。  
第一种形式中，标题行表示列的含义。可从字面上了解到大多数列的含义。但是，  
state 列包含单个字符代码，含义如下： r 表示运行、 s 表示已暂停、 q 表示已排队、  
w 表示等待。有关 qstat 输出格式的详细介绍，请参阅 qstat (1) 手册页。  
第二种形式分为两部分：第一部分显示所有可用队列的状态。第二部分名为  
PENDING JOBS，显示 sge\_qmaster 作业假脱机区域的状态。队列部分的第一行定义  
所列出的相应队列的列含义。队列用水平线分隔。队列中运行的作业会打印在相关队列  
下，其格式和 qstat 命令的第一种形式相同。在输出的第二部分，暂挂作业也以相同  
于 qstat 第一种形式的格式打印。  
队列说明的以下列需要较多的解释。  
■ qtype – 队列类型。队列类型为 B（批处理）或 I（交互式）。  
■ used/free – 队列中已用的和闲置的作业位置数。  
**86** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ states – 队列的状态。有关队列状态的详细信息，请参阅 qstat(1) 手册页。  
qstat(1) 手册页包含 qstat 输出格式的详细介绍。  
在第三种形式中，分配给作业的用量和票券值包含在以下列中：  
■ cpu/mem/io – 当前累积的 CPU、内存和 I/O 用量。  
■ tckts/ovrts/otckt/ftckt/stckt – 这些值如下：  
tckts – 分配给作业的票券总数  
ovrts – 通过 qalter -ot 分配的越权票券  
otckt – 通过越权策略分配的票券  
ftckt – 通过职能策略分配的票券  
stckt – 通过基于份额的策略分配的票券  
另外，如果适用，则限期初始时间显示在 “deadline” 列中。 “share” 列显示每个作业的  
当前资源份额（与群集中所有作业产生的用量相比）。有关更多信息，请参阅 qstat(1)  
手册页。  
qstat 命令的各种其他选项增强了该功能。使用 -r 选项显示已提交作业的资源需求。  
而且，输出可以限制到特定的用户或特定的队列。可以使用 -l 选项指定资源需求，  
qsub 命令在第 61 页中的 “定义资源需求” 中进行了介绍。使用资源需求时，仅显示与  
qstat 指定的资源需求相匹配的队列以及在这些队列中运行的作业。  
注 **–** 已增强 qstat 命令，以便管理员和使用者能够定义包含有用选项的文件（请参阅  
sge\_qstat(5)）。群集的 sge\_qstat 文件一般位于  
$xxQS\_NAME\_Sxx\_ROOT/$xxQS\_NAME\_Sxx\_CELL/common/sge\_qstat，用户的  
专用文件位于 $HOME/.sge\_qstat。主目录请求文件具有最高优先级别，其次是群集  
全局文件。可以使用命令行取代文件中包含的标志。  
示例 4–2 和示例 4–1 显示了qstat 和 qstat -f 命令的输出示例。  
示例 **4–1** qstat -f 输出示例  
queuename qtype used/free load\_avg arch states  
dq BIP 0/1 99.99 sun4 au  
durin.q BIP 2/2 0.36 sun4  
231 0 hydra craig r 07/13/96 20:27:15 MASTER  
232 0 compile penny r 07/13/96 20:30:40 MASTER  
dwain.q BIP 3/3 0.36 sun4  
230 0 blackhole don r 07/13/96 20:26:10 MASTER  
233 0 mac elaine r 07/13/96 20:30:40 MASTER  
234 0 golf shannon r 07/13/96 20:31:44 MASTER  
fq BIP 0/3 0.36 sun4  
第 4 章 • 监视和控制作业与队列 **87**  
示例 **4–1** qstat -f 输出示例 （续）  
################################################################################  
- PENDING JOBS - PENDING JOBS - PENDING JOBS - PENDING JOBS - PENDING JOBS -  
################################################################################  
236 5 word elaine qw 07/13/96 20:32:07  
235 0 andrun penny qw 07/13/96 20:31:43  
示例 **4–2** qstat 输出示例  
job-ID prior name user state submit/start at queue function  
231 0 hydra craig r 07/13/96 durin.q MASTER  
20:27:15  
232 0 compile penny r 07/13/96 durin.q MASTER  
20:30:40  
230 0 blackhole don r 07/13/96 dwain.q MASTER  
20:26:10  
233 0 mac elaine r 07/13/96 dwain.q MASTER  
20:30:40  
234 0 golf shannon r 07/13/96 dwain.q MASTER  
20:31:44  
236 5 word elaine qw 07/13/96  
20:32:07  
235 0 andrun penny qw 07/13/96 20:31:43  
用 qdel 和 qmod 控制作业  
要从命令行控制作业，请键入带有适当自变量的以下命令之一：  
% qdel *arguments*  
% qmod *arguments*  
无论作业是在运行还是假脱机，请使用 qdel 命令取消作业。使用 qmod 命令暂停或恢  
复（取消暂停）已经运行的作业。  
对于这两个命令，您都需要了解作业的标识号，成功执行 qsub 命令后会显示该编号。  
如果忘记该编号，请使用 qstat 命令进行检索。请参阅第 86 页中的 “用 qstat 监视  
作业” 。  
此处是 qdel 和 qmod 命令的几个示例：  
% qdel *job-id*  
% qdel -f *job-id1, job-id2*  
% qmod -s *job-id*  
% qmod -us -f *job-id1, job-id2*  
% qmod -s *job-id.task-id-range*  
必须是作业的拥有者或 Grid Engine 管理人员或操作员才能删除、暂停或恢复作业。请  
参阅第 33 页中的 “管理人员、操作员和拥有者” 。  
**88** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
可以对这两个命令使用 -f（强制）选项将作业状态的更改注册到 sge\_qmaster，而不  
必使用 sge\_execd。例如，由于网络问题而无法访问 sge\_execd 时，可以使用该强  
制选项。只有管理员才能使用 -f 选项。但是，对于 qdel，如果在群集配置  
qmaster\_params 条目中设置了 ENABLE\_FORCED\_QDEL 标志，则用户可以强制删除  
他们的作业。有关更多信息，请参阅 sge\_conf(5) 手册页。  
通过电子邮件监视作业  
在命令行键入带有适当自变量的以下命令：  
% qsub *arguments*  
qsub -m 命令要求在特定事件发生时为提交作业的用户发送电子邮件或将电子邮件发送  
到 -M 标志中指定的地址。有关该标志的介绍，请参阅 qsub(1) 手册页。 -m 选项的自变  
量指定这样的事件。以下是可用的自变量：  
■ b – 在作业开始时发送电子邮件。  
■ e – 在作业结束时发送电子邮件。  
■ a – 作业重新调度或中止时（例如，使用 qdel 命令）发送电子邮件。  
■ s – 作业暂停时发送电子邮件。  
■ n – 不发送电子邮件。 n 是默认选项。  
使用由一个或多个字母自变量组成的字符串为 -m 选项指定多个选项。例如， -m be 指  
定在作业开始和结束时发送电子邮件。  
还可以使用 “Submit Job” 对话框配置这些电子邮件事件。请参阅第 57 页中的 “用  
QMON 提交高级作业” 。  
监视和控制队列  
如第 34 页中的 “显示队列和队列属性” 中所述，队列的拥有者有权暂停或恢复队列、  
禁用或启用队列。 如果有重要的工作需要特定的计算机，拥有者可能需要暂停或禁用队  
列，因为这些计算机会受到后台运行的作业的强烈影响。  
控制队列有两种方式：  
■ 使用 “QMON Queue Control” 对话框  
■ 使用 qmod 命令  
用 QMON 监视和控制队列  
在 “QMON Main Control” 窗口中，单击 “Queue Control” 按钮。显示 “Cluster Queues”  
对话框。  
第 4 章 • 监视和控制作业与队列 **89**  
监视和控制群集队列  
“Cluster Queue” 选项卡可提供为群集定义的所有群集队列的快速概览。 “Cluster  
Queue” 选项卡还提供了暂停和恢复群集队列、禁用或启用群集队列以及配置群集队列  
的方法。  
“Cluster Queue” 对话框中显示的信息会定期更新。单击 “Refresh” 可强制进行更新。  
**90** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
若要选择一个群集队列，请单击它。  
单击 “Delete”、 “Suspend”、 “Resume”、 “Disable” 或 “Enable” 在您选择的群集队列  
上执行相应的操作。暂停/恢复和禁用/启用操作需要相应的 sge\_execd 通知。如果无  
法进行通知，可以单击 “Force” 强制更改 sge\_qmaster 内部状态。例如，由于主机关  
闭而无法通知。  
执行暂停/恢复和禁用/启用的操作需要具有群集队列拥有者、 Grid Engine 管理人员和  
操作员权限。有关详细信息，请参阅第 33 页中的 “管理人员、操作员和拥有者” 。  
暂停的群集队列无法接受更多的作业。暂停队列中运行的作业也将暂停，如第 77 页中  
的 “用 QMON 监视和控制作业” 中所述。一旦该队列恢复，群集队列及其作业便取消暂  
停。  
注 **–** 如果暂停群集队列中的作业被明确暂停，队列恢复时该作业无法相应地恢复。必须  
明确地恢复该作业。  
禁用的群集队列是关闭的。不过，正在该队列中运行的作业可继续运行。禁用群集队列  
常用于“消耗”队列。群集队列启用后，将再次具备运行作业的条件。没有对当前运行  
的作业执行操作。  
在队列列表中错误状态用红色字体表示。单击 “Clear Error” 从队列中删除错误状态。  
单击 “Reschedule” 以重新调度所选择的群集队列中当前运行的作业。  
若要配置群集队列和队列实例，请在 “Cluster Queue” 对话框中单击 “Add” 或  
“Modify”。有关详细信息，请参阅《 N1 Grid Engine 6 管理指南》中的“用 QMON 配  
置队列”。  
单击 “Done” 关闭对话框。  
群集队列状态  
群集队列表格中的每一行代表一个群集队列。对于每一个群集队列，该表格列出以下信  
息：  
■ Cluster Queue – 群集队列的名称。  
■ Load – 所有群集队列主机的平均标准负荷的平均值。只考虑具有负荷值的主机。  
■ Used – 当前使用的作业位置数。  
■ Avail – 当前可用的作业位置数。  
■ Total – 总的作业位置数。  
■ aoACD – 至少处于以下状态之一的队列实例数量：  
■ a – 阈值警告  
■ o – 孤立的  
■ A – 暂停阈值警告  
第 4 章 • 监视和控制作业与队列 **91**  
■ C – 由日历暂停  
■ D – 由日历禁用  
■ cdsuE – 至少处于以下状态之一的队列实例数量：  
■ c – 配置模糊  
■ d – 禁用  
■ s – 暂停  
■ u – 未知  
■ E – 错误  
■ s – 处于暂停状态的队列实例数量。  
■ A – 当前已超过一个或多个暂停阈值的队列实例的数量。没有更多的作业  
■ S – 通过从属于另外一个队列而暂停的队列实例数量。  
■ C – 由 Grid Engine 系统日历自动暂停的队列实例的数量。  
■ u – 处于未知状态的队列实例的数量。  
■ a – 当前已超过一个或多个负荷阈值的队列实例的数量。  
■ d – 处于禁用状态的队列实例的数量。  
■ D – 由 Grid Engine 系统日历自动禁用的队列实例的数量。  
■ c – 配置模糊的队列实例的数量。  
■ o – 处于孤立状态的队列实例的数量。  
■ E – 处于错误状态的队列实例的数量。  
有关群集队列及其状态的完整信息，请参阅 qstat(1) 手册页。  
监视和控制作业实例  
“Cluster Instance” 选项卡提供了与所选群集队列相关联的所有队列实例的快速概述。  
“Cluster Instance” 选项卡还提供了暂停、恢复、禁用和启用队列实例的方法。  
**92** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
若要选择一个队列实例，请单击它。  
单击 “Suspend”、 “Resume”、 “Disable” 或 “Enable” 在您选择的队列实例上执行相应  
的操作。暂停/恢复和禁用/启用操作需要相应的 sge\_execd 通知。如果无法进行通知  
（例如，由于主机关闭），可以单击 “Force ” 强制更改 sge\_qmaster 内部状态。  
执行暂停/恢复和禁用/启用的操作要求具有队列拥有者、管理人员或操作员权限。请参  
阅第 33 页中的 “管理人员、操作员和拥有者” 。  
第 4 章 • 监视和控制作业与队列 **93**  
暂停队列实例无法执行更多的作业。正在暂停队列实例中运行的作业也将暂停，如第  
77 页中的 “用 QMON 监视和控制作业” 中所述。一旦队列实例恢复，队列实例机及其作  
业便取消暂停。  
注 **–** 如果暂停队列实例中的作业被明确地暂停，队列实例恢复时该作业不能相应地恢  
复。必须明确地恢复该作业。  
禁用的队列实例是关闭的。然而，正在该队列实例中执行的作业可继续运行。禁用队列  
实例常用于“消耗”队列实例。队列实例启用之后，便再次具备了运行作业的条件。没  
有对当前运行的作业执行操作。  
队列实例状态  
队列实例表格中的每行都代表一个队列实例。对于每个队列实例，此表格列出以下信息  
： ■  
Queue – 队列实例的名称。  
■ qtype – 队列实例的类型，可以为 B（批处理）、 I （交互式）或 P （并行）。  
■ used/total – 已用的作业位置数和总的作业位置数。  
■ load\_avg – 队列实例主机的平均负荷。  
■ arch – 队列实例主机的体系结构  
■ states – 队列实例的状态  
有关队列状态的列表，请参阅”第 91 页中的 “群集队列状态” 。有关队列实例及其状态  
的完整信息，请参阅 qstat(1) 手册页。  
显示队列实例属性  
要检索队列实例的当前属性、负荷信息、资源使用信息，请选择队列实例，然后单击  
“Load”。该信息还明确地包含具有该队列示例的计算机的信息。 显示以下窗口：  
**94** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
“Attribute” 列列出了附加到队列实例的全部属性，包含继承自主机或全局群集的属性。  
“Slot-Limits/Fixed Attributes” 列显示了定义为每个队列实例位置限制或固定资源属性  
的属性值。  
“Load(scaled)/Consumable” 列显示了有关报告和调节负荷参数的信息。该列还根据可  
使用资源设备显示了可用资源容量的信息。请参阅《 N1 Grid Engine 6 管理指南》中  
的“负荷参数”和《 N1 Grid Engine 6 管理指南》中的“可使用资源”。  
如果负荷属性配置为可使用资源，负荷报告和可使用容量相互间可以越权。显示作业分  
派运算法则中使用的两者的最小值。  
注 **–** 所显示的负荷和可傅资源值不考虑负荷调整，如第 25 页中的 “执行主机” 所述。  
筛选群集队列和队列实例  
“Customize” 按钮可以筛选您希望显示的群集队列和队列实例。  
下图只显示当前配置为模糊的队列实例（筛选的选择）。  
第 4 章 • 监视和控制作业与队列 **95**  
单击 “Queue Customize” 对话框中的 “Save” 将设置存储到主目录下的  
.qmon\_preferences 文件中，以便以后调用 QMON 时进行标准激活。  
用 qmod 控制队列  
可以使用 qmod 命令暂停或恢复队列。可以使用 qmod 禁用或启用队列。  
键入带有适当自变量的以下命令：  
% qmod *arguments*  
下面的命令为如何使用 qmod 的示例：  
% qmod -s *q-name*  
% qmod -us -f *q-name1, q-name2*  
% qmod -d *q-name*  
% qmod -e *q-name1, q-name2, q-name3*  
qmod -s 暂停队列。 qmod –us –f 恢复（取消暂停）两个队列。 qmod -d 禁用队列。  
qmod -e 启用三个队列。  
在相应的 sge\_execd 无法访问（例如，由于网络问题）时， -f 选项强制将状态更改注  
册到 sge\_qmaster。  
执行暂停、恢复队列以及禁用和启用队列时需要具有队列拥有者、管理人员或操作员权  
限。请参阅第 33 页中的 “管理人员、操作员和拥有者” 。  
**96** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 可以使用带有 crontab 或 at 作业的 qmod 命令。  
使用点校验作业  
本节将介绍两种点校验作业：  
■ 用户级别  
■ 内核级别  
用户级别点校验  
很多应用程序（尤其是使用大量 CPU 时间的应用程序）使用点校验和重启机制来提高  
容错性能。在运算法则的特定阶段，状态信息和处理数据的重要部分重复写入一个或多  
个文件。如果中止应用程序，重启文件可在以后的时间处理并重启。该文件可以达到与  
校验点以前一致的状态。因为用户通常需要将重启文件移至合适的位置，所以将这种点  
校验称为用户级别点校验。  
对于不具有集成的用户级别的点校验的应用程序，可使用点校验库作为替代。点校验库  
可以由硬件供应商或公共域提供。 University of Wisconsin 的 Condor 项目就是一个示  
例。通过重新将应用程序和这样的库链接，无需源代码更改便可为应用程序安装点校验  
机制。  
内核级别点校验  
有些操作系统在操作系统内核内部提供了点校验支持。对于此种情况，不需要准备应用  
程序，也不需要重新链接应用程序。内核级别点校验通常适用于单个处理，也适用于整  
个完整的进程结构。因此，相互依赖的进程层次可以在任何时间执行点校验和重新启  
动。通常用户命令和 C 库接口都可以用来启动点校验。  
Grid Engine 系统支持操作系统点校验（如果有）。有关当前支持的内核点校验工具的  
信息，请参阅《 N1 Grid Engine 6 软件发行说明》。  
第 4 章 • 监视和控制作业与队列 **97**  
迁移点校验作业  
因为其重新启动性能确保仅有少量已经完成的工作必须重复，所以点校验作业在任何时  
候都是间断的。该能力用于在 Grid Engine 系统中构建迁移和动态负荷平衡机制。如果  
请求，点校验作业可按照命令中止。作业迁移到 Grid Engine 系统中的其他计算机，这  
样就可以动态平衡群集负荷。点校验作业将根据以下原因中止或迁移：  
■ 通过执行 qmod 或 QMON 命令，明确暂停执行队列或作业。  
■ 由于超过队列的暂停阈值而自动暂停作业运行处的作业或队列。作业的点校验情形的  
指定包括暂停情况。有关更多信息，请参阅《 N1 Grid Engine 6 管理指南》中的“配  
置负荷和暂停阈值”和第 99 页中的 “从命令行提交、监视或删除点校验作业” 。  
正在迁移的作业将移回到 sge\_qmaster。随后作业将被分派到另外一个合适的队列  
（如果存在这样的队列）。在这种情况下， qstat 输出显示为 R 状态。  
编写点校验作业脚本  
内核级别点校验 shell 脚本与普通的 shell 脚本一样。  
用户级别的 shell 脚本与普通批处理脚本的唯一不同在于恰当处理重新启动的能力。在  
重新启动时，设置点校验作业的环境变量 RESTARTED。使用该变量可以跳过只需要在  
初始调用时执行的作业脚本部分。  
透明的点校验作业脚本如示例 4–3 所示。  
示例 **4–3** 点校验作业脚本示例  
#!/bin/sh  
#Force /bin/sh in Grid Engine  
#$ -S /bin/sh  
# Test if restarted/migrated  
if [ $RESTARTED = 0 ]; then  
# 0 = not restarted  
# Parts to be executed only during the first  
# start go in here  
set\_up\_grid  
fi  
# Start the checkpointing executable  
fem  
#End of scriptfile  
迁移用户级别的点校验作业时，该作业脚本从开始重新启动。用户负责将 shell 脚本的  
程序流定向到作业间断的位置。这样做可以跳过脚本中必须多次执行的行。  
**98** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 内核级别点校验作业可在任何时间中断。功能强大的 shell 脚本会在最近的点校验处  
重新启动。因此， RESTARTED 环境变量与内核级别点校验作业无关。  
从命令行提交、监视或删除点校验作业  
键入带有适当选项的以下命令：  
# qsub *options arguments*  
除 qsub -ckpt 和 qsub -c 命令外， 提交点校验作业的操作与普通批处理脚本相同。  
这两个命令请求点校验机制。这两个命令还定义必须为作业生成点校验的场合。  
-ckpt 选项采用一个自变量，该自变量就是要使用的点校验环境的名称。请参阅《 N1  
Grid Engine 6 管理指南》中的“配置点校验环境”。  
不需要 -c 选项。 -c 也采用一个自变量。使用 -c 选项越权在点校验环境配置中定义的  
when 参数。有关详细信息，请参阅 checkpoint(5) 手册页。  
-c 选项的自变量可以是以下任一单字母，也可是以下字母的组合。自变量也可以是一个  
时间值。  
■ n – 不执行检验点。 n 具有最高的优先级别。  
■ s – 只有主机上的 sge\_execd 关闭时才能生成校验点。  
■ m – 按照相应的队列配置中定义的最小 CPU 时间间隔生成校验点。请参阅  
queue\_conf(5) 手册页中的 min\_cpu\_interval 参数 。  
■ x – 作业暂停时生成校验点。  
■ interval – 按照给定的时间间隔生成校验点，但频繁程度不能超过  
min\_cpu\_interval 的定义。时间值必须按 *hh:mm:ss* 格式指定。该格式指定两位  
的小时、分钟和秒，之间以冒号分隔。  
监视校验作业与监视常规作业不同。点校验作业可以不时迁移。因此点校验作业无法绑  
定于单个队列。但是，其专有的作业标识编号和作业名称保持不变。  
删除点校验作业的操作与第 86 页中的 “从命令行监视和控制作业” 中所介绍操作相  
同。  
用 QMON 提交点校验作业  
按照第 57 页中的 “用 QMON 提交高级作业” 中的说明操作，同时注意以下信息。  
使用 QMON 提交点校验作业与提交普通批处理作业相同，只是需要另行指定适当的点校  
验环境。如第 57 页中的 “用 QMON 提交高级作业” 中所介绍， “Submit Job” 对话框提  
供了与作业相关联的点校验环境字段。与该字段相邻的按钮可打开以下的选择对话框。  
第 4 章 • 监视和控制作业与队列 **99**  
此处，可从可用的点校验对象列表中选择适当的点校验环境。有关您的站点所安装的点  
校验环境的属性信息，请向系统管理员咨询。 或者请参阅《 N1 Grid Engine 6 管理指  
南》中的“配置点校验环境”。  
点校验的文件系统要求  
当写入基于点校验库的用户级别点校验或内核级别点校验时，由进程或由要执行点校验  
的作业覆盖的虚拟内存的完整映像必须进行转储。为此，必须具有足够的可用磁盘空  
间。如果设置了点校验环境配置参数 ckpt\_dir，点校验信息就会转储到 ckpt\_dir 下  
的作业专用的位置。如果 ckpt\_dir 设置为 NONE， 则会使用点校验作业开始位置的目  
录。有关点校验环境配置的详细信息，请参阅 checkpoint(5) 手册页。  
注 **–** 如果 ckpt\_dir 设置为 NONE，您应使用 qsub -cwd 脚本启动点校验作业。  
为成功迁移并重新启动作业，点校验文件和重新启动文件必须对所有的计算机可视。文  
件的可视性是文件系统组织方式的附加要求。因此需要 NFS 或类似的文件系统。 请向  
群集管理员咨询以了解您的站点是否满足此要求。  
如果您的站点没有运行 NFS，您需要在 shell 脚本的起始处明确指定转移重新启动文  
件。例如，在用户级别点校验作业的情况下，可以使用 rcp 或 ftp。  
**100** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
第 **5** 章  
统计和报告  
本章包括以下主题：  
■ 第 101 页中的 “启动 Accounting and Reporting Console”  
■ 第 104 页中的 “创建和运行简单查询”  
■ 第 116 页中的 “创建和运行高级查询”  
■ 第 118 页中的 “高级查询的后期绑定”  
启动 Accounting and Reporting Console  
Accounting and Reporting Console 与 N1 Grid Engine 6 软件是分别安装的。有关安装  
过程的详细信息，请参阅《 N1 Grid Engine 6 安装指南》中的第 8 章 “安装 Accounting  
and Reporting Console”。另外，必须启用 Grid Engine 系统的搜集报告信息的功能。  
有关如何启用报告数据搜集的详细信息，请参阅 《 N1 Grid Engine 6 管理指南》中  
的“报告统计信息 (ARCo)”。  
▼ 如何启动 Accounting and Reporting Console  
**1.** 启动 **Web** 浏览器。  
**2.** 通过在浏览器中访问以下 **URL**，可连接到 **Sun Web Console**：  
在以下示例中， *hostname* 是已经安装了统计和报告软件的主机。  
https://*hostname*:6789  
**3.** 用您的 **UNIX** 帐户登录，然后选择 **“N1 Grid Engine 6 ARCo”**。  
步骤  
**101**  
**4.** 登录成功后，请选择 **N1 Grid Engine 6 ARCo** 应用程序。界面会重新定向到  
**“Overview”** 页面，该页面将显示预定义的 **ARCo** 查询列表。  
**102** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
注 **–** 可直接连接到 ARCo 应用程序的链接为  
https://hostname:6789/console/login/Login?  
redirect\_url=%22/reporting/arcomodule/Index%22)  
按下名为 “Result List” 的选项卡，将看到所有存储的 “Query Results”。单击  
“Query List” 将返回 “Query List” 导航页面。  
第 5 章 • 统计和报告 **103**  
创建和运行简单查询  
查询可定义您希望检索的数据集。可以创建简单查询，系统将生成相应的 SQL 查询字符  
串。或者，如果了解 SQL 并希望亲自编写查询，则可以创建高级查询。  
▼ 如何创建简单查询  
**1.** 请转到 **“Query List”** 并按 **“New Simple”** 按钮。  
以下屏幕包含三个显示一般信息（如查询类别和说明）的选项卡。此信息是可选的。  
您可以在 “Simple Query” 选项卡中定义查询，在 “View” 选项卡中定义查询结果的  
显示方式。  
步骤  
**104** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
单击 “Simple Query” 选项卡可以显示 “Query” 定义页面。该页面分为：  
■ Table/View 下拉菜单，用来选择预定义查询的数据库表或视图  
■ Field List，将您感兴趣的所有字段在一行中列出  
■ 用于为查询定义筛选条件的 “Filter List”  
■ 用于限制查询结果条目数量的 “Row Limit” 文本框。  
第 5 章 • 统计和报告 **105**  
下面简要描述了创建简单查询时所涉及的几个步骤。  
**2.** 从表格列表中选择一个表格。  
**3.** 定义要查看的字段。  
“Field Function” 介绍该字段的功能。以下表格列出了支持的 “Field Function” 值。  
功能 说明  
VALUE 使用字段的当前值  
SUM 将字段的值累加  
COUNT 字段值计数  
MIN 获取字段的最小值  
MAX 获取字段的最大值  
AVG 获取字段的平均值  
■ “Field Name” 是选中的表格中的一个字段。  
■ “User Defined Name” 允许结果显示有含义的名称。  
■ 如果需要，可使用 “Sort” 定义每个字段的排序顺序。  
**4.** （可选的）定义筛选器。  
在定义筛选器之前必须至少指定一个字段。  
**106** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ 除第一个筛选器外，所有筛选器都需要选择 “AND/OR”。这表示与上一个筛选  
器的逻辑关系。  
■ “Field Name” 是需要筛选的字段名称。如果字段具有用户定义的名称，该名称会  
在选择列表中显示。否则会显示系统生成的名称。  
■ “Condition” 字段指定用于对数据库中的值进行筛选的操作符。 以下表格列出了  
支持的操作符。  
筛选器 符号 说明 要求的数量  
Equal = 必须等于要求的值 1  
Not Equal <>, != 必须不等于要求的值 1  
Less Than < 必须小于要求的值 1  
Less Than or Equal <=, ≤必须小于或等于要求的值 1  
Greater Than version="1 必须大于要求的值 1  
Greater Than or  
Equal  
>=, ≥必须大于或等于要求的值 1  
空 值必须为空 0  
Not Null 值必须非空 0  
Between 值必须位于指定的范围内 2  
In 值必须等于指定列表中的某个元素 1 或更大  
Like 值必须包含给定的要求值 1  
“Requirement” 包含用于筛选查询返回值的值。 以下列表包含 “Requirement” 字段  
的一些示例。  
1 AND 100 用于 Between 的情形  
d% 用于 Like 的情形  
%d% 用于 Like 的情形  
%d%e% 用于 Like 的情形  
第 5 章 • 统计和报告 **107**  
Wert-1’, Wert-2’, ... , ’Wert-*n* 用于 In 的情形  
**5.** （可选的）限制数据集数量。  
如果希望限制数据集数量，请选择 “Limit Query To First” 复选框。然后键入希望返  
回的数据集数量。  
**6.** 单击 **“Save”** 保存查询。  
下图显示 “Save this Query As” 屏幕。使用 “Query Name” 字段为查询命名，然后  
单击 “Ok”。  
保存查询后，将返回到已修改的 “Simple Query” 屏幕。  
▼ 如何创建视图配置  
**1.** 要更改查询的视图配置，请选择 **“View”** 选项卡。  
为已保存的查询创建视图：  
■ 从 “Overview” 页面的上的 “Query List” 中选择该查询  
■ 按 “Edit” 按钮  
■ 单击 “View” 选项卡。  
屏幕上即会显示该查询的当前视图配置。  
**2.** 请说明您所希望的查看查询结果的方式。  
您可以向视图配置中添加三个不同的部分，以确定是否显示与该查询有关的其他信息  
以及这些信息的显示顺序。  
步骤  
**108** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
通过顶部的跳转链接，可以跳转到相应的部分（如果可用）。可能的部分有  
“Database Table”、 “Pivot Table” 和 “Graphic”。 “View Configuration” 部分始终可  
见并允许显示在 “Common” 选项卡中输入的查询说明、 “Filter List” 中的筛选条  
件，以及查询定义的 SQL 语句或用于高级查询的 SQL 选项卡的内容。  
按下 “Add Database”、 “Add Pivot”、 “Add Graphic” 将添加相应的部分。  
对有些查询，只有部分视图选项具有意义。例如，如果只有两个可选择的列， pivot  
就会失去意义。  
为 “Database Table” 添加并选择您要显示在 “Name” 下的列，并调整其 “Type” 和  
“Format”。列的添加顺序就是列的显示顺序。为该报告所做的选择不会影响应用到  
数据的筛选器。  
为 “Pivot Table” 添加 pivot 列、行和数据条目，然后选择列的 “Name”、 “Type” 和  
“Format”。要将某个条目转换为其他 pivot 类型，请在 “Pivot Type” 下选择该条  
目。  
第 5 章 • 统计和报告 **109**  
对于 Graphic 部分，可以将查询数据附加到其他图表类型。在 “Diagram Type” 下拉  
菜单中，您可以使用以下图表类型：  
■ 条形图，  
■ 条形图 (3D)  
■ 堆叠式条形图  
■ 堆叠式条形图 (3d)  
■ 饼图，饼图 3D  
■ 线形图  
■ 线形图堆叠线  
有三种不同的图表类型：  
■ 条形图  
■ 饼图  
■ 线形图  
条形图和饼图可以具有 3D 效果。可以使用堆叠方式绘制条形图和线形图（对 Y 轴上  
的值进行汇总）。  
**110** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
**3.** 单击 **“Save”** 或 **“Save As”** 可以将 **“View”** 配置保存为查询。  
**4.** 单击 **“Run”** 运行查询。  
▼ 定义图数据源的方法  
定义图数据源的方法有两种。  
**1.** 列数据源：将某列所有的值作为数据源添加进来。数据源的名称即为列标题  
**2.** 行数据源：所有列的值均定义该数据源。数据源的名称由标签列的值定义。数据源的  
值由值列定义。  
“Accounting per Department” 查询的结果分三列显示在表格中，分 别为： time、 department 和 cpu。  
步骤  
示例 **5–1**  
第 5 章 • 统计和报告 **111** 显示结果 - 选择配置  
要将结果显示为饼图，请选择以下配置： 显示结果 - 饼图  
结果将为多个饼图  
示例 **5–2**  
示例 **5–3**  
**112** N1 Grid Engine 6 用户指南 • 2005 年 7 月 所有部门的 Cpu、 Io 和 Mem 用量  
统计所有部门的 cpu、 io 和 mem 用量的查询： 显示结果 - 选择配置  
要将结果显示为条形图，请选择以下配置  
示例 **5–4**  
示例 **5–5**  
第 5 章 • 统计和报告 **113** 显示结果 - 条形图  
结果将为一张条形图，其中有三个条形，分别用来表示每个部门：  
示例 **5–6**  
**114** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
▼ 如何运行简单查询  
● 您可以运行刚创建的简单查询。也可以运行以前保存的简单查询。  
■ 若要运行刚创建的查询，请在 **“Simple Query”** 屏幕上单击 **“Run”**。  
■ 若要运行以前保存的查询，请从 **“Query List”** 屏幕中选择查询，然后单击  
**“Run”**。  
▼ 如何编辑简单查询  
**1.** 在 **”Query List“** 屏幕的列表中选择一个查询。  
**2.** 单击 **“Edit”**。  
将显示选定的 “Simple Query” 屏幕。  
**3.** 对 **“Simple Query”** 屏幕进行更改与创建简单查询一样简单，只需浏览选项卡并输入  
更改后的值即可。  
步骤  
步骤  
第 5 章 • 统计和报告 **115**  
**4.** 保存或运行更改的查询。  
创建和运行高级查询  
必须具有编写 SQL 查询的经历，才能使用 Accounting and Reporting Console 的这一  
功能。  
▼ 如何创建高级查询  
**1.** 在 **“Query List ”** 屏幕上单击 **“New Advanced Query”**。  
**2.** 在文本框中输入 **SQL** 查询。  
与可用于 “Simple Query” 的操作相比，这是唯一的不同点。  
**3.** 保存或运行查询。  
■ 要保存查询，请单击 **“Save”**。  
步骤  
**116** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ 要运行查询，请单击 **“Run”**。  
▼ 如何运行高级查询  
● 您可以运行刚刚创建的高级查询，也可以运行以前保存的高级查询。  
■ 要运行刚创建的查询，请单击 **“Advanced Query”** 屏幕中的 **“Run”**。  
■ 要运行以前保存过的查询，请在 **“Query List”** 屏幕中选择查询，然后单击  
**“Run”**。  
▼ 如何编辑高级查询  
**1.** 在 **”Query List“** 屏幕的列表中选择一个查询。  
**2.** 单击 **“Edit”**。  
屏幕上将显示已完成的 “Advanced Query”。  
**3.** 更改 **SQL** 查询。  
**4.** 保存或运行更改的查询。  
步骤  
步骤  
第 5 章 • 统计和报告 **117**  
■ 要保存更改的查询，请单击 **“Save”**。  
■ 要运行更改的查询，请单击 **“Run”**。  
高级查询的后期绑定  
高级查询中后期绑定的语法为：  
LATEBINDING{ <column>;<operator>;<default value> }  
<column> name if the latebinding  
<operator> a SQL operator (e.g. = < > in .. )  
<value> default value (e.g. ’localhost’ )  
示例 **5–7** 后期绑定示例  
select hostname from sge\_host where LATEBINDING{hostname, like, ’a%’}  
select hostname from sge\_host where LATEBINDING{hostname, in, (’localhost’, ’foo.bar’)}  
**118** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
第 **6** 章  
错误消息和错误诊断  
本章介绍了 Grid Engine 系统错误消息的收发过程，并提供了如何解决各种常见问题的  
技巧。  
■ 第 119 页中的 “软件如何检索错误报告”  
■ 第 125 页中的 “诊断问题”  
■ 第 126 页中的 “常见问题的错误诊断”  
软件如何检索错误报告  
Grid Engine 软件通过将错误消息记录到特定的文件，或发送邮件，或同时采用这两种  
方式来报告错误和警告。日志文件包括消息文件和作业的 STDERR 输出。  
作业一旦启动，作业脚本的标准错误 (STDERR) 输出即被重定向到一个文件。定向时使  
用默认的文件名和位置，或者可以使用 qsub 命令的特定选项来指定文件名和位置。有  
关详细信息，请参阅 Grid Engine 系统手册页。  
sge\_qmaster、 sge\_schedd 和 sge\_execds 均有单独的消息文件。这些文件具有相  
同的文件名： messages。 sge\_qmaster 日志文件驻留在主控主机的假脱机目录下。  
sge\_schedd 消息文件驻留在调度程序的假脱机目录下。执行守护进程的日志文件驻留  
在执行守护进程的假脱机目录下。有关假脱机目录的详细信息，请参阅《 N1 Grid  
Engine 6 安装指南》中的“根目录下的假脱机目录”。  
每条消息占有文件中的单独一行。每条消息由竖线分隔号 (|) 细分为五部分。  
每条消息都由以下部分组成：  
1. 第一个组成部分是消息的时间戳。  
2. 第二个组成部分指定生成消息的 Grid Engine 系统守护进程。  
3. 第三个组成部分是运行守护进程的主机的名称。  
4. 第四部分是消息类型。消息可具有如下类型之一：  
**119**  
■ N 表示 notice – 用于提供信息  
■ I 表示 info – 用于提供信息  
■ W 表示 warning  
■ E 表示 error – 检测到错误情况  
■ C 表示 critical – 能导致程序中止  
在群集配置中使用 loglevel 参数在全局范围内或本地范围内指定希望记录的消息  
类型。  
5. 第五个组成部分是消息文本。  
注 **–** 如果由于某些原因导致无法访问错误日志文件时， Grid Engine 系统会尝试在相  
应主机上的 /tmp/sge\_qmaster\_messages、 /tmp/sge\_schedd\_messages 或  
/tmp/sge\_execd\_messages 文件中记录错误消息。  
在某些情况下， Grid Engine 系统通过电子邮件将错误事件通知给用户、管理员或两  
者。 Grid Engine 系统发送的电子邮件不包含消息主体。消息正文都包含在邮件标题字  
段中。  
不同的错误或退出代码所表示的结果  
下表列出了不同的与作业相关的错误代码或退出代码所表示的结果。这些代码对于每种  
作业类型都有效。  
表 **6–1** 作业相关的错误或退出代码  
脚本**/**方法 退出或错误代码 结果  
作业脚本 0 成功  
99 重新列队  
Rest 成功：统计文件中的退出代码  
前导脚本/收尾脚本 0 成功  
99 重新列队  
Rest 队列错误状态，作业已重新排队  
以下表格列出了与并行环境 (PE) 配置相关的作业的退出代码或错误代码的结果。  
**120** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
表 **6–2** 并行环境相关的错误或退出代码  
脚本**/**方法 退出或错误代码 结果  
并行环境启动 0 成功  
Rest 队列设置为错误状态，作业已重新排队  
并行环境停止 0 成功  
Rest 队列设置为错误状态，作业未重新排队  
以下表格列出了与队列配置相关的作业的错误代码或退出代码的结果。 仅当覆盖了相应  
的方法后，这些代码才有效。  
表 **6–3** 队列相关的错误或退出代码  
脚本**/**方法 退出或错误代码 结果  
作业启动程序 0 成功  
Rest 成功，无其他特殊意义  
暂停 0 成功  
Rest 成功，无其他特殊意义  
恢复 0 成功  
Rest 成功，无其他特殊意义  
终止 0 成功  
Rest 成功，无其他特殊意义  
以下表格列出了与点校验相关的作业的错误代码或退出代码的结果。  
表 **6–4** 点校验相关的错误或退出代码  
脚本**/**方法 退出或错误代码 结果  
点校验 0 成功  
Rest 成功。然而对于内核校验点，这表示校验点不成功。  
第 6 章 • 错误消息和错误诊断 **121**  
表 **6–4** 点校验相关的错误或退出代码 （续）  
脚本**/**方法 退出或错误代码 结果  
移植 0 成功  
Rest 成功。然而对于内核校验点，这表示校验点不成功。将  
发生迁移。  
重新启动 0 成功  
Rest 成功，无其他特殊意义  
清除 0 成功  
Rest 成功，无其他特殊意义  
对于成功运行的作业， qacct -j 命令在 failed 字段显示值 0，输出结果在  
exit\_status 字段显示作业的退出状态。然而守护进程可能无法成功运行作业。例  
如，收尾脚本可能失败，或者守护进程无法启动作业。这种情况下， failed 字段显示  
下表中列出的代码值之一。  
表 **6–5** qacct -j failed 字段代码  
代码 说明 统计有效 对于作业的含义  
0 No failure t 作业正常运行、退出  
1 Presumably before job f 无法启动作业  
3 Before writing config f 无法启动作业  
4 Before writing PID f 无法启动作业  
5 On reading config file f 无法启动作业  
6 Setting processor set f 无法启动作业  
7 Before prolog f 无法启动作业  
8 In prolog f 无法启动作业  
9 Before pestart f 无法启动作业  
10 In pestart f 无法启动作业  
11 Before job f 无法启动作业  
12 Before pestop t 作业已运行，在调用 PE 停止过程之前失败  
13 In pestop t 作业已运行， PE 停止过程失败  
14 Before epilog t 作业已运行，在调用收尾脚本之前失败  
**122** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
表 **6–5** qacct -j failed 字段代码 （续）  
代码 说明 统计有效 对于作业的含义  
15 In epilog t 作业已运行，调用收尾脚本时失败  
16 Releasing processor set t 作业已运行，无法释放处理器设置  
24 Migrating （点校验作  
业）  
t 作业已运行，将迁移作业  
25 Rescheduling t 作业已运行，将重新调度作业  
26 Opening output file f 无法启动作业，无法打开 stderr/stdout 文件  
27 Searching requested  
shell  
f 无法启动作业， shell 未找到  
28 Changing to working  
directory  
f 无法启动作业，更改到启动目录时产生错误  
100 Assumedly after job t 作业已运行，作业由信号中止  
“代码”列列出了 failed 字段的值。“说明”列列出了出现在 qacct -j 输出中的文  
本。如果 acctvalid 设置为 t，则作业统计值有效。如果 acctvalid 设置为 f，则  
统计记录的资源用量值无效。“对于作业的含义”列表明作业是否运行。  
在调试模式下运行 Grid Engine 系统程序  
对于一些严重错误，错误日志机制可能无法输出足够的信息以识别问题。因此， Grid  
Engine 系统提供了一种在调试模式下几乎可运行所有辅助程序和守护进程的能力。不同  
的调试级别所提供信息的广度和深度都不尽相同。调试级别的范围是从 0 到 10，其中级  
别 10 可提供最详细的信息而级别 0 将关闭调试。  
要设置调试级别， Grid Engine 系统分发版本提供了 .cshrc 或 .profile 资源文件扩  
展。对于 csh 或 tcsh 用户，包含了文件 *sge-root*/util/dl.csh。对于 sh 或 ksh 用  
户，相应的文件名为 *sge-root*/util/dl.sh。必须将文件源放入标准资源文件中。对于  
csh 或 tcsh 用户， .cshrc 文件中包含以下一行：  
source *sge-root/*util/dl.csh  
作为 sh 或 ksh 用户，请在 .profile 文件中包含以下行：  
. *sge-root*/util/dl.sh  
注销并重新登录后，便可以使用以下命令设置调试级别：  
% dl *level*  
如果 *level* 大于 0，则启动 Grid Engine 系统命令可强制该命令将跟踪输出写入到  
STDOUT。跟踪输出可包含警告消息、状态消息、错误消息以及内部调用的程序模块的  
名称。消息还包含行数信息（有助于错误报告），这取决于所指定的调试级别。  
第 6 章 • 错误消息和错误诊断 **123**  
注 **–** 要查看调试跟踪，应使用带有大的滚动行缓冲区的窗口。例如，可使用 1000 行的滚  
动行缓冲区。  
注 **–** 如果窗口是 xterm，则可能希望使用 xterm 记录机制以后检查跟踪输出。  
如果以调试模式运行某个 Grid Engine 系统守护进程，守护进程将保持其终端连接以写  
入跟踪输出。可键入所使用的终端仿真程序的中断字符中止终端连接。例如，可使用  
Control-C。  
要关闭调试模式，请将调试级别设置为 0。  
设置 dbwriter 调试级别  
sgedbwriter 脚本启动 dbwriter 程序。该脚本的路径为  
*sge\_root*/dbwriter/bin/sgedbwriter。 sgedbwriter 脚本读取 dbwriter 配置  
文件 dbwriter.conf。该配置文件的路径为  
*sge\_root*/*cell*/common/dbwriter.conf。该配置文件设置 dbwriter 的调试级别。例  
如：  
#  
# Debug level  
# Valid values: WARNING, INFO, CONFIG, FINE, FINER, FINEST, ALL  
#  
DBWRITER\_DEBUG=INFO  
可使用 dbwriter 命令的 –debug 选项更改 dbwriter 所产生的消息数量。通常，应  
使用默认的调试级别，即 info。如果使用更详细的调试级别，则在很大程序上增加  
dbwriter 所产生的数据输出量。  
可以指定以下调试级别：  
warning 仅显示严重的错误和警告。  
info 添加许多信息消息。 “info” 是默认的调试级别。  
config 提供与 dbwriter 配置相关（例如，关于规则处理）的附加信息。  
fine 产生更多信息。如果选择此调试级别，则输出 dbwriter 所运行的所有  
SQL 语句。  
finer 用于调试。  
finest 用于调试。  
all 显示所有级别的信息。用于调试。  
**124** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
诊断问题  
Grid Engine 系统提供了几种帮助您诊断错误的方法。以下几节将概述它们的用法。  
未分派的暂挂作业  
有时，一个暂挂暂挂作业很明显是能够运行的，但是未分派该作业。为诊断其原因，  
Grid Engine 系统提供了一对实用程序和选项 qstat -j *job-id* 和 qalter-w v  
*job-id*。  
■ qstat -j *job-id*  
启用后， qstat -j *job-id* 将提供一个原因列表，解释最近一次调度中未分派某些作  
业的原因。这种监视是可启用或禁用的。可能希望禁用监视程序，因为它会在  
schedd 守护进程和 qmaster 之间产生您不希望的通信系统开销。请参阅  
sched\_conf(5) 手册页中的 schedd\_job\_info 。下例显示了 ID 为 242059 的作  
业的输出：  
% qstat -j 242059  
scheduling info: queue "fangorn.q" dropped because it is temporarily not available  
queue "lolek.q" dropped because it is temporarily not available  
queue "balrog.q" dropped because it is temporarily not available  
queue "saruman.q" dropped because it is full  
cannot run in queue "bilbur.q" because it is not contained in its hard queuelist (-q)  
cannot run in queue "dwain.q" because it is not contained in its hard queue list (-q)  
has no permission for host "ori"  
此信息直接由 schedd 守护进程生成。此信息的生成将考虑群集的当前用量。有时  
此信息并不能提供您所希望得到的信息。例如，如果其他用户的作业已占用了所有队  
列位置数，则无法为所需要的作业生成详细消息。  
■ qalter -w v *job-id*  
此命令可列出原则中作业不可分派的原因。为实现此目的，将执行一个不带优先级的调  
试运行操作。所有可用 的资源以及所有位置数都可用于此作业。类似的，将忽略所有负  
荷值，因为这些值是不同的。  
报告为错误状态 E 的作业或队列  
作业或队列错误由 qstat 输出中的大写 E 表示。  
由于特定作业的原因， Grid Engine 系统尝试运行一个作业失败时，该作业进入错误状  
态。  
第 6 章 • 错误消息和错误诊断 **125**  
由于特定队列的原因， Grid Engine 系统尝试运行一个作业失败时，该队列进入错误状  
态。  
Grid Engine 系统为用户和管理员提供了作业执行错误的一组可能原因，以便他们搜集  
诊断信息。队列和作业错误状态都由作业执行失败而引起。因此这些诊断方法适用于这  
两种错误状态类型。  
■ 用户中止邮件。 如果作业是用 qsub -m a 命令提交的，则将中止邮件发送到 -M  
*user*[@*host*] 选项所指定的地址。中止邮件包含关于作业错误的诊断信息。中止邮件  
是建议用户使用的信息源。  
■ **qacct** 统计。 如果没有可用的中止邮件，用户可以运行 qacct -j 命令。该命令从  
Grid Engine 系统的作业统计功能获取有关作业错误的信息。  
■ 管理员中止邮件。 管理员可通过指定适当的邮件地址预订有关作业执行问题的管理员  
邮件。请参阅 sge\_conf(5) 手册页的 administrator\_mail 下的信息。管理员邮  
件比用户中止邮件包含更详细的诊断信息。当作业执行错误频繁发生时，建议的方法  
是使用管理员邮件。  
■ 消息文件。 如果未收到管理员邮件，则应首先检查 qmaster messages 文件。通过  
搜索适当的作业 ID 可以查找到与特定作业相关的条目。在默认安装的情况下，  
qmaster messages 文件为 *sge-root*/*cell*/spool/qmaster/messages。  
有时可从作业启动处的 execd 守护进程中查找到附加的信息。使用 qacct -j *job-id*  
搜索启动作业的主机，并在 *sge-root*/*cell*/spool/*host*/messages 中搜索作业 ID。  
常见问题的错误诊断  
本节提供的信息有助于诊断常见问题的原因并采取措施。  
■ 问题 — 作业的输出文件显示 Warning: no access to tty; thus no job  
control in this shell...。  
■ 可能的原因 — 一个或多个登录文件中包含 stty 命令。只有存在终端时，这些命  
令才有用。  
■ 可能的解决方法 — 不要将终端与批量作业相关联。必须从登录文件中删除所有的  
stty 命令，或用 if 语句将这些命令包括起来。处理 if 语句前应该检查终端。  
下例显示了一个 if 语句：  
/bin/csh:  
stty -g # checks terminal status  
if ($status == 0) # succeeds if a  
terminal is present  
*<put all* stty *commands in here>*  
endif  
■ 问题 — 作业标准错误日志文件显示 ‘tty‘: Ambiguous。但是在作业脚本所调用  
的用户 shell 中不存在对 tty 的引用。  
**126** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
■ 可能的原因 — 默认情况下， shell\_start\_mode 是 posix\_compliant。因  
此，所有的作业脚本都使用在队列定义中指定的 shell 运行 。而不使用在作业脚  
本的第一行中指定的 shell 运行。  
■ 可能的解决方法 — 使用 qsub 命令的 -S 标志，或将 shell\_start\_mode 更改  
为 unix\_behavior。  
■ 问题 — 可以在命令行运行作业脚本，但是使用 qsub 命令运行作业脚本时失败。  
■ 可能的原因 — 可能为作业设置了进程限制。要测试是否测试了限制，请编写一个  
执行 limit 和 limit -h 功能的测试脚本。在 shell 提示符下交互式运行这两种  
功能，并使用 qsub 命令比较结果。  
■ 可能的解决方法 — 删除（在 shell 中设置限制的）配置文件中的所有命令。  
■ 问题 — 执行主机报告的负荷为 99.99。  
1. 可能的原因 — execd 守护进程未在此主机上运行。  
可能的解决方法 — 以 root 用户的身份运行  
$SGE\_ROOT/default/common/’rcsge’ 脚本以便在执行主机上启动 execd 守  
护进程。  
2. 可能的原因 — 未正确指定默认域。  
可能的解决方法 — 以 Grid Engine 系统管理员的身份运行 qconf -mconf 命  
令，并将 default\_domain 变量更改为 none。  
3. 可能的原因 — qmaster 主机所看到的执行主机名称与执行主机所看到的自己的  
名称不同。  
可能的解决方法 — 如果使用 DNS 解析计算机群集的主机名，则请配置  
/etc/hosts 和 NIS，使其作为主要主机名称返回全称域名 (FQDN)。当然还可  
以定义和使用较短的别名，例如 168.0.0.1 myhost.dom.com myhost。  
如果未 使用 DNS，请确保所有的 /etc/hosts 文件和 NIS 表格都是一致的，例  
如， 168.0.0.1 myhost.corp myhost或168.0.0.1 myhost。  
■ 问题 — 每过 30 秒，便有一条类似于以下消息的警告打印到  
*cell*/spool/*host*/messages：  
Tue Jan 23 21:20:46 2001|execd|meta|W|local  
configuration meta not defined - using global configuration  
但 *cell*/common/local\_conf 包含每一个带有 FQDN 的主机的文件。  
■ 可能的原因 — 计算机 meta 上的主机名解析返回的是短名称，但是主控主机上返  
回的是带有 FQDN 的 meta。  
■ 可能的解决方法 — 请确保所有 /etc/hosts 文件和 NIS 表在这方面是一致的。  
在此示例中，类似如下的文本行会错误地包含在主机 meta 的 /etc/hosts 文件  
中。  
168.0.0.1 meta meta.your.domain  
该行的正确内容应为：  
168.0.0.1 meta.your.domain meta.  
第 6 章 • 错误消息和错误诊断 **127**  
■ 问题 — 偶尔可在守护进程的 messages 文件中看到 CHECKSUM ERROR、 WRITE  
ERROR 或 READ ERROR 信息 。  
■ 可能的原因 — 只要这些消息出现的时间间隔大于一秒，就不需要采取任何措施。  
该信息典型的情况是每天出现 1 到 30 次。  
■ 问题 — 作业在特定的队列中完成并在 qmaster/messages 中返回以下消息：  
Wed Mar 28 10:57:15 2001|qmaster|masterhost|I|job 490.1  
finished on host exechost  
然后在执行主机的 exechost/messages 文件中可以看到以下错误消息：  
Wed Mar 28 10:57:15 2001|execd|exechost|E|can’t find directory  
"active\_jobs/490.1" for reaping job 490.1  
Wed Mar 28 10:57:15 2001|execd|exechost|E|can’t remove directory  
"active\_jobs/490.1": opendir(active\_jobs/490.1) failed:  
Input/output error  
■ 可能的原因 — 正在卸载已自动装入的 $SGE\_ROOT 目录，从而导致 sge\_execd  
守护进程丢失其当前的工作目录。  
■ 可能的解决方法 — 使用 execd 主机的本地假脱机目录。使用 qmon 或 qconf 命  
令设置参数 execd\_spool\_dir。  
■ 问题 — 当使用 qrsh 实用程序提交交互式作业时，会得到以下错误消息：  
% qrsh -l mem\_free=1G error: error: no suitable queues  
但对于使用 qsub 命令提交的批处理作业，队列是可用的。可使用 qhost -l  
mem\_free=1G 和 qstat -f -l mem\_free=1G 查询这些队列。  
■ 可能的原因 — 消息 error: no suitable queues 由 -w e 提交选项引起，默  
认情况下，该选项适用于交互式作业（例如 qrsh）。请参阅 qrsh(1) 手册页中  
的 -w e。如果 qmaster 无法确定地知道在当前的群集配置中该作业是可分派  
的，此选项就会导致提交命令的失败。该机制的目的在于提前拒绝作业的请求，  
以防该请求无法获得准许。  
■ 可能的解决方法 — 这种情况下， mem\_free 已配置为可使用的资源，但是尚未  
指定每台主机可用的内存数量。执行此检查时特意未考虑内存负荷值（该值不时  
变化）因此，内存负荷值无法视为群集配置的一部分。您可以执行以下之一：  
■ 全局忽略该检查通过使用 -w n 选项明确替代 qrsh 的默认选项 -w e。还可  
将此命令放到 *sge-root*/*cell*/common/cod\_request 下。  
■ 如果希望将 mem\_free 作为可使用的资源管理，请使用 qconf -me  
*hostname* 在 host\_conf 的 complex\_values 中指定主机的 mem\_free 容  
量。  
■ 如果不希望将 mem\_free 作为可使用的资源进行管理，请在 complex(5) 的  
consumable 列中使用 qconf -mc *hostname* 再次将其设为不可使用的资  
源。  
■ 问题 — qrsh 没有分派到其驻留的相同节点。从 qsh shell 得到类似如下内容的消息  
：  
**128** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
host2 [49]% qrsh -inherit host2 hostname  
error: executing task of job 1 failed:  
host2 [50]% qrsh -inherit host4 hostname  
host4  
■ 可能的原因 — gid\_range 不够。 gid\_range 应定义为一个范围，而不是单个  
数字。 Grid Engine 系统为主机上的每个作业分配一个独有的 gid。  
■ 可能的解决方法 — 用 qconf -mconf 命令或 QMON 调整 gid\_range。建议的  
范围如下：  
gid\_range 20000-20100  
■ 问题 — qrsh -inherit -V 在并行作业中不工作。得到以下消息：  
cannot get connection to "qlogin\_starter"  
■ 可能的原因 — 此问题发生在嵌套调用 qrsh 时。此问题由 -V 选项导致。首次  
qrsh -inherit 调用会设置环境变量 TASK\_ID。 TASK\_ID 是并行作业中高度  
集成任务的 ID 。第二次调用 qrsh -inherit 时使用该环境变量注册任务。当  
命令试图启动与已经运行的首次任务具有相同 ID 的任务时，该命令失败。  
■ 可能的解决方法 — 可在调用 qrsh -inherit 之前取消 TASK\_ID 的设置，或使  
用 -v 选项替代 -V。此选项只导出真正需要的环境变量。  
■ 问题 — qrsh 看起来根本不工作。生成类似如下内容的消息：  
host2$ qrsh -verbose hostname  
local configuration host2 not defined - using global configuration  
waiting for interactive job to be scheduled ...  
Your interactive job 88 has been successfully scheduled.  
Establishing /share/gridware/utilbin/solaris64/rsh session  
to host exehost ...  
rcmd: socket: Permission denied  
/share/gridware/utilbin/solaris64/rsh exited with exit code 1  
reading exit code from shepherd ...  
error: error waiting on socket for client to connect:  
Interrupted system call  
error: error reading return code of remote command  
cleaning up after abnormal exit of  
/share/gridware/utilbin/solaris64/rsh  
host2$  
■ 可能的原因 — 未正确设置 qrsh 的权限。  
■ 可能的解决方法 — 检查 $SGE\_ROOT/utilbin/ 目录中文件的权限。（注意，  
rlogin 和 rsh 必须为 setuid 并且由 root 用户拥有。）  
-r-s--x--x 1 root root 28856 Sep 18 06:00 rlogin\*  
-r-s--x--x 1 root root 19808 Sep 18 06:00 rsh\*  
-rwxr-xr-x 1 sgeadmin adm 128160 Sep 18 06:00 rshd\*  
第 6 章 • 错误消息和错误诊断 **129**  
注 **–** *sge-root* 目录也需要使用setuid 选项的 NFS 安装方式。如果 *sge-root* 从提交  
客户机采用 nosuid 安装， qrsh 及其相关联的命令将不工作。  
■ 问题 – 当您尝试启动分布式 make 操作时， qmake 退出并返回以下错误消息：  
qrsh\_starter: executing child process  
qmake failed: No such file or directory  
■ 可能的原因 — Grid Engine 系统在执行主机上启动了一个 qmake 实例。如果  
Grid Engine 系统环境（尤其是 PATH 变量）没有在用户的 shell 资源文件  
（ .profile 或 .cshrc）中设置，则该 qmake 调用失败。  
■ 可能的解决方法 — 使用 -v 选项将 PATH 环境变量导出到 qmake 作业。典型的  
qmake 调用如下：  
qmake -v PATH -cwd -pe make 2-10 --  
■ 问题 — 使用qmake 实用程序时，得到以下错误消息：  
waiting for interactive job to be scheduled ...timeout (4 s)  
expired while waiting on socket fd 5  
Your "qrsh" request could not be scheduled, try again later.  
■ 可能的原因 — 未正确设置 shell 的 ARCH 环境变量，而 qmake 是从该 shell 中调  
用的。  
■ 可能的解决方法 – 将 ARCH 变量正确地设置为与群集中可用主机相匹配的受支持  
的值，或在提交时指定正确的值（例如， qmake -v ARCH=solaris64  
...）。  
典型的 Accounting and Reporting Console 错误  
问题**:** Sun Web Console 2.0.3 版安装失败，并显示以下错误消息：  
# ./inst\_reporting  
...  
Register the N1 SGE reporting module in the webconsole  
Registering com.sun.grid.arco\_6u3.  
Starting Sun(TM) Web Console Version 2.0.3...  
Ambiguous output redirect.  
解决方法**:** 只有以 /bin/sh 作为其登录 shell 的 noacces 用户，才能安装该版本 Sun  
Web Console。必须使用以下命令添加用户：  
# useradd -u 60002 -g 60002 -d /tmp -s /bin/sh -c "No Access User" noaccess  
**130** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
问题**:** 简单查询定义的 “Table/View” 下拉菜单不包含任何条目，而是在数据库中定义这  
些表格。  
解决方法**:** 将 Oracle 用作数据库时，通常会发生此问题。在安装报告模块的过程中指定  
了错误的数据库模式名称。对于 Oracle，数据库模式名称与 dbwriter 所使用的数据库  
用户名相同（默认名称为 arco\_write）。对于 Postgres，数据库模式名称应为  
public。  
问题**:** “Connection refused” 消息。  
解决方法**:** smcwebserver 可能是关闭的。启动或重新启动 smcwebserver。  
问题**:** 查询列表或结果列表为空。  
解决方法**:** 原因可能是以下之一：  
■ 数据库已关闭。启动或重新启动数据库。  
■ 没有更多可用的数据库。增加允许连接到数据库的数量。  
■ 在应用程序的配置文件中存在一个错误。检查配置，查找错误的数据库用户、错误的  
用户密码或者错误的数据库类型，然后重新启动应用程序。  
■ 没有可用的查询。如果查询目录 /var/spool/arco/queries 不是空的，则可能  
发生以下错误：  
■ XML 文件中的查询在语法上是不正确的。在 XML 解析程序中检查错误消息的日  
志文件。  
■ noaccess 用户在查询目录中没有读或写权限。  
问题**:** 可使用数据库表的列表为空。  
解决方法**:** 原因可能是以下之一：  
■ 数据库已关闭。启动或重新启动数据库。  
■ 没有更多可用的数据库。增加允许连接到数据库的数量。  
■ 在应用程序的配置文件中存在一个错误。检查配置，查找错误的数据库用户、错误的  
用密码令或者错误的数据库类型，然后重新启动应用程序。  
问题**:** 可选字段的列表为空。  
解决方法**:** 为选择表格。从列表中选择一个表格。  
问题**:** 筛选器列表为空。  
解决方法**:** 没有选择的字段。请至少定义一个字段。  
问题**:** 排序列表为空。  
解决方法**:** 没有选择的字段。请至少定义一个字段。  
问题**:** 未使用已定义的筛选器。  
解决方法**:** 筛选器可能处于不活动状态。修改未使用的筛选器并使其处于活动状态。  
第 6 章 • 错误消息和错误诊断 **131**  
问题**:** 高级查询中最近的绑定已忽略，但执行遇到错误。  
解决方法**:** 最近的捆绑宏具有语法上的错误。高级查询的最新捆绑宏的正确语法是：  
latebinding{attribute;operator}  
latebinding{attribute;operator;defaultvalue}  
问题**:** breadcrumb 用于移回，但却显示登录屏幕：  
解决方法**:** 会话超时。再次登录，或者在 app.xml 中增加会话时间。  
问题**:** 已定义视图配置，但却显示默认配置。  
解决方法**:** 已定义的视图配置未设置为可视的。打开视图配置并定义要使用的视图配置。  
问题**:** 已定义视图配置，但却显示最近的配置。  
解决方法**:** 已定义的视图配置未设置为可视的。打开视图配置并定义要使用的视图配置。  
问题**:** 执行查询需要很长时间。  
解决方法**:** 巨大的数据库会导致该结果。 设置结果限制，或者扩展筛选器条件。  
**132** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
附录 **A**  
数据库模式  
本附录通过一系列的表格详细介绍了数据库的模式。包括以下主题：  
模式表格  
sge\_job  
在 sge\_job 表格中，每个阵列任务对应着一条记录（非阵列作业的阵列任务号均为  
1），每个从紧密集成的并行作业中启动的并行任务也对应着一条记录。  
对于 N1GE 6.0 系统，会在调度作业、阵列任务或并行任务时创建记录，并在作业执行  
过程中更新记录。  
关于 N1GE 作业、阵列作业、并行作业及其区别的简要说明，请参阅《 Sun Grid  
Engine 用户指南》。其中的词汇表将有助于您了解本软件。  
栏 类型 说明  
j\_id 整数 唯一的记录标识符  
j\_job\_number 整数 JOB\_ID  
j\_task\_number 整数 阵列任务 ID。  
j\_pe\_taskid 文本 紧密集成的并行任务的任务  
ID。  
**133**  
栏 类型 说明  
j\_job\_name 文本 作业名称（使用提交选项 -N  
设置的脚本名称或值）  
j\_group 文本 执行作业的主要组的 UNIX 组  
名。  
引用组表格。  
j\_owner 文本 正在运行作业的 UNIX 用户帐  
户。  
引用用户表格。  
j\_account 文本 使用提交选项 -A 设置的帐户字  
符串。  
j\_priority 整数 使用提交选项 -p 设置的或在队  
列配置中指定的优先级。  
j\_submission\_time 时间戳 任务的提交时间。  
j\_project 文本 项目（仅适用于企业版的 Sun  
Grid Engine）  
引用项目表格。  
j\_department 文本 部门（仅适用于企业版的 Sun  
Grid Engine）  
引用部门表格。  
sge\_job\_usage  
sge\_job\_usage 表格包含作业资源随时间流逝的使用情况。  
对于 N1GE 5.3 系统，每个已完成的作业、阵列任务和并行任务只有一个记录。  
ju\_curr\_time 列包含作业的结束时间（ sge\_job 中的 j\_end\_time）。  
对于 N1GE 6.0 系统，还会存储联机使用情况；这将为存储在 sge\_job 中的一个作业、  
阵列任务和并行任务产生多个记录。监视随时间流逝作业的资源使用情况  
(ju\_curr\_time)；每个作业、阵列任务或并行任务的最后一条记录中均包含可用来统  
计的总使用情况；此记录的 ju\_curr\_time 将等于 sge\_job 的 j\_end\_time。  
栏 类型 说明  
ju\_id 整数 唯一的记录标识符  
ju\_parent 整数 引用 sge\_job 表格  
ju\_curr\_time 整数 使用的当前时间  
**134** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
栏 类型 说明  
ju\_qname 文本 正在运行作业的队列名称。在  
N1GE 6.0 系统中，这将是群集  
队列名称。  
引用队列表格中的队列。  
ju\_hostname 文本 正在运行作业的主机名称。  
引用主机表格中的主机。  
ju\_start\_time 时间戳 作业开始的时间。  
ju\_end\_time 时间戳 作业完成的时间。  
ju\_failed 整数 if != 0 表示故障  
ju\_exit\_status 整数 作业的退出状态  
ju\_granted\_pe 文本 为该作业选择的并行环境。  
ju\_slots 整数 分派到作业的位置数。  
ju\_state 文本 作业状态  
ju\_ru\_wallclock 整数 end\_time – start\_time  
ju\_ru\_utime 双精度型 使用的用户时间  
ju\_ru\_stime 双精度型 使用的系统时间  
ju\_ru\_maxrss 整数 最大驻留集大小  
ju\_ru\_ixrss 整数 当前为 0  
ju\_ru\_ismrss 整数  
ju\_ru\_idrss 整数 整数驻留集大小  
ju\_ru\_isrss 整数 当前为 0  
ju\_ru\_minflt 整数 未要求物理 I/O 的页面错误  
ju\_ru\_majflt 整数 要求物理 I/O 的页面错误  
ju\_ru\_nswap 整数 交换  
ju\_ru\_inblock 整数 块输入操作  
ju\_ru\_oublock 整数 块输出操作  
ju\_ru\_msgsnd 整数 发送的消息  
ju\_ru\_msgrcv 整数 收到的消息  
ju\_ru\_nsignals 整数 收到的信号  
ju\_ru\_nvcsw 整数 自动环境切换  
附录 A • 数据库模式 **135**  
栏 类型 说明  
ju\_ru\_nivcsw 整数 偶然环境切换  
ju\_cpu 双精度型 CPU 时间用量（以秒表示）。  
ju\_mem 双精度型 整体内存用量（以千兆字节秒  
表示）。  
ju\_io 双精度型 在输入/输出操作中传输的数据  
量。  
ju\_iow 双精度型 IO 等待时间（以秒表示）。  
ju\_maxvmem 双精度型 最大 VMEM 大小（以字节表  
示）。  
sge\_job\_request  
作业请求的存储资源。  
目前可处理两种类型的请求（ qsub 选项）：  
1. -l 资源请求，例如 -l arch=solaris,mem\_total=100M  
系统会为每个请求创建一条记录。  
2. -q 队列请求，例如 -q balrog。 q  
所创建的记录包含 “queue” 变量和请求内容变量。  
栏 类型 说明  
jr\_id 整数 唯一的记录标识符  
jr\_parent 整数 引用 sge\_job 表格  
jr\_variable 文本 请求的属性定义集变量的名称  
jr\_value 文本 请求的值  
sge\_job\_log  
sge\_job\_log 表格包含作业日志信息。  
栏 类型 说明  
jl\_id 整数 唯一的记录标识符  
**136** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
栏 类型 说明  
jl\_parent 整数 引用 sge\_job 表格  
jl\_time Unix 时间戳 生成作业登录项的时间。  
jl\_event 文本  
jl\_job\_number 整数  
jl\_task\_number 整数  
jl\_pe\_task\_id 文本  
jl\_state 文本 报告事件后的作业状态  
jl\_user 文本 启动事件操作的用户  
jl\_host 文本 启动事件操作的主机  
jl\_state\_time Unix 时间戳 描述作业处于某种状态的时  
间，请参阅下面的说明  
jl\_message 文本 解释所发生事件的消息  
sge\_share\_log  
sge\_share\_log 表格包含关于 N1GE(EE) 份额树配置和用量的信息。  
有关更多信息，请参阅 N1GE 手册 sharetree(5)。  
栏 类型 说明  
sl\_id 整数 共享日志记录的唯一标识符  
sl\_curr\_time 时间戳 当前时间  
sl\_usage\_time 时间戳 使用时间  
sl\_node 文本 份额树的节点名称  
sl\_user 文本 用户名称（作业拥有者）  
引用用户表格。  
sl\_project 文本 项目名称  
引用项目表格。  
sl\_shares 整数 份额树中配置的份额  
sl\_job\_count 整数 份额树策略涉及的作业数  
附录 A • 数据库模式 **137**  
栏 类型 说明  
sl\_level 双精度型 此树的级别内的份额（以百分  
比表示）  
sl\_total 双精度型 在整个份额树中的总份额（以  
百分比表示）  
sl\_long\_target\_share 双精度型 所需长期份额（以百分比表  
示）  
sl\_short\_target\_share 双精度型 所需短期份额（以百分比表  
示）  
sl\_actual\_share 双精度型 实际份额（以百分比表示）  
sl\_usage 双精度型 可以配置的组合用量、 CPU 权  
重、内存和 IO  
sl\_cpu 双精度型 CPU 用量（以秒表示）  
sl\_mem 双精度型 整体内存用量（以千兆字节秒  
表示）  
sl\_io 双精度型 在输入/输出操作中传输的数据  
量。  
sl\_ltcpu 双精度型 长期 CPU  
sl\_ltmem 双精度型 长期内存  
sl\_ltio 双精度型 长期 IO  
sge\_host  
sge\_host 表格列出了群集中的所有主机。  
栏 类型 说明  
h\_id 整数 唯一的主机标识符  
h\_hostname 文本 主机名。  
**138** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
sge\_host\_values  
sge\_host\_values 表格存储着随时都会更改的主机变量值，例如负荷平均值。  
此外，还将存储派生的主机值，例如每小时的平均值、总值等。  
栏 类型 说明  
hv\_hostname 文本 引用主机表格。  
hv\_time\_start 时间戳 值开始生效的时间。  
hv\_time\_end 时间戳 值失效的时间。  
hv\_variable 文本 变量名，例如 load\_avg。  
hv\_value 文本 变量值，例如 0.34。  
hv\_dvalue 双精度 数字变量值。  
hv\_dconfig 双精度 可使用时：可使用的最大可用  
值（配置的值）。  
sge\_queue  
sge\_queue 表格列出了在群集中配置的所有队列。  
栏 类型 说明  
q\_id 整数 唯一的队列标识符  
q\_qname 文本 队列名称  
q\_hostname 文本 主机名称  
sge\_queue\_values  
sge\_queue\_values 表格存储着随时都会更改的队列变量值，例如空闲插槽数。  
此外，还将存储派生的队列值，例如每小时的平均值、总值等。  
栏 类型 说明  
qv\_parent 整数 引用 sge\_queue 表格中的  
q\_id。  
附录 A • 数据库模式 **139**  
栏 类型 说明  
qv\_time\_start 时间戳 值开始生效的时间。  
qv\_time\_end 时间戳 值失效的时间。  
qv\_variable 文本 变量名，例如 slots。  
qv\_value 文本 变量值，例如 5。  
qv\_dvalue 双精度 数字变量值  
qv\_dconfig 双精度 可使用时：可使用的最大可用  
值（配置的值）。  
sge\_department  
列出在数据库中引用的所有部门。  
栏 类型 说明  
d\_id 整数 唯一的部门标识符  
d\_department 文本 部门名称。  
表 9： sge\_department 表格  
sge\_department\_values  
sge\_department\_values 表格存储着随时都会更改的部门相关变量的值。当前这些值均  
是派生的，例如每小时的平均值、总值等。  
栏 类型 说明  
dv\_parent 整数 引用 sge\_department 表格中的 d\_id。  
dv\_time\_start 时间戳 值开始生效的时间。  
dv\_time\_end 时间戳 值失效的时间。  
dv\_variable 文本 变量名，例如 h\_sum\_jobs  
dv\_value 文本 变量值，例如 5。  
dv\_dvalue 双精度 数字变量值。  
dv\_dconfig 双精度 可使用时：可使用的最大可用值（配置的值）。  
**140** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
sge\_project  
列出数据库中引用的所有项目。  
栏 类型 说明  
p\_id 整数 唯一的项目标识符  
p\_project 文本 项目名称。  
sge\_project\_values  
sge\_project\_values 表格存储着随时都会更改的项目相关变量的值。当前这些值均是派  
生的，例如每小时的平均值、总值等。  
栏 类型 说明  
pv\_parent 整数 引用 sge\_queue 表格中的  
q\_id。  
pv\_time\_start 时间戳 值开始生效的时间。  
pv\_time\_end 时间戳 值失效的时间。  
pv\_variable 文本 变量名，例如 h\_avg\_cpu  
pv\_value 文本 变量值，例如 345.5  
pv\_dvalue 双精度 数字变量值。  
pv\_dconfig 双精度 可使用时：可使用的最大可用  
值（配置的值）。  
sge\_user  
列出数据库中引用的所有用户。  
栏 类型 说明  
u\_id 整数 唯一的用户 ID  
u\_user 文本 用户名。  
附录 A • 数据库模式 **141**  
sge\_user\_values  
sge\_user\_values 表格存储着随时都会更改的用户相关变量的值。这些值是当前派生的队  
列值，例如每小时的平均值、总值等。  
栏 类型 说明  
uv\_parent 整数 引用 sge\_queue 表格中的  
q\_id。  
uv\_time\_start 时间戳 值开始生效的时间。  
uv\_time\_end 时间戳 值失效的时间。  
uv\_variable 文本 变量名，例如 h\_sum\_cpu  
uv\_value 文本 变量值，例如 23.2  
uv\_dvalue 双精度 数字变量值。  
uv\_dconfig 双精度 可使用时：可使用的最大可用  
值（配置的值）。  
sge\_group  
列出数据库中引用的所有用户组。  
栏 类型 说明  
g\_id 整数 唯一的组 ID  
g\_group 文本 组名。  
表 15： sge\_group 表格  
sge\_group\_values  
sge\_group\_values 表格存储着随时都会更改的组相关变量的值。这些值均是当前派生  
的，例如每小时的平均值、总值等。  
栏 类型 说明  
gv\_parent 整数 引用 sge\_queue 表格中的  
q\_id。  
**142** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
栏 类型 说明  
gv\_time\_start 时间戳 值开始生效的时间。  
gv\_time\_end 时间戳 值失效的时间。  
gv\_variable 文本 变量名，例如 h\_sum\_jobs。  
gv\_value 文本 变量值，例如 53  
gv\_dvalue 双精度 数字变量值。  
gv\_dconfig 双精度 可使用时：可使用的最大可用  
值（配置的值）。  
预定义视图列表  
view\_accounting  
作业、阵列任务和紧密集成任务的统计记录。仅包含已完成的作业。  
栏 类型 说明  
job\_number 整数 作业号  
task\_number 整数 阵列任务 ID  
pe\_taskid 文本 紧密集成的并行任务的 ID  
名称 文本 作业名称（使用提交选项 -N  
设置的脚本名称或值）  
groupname 文本 执行作业的主要组的 UNIX 组  
名。引用组表格。  
username 文本 正在运行作业的 UNIX 用户帐  
户。引用用户表格。  
客户资格 文本 使用提交选项 -A 设置的帐户字  
符串  
项目 文本 引用项目表格中的项目  
部门 文本 引用部门表格中的部门  
submission\_time 时间戳 作业的提交时间  
附录 A • 数据库模式 **143**  
栏 类型 说明  
start\_time 时间戳 作业开始的时间  
end\_time 时间戳 作业完成的时间  
wallclock\_time 整数 end\_time - start\_time  
cpu 双精度型 CPU 时间用量（以秒表示）  
io 双精度型 在输入/输出操作中传输的数据  
量  
iow 双精度型 IO 等待时间（以秒表示）  
maxvmem 双精度型 最大 VMEM 大小（以字节表  
示）  
wait\_time 整数 start\_time - submission\_time  
turnaround\_time 整数 end\_time - submission\_time  
view\_job\_times  
与 view\_accounting 相同，但未列出紧密集成的并行作业的任务。  
view\_jobs\_completed  
每小时完成的作业数，每小时记录一次。  
栏 类型 说明  
完成 整数 完成的作业  
time 时间戳 作业完成的时间  
view\_job\_log  
作业日志（例如 “Submission”、状态更改情况、作业完成情况）。  
栏 类型 说明  
job\_number 整数 作业号  
task\_number 整数 阵列任务 ID  
**144** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
栏 类型 说明  
pe\_taskid 文本 紧密集成的并行任务的 ID  
名称 文本 作业名称（使用提交选项 -N  
设置的脚本名称或值）  
username 文本 执行作业的主要组的 UNIX 组  
名。引用组表格  
客户资格 文本 正在运行作业的 UNIX 用户帐  
户。引用用户表格  
项目 文本 项目。引用项目表格  
部门 文本 部门。引用部门表格  
time 时间戳 生成作业日志项的时间。  
事件 文本 记录的事件  
状态 文本 报告事件后的作业状态  
启动器 (initiator) 文本 启动事件操作的用户  
主机 文本 启动事件操作的主机  
消息 文本 解释所发生事件的消息  
view\_department\_values  
部门特定变量。  
栏 类型 说明  
部门 文本 部门名称  
time\_start 时间戳 值开始生效的时间  
time\_end 时间戳 End time for the validity of a  
value  
variable 文本 变量名，例如 h\_sum\_jobs  
str\_value 文本 变量值，例如 5  
num\_value 双精度 数字变量值  
num\_config 双精度 可使用时：可使用的最大可用  
值（配置的值）  
附录 A • 数据库模式 **145**  
view\_group\_values  
组特定变量  
栏 类型 说明  
groupname 文本 组名  
time\_start 时间戳 值开始生效的时间  
time\_end 时间戳 值失效的时间  
variable 文本 变量名，例如 h\_sum\_jobs  
str\_value 文本 变量值，例如 53  
num\_value 双精度 数字变量值  
num\_config 双精度 可使用时：可使用的最大可用  
值（配置的值）  
view\_host\_values  
主机特定变量  
栏 类型 说明  
**hostname** 文本 主机名  
time\_start 时间戳 值开始生效的时间  
time\_end 时间戳 值失效的时间  
variable 文本 变量名，例如 load\_avg  
str\_value 文本 变量值，例如 0.34  
num\_value 双精度 数字变量值  
num\_config 双精度 可使用时：可使用的最大可用  
值（配置的值）  
**146** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
view\_project\_values  
项目特定变量  
栏 类型 说明  
项目 文本 项目名称  
time\_start 时间戳 值开始生效的时间  
time\_end 时间戳 值失效的时间  
variable 文本 变量名，例如 h\_avg\_cpu  
str\_value 文本 变量值，例如 345.5  
num\_value 双精度 数字变量值  
num\_config 双精度 可使用时：可使用的最大可用  
值（配置的值）  
view\_queue\_values  
队列特定变量  
栏 类型 说明  
qname 文本 队列名称  
hostname 文本 主机名称  
time\_start 时间戳 值开始生效的时间  
time\_end 时间戳 值失效的时间  
variable 文本 变量名，例如 slots  
str\_value 文本 变量值，例如 5  
num\_value 双精度 数字变量值  
num\_config 双精度 可使用时：可使用的最大可用  
值（配置的值）  
附录 A • 数据库模式 **147**  
view\_user\_values  
用户特定变量。  
栏 类型 说明  
username 文本 Name of the user  
time\_start 时间戳 值开始生效的时间  
time\_end 时间戳 值失效的时间  
variable 文本 变量名，例如 h\_sum\_cpu  
str\_value 文本 变量值，例如 23.2  
num\_value 双精度 数字变量值  
num\_config 双精度 可使用时：可使用的最大可用  
值（配置的值）  
派生值列表  
数据库中存储的派生值可以极大地缩短查询处理时间。报告数据库每小时对值进行一次  
汇总（总值、平均值、最小值、最大值）。一段时间后（例如一年），可以将这些值进  
一步压缩为日、周或月值。  
可提供以下派生的值：  
表格 变量 说明  
sge\_host\_values h\_sum\_cpu、 d\_sum\_cpu、 m\_sum\_cpu 每台主机在每小时、每天、每月的 CPU 用量  
sge\_user\_values h\_sum\_cpu、 d\_sum\_cpu、 m\_sum\_cpu 每个用户在每小时、每天、每月的 CPU 用量  
sge\_group\_values h\_sum\_cpu、 d\_sum\_cpu、 m\_sum\_cpu 每个组在每小时、每天、每月的 CPU 用量  
sge\_department\_values h\_sum\_cpu、 d\_sum\_cpu、 m\_sum\_cpu 每个部门在每小时、每天、每月的 CPU 用量  
sge\_project\_values h\_sum\_cpu、 d\_sum\_cpu、 m\_sum\_cpu 每个项目在每小时、每天、每月的 CPU 用量  
sge\_host\_values h\_avg\_load 平均主机负荷  
sge\_host\_values h\_max\_load 最大主机负荷  
可以在配置文件中指定生成派生值的规则。  
**148** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
词汇表  
管理主机  
**(administration host)**  
管理主机是有权执行 Grid Engine 系统的管理活动的主机。  
访问权限列表 **(access**  
**list)**  
是包含多个用户和 UNIX 组的列表，可允许或拒绝列表中的用户对资源  
（如队列或主机）的访问。用户和组可以属于多个访问权限列表，同一  
个访问权限列表可用于不同的环境。  
阵列作业 **(array job)** 由多个相互独立的雷同任务组成的作业。每个任务类似于一个单独的作  
业。阵列作业任务仅依靠其独特的任务标志符（整数数字）相互区别。  
校园网格 **(campus grid)** 允许同一组织内的多个项目或多个部门共享计算资源的网格。  
单元 **(cell)** 具有独立配置和独立主机的独立群集。单元可以用来松散地耦合独立的  
管理单元。  
点校验 **(checkpointing)** 一种将作业执行状态保存到校验点的过程，从而允许中止作业以及随后  
继续作业，而不会丢失信息和已完成的工作。 恢复执行之前，校验点从  
一个主机移动到另一个主机的过程称为迁移。  
点校验环境  
**(checkpointing**  
**environment)**  
定义与特定的点校验方法相关联的事件、界面和操作的 Grid Engine 系  
统配置实体。  
群集 可执行 Grid Engine 系统功能的计算机（称为主机）的集合。  
群集网格 **(cluster grid)** 网格的最简单形式，由一些协同工作的计算机主机组成，向单个项目或  
部门中的用户提供单点访问。  
群集队列 **(cluster queue)** 含有可并行运行的同一类作业的容器。队列决定着作业的的某些属性，  
例如，是否可迁移。作业在运行的整个过程中，始终与队列相关联。 与  
队列的关联性可以影响针对作业发生的一些事件。例如，如果暂停队  
列，则与该队列相关联的作业均被暂停。  
属性定义集 **(complex)** 与队列、主机或者整个群集相关联的资源属性定义的集合。  
部门 **(department)** 在 Grid Engine 系统的职能和越权调度策略中进行相同处理的用户和组  
的列表。 用户和组可以只属于一个部门。  
**149**  
权益 **(entitlement)** 与份额一样。规划用于特定作业、用户、用户组或项目的资源量。  
执行主机 **(execution**  
**host)**  
有权运行 Grid Engine 系统作业的系统。这些系统具有队列实例，并运  
行执行守护进程 sge\_execd。  
职能策略 **(functional**  
**policy)**  
为作业、用户、用户组和项目分配特定的重要性级别的策略。例如，通  
过职能策略，高优先级的项目及其所有作业与低优先级项目相比，可获  
得较高的资源份额。  
全局网格 **(global grid)** 跨越组织创建巨型虚拟系统的校园网格的集合。  
网格 **(grid)** 执行任务的计算资源的集合。用户将网格视为单个计算资源。  
组 **(group)** UNIX 组。  
硬性资源需求 **(hard**  
**resource requirements)**  
开始执行作业之前必须分配的资源。与软性资源需求相对应。  
主机 可执行 Grid Engine 系统功能的系统。  
作业 **(job)** 用户发出的对于网格计算资源的请求。  
批处理作业 **(batch job)** 批处理作业是不需用户介入也不需要访问终端就可运行的 UNIX shell 脚  
本。  
交互式作业 **(interactive**  
**job)**  
交互式作业是以 qrsh、 qsh 或 qlogin 命令开始的会话，该会话打开  
一个用于用户交互或提供一个等同于远程登录会话的 *xterm* 窗口。  
作业类 **(job class)** 在某种程度上相同并以类似方式处理的一组作业。作业类通过相应作业  
的相同需求以及适合这些作业的队列的特征来定义。  
管理人员**(manager)** 可以全面控制 Grid Engine 软件的用户。主控主机以及任何声明为管理  
主机的计算机的超级用户，均具有管理人员权限。管理人员权限也可以  
分配给非超级用户帐户。  
迁移 **(migration)** 在恢复执行作业之前将点校验作业从一个主机移动到另一个主机的过  
程。  
操作员 **(operator)** 这类用户除不能修改配置之外，可以执行与管理人员相同的命令。操作  
员维护操作。  
越权策略 **(override**  
**policy)**  
该策略通常用于替代职能策略和基于份额的策略的自动资源权益管理。  
群集管理员可以通过修改自动策略的执行，将越权分配给作业、用户、  
用户组和项目。  
拥有者 **(owner)** 此类用户可以暂停或恢复、禁用或启用其拥有的队列。通常，用户是驻  
留在他们工作站上的队列实例的拥有者。  
并行环境 **(parallel**  
**environment)**  
Grid Engine 系统的一种配置，该配置为 Grid Engine 软件定义必要的  
界面以使其正确地处理并行作业。  
并行作业 **(parallel job)** 由多个密切相关的任务组成的一个作业。这些任务可以分布于多个主机  
之上。并行作业通常使用如共享内存或消息传递 (MPI、 PVM) 这样的通  
信工具同步任务并使任务相关联。  
**150** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
策略 **(policy)** 管理员用来定义 Grid Engine 系统行为的规则和配置的集合。策略由系  
统自动执行。  
优先级 **(priority)** 一项作业相对于其他作业的重要性。  
项目**(project)** Grid Engine 系统的项目。  
资源 运行作业所需要使用或占用的计算设备。典型示例如：内存、 CPU、  
I/O 带宽、文件空间、软件许可证等等。  
主控主机 **(master host)** 主控主机是全部群集活动的中心。其上运行着主守护进程  
sge\_qmaster 和调度程序守护进程 sge\_schedd。默认情况下，主控  
主机也是管理主机和提交主机。  
份额 **(share)** 与权益相同。规划用于某个作业、用户或者项目的资源量。  
基于份额的策略  
**(share-based policy)**  
允许以层次形式定义用户、项目和任意组的权益的策略。 比如说企业可  
以再分为分支机构、部门、部门内的活动项目、项目的用户组、用户组  
中的用户。基于份额的分层结构称为份额树，一旦定义了份额树， Grid  
Engine 软件便可以自动执行权益分配。  
份额树 **(share-tree)** 基于份额策略的分层结构定义。  
软性资源需求 **(soft**  
**resource requirements)**  
作业需要的资源，但不一定要在作业开始之前分配。在可用的基础上分  
配给作业。与硬性资源需求相对应。  
提交主机 **(submit host)** 提交主机只允许提交和控制批处理作业。具体地说，用户登录提交主机  
后可以使用 qsub 提交作业、使用 qstat 控制作业状态、还可以使用  
Grid Engine 系统 OSF/1 Motif 图形用户界面 QMON。  
暂停 **(suspension)** 将正在运行的作业留置在执行主机上的过程（与作业中止处的点校验相  
对应）。暂停的的作业仍需要使用部分资源，如交换内存或文件空间。  
票券 **(ticket)** 用于资源份额定义的通用单位 。作业、用户、项目或其他组件具有的票  
券份额越多，它就越重要。如果一个作业的票券数是另一个的两倍，则  
该作业就有权使用两倍的资源。  
用量 **(usage)** “资源使用”的另一种说法。用量是 CPU 使用时间、内存的超时占用  
和 I/O 执行数量的加权总数，这个总数可由管理员进行配置。  
用户 **(user)** 至少在一台提交主机和一台执行主机上以有效登录 ID 登录后，便可以  
向网格提交并运行作业的人。  
用户组 **(userset)** 一个访问权限列表或一个部门。  
**151**  
**152** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
索引  
数字和符号  
“Main Control” 窗口, 29-30  
$pe\_hostfile, 52  
3rd\_party 文件, 69, 70, 72  
**A**  
access-allowed-list, 32  
access-denied-list, 32  
Accounting and Reporting Console, 101-104  
编辑查询  
简单, 115-116  
查询  
高级, 116-117, 117  
简单, 104-108, 115  
创建查询  
简单, 104-108  
高级查询  
创建, 116-117  
运行, 117  
简单查询  
编辑, 115-116  
创建, 104-108  
运行, 115  
启动, 101-104  
运行查询  
高级, 116-117, 117  
简单, 115  
ACL, 32  
act\_qmaster 文件, 37  
ARC, 51  
at 作业, 97  
**B**  
-b qrsh 选项, 60, 70  
**C**  
C, 紧急消息, 119  
-C qsub 选项, 50  
-c qsub 选项, 99  
-c qtcsh 选项, 71  
C 程序集成, 71  
cell, 51  
ckpt\_dir, 100  
-clear qsub 选项, 61  
crontab 作业, 97  
csh shell, 47  
.cshrc 文件, 48  
**D**  
-d qmod 选项, 96  
DISPLAY 变量, 65  
dl, 123  
**E**  
E, 错误消息, 119  
-e qmod 选项, 96  
ENVIRONMENT, 51  
/etc/login 文件, 48  
-ext qstat选项, 75  
**153**  
**F**  
-f qdel 选项, 89  
-f qmod 选项, 89, 96  
**G**  
gmake, 72  
-j, 73  
**H**  
-hold\_jid qsub 选项, 64  
HOME, 49, 51  
HOSTNAME, 49, 51  
**I**  
I, 信息消息, 119  
-inherit qmake 选项, 73  
-inherit qrsh 选项, 70, 72  
**J**  
-j gmake 选项, 73  
-j qmake 选项, 73  
Java 程序集成, 71  
JOB\_ID, 49, 51  
JOB\_NAME, 49, 51  
**K**  
ksh shell, 47  
**L**  
-l qstat 选项, 87  
login\_shells, 47  
.login 文件, 48  
LOGNAME, 51  
**M**  
-m a qsub 选项, 82  
-M qsub 选项, 89  
-m qsub选项, 89  
MAIL, 51  
make, 72  
makefile, 并行处理, 72-73  
**N**  
N, 注意消息, 119  
newgrp, 32  
NFS 网络文件系统, 100  
NHOSTS, 51  
-noshell qrsh 选项, 70  
-nostdin qrsh 选项, 70  
-now no qlogin 选项, 66  
-now no qrsh 选项, 66  
-now no qsh 选项, 66  
-now no qsub 选项, 66  
-now qrsh 选项, 70  
NQUEUES, 51  
NSLOTS, 51  
**O**  
-ot qalter 选项, 87  
**P**  
-p qsub 选项, 24, 74  
PATH, 51, 52  
PE, 52  
PE\_HOSTFILE, 52  
-pe qmake 选项, 73  
project\_lists, 33  
**Q**  
qacct, 27  
qalter, 27, 79  
-ot, 87  
一致性检查, 81  
qconf, 27  
**154** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
qconf （续）  
-sc, 38  
-se, 37  
-sel, 37  
-sh, 37  
-sm, 33  
-so, 33  
-sq, 33, 35  
-sql, 34  
-ss, 37  
-su, 32  
-sul, 32  
qdel, 27  
-f, 89  
取消作业, 88  
qhold, 27  
qhost, 27, 37  
qlogin, 27, 65, 68-69  
-now no, 66  
qmake, 27, 72  
-inherit, 73  
-j, 73  
-pe, 73  
交互式用法, 73  
批处理用量, 73  
qmod, 27  
-d, 96  
-e, 96  
-f, 89, 96  
-s, 88, 96  
-us, 88, 96  
监视作业, 88  
禁用队列, 96  
使用 crontab 或 at, 97  
暂停队列, 96  
qmon, 27  
QMON  
“Main Control” 窗口, 29-30  
定制, 28, 30, 83, 96  
首选项, 83, 96  
优先权, 30  
与嵌入的脚本自变量, 50  
.qmon\_preferences 文件, 30, 83, 96  
Qmon 文件, 30  
QMON 资源文件, 30  
qresub, 27  
qrls, 27  
qrsh, 27, 65, 69-70  
qrsh （续）  
-b, 60, 70  
-inherit, 70, 72  
-noshell, 70  
-nostdin, 70  
-now, 70  
-now no, 66  
-verbose, 70  
qrshmode, 72  
qselect, 27  
qsh, 27, 65, 68  
-now no, 66  
qstat, 25, 27  
-ext, 75  
-l, 87  
-r, 87  
资源需求, 87  
qsub, 25, 27  
-C, 50  
-c, 99  
-clear, 61  
-cwd 点校验作业, 100  
-hold\_jid, 64  
-M, 89  
-m, 89  
-m a, 82  
-now no, 66  
-p, 24, 74  
-t, 65  
脚本中的自变量, 50  
.qtask 文件, 70, 71  
qtcsh, 27, 70-72  
-c, 71  
QUEUE, 52  
queue\_sort\_method, 75  
**R**  
-r qstat, 选项, 87  
remsh, 69  
REQUEST, 52  
RESTARTED, 52  
rlogin, 65, 69  
rsh, 65, 69-70  
**155**  
**S**  
-s qmod 选项, 88, 96  
-sc qconf 选项, 38  
schedd\_job\_info, 81  
-se qconf 选项, 37  
-sel qconf 选项, 37  
seq\_no, 75  
SGE\_BINARY\_PATH, 51  
SGE\_CELL, 51  
SGE\_CKPT\_DIR, 51  
SGE\_CKPT\_ENV, 51  
sge\_execd, 25, 26  
SGE\_JOB\_SPOOL\_DIR, 51  
SGE\_O\_HOME, 51  
SGE\_O\_HOST, 51  
SGE\_O\_LOGNAME, 51  
SGE\_O\_MAIL, 51  
SGE\_O\_PATH, 51  
SGE\_O\_SHEL, 51  
SGE\_O\_TZ, 51  
SGE\_O\_WORKDIR, 51  
sge\_qmaster, 25, 26  
.sge\_request 文件, 60  
sge\_request 文件, 60  
SGE\_ROOT, 51  
sge\_schedd, 25  
SGE\_STDERR\_PATH, 51  
SGE\_STDOUT\_PATH, 51  
SGE\_TASK\_ID, 51, 64  
-sh qconf 选项, 37  
sh shell, 47  
SHELL, 51  
shell, 52  
shell\_start\_mode, 49  
shell 队列参数, 49  
shell 脚本, 47-48  
Grid Engine 系统扩展, 48-52  
示例, 48  
-sm qconf 选项, 33  
-so qconf 选项, 33  
-sq qconf 选项, 33, 35  
-sql qconf 选项, 34  
-ss qconf 选项, 37  
stderr 重定向  
重定向  
stderr, 119  
-su qconf 选项, 32  
-sul qconf 选项, 32  
**T**  
-t qsub 选项, 65  
TASK\_ID, 49  
tcsh shell, 70  
tcsh shell, 47  
telnet, 65, 69  
TMP, 52  
TMPDIR, 52  
TZ, 52  
时区, 51  
**U**  
unix\_behavior, 49  
-us qmod 选项, 88, 96  
USER, 49, 52  
user\_lists, 32  
**V**  
-verbose qrsh 选项, 70  
**W**  
W, 警告消息, 119  
**X**  
XAPPLRESDIR, 30  
.Xdefaults 文件, 30  
.xinitrc 文件, 30  
xproject\_lists, 33  
xrdb, 30  
.Xresources 文件, 30  
xterm, 65  
xuser\_lists, 32  
变  
变量, 环境, 51-52  
**156** N1 Grid Engine 6 用户指南 • 2005 年 7 月  
标  
标准错误, 48, 49  
标准输出, 48, 49  
并  
并行环境, 150  
并行作业, 51, 150  
部  
部门, 149  
操  
操作员, 31, 150  
列出, 33  
策  
策略, 20, 74-75, 151  
管理, 21  
管理票券策略, 23  
基于份额, 23, 74, 151  
基于票券的, 74-75  
紧急程度, 23, 24, 74  
类型, 23  
越权, 23, 74, 150  
职能, 23, 53, 74, 150  
处  
处理并行 makefile, 72-73  
处理层次, 点校验, 97  
磁  
磁盘空间, 点校验的要求, 100  
从  
从属性  
作业, 54, 64  
错  
错误  
队列状态, 91  
作业状态, 81  
错误报告, 123  
错误消息, 119  
单  
单元, 149  
登  
登录 shell, 47  
点  
点校验, 97-100, 149  
处理层次, 97  
磁盘空间要求, 100  
和重新启动, 52  
内存要求, 100  
内核级别, 97  
迁移作业, 98  
文件系统要求, 100  
用户级别, 97  
点校验环境, 149  
点校验目录, 100  
电  
电子邮件, 89, 119  
错误邮件格式, 120  
作业结束时发送, 89  
作业开始时发送, 89  
作业暂停时发送, 89  
作业中止时发送, 89  
**157**  
调  
调度  
策略, 74-75  
票券, 74-75  
调度程序守护进程, 25, 26  
调试模式, 123  
跟踪输出, 123  
定  
定制  
QMON, 28, 30, 83, 96  
动  
动态负荷平衡, 98  
队  
队列, 26, 149  
shell 参数, 49  
错误状态, 91  
负荷参数, 95  
和作业, 22-23  
恢复, 91, 93  
禁用, 91, 93  
可使用资源, 94, 95  
列表, 34  
配置, 91  
批处理, 75  
启用, 91, 93  
属性列表, 34-35  
位置限制, 95  
系统负荷, 94  
显示, 34-36  
序号, 75  
选择, 75  
拥有者, 31, 33  
用 qmod 禁用队列, 96  
用 qmod 暂停, 96  
暂停, 91, 93  
资源属性, 94, 95  
队列实例, 92-94  
恢复, 93  
禁用, 93  
队列实例 （续）  
配置, 91  
启用, 93  
暂停, 93  
访  
访问权限列表, 32, 149  
分  
分派作业, 使用命名的队列请求, 75  
份  
份额, 151  
份额树, 151  
负  
负荷参数, 95  
负荷调整, 95  
负荷管理, 21  
负荷平衡, 动态, 98