



Sun™ ONE Grid Engine 5.3 和 Sun™ ONE Grid Engine，企業版 5.3 參考手冊

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054 U.S.A.
650-960-1300

文件編號：816-7479-10
2002 年 9 月，修訂版 A

請將您對本文件的意見寄到：docfeedback@sun.com

Copyright 2002 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 版權所有。

本產品或文件按照限制其使用、複製、分發和反編譯的授權許可進行分發。未經 Sun 及其授權許可頒發機構的書面授權，不得以任何方式、任何形式複製本產品或本文件的任何部分。協力廠商軟體，包括字型技術，由 Sun 供應商提供許可和版權。

本產品的某些部分從 Berkeley BSD 系統衍生而來，經 University of California 許可授權。UNIX 是在美國和其他國家註冊的商標，經 X/Open Company, Ltd. 獨家許可授權。

Sun、Sun Microsystems、Sun 標誌、AnswerBook2、docs.sun.com 和 Solaris 是 Sun Microsystems, Inc. 在美國和其他國家的商標、註冊商標或服務標記。所有的 SPARC 商標都按授權許可使用，是 SPARC International, Inc. 在美國和其他國家的商標或註冊商標。具有 SPARC 商標的產品都基於 Sun Microsystems, Inc. 開發的架構。「能源之星」標誌是 EPA 的註冊商標。OPEN LOOK 和 Sun™ 圖形使用者介面是 Sun Microsystems, Inc. 為其使用者和授權許可持有人開發的。Sun 承認 Xerox 在為電腦行業研究和開發可視或圖形使用者介面方面所作出的先行努力。Sun 以非獨佔方式從 Xerox 獲得 Xerox 圖形使用者介面的授權許可，該授權許可涵蓋實施 OPEN LOOK GUI 且遵守 Sun 的書面許可協議的授權許可持有人。

本資料按「現有形式」提供，不承擔明確或隱含的條件、陳述和保證，包括對特定目的或非侵害性的商業活動和適用性的任何隱含保證，除非這種不承擔責任的聲明是不合法的。



請回收



Adobe PostScript

目錄

1. Sun Grid Engine 5.3 和 Sun Grid Engine，企業版 5.3 線上說明手冊	1
簡介	1
sge_intro(1)	1
sge_ckpt(1)	4
qacct(1)	6
qconf(1)	10
qdel(1)	30
qhold(1)	33
qhost(1)	35
qmake(1)	40
qmod(1)	44
qmon(1)	47
qrls(1)	55
qselect(1)	57
qstat(1)	60
qtcsh(1)	68
submit(1)	71
access_list(5)	99

accounting(5) 100
calendar_conf(5) 102
checkpoint(5) 105
sge_request(5) 108
sge_aliases(5) 110
sge_conf(5) 111
sge_h_aliases(5) 125
sge_pe(5) 126
complex(5) 129
host_conf(5) 134
hostgroup(5) 137
project(5) 139
qtask(5) 140
queue_conf(5) 141
sched_conf(5) 150
share_tree(5) 154
user(5) 155
usermapping(5) 156
sge_commd(8) 158
sge_execd(8) 161
sge_qmaster(8) 164
sge_schedd(8) 166
sge_shadowd(8) 168
sge_sheperd(8) 170
sgecommdcnt1(8) 172

表格

表 1	使用者介面程式及說明	2
表 2	qacct 選項	7
表 3	qacct 環境變數	9
表 4	qconf 選項	10
表 5	qconf 環境變數	29
表 6	qdel 選項	31
表 7	qdel 環境變數	32
表 8	qhold 選項	34
表 9	qhold 環境變數	34
表 10	qhost 選項	36
表 11	qhost 環境變數	39
表 12	qmake 環境變數	42
表 13	qmod 選項	44
表 14	qmod 環境變數	46
表 15	qmon 選項	48
表 16	qmon 環境變數	54
表 17	qrls 選項	56
表 18	qrls 環境變數	56
表 19	qselect 選項	58

表 20	qselect 環境變數	59
表 21	qstat 選項	61
表 22	qstat 環境變數	67
表 23	qtcsch 選項	69
表 24	qtsch 環境變數	70
表 25	submit 指令選項	73
表 26	submit 環境變數	92
表 27	額外的 qsub、qsh、以及 qlogin 環境變數	93
表 28	藉由 submit 指令設定作業環境中的變數	93
表 29	sge_cmmd 選項	159
表 30	sge_cmmd 環境變數	160
表 31	sge_execd 選項	161
表 32	sge_execd 環境變數	162
表 33	sge_qmaster 選項	164
表 34	sge_qmaster 環境變數	165
表 35	sge_schedd 選項	167
表 36	sge_schedd 環境變數	167
表 37	sge_shadowd 環境變數	169
表 38	sgecommdcntl 選項	173
表 39	sgecommdcntl 環境變數	174

前言

《*Sun Grid Engine 5.3 和 Sun Grid Engine，企業版 5.3 參考手冊*》是包含兩項產品的命令說明檔 (man) 編譯結果。本手冊包含兩項產品的最常見資訊。但是當此資訊僅為一特定產品時，就會特別標示出來。

本書編排架構

除了本序言與內容圖表外，本手冊僅包含一個章節。

- Sun Grid Engine 5.3 和 Sun Grid Engine，企業版 5.3 線上說明手冊

使用 UNIX 指令

本文件不包含基本 UNIX[®] 指令及程序的相關資訊，如關閉系統、啓動系統及配置裝置等。

請參考下列資訊：

- 《*Solaris Handbook for Sun Peripherals*》
- 關於 Solaris[™] 作業環境的 AnswerBook2[™] 線上說明文件
- 其他隨著系統一起收到的軟體說明文件

排印慣例

字型	意義	範例
AaBbCc123	指令、檔案和目錄名稱；電腦螢幕輸出。	編輯您的 .login 檔案。 使用 <code>ls -a</code> 列出所有檔案。 % You have mail.
AaBbCc123	您鍵入的內容，與電腦螢幕輸出不同	% su Password:
<i>AaBbCc123</i>	書名、新字或專有名詞、要強調的文字。用真實名稱或數值替換指令行變數。	請閱讀《 <i>使用者指南</i> 》的第六章。 這些是 <i>類別</i> 選項。 您 <i>必須</i> 是超級使用者才能執行此項操作。 若要刪除某個檔案，請鍵入 <code>rm 檔案名稱</code> 。

Shell 提示符號

Shell	提示符號
C shell	<i>機器名稱</i> %
C shell 超級使用者	<i>機器名稱</i> #
Bourne shell 和 Korn shell	\$
Bourne shell 和 Korn shell 超級使用者	#

相關說明文件

適用文件	標題	文件編號
管理與使用	《 <i>Sun Grid Engine</i> ，企業版 5.3 管理和使用者指南》	816-7474-10
管理與使用	《 <i>Sun Grid Engine 5.3 管理和使用者指南</i> 》	816-7467-10

線上存取 Sun 出版文件

部分精選之 Sun 系統說明文件位於：

<http://www.sun.com/products-n-solutions/hardware/docs>

整套的 Solaris 說明文件及其他主題的說明文件則位於：

<http://docs.sun.com>

在該網站上，您也可以找到如何訂購本指南印刷版的相關資訊。

Sun 歡迎您的指教

Sun 一直致力於改善相關的說明文件，因此竭誠歡迎您提出批評和建議。您可以將意見透過電子郵件寄給 Sun：

docfeedback@sun.com

請將文件編號 (816-7479-10) 放到您 email 的主題中。

Sun Grid Engine 5.3 和 Sun Grid Engine，企業版 5.3 線上說明手冊

簡介

本文件所內含之線上手冊與銷售 Sun™ Grid Engine 時所包含之手冊相同。請注意本文件內所列之某些指令只適用於 Sun Grid Engine，企業版 5.3 的產品，而這些指令將於其個別說明中指出。

sg_e_intro(1)

名稱

Sun Grid Engine 簡介 — 可在遠端機器上執行 UNIX 作業之工具

說明

Sun Grid Engine 5.3 和 Sun Grid Engine，企業版 5.3 軟體 — Sun Grid Engine 在本手冊大部分之說明中皆意指這二項產品 — 每個產品的功能均可於協同作業之工作站儲存區內執行 UNIX 批次作業（shell script）。當某些作業站處於閒置或只有輕微的負載情況下，有時作業會送至這些工作站的佇列中等待並且遠端執行。作業量會相對應於每一台機器之負載狀況，以及作業資源的需求而分散於叢集內之工作站間。

支援使用者等級之檢查點程式，並提供透明 (transparent) 之檢查點機制 (請參閱 `sge_ckpt(1)`)。檢查點作業可以不經由使用者介入而依負載需求移轉於工作站之間。除此之外，批次作業、互動式作業以及平行作業皆可提交 Sun Grid Engine 來處理。

使用者介面

Sun Grid Engine 使用者介面由幾種程式所組成，將分別於表 1 中說明。

表 1 使用者介面程式及說明

程式名稱	說明
<code>qacct(1)</code>	自叢集日誌檔案中取出仲裁計算 (arbitrary accounting) 的資訊。
<code>qalter(1)</code>	變更已提交作業之特性。
<code>qconf(1)</code>	提供配置、修改、刪除與查詢佇列以及叢集組態功能之使用者介面。
<code>qdel(1)</code>	提供使用者 / 操作員 / 管理員取消作業之工具。
<code>qhold(1)</code>	暫緩執行已提交之作業。
<code>qhost(1)</code>	顯示關於 Sun Grid Engine 執行主機的狀態資訊。
<code>qlogin(1)</code>	初始化遠程登入或是類似之登入階段作業時，會自動選取低負載與合適之主機來執行。
<code>qmake(1)</code>	替換標準 UNIX 指令 <code>make</code> 之工具。其藉由將每個獨立的 <code>make</code> 步驟分散至合適的機器叢集內，來延伸 <code>make</code> 的功能。
<code>qmod(1)</code>	允許佇列所有人 (owner) 暫停以及啟動與其機器相關聯的所有佇列 (所有目前在此佇列作用中的處理程序也一併標上記號)，或是允許其暫停與啟動所擁有佇列內正在執行的作業。
<code>qmon(1)</code>	提供所有的 Sun Grid Engine 功能一個 Motif 指令介面。藉由變更對應的佇列圖示顏色，以便在線上顯示所有或個人化選取之配置佇列的狀態。
<code>qresub(1)</code>	藉由複製目前正在執行或擱置之作業來建立新作業。
<code>qrsl(1)</code>	釋放先前指定暫緩之作業，例如透過 <code>qhold(1)</code> 指令 (請參閱上述說明)。
<code>qrsh(1)</code>	此程式可以應用在多種不同的用途，像是透過 Sun Grid Engine 提供的可遠端執行互動式應用程式之功能，可與標準 UNIX 工具 <code>rsh</code> 進行比較；允許提交執行中之批次作業；支援終端機 I/O (標準 / 錯誤輸出以及標準輸入) 與終端機控制；提供一個可於作業結束前保持作用中的批次作業提交客戶端；或允許 Sun Grid Engine 控制的遠端執行之平行作業的工作。
<code>qselect(1)</code>	列出對應特定選取標準的佇列名稱清單。 <code>qselect</code> 的輸出通常會傳輸到其他 Sun Grid Engine 的指令，以將動作套用在選取之佇列集上。

表 1 使用者介面程式及說明 (接上頁)

程式名稱	說明
qsh(1)	於低負載之主機上開啓互動式 shell (在 xterm(1) 的環境下)。任何類型的互動式作業都可在此 shell 執行。
qstat(1)	提供此指令所有與叢集相關的作業和佇列的狀態清單。
qsub(1)	提交 Sun Grid Engine 作業之使用者介面。
qtcsh(1)	可完全相容地替換廣為人知且已廣泛使用的 UNIX C-Shell (csh) 所衍生出的 tcsh。透過 Sun Grid Engine，此程式提供了具延伸功能的指令 shell，可將指定之應用程式透過性地分散執行在合適且只有輕微負載之主機上。

同時參閱

sge_ckpt(1)、qacct(1)、qalter(1)、qconf(1)、qdel(1)、qhold(1)、qghost(1)、qlogin(1)、qmake(1)、qmod(1)、qmon(1)、qresub(1)、qrls(1)、qrsh(1)、qselect(1)、qsh(1)、qstat(1)、qsub(1)、qtcsh(1)、《Sun Grid Engine 5.3 管理和使用者指南》與《Sun Grid Engine 企業版 5.3 管理和使用者指南》

版權

Copyright 2002 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 版權所有。

本產品或文件已授權限制其使用、複製、散佈以及反編譯。本產品或文件之任一部分若未事先得知 Sun 與其授權者 (若有的話) 之書面授權同意，不得以任何形式及任何工具進行複製。協力廠商，包含字型技術，由 Sun 的供應商提供版權保護以及授權。

此產品之部分內容來自於 Berkeley BSD 系統，且經由加州大學授權。UNIX 是在美國和其他國家註冊的商標，經 X/Open Company, Ltd. 獨家許可授權。

Sun、Sun Microsystems、Sun 標誌、AnswerBook2、docs.sun.com 和 Solaris 是 Sun Microsystems, Inc. 在美國和其他國家的商標、註冊商標或服務標記。所有的 SPARC 商標都按授權許可使用，是 SPARC International, Inc. 在美國和其他國家的商標或註冊商標。具有 SPARC 商標的產品都基於 Sun Microsystems, Inc. 開發的架構。

OPEN LOOK 和 Sun™ 圖形使用者介面是 Sun Microsystems, Inc. 為其使用者和授權許可持有人開發的。Sun 承認 Xerox 在為電腦行業研究和開發可視或圖形使用者介面方面所作出的先行努力。Sun 以非獨佔方式從 Xerox 得知 Xerox 圖形使用者介面的授權許可，該授權許可涵蓋實施 OPEN LOOK GUI 且遵守 Sun 的書面許可協議的授權許可持有人。

本資料按「現有形式」提供，不承擔明確或隱含的條件、陳述和保證，包括對特定目的或非侵害性的商業活動和適用性的任何隱含保證，除非這種不承擔責任的聲明是不合法的。

sge_ckpt(1)

名稱

Sun Grid Engine 檢查點 — Sun Grid Engine 檢查點機制與檢查點支援

說明

Sun Grid Engine 支援二個等級之檢查點：使用者等級與作業系統所提供之透通性等級。使用者等級檢查點意指那些在某幾個特定的時間點或演算步驟之中寫入重新啟動的檔案，且在重新啟動之後適當的處理這些重新啟動檔，以執行所屬檢查點作業之應用程式。

透通性檢查點必定是由作業系統所提供，且通常與作業系統之核心整合在一起。Softway 為 SGI IRIX 平台所開發之 Hibernator 套裝軟體即為核心整合檢查點工具之一例。

Sun Grid Engine 系統須經由使用 `qsub(1)` 指令內之 `-ckpt` 選項來識別檢查點作業。此旗號之引數意指所謂的檢查點環境，其定義了欲使用之檢查點方式的屬性（若需詳細資訊請參閱 `checkpoint(5)`）。檢查點環境是藉由 `qconf(1)` 來設定，其選項有 `-ackpt`、`-dckpt`、`-mckpt` 與 `-sckpt`。在參照的檢查點環境中可以使用 `qsub(1)` 的 `-c` 選項來覆寫 `when` 的屬性。

若是佇列為檢查點類型，作業須標有檢查點屬性旗號（請參閱 `qsub(1)` 之 `-ckpt` 選項）才能被允許在這類佇列中執行。與一般批次作業之運作方式不同的是，檢查點作業會在某些條件下中斷，然而批次作業或互動式作業只會暫停、甚至根本不受影響。這些條件如下：

- 若是指定 `x` 狀態指定器 `occasion specifier`（請參閱 `qsub(1)-c` 與 `checkpoint(5)`）於一項作業，那麼叢集管理或佇列所有人會透過 `qmod(1)` 下達明確的暫停佇列或作業的指令。
- 平均載入值超過為對應的佇列所配置之移轉臨界值（請參閱 `queue_conf(5)`）。
- 關閉負責檢查點作業之 Sun Grid Engine 執行常駐程式 `sge_execd(8)`。

在檢查點中斷之後，除非有明確的使用者需求將其提交給特定的佇列，否則作業會移轉至其他佇列。作業的移轉會導致動態的負載平衡。

注意 – 中斷檢查點作業將會釋放當時作業所佔用的所有資源（記憶體、置換空間）。此點與暫停一般作業時仍會佔用置換空間的情形不同。

限制

當一項作業移轉至另一台機器之佇列時，同時間不會有任何檔案自動地傳輸至該台機器。這表示一整體作業從頭到尾所有使用到的檔案，包含重新啟動檔、可執行檔以及暫存檔，都必須被明確地顯示出來或傳輸出去（例如，置於作業 script 的開頭部份）。

就透通性檢查點作業之磁碟空間使用方式而言，也有一些實際上的限制。透通性檢查點應用程式內的檢查點通常藉由作業系統儲存在檢查點檔案或目錄中。此檔案或目錄包含處理所需之所有的文字、資料以及堆疊空間，另有一些附加的控制資訊在內。這表示若作業使用非常大量的虛擬位址空間，將會產生極大的檢查點檔案。亦即實際執行作業之作業站，其可用的磁碟空間可能所剩無幾。因此透通性檢查點作業不可能總是傳輸至機器上，即使該機器處於閒置狀態中。因為需要大量虛擬記憶體之作業必須等待一台不但處於閒置狀態、並且有充足的可用磁碟空間之機器來處理，只是這類作業恐須忍受冗長的回應時間之苦。

同時參閱

`sge_intro(1)`、`qconf(1)`、`qmod(1)`、`qsub(1)`、`checkpoint(5)`、《*Sun Grid Engine 5.3 管理和使用者指南*》與《*Sun Grid Engine 企業版 5.3 管理和使用者指南*》

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qacct (1)

名稱

qacct — Sun Grid Engine 用法之報告與計算。

指令概要

```
qacct [ -A 計算 ] [ -b 開始時間 ] [ -d 天數 ]  
[ -e 結束時間 ] [ -g [ 群組 id | 群組名稱 ] ]  
[ -h [ 主機名稱 ] ] [ -help ] [ -history 歷程路徑 ]  
[ -j [ 作業 id | 作業名稱 ] ] [ -l 屬性=數值,... ] [ -nohist ]  
[ -o [ 所有人 ] ] [ -pe [ 平行環境名稱 ] ] [ -q [ 佇列名稱 ] ]  
[ -slots [ 插槽數 ] ] [ -t 工作 id 範圍清單 ]  
[ -P [ 專案 ] ] [ -D [ 部門 ] ] [ -f 計算檔名稱 ]
```

說明

qacct 公用程式會掃描計算資料檔（請參閱 `accounting(5)`），並且產生資訊摘要，包含各類別主機名稱、佇列名稱、群組名稱、所有人名稱、作業名稱、作業識別碼以及電腦時鐘時間 (wall-clock time)、CPU 時間及系統時間，還有使用 `-l` 開關 (switch) 所特定之符合資源需求的佇列，也可產生各類型之間的組合資訊摘要。或者也可使用 `-j` 開關來列出所有或特定作業的清單。例如，搜尋標準可以包含一個佇列與所有人之摘要，但是在同一需求內不能針對二個佇列。

選項

表 2 中列出 qacct 之選項。

表 2 qacct 選項

選項	說明
-A 計算	摘要作業的計算。
-b 開始時間	欲摘要作業的最早啓始時間，以 <code>[[CC]YY]MMDDhhmm[.SS]</code> 之格式表示。同時參閱 -d 選項。
-d 天數	摘要天數以及列出計算資訊於其上。若是與 -b 開始時間 選項（請參閱上述說明）一起使用的話，在開始時間至開始時間+ 天數 期間內的作業將列入計算。若是與 -e 結束時間 選項（請參閱下述說明）一起使用的話，計算啓始自結束時間- 天數。
-e 結束時間	欲摘要作業的最近啓始時間，以 <code>[[CC]YY]MMDDhhmm[.SS]</code> 之格式表示。同時參閱 -d 選項。
[-f 計算檔名稱]	欲使用的計算檔。若是省略，則處理系統預設的計算檔。
-g [群組 id / 群組名稱]	在計算內包含作業所有人的數值系統群組識別碼或字母與數字組合的群組名稱。若是省略 群組 id / 群組名稱，則所有群組都將被計算在內。
-h [主機名稱]	無大小寫分別之主機名稱取決於所要求的計算資訊。若是省略名稱，則會分別列出每個主機的所有資訊。
-help	顯示 qacct 指令的說明資訊。
-history 歷程路徑	目錄路徑是歷程佇列與聚合體配置資料之所在位置，其搭配 -l 開關使用於符合資源需求的配對。若是沒有設定後者，則忽略此選項。
-j [[作業名稱 / 作業 id]]	執行列出計算資訊之作業名稱或識別碼。若皆未給予名稱與識別碼時，則會列出所有的作業。此選項變更了 qacct 的輸出格式。若是啓動它，CPU 時間將不再累計，而以列出「原始」的計算資訊於格式化的表格內來替代。請參閱 <code>accounting(5)</code> 以得知顯示資訊的說明。
-l 屬性 = 數值 ...	計算作業正在其中執行的佇列必須符合資源需求規格。該配對是使用歷程資料來執行，換言之，其反映了作業啓始當時，佇列以及聚合體配置的情況。資源需求與 <code>qsub(1)</code> 內的說明相當類似。主要的差別在於它不包含在被使用的歷程配置資料之內，所以並不是每個變更負載資訊都會提出需求。

表 2 qacct 選項 (接上頁)

選項	說明
-nohist	只有與 -l 選項共同使用時才有用。這會強制 qacct 於資源需求配對時不使用歷程佇列以及聚合體配置資料，而從 sge_qmaster(8) 擷取實際佇列與聚合體配置來替代。請注意這可能會導致混淆的統計結果，因為目前的佇列以及聚合體配置與之前作業的有效情形可能有極大的不同。也請注意所有參照於計算檔案中的主機都須開機執行以得知結果。
-o [所有人]	組合計算統計之作業所有人名稱。若是省略選擇性的「所有人」引數，則會產生目前計算檔案內所有作業所有人的計算數據 (accounting statistics) 清單。
-pe [平行環境名稱]	已摘要用法的平行環境名稱。若是未指定 平行環境名稱，則會分別列出每個平行環境的計算資料。
-q [佇列名稱]	已摘要用法的佇列名稱。若是未指定 佇列名稱，則會分別列出每個佇列的計算資料。
-slots [插槽數]	已摘要用法的佇列插槽數。若是未指定 插槽數，則分別列出每個佇列插槽數的計算資料。
-t 工作 id 範圍清單	只有與上述說明的 -j 選項共同使用時才有效。 -t 開關會指定計算資訊應該列出之陣列作業的工作範圍。工作 id 範圍清單的語法以及語義與在 qsub(1) 中說明的 -t 選項完全相同。請同時參閱上述指令說明以得知作業陣列更多資訊。
-P [專案]	已摘要用法的專案名稱。若是未指定 專案，則會分別列出每個所有人專案的計算資料。只有在執行 Sun Grid Engine，企業版模式時才使用專案。
-D [部門]	已摘要用法的部門名稱。若是未指定 部門，則會分別列出每個所有人部門的計算資料。只有在執行 Sun Grid Engine，企業版模式時才使用部門。

環境變數

表 3 說明與 `qacct` 相關聯的環境變數。

表 3 `qacct` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qacct</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中所指定的儲存格名稱，若是有設定的話。預設儲存格的名稱；例如，<code>default</code>
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，請指定特定 <code>sge_commd(8)</code> 所執行的主機，以便用於 <code>qacct</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

檔案

- Sun Grid Engine 預設計算檔案 — `<sge_root>/<儲存格>/common/accounting`
- Sun Grid Engine 預設歷程資料庫 — `<sge_root>/<儲存格>/common/history`
- Sun Grid Engine 主要主機檔案 — `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

`sge_intro(1)`、`qsub(1)`、`accounting(5)`、`sge_qmaster(8)`、`sge_commd(8)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qconf (1)

名稱

qconf — Sun Grid Engine 佇列配置

說明

qconf 允許系統管理員新增、刪除以及修改目前的 Sun Grid Engine 配置，包含佇列管理、主機管理、聚合體管理以及使用者管理。qconf 同時允許您檢查現有佇列目前的佇列配置。

選項

除非另外註明，否則所有有效帳戶的使用者皆可使用列於表 4 的選項與其對應的操作。

表 4 qconf 選項

選項	說明
-Aattr 物件規格 <i>fname</i> 物件實例,...	<i>新增物件屬性</i> — 與 -aattr 類似（請參閱下述說明），但是從名為 <i>fname</i> 的檔案內取得欲增強物件屬性的規格。與 -aattr 不同的是其可增強多個屬性。物件規格須列於 <i>fname</i> 內，而對應物件的檔案格式則列於其後（以佇列為例，請參閱 <code>queue_conf(5)</code> ）。此選項需要 root/ 管理員的特權。
-Ac 聚合體名稱 <i>fname</i>	<i>新增聚合體</i> — 以 <i>fname</i> 中的定義來新增聚合體 <i>聚合體名稱</i> 至 Sun Grid Engine 叢集中。於 <code>complex(5)</code> 中可得知聚合體規格的格式說明。需要 root 或管理員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-Acal <i>fname</i>	<i>新增行事曆</i> – 新增行事曆定義至 Sun Grid Engine 環境中。在 Sun Grid Engine 中是使用行事曆來定義佇列可用與不可用之排程表。於 <code>calendar_conf(5)</code> 中以得知行事曆定義的格式說明。行事曆定義來自於 <i>fname</i> 檔案。需要 root/ 管理員的特權。
-Ackpt <i>fname</i>	<i>新增檢查點環境</i> – 以 <i>fname</i> 中的定義來新增檢查點環境至支援的檢查點清單中 (請參閱 <code>checkpoint(5)</code>)。需要 root 或管理員的特權。
-Aconf 檔案清單	<i>新增配置</i> – 新增以逗號分隔的檔案清單中所指定的叢集配置 (請參閱 <code>sge_conf(5)</code>)。需要 root 或管理員的特權。
-Ae <i>fname</i>	<i>新增執行主機</i> – 將 <i>fname</i> 中所定義的執行主機新增至 Sun Grid Engine 叢集中。於 <code>host_conf(5)</code> 中可得知執行主機規格的格式說明。需要 root 或管理員的特權。
-Ap <i>fname</i>	<i>新增 PE 配置</i> – 將 <i>fname</i> 中所定義的平行環境 (PE) 新增至 Sun Grid Engine 叢集中。需要 root 或管理員的特權。
-Aprj <i>fname</i>	<i>新增專案</i> – 只有 Sun Grid Engine, 企業版系統支援此選項, Sun Grid Engine 系統並沒有這個選項。將 <i>fname</i> 中所定義的專案說明新增至登錄專案的清單中 (請參閱 <code>project(5)</code>)。需要 root 或管理員的特權。
-Aq <i>fname</i>	<i>新增佇列</i> – 將 <i>fname</i> 中定義的佇列新增至 Sun Grid Engine 叢集中。需要 root 或管理員的特權。
-Au <i>fname</i>	<i>新增 ACL</i> – 新增使用者存取清單 (ACL) 至 Sun Grid Engine 中。使用者清單用於佇列使用認證。需要 root/ 管理員 / 操作員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-Dattr 物件規格 <i>fname</i> 物件實例,...	刪除物件屬性 — 與 -dattr 類似 (請見下述說明), 但是要刪除項目之屬性清單的定義包含在名為 <i>fname</i> 的檔案內。與 -dattr 不同的是其可以修改多個屬性。其規格須列於 <i>fname</i> 內, 而對應物件的檔案格式則列於其後 (以佇列為例, 請參閱 queue_conf(5))。需要 root/ 管理員的特權。
-Mattr 物件規格 <i>fname</i> 物件實例,...	修改物件屬性 — 與 -mattr 類似 (請參閱下面說明), 但是從名為 <i>fname</i> 的檔案內取得欲修改物件屬性的規格。與 -mattr 不同的是其可以修改多個屬性。其規格須列於 <i>fname</i> 內, 而對應物件的檔案格式則列於其後 (以佇列為例, 請參閱 queue_conf(5))。需要 root/ 管理員的特權。
-Mc 聚合體名稱 <i>fname</i>	修改聚合體 — 以 <i>fname</i> 檔之內容取代指定的聚合體。引數檔案必須符合 complex(5) 中所指定的格式。需要 root 或管理員的特權。
-Mcal <i>fname</i>	修改行事曆 — 以 <i>fname</i> 中指定的定義來取代行事曆定義。引數檔案必須符合在 calendar_conf(5) 中說明的格式。需要 root 或管理員的特權。
-Mckpt <i>fname</i>	修改檢查點環境 — 使用 <i>fname</i> 中的定義取代現有的檢查點環境 (請參閱 checkpoint(5))。在 <i>fname</i> 中的名稱屬性必須符合現有的檢查點環境。需要 root 或管理員的特權。
-Me <i>fname</i>	修改執行主機 — 以 <i>fname</i> 中的內容取代指定主機的執行主機配置, 其必須符合在 host_conf(5) 中所定義的格式。需要 root 或管理員的特權。
-Mp <i>fname</i>	修改 PE 配置 — 與 -mp 相同 (請見下述說明), 但是此檔案 <i>fname</i> 則被視為包含變更配置, 以替代啟動編輯器來修改 PE 配置。請參考 sge_pe(5) 以得知 PE 配置格式的詳細資訊。需要 root 或管理員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-Mprj <i>fname</i>	<i>修改專案配置</i> — 與 -mprj 相同 (請見下述說明), 但是檔案 <i>fname</i> 則被視為包含變更配置, 以替代啟動編輯器來修改專案配置。請參閱 project(5) 以得知專案配置格式的詳細資訊。需要 root 或管理員的特權。
-Mq <i>fname</i>	<i>修改佇列配置</i> — 與 -mq 相同 (請見下述說明), 但是檔案 <i>fname</i> 則被視為包含變更配置, 以替代啟動編輯器來修改佇列配置。請參閱 queue_conf(5) 以得知佇列配置格式的詳細資訊。需要 root 或管理員的特權。
-Mqattr <i>fname</i> 佇列名稱, ... 此指令已停用, 改以 -Mattr 來替代。	<i>修改佇列屬性</i> — 允許以單一指令在多重佇列中變更選取的佇列配置屬性, 將套用在以逗號分隔的佇列名稱清單內的所有佇列, 以及在 <i>fname</i> 中所包含的佇列屬性定義。而沒有包含在 <i>fname</i> 中的佇列屬性將保持不變。 除了 <i>佇列名稱</i> 與 <i>佇列主機名稱</i> 之外, 所有的佇列屬性都可以修改。請參閱 queue_conf(5) 以得知佇列配置格式的詳細資訊。需要 root 或管理員的特權。
-Mu <i>fname</i>	<i>修改 ACL</i> — 以定義在 <i>fname</i> 中的使用者存取清單 (ACL) 來取代現有任何使用相同名稱的 ACL。請參閱 access_list(5) 以得知 ACL 配置格式的詳細資訊。需要 root 或管理員的特權。
-Rattr 物件規格 <i>fname</i> 物件實例, ...	<i>替換物件屬性</i> — 與 -rattr 類似 (請見下述說明), 但是欲替換內容的清單屬性定義將被包含在名為 <i>fname</i> 的檔案中。與 -rattr 不同的是其可以修改多重屬性。其規格須列於 <i>fname</i> 內, 而對應物件的檔案格式則列於其後 (以佇列為例, 請參閱 queue_conf(5))。需要 root/ 管理員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-aattr <i>obj_spec attr_name val obj_instance,...</i>	<p>新增物件屬性 – 允許以單一指令，將規格新增至物件之多個實例中的單一配置清單屬性。目前支援的物件為佇列以及主機配置，如同<i>物件規格</i>中所指定的<i>佇列</i>或<i>主機</i>。佇列的<i>load_threshold</i>參數為清單屬性之一例。項目可使用 -aattr 選項來新增至這類的清單中，同時可以使用 -dattr 選項來刪除，使用 -mattr 選項來修改，以及使用 -rattr 選項來替換。欲增強的配置屬姓名稱是使用<i>屬姓名稱</i>來指定，而數值則列於其後成為<i>名稱=數值</i>配對。在指令的結尾部分則指定變更須套用之以逗號分隔的物件實例清單（例如：佇列清單）。</p> <p>須套用下述的限制：針對<i>主機</i>物件，則無法修改<i>載入值</i>屬性（請參閱 <i>host_conf(5)</i>）。</p> <p>需要 root 或管理員的特權。</p>
-ac <i>聚合體名稱</i>	<p>新增聚合體 – 新增聚合體至 Sun Grid Engine 環境中。聚合體項目包含一項或多項資源，而提交至系統的作業可能會要求這些項目。<i>complex(5)</i> 線上說明手冊包含關於聚合體定義規格的詳細資訊。</p> <p>當使用 -ac 選項時，在指令選項中需指定聚合體名稱。然後 qconf 將會開啓一個暫存檔案並且啓動由環境變數 EDITOR 所指定的文字編輯器（若未設定 EDITOR 的話，預設編輯器則為 vi(1)）。在輸入聚合體定義並關閉編輯器後，將使用 <i>sge_qmaster(8)</i> 來檢查以及登錄新增的聚合體。需要 root/ 管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-acal 行事曆名稱	<p><i>新增行事曆</i> – 新增行事曆定義至 Sun Grid Engine 環境中。在 Sun Grid Engine 中是使用行事曆來定義佇列可用與不可用之排程表。於 calendar_conf(5) 中以得知行事曆定義的格式說明。使用在選項引數 qconf 中所指定的行事曆名稱，將會開啓一個暫存檔案並且啓動由環境變數 EDITOR 所指定的文字編輯器 (若是未設定 EDITOR 的話，預設編輯器則為 vi(1))。在輸入行事曆定義並關閉編輯器後，將使用 sge_qmaster(8) 來檢查以及登錄新增的行事曆。需要 root/ 管理員的特權。</p>
-ackpt 檢查點名稱	<p><i>新增檢查點環境</i> – 將檢查點名稱下的檢查點環境新增至 Sun Grid Engine 維護的檢查點環境清單，以及可用來提交檢查點作業 (請參閱 checkpoint(5) 以得知檢查點環境定義格式的詳細資訊)。qconf 擷取預設的檢查點環境配置並且執行 vi(1) (若有設定 EDITOR 環境變數的話，則執行 \$EDITOR) 來允許您自訂檢查點環境配置。在退出編輯器之後，將藉由 sge_qmaster(8) 來登錄檢查點環境。需要 root/ 管理員的特權。</p>
-aconf 主機...	<p><i>新增配置</i> – 依序新增叢集配置 (請參閱 sge_conf(5))，針對以逗號分隔之檔案清單內的主機。針對每個主機，都會呼叫一個編輯器 (\$EDITOR 所指定的或是 vi(1)) 然後可輸入該主機的配置。在儲存檔案並退出文字編輯器後，將使用 sge_qmaster(8) 來登錄配置。需要 root 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-ae [主機範本]	<p><i>新增執行主機</i> — 新增主機至 Sun Grid Engine 執行主機的清單中。若是配置佇列於一台主機上，則此主機會自動新增至 Sun Grid Engine 的執行主機清單中。明確地新增執行主機可提供藉由使用執行主機登錄來指定參數的優點，如負載比例值。然而，這些參數可以在之後的任何時間，透過 -me 選項來修改（依其預設值），請見下述說明。</p> <p>若是提出 <i>主機範本</i> 引數，qconf 會自 sge_qmaster(8) 中擷取指定的執行主機配置，或是自通用範本中擷取。之後範本會儲存在一檔案中，並且 qconf 會執行 vi(1)（若有設定 EDITOR 環境變數的話，則執行 \$EDITOR 所指定的文字編輯器）來變更檔案中的項目。於 host_conf(5) 中可得知執行主機規格的格式說明。當變更已在編輯器中儲存且退出編輯器時，將使用 sge_qmaster(8) 來登錄新增的執行主機。此選項需要 root/ 管理員的特權。</p>
-ah 主機名稱,...	<p><i>新增管理主機</i> — 新增主機 <i>主機名稱</i> 至 Sun Grid Engine 的信任主機清單中（若要執行 Sun Grid Engine 管理指令，主機必須列於此清單中，除了在 sge_qmaster(8) 節點上執行的 qconf 指令之外）。預設的 Sun Grid Engine 安裝程序通常會自動地將所有指定的執行主機（請參閱上方的 -ae 選項）新增至 Sun Grid Engine 的信任主機清單 (trusted host list) 中。需要 root 或管理員的特權。</p>
-am 使用者,...	<p><i>新增管理員</i> — 新增指定的使用者至 Sun Grid Engine 管理員清單中。需要 root 或管理員的特權。</p>
-ao 使用者,...	<p><i>新增操作員</i> — 新增指定的使用者至 Sun Grid Engine 操作員清單中。需要 root/ 管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-ap 平行環境名稱	<p><i>新增 PE</i> – 將平行環境名稱名稱下的平行環境 (PE) 說明新增至 Sun Grid Engine 所維護的 PE 清單中，並可供提交平行作業使用 (請參閱 <code>sge_pe(5)</code> 以得知 PE 定義格式的詳細資訊)。<code>qconf</code> 擷取預設的 PE 配置並且執行 <code>vi(1)</code> (若有設定 <code>EDITOR</code> 環境變數的話，則執行 <code>\$EDITOR</code>) 以允許您自訂 PE 配置。在退出編輯器之後，將使用 <code>sge_qmaster(8)</code> 來登錄 PE。需要 <code>root/</code> 管理員的特權。</p>
-apri	<p><i>新增專案</i> – 只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統則沒有這個選項。 新增專案說明至已登錄專案的清單中 (請參閱 <code>project(5)</code>)。<code>qconf</code> 擷取範本專案配置並且執行 <code>vi(1)</code> (若有設定 <code>EDITOR</code> 環境變數的話，就執行 <code>\$EDITOR</code>) 以允許您自訂新增的專案。在退出編輯器之後，將使用 <code>sge_qmaster(8)</code> 來登錄範本。需要 <code>root</code> 或管理員的特權。</p>
-aq [佇列範本]	<p><i>新增佇列</i> – <code>qconf</code> 會擷取預設的佇列配置 (請參閱 <code>queue_conf(5)</code>)，或是佇列<i>佇列配置</i>的配置 (若有提出選擇性的引數) 並且執行 <code>vi(1)</code> (若有設定 <code>EDITOR</code> 環境變數的話，則執行 <code>\$EDITOR</code>) 以允許您自訂佇列配置。在退出編輯器之後，將使用 <code>sge_qmaster(8)</code> 來登錄佇列。最低配置需求只要設定佇列名稱以及佇列主機名稱即可。需要 <code>root</code> 或管理員的特權。</p>
-as 主機名稱,...	<p><i>新增提交主機</i> – 新增主機主機名稱至只允許提交 Sun Grid Engine 作業以及控制其運作方式的主機清單中。需要 <code>root</code> 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-astnode 節點路徑=共享,...	<p>新增共享樹節點—只有 Sun Grid Engine，企業版系統支援此選項。Sun Grid Engine 系統則沒有這個選項。新增指定的共享樹節點至共享樹（請參閱 share_tree(5)）。節點路徑是一階層式的路徑 ([/] 節點名稱 [[/.] 節點名稱...]) 用來指定共享樹中新增結點的位置。節點路徑的基本名稱即為新增節點的名稱。該節點將初始化為指定共享的數目。需要 root 或管理員的特權。</p>
-astree	<p>新增共享樹—只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統則沒有這個選項。新增共享樹的定義至系統中（請參閱 share_tree(5)）。擷取共享樹範本並且啟動編輯器（為 vi(1) 或是 \$EDITOR 所指定的編輯器）來修改共享樹的定義。在退出編輯器之後，將使用 sge_qmaster(8) 來登錄修改之資料。需要 root 或管理員的特權。</p>
-au 使用者,...acl 名稱,...	<p>新增使用者至 ACL—新增使用者至 Sun Grid Engine 使用者存取清單 (ACL)。使用者清單用於佇列使用認證。需要 root/ 管理員 / 操作員的特權。</p>
-Auser fname	<p>新增使用者—（只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統則沒有這個選項。）將定義在 fname 中的使用者新增至 Sun Grid Engine，企業版叢集內。於 user(5) 中可得知使用者規格格式的說明。需要 root 或管理員的特權。</p>
-auser	<p>新增使用者—只有 Sun Grid Engine，企業版系統支援此選項。Sun Grid Engine 系統則沒有這個選項。新增使用者至已登錄的使用者清單中（請參閱 user(5)）。此指令會啟動一個編輯器（為 vi(1) 或是 EDITOR 環境變數所指定的編輯器）來編輯範本使用者。在變更項目以及退出編輯器之後將登錄新增的使用者。需要 root 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-clearusage	清除共享樹的使用 — (只有 Sun Grid Engine, 企業版系統支援此選項。Sun Grid Engine 系統則沒有這個選項。) 自共享樹清除所有的使用者以及專案使用記錄。所有的使用記錄都將初始化為零。
-cq 佇列名稱,...	清除佇列 — 自未完成的作業中清除佇列。主要為一開發工具。需要 root/ 管理員 / 操作員的特權。
-dattr obj_spec attr_name val obj_instance,...	刪除物件中的屬性 — 允許以單一指令, 刪除物件多個實例中的單一配置清單屬性之規格。目前支援的物件為佇列以及主機配置, 如同物件規格中所指定的佇列或主機。佇列的載入臨界值參數為清單屬性之一例。可使用 -aattr 選項來新增項目至這類的清單中, 同時可以使用 -dattr 選項來刪除, 使用 -mattr 選項來修改, 以及使用 -rattr 選項來替換項目。欲修改的配置屬性名稱是以屬性名稱來指定, 而屬性值則列於其後用來定義欲刪除的屬性清單項目的名稱。在指令的結尾部分則指定變更須套用之以逗號分隔的物件實例清單 (例如: 佇列清單)。須套用下述的限制: 針對主機物件, 則無法修改載入值屬性 (請參閱 host_conf(5))。需要 root 或管理員的特權。
-dc 聚合體名稱,...	刪除聚合體 — 自 Sun Grid Engine 中刪除聚合體。此選項需要 root/ 管理員 / 操作員的特權。
-dcal 行事曆名稱,...	刪除行事曆 — 自 Sun Grid Engine 中刪除指定的行事曆定義。需要 root/ 管理員的特權。
-dckpt 檢查點名稱	刪除檢查點環境 — 刪除指定的檢查點環境。需要 root/ 管理員的特權。
-dconf 主機,...	刪除配置 — 從配置清單中刪除指定主機的配置項目。需要 root 或管理員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-de 主機名稱,...	刪除執行主機 — 從 Sun Grid Engine 執行主機清單中刪除主機。需要 root/ 管理員的特權。
-dh 主機名稱,...	刪除管理主機 — 從 Sun Grid Engine 信任的主機清單中刪除主機。無法從管理主機清單中移除正在執行 sge_qmaster(8) 的主機。這個選項需要 root/ 管理員的特權。
-dm 使用者[, 使用者...]	刪除管理員 — 管理員清單中刪除管理員。需要 root/ 管理員的特權。
-do 使用者[, 使用者...]	刪除操作員 — 從操作員清單中刪除操作員。需要 root/ 管理員的特權。
-dp 平行環境名稱	刪除平行環境 — 刪除指定的平行環境 (PE)。需要 root/ 管理員的特權。
-dprj 專案,...	刪除專案 — 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統則沒有這個選項。刪除指定的專案。需要 root/ 管理員的特權。
-dq 佇列名稱,...	刪除佇列 — 移除指定的佇列。將允許作用中的作業執行至完成。需要 root/ 管理員的特權。
-ds 主機名稱,...	刪除提交主機 — 從 Sun Grid Engine 提交主機清單中移除主機。需要 root/ 管理員的特權。
-dstnode 節點路徑,...	刪除共享樹節點 — 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。刪除指定的共享樹節點。該節點路徑為階層式的路徑 ([/] 節點名稱 [/.] 節點名稱...]) 以指定在共享樹中要刪除的節點位置。需要 root 或管理員的特權。
-dstree	刪除共享樹 — 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。刪除目前的共享樹。需要 root 或管理員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-du 使用者,... acl 名稱,...	從 ACL 刪除使用者 – 從一個或更多的 Sun Grid Engine 使用者存取清單 (ACL) 中刪除一個或更多的使用者。需要 root/ 管理員 / 操作員的特權。
-dul acl 名稱,...	刪除使用者清單 – 從系統中刪除一個或多個使用者清單。需要 root/ 管理員 / 操作員的特權。
-duser 使用者,...	刪除使用者 – 只有 Sun Grid Engine, 企業版系統支援這個選項, Sun Grid Engine 系統並沒有這個選項。從登錄的使用者清單中刪除指定的使用者。需要 root 或管理員的特權。
-help	列出所有選項清單。
-k{m s e[j] { 主機,... all}}	關閉 Sun Grid Engine 軟體 – 用以關閉 Sun Grid Engine 元件 (常駐程式)。在表格中 -km sge_qmaster(8) 會強制在控制模式中終止。以相同的方式 -ks 開關 (switch) 會造成 sge_schedd(8) 的終止。藉由 -ke 選項, 將會初始化目前已登錄, 並且正在執行之 sge_execd(8) 處理程序的關閉動作。若是指定 -kej 來代替, 所有正在執行主機上執行的作業將會在相對應的 sge_execd(8) 終止之前中斷。以逗號分隔的主機清單會指定 -ke 以及 -kej 選項所使用的執行主機。若是指定此關鍵字, all, 以替代主機清單, 將會關閉所有正在執行的 sge_execd(8) 處理程序。需要 root 或管理員的特權。
-kec {id,... all}	清除事件客戶端 – 用來關閉 sge_qmaster(8) 中所登錄的事件客戶端。以逗號分隔的事件客戶端清單會指定 -kec 選項所使用的事件客戶端。若是指定此關鍵字, all, 以替代事件客戶端清單, 則將會終止所有正在執行的事件客戶端, 除了諸如 sge_schedd(8) 的特殊用戶端之外。需要 root 或管理員的特權。

表 4 qconf 選項 (接上頁)

選項	說明
-mattr <i>obj_spec attr_name val obj_instance,...</i>	<p><i>修改物件屬性</i> – 允許使用單一指令，變更物件之多重實例中的單一配置屬性。目前支援的物件為佇列以及主機配置，如同<i>佇列</i>所指定的佇列，或是<i>物件規格</i>中所指定的主機。請注意：-mattr <i>queue attr_name val q_name, ...</i> 相同於 -mqattr <i>attr_name val q_name,...</i> (請參閱下述說明)。針對回溯相容性，有較新的版本可以使用。</p> <p>將使用該屬性設定值之後的<i>屬姓名稱</i>來指定欲修改的配置屬姓名稱。若是該屬性為清單，像是佇列<i>載入臨界值</i>，則<i>數值</i>可以是一對<i>名稱 = 數值</i>，且不論是那一種情況下只會變更清單中相對應的項目。請參閱 -aattr、-dattr、以及 -rattr 選項以得知進一步變更方法的說明，特別像是清單屬性。</p> <p>在指令的結尾部分則指定變更須套用之以逗號分隔的物件實例清單 (例如：佇列清單)。</p> <p>須套用下述的限制：針對佇列物件，則無法修改<i>佇列名稱</i>以及<i>佇列主機名稱</i>屬性 (請參閱 <code>queue_conf(5)</code>)。針對主機物件，則不能修改<i>主機名稱</i>、<i>載入值</i>、以及<i>處理器</i>屬性 (請參閱 <code>host_conf(5)</code>)。</p> <p>需要 root 或管理員的特權。</p>
-mc <i>聚合體名稱</i>	<p><i>修改聚合體</i> – 擷取指定的聚合體選項 (請參閱 <code>complex(5)</code>)、執行一個編輯程式 (為 <code>vi(1)</code> 或是 <code>\$EDITOR</code> 所指定的編輯程式)、並且使用 <code>sge_qmaster(8)</code> 在退出編輯程式後登錄變更的聚合體配置。需要 root 或管理員的特權。</p>
-mcal <i>行事曆名稱</i>	<p><i>修改行事曆</i> – 擷取指定的行事曆定義 (請參閱 <code>calendar_conf(5)</code>)、執行一個編輯程式 (為 <code>vi(1)</code> 或是 <code>\$EDITOR</code> 所指定的編輯程式)、並且使用 <code>sge_qmaster(8)</code> 在退出編輯程式後登錄變更的行事曆定義。需要 root 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-mckpt 檢查點名稱	<p><i>修改檢查點環境</i> – 擷取指定檢查點環境目前的配置、執行一個編輯程式 (為 vi(1) 或是 \$EDITOR 環境變數所指定的編輯程式)、並且使用 sge_qmaster(8) 登錄新增的配置。請參閱 checkpoint(5) 以得知檢查點環境配置格式的詳細資訊。需要 root 或管理員的特權。</p>
-mconf [主機,... 全域]	<p><i>變更配置</i> – 擷取指定主機的配置、執行一個編輯程式 (為 vi(1) 或是 \$EDITOR 所指定的編輯程式)、並且使用 sge_qmaster(8) 在退出編輯程式後登錄變更的配置。若是忽略選擇性的主機引數, 或是指定此特殊的主機名稱, global, 便會修改儲存格的全域配置。在 sge_conf(5) 中有說明主機配置的格式。需要 root 或管理員的特權。</p>
-me 主機名稱	<p><i>修改執行主機</i> – 擷取指定執行主機目前的配置、執行編輯程式 (為 vi(1) 或是 \$EDITOR 環境變數所指定的編輯程式)、並且使用 sge_qmaster(8) 在退出編輯程式後登錄變更的配置。在 host_conf(5) 中有說明執行主機配置的格式。需要 root 或管理員的特權。</p>
-mp 平行環境名稱	<p><i>修改平行環境配置</i> – 針對指定的平行環境 (PE) 擷取目前的配置、執行編輯程式 (為 vi(1) 或是 EDITOR 環境變數所指定的編輯程式)、並且使用 sge_qmaster(8) 登錄新增的配置。請參閱 sge_pe(5) 以得知平行環境 (PE) 配置格式的詳細資訊。需要 root 或管理員的特權。</p>
-mprj 專案	<p><i>修改專案</i> – 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。擷取特定專案的資料 (請參閱 project(5)), 並且啟動一個編輯程式 (為 vi(1) 或是 \$EDITOR 所指定的編輯程式) 以修改專案定義。在退出編輯程式後, 會登錄變更的資料。需要 root 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-mq 佇列名稱	<p><i>修改佇列配置</i> – 擷取指定佇列目前的配置、執行編輯程式 (為 vi(1) 或是 EDITOR 環境變數所指定的編輯程式)、並且使用 sge_qmaster(8) 登錄新增的配置。請參閱 queue_conf(5) 以得知佇列配置格式的詳細資訊。需要 root 或管理員的特權。</p>
-mqattr attr_name val q_name,... 此選項已停用，請使用 -matr 來替代。	<p><i>修改佇列屬性</i> – 允許使用單一指令，變更多重佇列中的單一佇列配置屬性。在以逗號分隔的佇列名稱清單中，其所包含之全部佇列的屬性名稱屬性值將會被數值覆寫。 所有佇列的屬性皆可被修改，除了佇列名稱與佇列主機名稱之外。請參閱 queue_conf(5) 以得知佇列配置格式的詳細資訊。需要 root 或管理員的特權。</p>
-msconf	<p><i>修改排程器配置</i> – 擷取目前的排程器配置 (請參閱 sched_conf(5))、執行編輯程式 (為 vi(1) 或是 \$EDITOR 所指定的編輯程式)、並且使用 sge_qmaster(8) 在退出編輯程式後登錄變更的配置。需要 root 或管理員的特權。 <></p>
-mstnode 節點路徑 = 共享....	<p><i>修改共享樹節點</i> – 只有 Sun Grid Engine，企業版系統才支援這個選項，Sun Grid Engine 系統並沒有這個選項。修改共享樹中指定的節點 (請參閱 share_tree(5))。該節點路徑為階層式的路徑 ([/] 節點名稱 [/.] 節點名稱 ...) 以指定在共享樹中現有節點的位置。該節點被設定為指定共享的數目。需要 root 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-mstree	<p><i>修改共享樹</i> – 只有 Sun Grid Engine，企業版系統才支援這個選項。Sun Grid Engine 系統並沒有這個選項。</p> <p>修改共享樹的定義 (請參閱 <code>share_tree(5)</code>)。擷取目前的共享樹並啟動編輯程式 (為 <code>vi(1)</code> 或是 <code>\$EDITOR</code> 所指定的編輯程式) 以修改共享樹的定義。在退出編輯程式後，將使用 <code>sge_qmaster(8)</code> 登錄修改的資料。需要 root 或管理員的特權。</p>
-mu <i>存取清單名稱</i>	<p><i>修改使用者存取清單</i> – 擷取指定使用者存取清單目前的配置、執行一個編輯程式 (為 <code>vi(1)</code> 或是 <code>EDITOR</code> 環境變數所指定的編輯程式)、並且使用 <code>sge_qmaster(8)</code> 以登錄新增的配置。需要 root 或管理員的特權。</p>
-ratrr <i>obj_spec attr_name val obj_instance...</i>	<p><i>置換物件屬性</i> – 允許使用單一指令，置換物件多重實例中的單一配置清單屬性。目前支援的物件為佇列以及主機配置，如同 <i>佇列</i> 或是 <i>物件規格</i> 中的 <i>主機</i>。該佇列載入 <i>臨界值</i> 參數為清單屬性的範例。使用 <code>-ratrr</code> 選項就可置換這類清單，當使用 <code>-aatrr</code> 時可以在其中新增項目，當使用 <code>-datrr</code> 則可在其中刪除項目，而當使用 <code>-matrr</code> 時可以在其中修改項目。</p> <p>將使用 <i>屬性名稱</i> 指定欲修改的配置屬性名稱，其後跟隨的 <i>數值</i> 定義該屬性的新設定。在指令的結尾部分則指定變更須套用以逗號分隔的物件實例清單 (例如：<i>佇列清單</i>)。</p> <p>須套用下述的限制：針對 <i>主機</i> 物件，無法修改 <i>載入值</i> 屬性 (請參閱 <code>host_conf(5)</code>)。</p> <p>需要 root 或管理員的特權。</p>
-Muser <i>fname</i>	<p><i>修改使用者</i> – 只有 Sun Grid Engine，企業版系統才支援這個選項，Sun Grid Engine 系統並沒有這個選項。</p> <p>修改 Sun Grid Engine，企業版叢集中 <code>fname</code> 內所定義的使用者。在 <code>user(5)</code> 中有說明使用者規格的格式。需要 root 或管理員的特權。</p>

表 4 qconf 選項 (接上頁)

選項	說明
-muser 使用者	修改使用者 – 只有 Sun Grid Engine，企業版系統才支援這個選項，Sun Grid Engine 系統並沒有這個選項。擷取特定使用者的資料 (請參閱 user(5))，並且啟動編輯程式 (為 vi(1) 或是 EDITOR 環境變數所指定的編輯程式) 以修改使用者定義。在退出編輯程式後，會登錄變更的資料。需要 root 或管理員的特權。
-sc 聚合體名稱,...	顯示聚合體 – 顯示一個或多個聚合體的配置。
-scal 行事曆名稱	顯示行事曆 – 顯示指定行事曆的配置。
-scall	顯示行事曆清單 – 顯示所有目前定義的行事曆清單。
-scl	顯示聚合體清單名稱 – 顯示所有目前配置的聚合體清單。
-sckpt 檢查點名稱	顯示檢查點環境 – 顯示指定檢查點環境的配置。
-sckptl	顯示檢查點環境清單 – 顯示所有目前配置的檢查點環境名稱清單。
-sconf [主機, ... 全域]	顯示配置 – 列出正在全域執行，或是指定主機上執行之叢集配置。若是忽略選擇性以逗號分隔之主機清單引數，或是指定特殊的字串 global，便會顯示全域儲存格配置。針對清單中任何其他的主機名稱，便會顯示全域配置與主機特定配置的合併程式。在 sge_conf(5) 中有說明主機配置的格式。
-sconf1	顯示配置清單 – 顯示可用配置的主機清單。特殊主機名稱 global 代表儲存格全域配置。
-se 主機名稱	顯示執行主機 – 顯示指定執行主機的定義。
-secl	顯示事件客戶端 – 顯示 Sun Grid Engine 事件客戶端清單。
-sel	顯示執行主機 – 顯示 Sun Grid Engine 執行主機清單。

表 4 qconf 選項 (接上頁)

選項	說明
-sep	顯示得知授權的處理器 (<i>licensed processors</i>) – 顯示每一執行主機得知授權的處理器數量與其總和的清單。
-sh	顯示管理主機 – 顯示 Sun Grid Engine 管理主機清單。
-sm	顯示管理員 – 顯示管理員清單。
-so	顯示操作員 – 顯示操作員清單。
-sp 平行環境名稱	顯示平行環境 (PE) 配置 – 顯示引數所指定之平行環境 (PE) 的定義。
-spl	顯示平行環境 (PE) 清單 – 顯示目前所有定義的平行環境 (PE) 清單。
-sprj 專案	顯示專案 – 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。顯示指定專案的定義 (請參閱 <code>project(5)</code>)。
-sprjl	顯示專案清單 – 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。顯示目前所有定義的專案清單。
-sq 佇列名稱 [, 佇列名稱 ...]	顯示佇列 – 顯示一個或多重佇列。
-sql	顯示佇列清單 – 顯示目前所有定義的佇列清單。
-ss	顯示提交主機 – 顯示 Sun Grid Engine 提交主機清單。
-ssconf	顯示排程器配置 – 以 <code>sched_conf(5)</code> 中說明的格式來顯示目前的排程器配置。
-sstnode 節點路徑 ...	顯示共享樹節點 – 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。顯示指定共享樹節點的名稱與共享數量 (請參閱 <code>share_tree(5)</code>)。該節點路徑為階層式的路徑 (<code>[/] 節點名稱 [/] 節點名稱 ...]</code>) 以指定共享樹中的節點位置。

表 4 qconf 選項 (接上頁)

選項	說明
-sstree	顯示共享樹 – 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。顯示共享樹的定義 (請參閱 <code>share_tree(5)</code>)。
-sss	顯示排程器狀態 – 顯示現在 Sun Grid Engine 排程器正在執行中的主機, 若是沒有排程器正在執行就會顯示錯誤訊息。
-su 存取清單名稱	顯示使用者 ACL – 顯示 Sun Grid Engine 使用者存取控制清單 (ACL)。
-sul	顯示使用者清單 – 顯示所有目前定義的 Sun Grid Engine 使用者存取控制清單 (ACL) 名稱。
-suser 使用者,...	顯示使用者 – 只有 Sun Grid Engine, 企業版系統才支援這個選項, Sun Grid Engine 系統並沒有這個選項。顯示指定使用者定義 (請參閱 <code>user(5)</code>)。
-suserl	顯示使用者 – 只有 Sun Grid Engine, 企業版系統才支援這個選項。Sun Grid Engine 系統並沒有這個選項。顯示所有目前定義的使用者清單。
-tsm	啟動排程器監控 – 該 Sun Grid Engine 排程器 <code>sge_schedd(8)</code> 會經由此選項強制將其下一個排程執行的作業追蹤訊息列至檔案 <code><sge_root>/<儲存格>/common/schedd_runlog</code> 。該訊息會指出在執行作業中, 尚未選取作業或佇列的原因。需要 root 或管理員的特權。

注意 – 使用在 `qstat(1)`-F 選項中說明的格式, 來顯示有關佇列資源可用性的作業需求之所以無效的原因 (請參閱 `qstat(1)` 線上說明手冊「輸出格式」章節中「完整格式」之說明。)

環境變數

表 5 中說明與 `qconf` 相關之環境變數。

表 5 `qconf` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若設定的話，請指定預設的 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qconf</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中所指定的儲存格名稱，若是有設定的話。預設儲存格的名稱；例如，<code>default</code>
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，則指定特定 <code>sge_commd(8)</code> 所執行的主機，以便用於 <code>qconf</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設僅可使用本端主機。

限制

修改佇列配置並不會影響執行中的佇列，而是在下一次執行佇列時才會產生影響（例如，下一個作業）。

檔案

- Sun Grid Engine 主要主機檔案 – `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

`sge_intro(1)`、`qstat(1)`、`checkpoint(5)`、`complex(5)`、`sge_conf(5)`、`host_conf(5)`、`sge_pe(5)`、`queue_conf(5)`、`sge_execd(8)`、`sge_qmaster(8)`、`sge_schedd(8)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qdel(1)

名稱

`qdel` — 自佇列中刪除 Sun Grid Engine 作業。

語法

```
qdel [ -f ] [ -help ] [ -verify ] [ 作業 / 工作id 清單 ]
```

```
qdel [ -f ] [ -help ] [ -verify ] -u 使用者清單 | -uall
```

說明

`Qdel` 提供給使用者、操作員或管理員一個可以刪除一或多項作業的方法。`Qdel` 依照作業識別碼提出的順序來刪除作業。

選項

表 6 中列出 `qdel` 之選項。

表 6 `qdel` 選項

選項	說明
<code>-f</code>	<p>對執行中的作業採取強制動作。作業會從登錄在 <code>sge_qmaster(8)</code> 的作業清單中刪除，即使控制作業的 <code>sge_execd(8)</code> 並未回應由 <code>sge_qmaster(8)</code> 所送出的刪除要求。</p> <p>若是叢集配置項目 <i>佇列主要參數</i> 包含 <code>ENABLE_FORCED_QDEL</code> 旗號（請參閱 <code>sge_conf(5)</code>）時，不是 Sun Grid Engine 管理員或操作員的使用者就只能使用 <code>-f</code> 選項（針對自己擁有的作業）。然而，對管理與非管理的使用者而言其運作方式並不相同。若是管理員，則作業會直接自 Sun Grid Engine 資料庫中刪除。否則，會先嘗試一般刪除；若一般刪除失敗時，才會執行強制取消的動作。</p>
<code>-help</code>	列出所有選項清單。
<code>-u 使用者名稱... -uall</code>	<p>刪除那些只在使用者名稱清單中指定的使用者所提交之作業。對管理員而言，可能會使用 <code>qdel -uall</code> 指令刪除所有使用者的所有作業。</p> <p>若是您使用 <code>-u</code> 或 <code>-uall</code> 開關 (switch)，則允許指定一個額外的 <i>作業 / 工作 id 清單</i>。</p>
<code>-verify</code>	不執行任何修改，但若是沒有提出 <code>-verify</code> 時，則只列出需完成的項目。
<code>job/ 工作 id 清單</code>	<p>以下列格式來指定：</p> <p><i>作業 id</i>[. <i>工作範圍</i>][<i>作業 id</i>[. <i>工作範圍</i>] ...]</p> <p>若提出指定在 <i>作業 id</i> 之後的 <i>工作範圍</i>，就會限制了 <code>qdel</code> 對陣列作業範圍操作的效力（若要得知作業陣列進一步的詳細資訊，請參閱 <code>qsub(1)</code> 的 <code>-t</code> 選項）。</p> <p>工作範圍指定器 (specifier) 的格式為 <i>n</i>[<i>-m</i>[:<i>s</i>]]。範圍可以是單一數字、簡單的範圍格式 <i>n-m</i>、或是一個步驟大小的範圍。</p> <p>有可能會使用此關鍵字 <code>all</code> 來替代 <i>作業 / 工作 id 清單</i> 以修改目前使用者的所有作業。</p>

環境變數

表 7 中說明與 `qdel` 相關之環境變數。

表 7 `qdel` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若設定的話，請指定預設的 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qdel</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中所指定的儲存格名稱，若是有設定的話。預設儲存格的名稱；例如，<code>default</code>
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 <code>tcp</code> 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，則指定特定 <code>sge_commd(8)</code> 所執行的主機，以使用於 <code>qdel</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

檔案

- Sun Grid Engine 主要主機檔案 — `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

`sge_intro(1)`、`qstat(1)`、`qsub(1)`、`sge_qmaster(8)`、`sge_execd(8)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qhold(1)

名稱

qhold – 暫緩 Sun Grid Engine 作業的執行。

語法

```
qhold [-h {u|o|s},...] [-help] [作業/工作 id 清單]
```

```
qhold [-h {u|o|s},...] [-help] -u 使用者清單 | -uall
```

說明

qhold 提供使用者 / 操作員 / 管理員一個工具以放置所謂的**暫停 (holds)** 至一個或多個作業中，來擱置其執行之排程。只要對作業指定任何類型的暫停，則暫停期間該作業沒有資格進入排程。

可使用 `qrls(1)` 或 `qalter(1)` 指令來移除暫停。

暫停有三種類型：

- 使用者 – 管理員、操作員與作業所有人可以指定及移除使用者暫停。
- 操作員 – 管理員與操作員可以指定及移除操作員暫停。
- 系統 – 只有管理員可以指定及移除系統暫停。

若是 `-h` 選項（請參閱下列說明）並沒有指定暫停類型，則假設預設值為使用者暫停。

指定暫停給作業有二種可選擇的方式，`qsub(1)` 或是 `qalter(1)` 指令（請參閱 `-h` 選項）。

選項

表 8 列出與 qhold 相關之選項

表 8 qhold 選項

選項	說明
-h {u o s},...	指定 u (使用者)、 o (操作員) 或 s (系統) 暫停或其組合至一項或多項作業。
-help	列出所有選項清單。
-u 使用者名稱,... -uall	只有使用者在使用者名稱清單中指定提交的那些作業會進行變更。對管理員而言，有可能使用 <code>qhold -uall</code> 指令針對所有使用者的所有作業設定暫停。 若是您使用 <code>-u</code> 或是 <code>-uall</code> 開關，則允許指定一個額外的 <i>作業/工作 id 清單</i> 。
job/ 工作 id 清單	以下列格式來指定： 作業 id[. 工作範圍], 作業 id [. 工作範圍], ...] 若提出指定在作業 id 之後的 <i>工作範圍</i> ，就會限制了 <i>qhold</i> 對陣列作業範圍操作的效力 (若要得知作業陣列進一步的詳細資訊，請參閱 <i>qsub(1)</i> 的 <code>-t</code> 選項)。 工作範圍指定器的格式為 <code>n[-m[:s]]</code> 。範圍可以是單一數字、簡單的範圍格式 <code>n-m</code> 、或是一個步驟大小的範圍。 有可能會使用關鍵字「all」替代 <i>作業/工作 id 清單</i> 以修改目前使用者的所有作業之暫停狀態。

環境變數

表 9 中說明與 qhold 相關之環境變數。

表 9 qhold 環境變數

選項	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位置。
SGE_CELL	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <i>qhold</i> 會使用 (以優先順序列出)： <ul style="list-style-type: none">• 指定於環境中的儲存格名稱 如果已設定的話，變數 <code>SGE_</code> 儲存格。• 預設儲存格的名稱；例如，<code>default</code>。

表 9 qhold 環境變數 (接上頁)

選項	說明
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 <code>tcp</code> 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
COMMD_HOST	若有設定，請指定特定 <code>sge_commd(8)</code> 所執行的主機，以使用於 <code>qhold</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

同時參閱

`sge_intro(1)`、`qalter(1)`、`qrls(1)`、`qsub(1)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qghost(1)

名稱

`qghost` – 顯示 Sun Grid Engine 主機、佇列以及作業的狀態。

語法

```
qghost [ -F [ 資源名稱,... ] [ -help ] [ -h 主機清單 ]  
[ -j ] [ -l 資源 = 數值,... ] [ -u 使用者,... ]
```

說明

qghost 顯示與佇列相關之 Sun Grid Engine 主機、佇列以及作業目前的狀態。選取選項允許您獲得關於指定主機、佇列、作業或使用者的資訊。若沒有選項，qghost 則會顯示無佇列或作業資訊的所有主機清單。

選項

表 10 列出與 qghost 相關之選項。

表 10 qghost 選項

選項	說明
-F [資源名稱,...]	qghost 會提出每台主機與所有資源相關的目前資源可用性詳細清單（若是省略選項引數的話），或是那些包含在資源名稱清單內的資源。若要得知進一步的詳細資訊，請參考下列「輸出格式」章節中「完整格式」之說明。
-help	列出所有選項清單。
-h 主機清單	列出包含在 主機清單 中所有主機的清單。
-j	列出所有在顯示主機擁有之佇列上執行的作業。此開關會間接地呼叫 -q。
-l 資源 [= 數值],...	依要求的資訊來定義主機所需求的資源。在主機上執行配對。
-q	顯示關於顯示主機擁有之佇列的資訊。
-u 使用者,...	只顯示與指定使用者清單中的使用者相關聯之作業以及佇列的資訊。

輸出格式

須依據 -q 或 -F 以及 -j 選項的存在與否，來區別三種輸出格式。

預設格式

在預設格式中 — 即指沒有 -q、-F、以及 -j 選項 — 於標題行之後會針對每台主機列出一行資訊，內容包括：

- 主機名稱
- 架構
- 處理器數量

- 負載
- 總記憶體
- 已使用的記憶體
- 總置換空間
- 已使用的置換空間

有選項的格式

若是您提供 `-q` 選項，每台主機的狀態行也包含該主機所擁有的每一個佇列的額外行資訊，包括：

- 佇列名稱
- 佇列類型 — 為 **B**（批次）、**I**（互動式）、**C**（檢查點）、**P**（平行）與 **T**（傳輸）其中之一，或是這些類型的組合
- 已使用以及可使用的作業插槽數量
- 佇列的狀態 — 為 **u**（不明的）意指若無法與對應的 `sge_execd(8)` 取得聯繫、**a**（警報）、**A**（警報）、**C**（暫停行事曆）、**s**（暫停）、**S**（次級）、**d**（停用）、**D**（停用）與 **E**（錯誤）其中之一，或是這些狀態的組合

若是狀態為 **a**（警報），表示定義於佇列配置（請參閱 `queue_conf(5)`）的載入臨界值清單中至少有一個載入臨界值目前已超過設定，這可以制止進一步的作業排程至該佇列中。

與此狀態不同的是，**A**（警報）狀態指出至少有一個佇列（請參閱 `queue_conf(5)`）的暫停臨界值目前已超過設定。這會導致在該佇列執行的作業會依序地暫停，直到沒有臨界值違反設定。

可以透過 `qmod(1)` 指令來指定與釋放佇列的 **s**（暫停）狀態以及 **d**（停用）狀態。暫停一個佇列會導致在該佇列執行的所有作業皆暫停。

D（停用）狀態以及 **C**（暫停行事曆）狀態指出透過 **Sun Grid Engine** 的行事曆工具（請參閱 `calendar_conf(5)`）使佇列自動地停用或暫停，而 **S**（次級）狀態指出該佇列已經成為另一個佇列的次級而被暫停（若要得知詳細資訊，請參閱 `queue_conf(5)`）。當佇列暫停時（無論原因為何），所有在該佇列執行的作業也將一併暫停。

若是佇列顯示 **E**（錯誤）狀態，表示該主機的 `sge_execd(8)` 無法在該主機上定位 `sge_shepherd(8)` 執行檔案以便啓動作業。請檢查該 `sge_execd(8)` 的錯誤日誌檔案，找出如何解決這個問題的線索。之後請透過 `qmod(1)` 指令的 `-c` 選項來手動啓動佇列。

若是使用 `-F` 選項，在主機狀態行之後將列出資源可用性資訊。針對每一項資源（依照在 `-F` 選項引數中所選取的資源，或若是省略選項引數時，則指所有的資源）會以下列格式顯示於單一資料行中：

- 一個字母之指定器 (**specifier**) 會指定目前資源可用性數值是由下述二個之中的哪一個所掌控：

- g – 叢集全域
- h – 主機總和；或是
- 第二種，一個字母之指定器會指定目前資源可用性數值是屬於下述二項何者之一：
 - l – 報告資源的載入值，
 - L – 在套用管理員所定義之負載比例後的資源載入值，
 - c – 衍生自可消耗資源工具的可用性（請參閱 `complexes(5)`）
 - v – 預設的聚合體配置值，絕不會被載入報告或是可消耗更新資料所覆寫
 - f – 固定的可用性定義，衍生自非消耗的聚合體屬性或是固定的資源限制
- 在冒號之後，會顯示資源名稱的資訊
- 在等號之後則是目前資源可用性數值

顯示的可用性數值以及其所衍生之來源，永遠都是所有可能組合裡的最小值。因此，舉例來說，一行的格式 "`qf:h_vmem=4G`" 意指一個佇列目前在虛擬記憶體中有 4 GB（十億位元組）的最大可用性，而此值是一個固定值（例如，在佇列配置中的資源限制），且是由佇列掌控，換言之，此主機的總數或許有比此值更多的虛擬記憶體可使用，但是佇列不允許使用更多的資源。相反的，一行 "`hl:h_vmem=4G`" 也是意指上限為 4 GB（十億位元組）的虛擬記憶體可用性上限，但是限制會是從目前主機報告的載入值所衍生出來。所以當佇列可能會允許有較高虛擬記憶體需求的作業，此特定佇列所常駐的主機目前也只有 4 GB 可用。

在佇列狀態行之後（若使用 `-j`），則針對目前在這個佇列執行的每個作業列出單一資料行。每個作業狀態行包含：

- 作業 id
- 作業名稱
- 作業所有人名稱
- 作業狀態 – 單一的 `t`（傳輸）、`r`（執行）、`R`（重新啟動）、`s`（暫停）、`S`（暫停）或 `T`（臨界值）（請參閱第 63 頁的「精簡格式（不使用 `-f` 以及 `-F`）」以得知詳細資訊）
- 啟動日期與時間以及作業的功能（「主控者 (MASTER)」或「從屬者 (SLAVE)」 – 若是平行作業才有意義）
- 作業的優先順序

環境變數

表 11 中說明與 qhost 相關之環境變數。

表 11 qhost 環境變數

變數名稱	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位置。
SGE_CELL	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <i>qstat</i> 會使用（以優先順序列出）： <ul style="list-style-type: none">• 指定於環境中的儲存格名稱 如果已設定的話，變數 SGE_CELL。• 預設儲存格名稱，例如 default。
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 sge_commd(8) 的 tcp 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
COMMD_HOST	若有設定，請指定特定 sge_commd(8) 所執行的主機，以使用於 qhost 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

檔案

- Sun Grid Engine 主要主機檔案 — `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

sge_intro(1)、qalter(1)、qconf(1)、qhold(1)、qmod(1)、qstat(1)、qsub(1)、queue_conf(5)、sge_commd(8)、sge_execd(8)、sge_qmaster(8)、sge_shepherd(8)

版權

請見 sge_intro(1) 以得知權利與權限的完整聲明。

qmake (1)

名稱

qmake — 由 Sun Grid Engine 排程的分散式平行 make。

語法

qmake [選項] -- [gmake 選項]

說明

qmake 是一平行、分散式的 make(1) 公用程式。藉由 Sun Grid Engine 軟體來執行平行 make 工作的排程。其是以 gmake (GNU make)，版本 3.78.1 為架構。Sun Grid Engine 以及 gmake 兩者皆可指定其指令行選項 而選項是以 -- 字元來分隔。

qmake 可以指定在 qsub(1) 或 qrsh(1) 中所有有效的 Sun Grid Engine 選項。請參閱 submit(1) 以得知所有 Sun Grid Engine 指令行選項的說明。make(1) 線上說明手冊中說明 gmake 的指令行語法。

qmake 的語法編譯檔案 (makefile) 與 gmake 相對應，且其說明於 《GNU Make Manual》中。

若是沒有在使用者的 .cshrc 或是 .profile shell 的資源檔內設置 Sun Grid Engine 環境、並且於平行環境中要求插槽的狀況下，則典型的 qmake 呼叫會使用 Sun Grid Engine 指令行選項 -cwd，在執行主機目前的作業目錄 -v 路徑下啓動已排程的 make (請參閱 sge_pe(5))。

若是沒有指定資源要求 (QS_NAME 指令行選項 -l)，則 qmake 會使用 ARCH 環境變數要求與提交主機相同的工作執行架構。若設定 ARCH，則會藉由插入選項 -l arch=\$ARCH 至指令行選項中，來要求 ARCH 中指定的架構。若未設定 ARCH，make 工作可以在任何可用的架構中執行。然而這對典型的 make (編譯) 作業而言是具危險性的，系統會輸出警告訊息。

範例

下列的範例，程式範例 1，說明在與提交主機相同的架構下，於 `compiling` 平行環境中會要求一到十之間的插槽。make 工作會繼承完整的呼叫 shell 環境。它將會執行由 Sun Grid Engine 軟體所授予的與插槽數一樣多的平行工作。

```
qmake -cwd -v PATH -pe compiling 1-10 --
```

程式範例 1

下列的範例，程式範例 2，說明在與提交主機相同的架構下，於 `make` 平行環境中要求一到四之間的插槽。

```
qmake -cwd -v PATH -- -j 4
```

程式範例 2

下列的範例，程式範例 3，要求在 `solaris` 架構的主機上執行三個平行 `make` 工作。此提交可以執行在任何架構的主機上。

```
qmake -cwd -v PATH -l arch=solaris -pe make 3
```

程式範例 3

下列的 shell script 是程式範例 4。

```
#!/bin/sh
qmake -inherit --
```

程式範例 4

您可以使用程式範例 5 的樣式來提交程式範例 4 的 shell script。

```
qsub -cwd -v PATH -pe make 1-10 [進一步的選項] <script>
```

程式範例 5

在程式範例 5 中，`qmake` 會繼承在平行環境下給予 `make` 所提交作業的資源。

環境變數

表 12 中說明與 `qmake` 相關之環境變數。

表 12 `qmake` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若有設定，則指定預設之 Sun Grid Engine 儲存格。關於 Sun Grid Engine 儲存格， <code>qmake</code> 會使用（依先後順序排列）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中指定的儲存格名稱，若是有設定的話預設儲存格的名稱；例如，<code>default</code>
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>ARCH</code>	提交主機的架構。若有設定此變數在提交環境中，則 <code>qmake</code> 會為執行作業要求所給與的架構（請參閱上述說明）。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 <code>tcp</code> 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，請指定特定 <code>sge_commd(8)</code> 所執行的主機，以便用於 <code>qmake</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

已知問題

下列是與 `qmake` 指令相關之已知問題。

使 NFS 伺服器效率變慢

極低的檔案伺服器效能可能會導致其次級檔案的問題。

例如：a 主機會編譯 a.c. 檔案成 a.o. 檔案，而 b 主機會編譯 b.c. 檔案成 b.o. 檔案，而 c 主機會從 a.o. 與 b.o. 檔案連結成 c 程式。若在極差的檔案伺服器效能的情形下，c 主機可能無法看到 a.o. 以及 b.o. 檔。

一項規則中含多個指令

若在一項規則中執行多個指令，編譯檔 (makefile) 必須保證這些指令是以一個指令行來處理。

例如，假設您輸入下列敘述：

```
libx.a:  
  cd x  
  ar ru libx.a x.o
```

在上述的例子中，若是指令以平行方式來執行（且可能在不同的主機上），則 `libx.a` 會建立失敗。寫入下列敘述來替代。

```
libx.a:  
  cd x ; ar ru libx.a x.o
```

另一個有效率的選擇是寫入下列敘述。

```
libx.a:  
  cd x ; \  
  ar ru libx.a x.o
```

同時參閱

`submit(1)`、`sge_pe(5)`、以及與 `make(1)`（GNU make manpage）相同而置於 `<sge_root>/3rd_party/qmake` 中的 *《The GNU Make Manual》*

版權

`qmake` 包含部分的 Gnu Make（`qmake`），其版權為 Free Software Foundation, Inc., Boston, MA, USA 所有，並且受到 Gnu General Public License 的保護。

若要得知進一步的權利與權限的聲明，請參閱 `sge_intro(1)` 以及在 `<sge_root>/3rd_party/qmake` 內所提供之資訊。

qmod(1)

名稱

qmod – 修改 Sun Grid Engine 佇列

語法

qmod [選項] [作業 / 工作 id 清單 | 佇列清單]

說明

qmod 允許將使用者分類為工作站的 *所有人*（請參閱 `queue_conf(5)` 以得知詳細資訊）以便在他 / 她的機器上修改 Sun Grid Engine 佇列的狀態，也可以修改他 / 她所擁有作業的狀態。管理員 / 操作員或是 `root` 可以針對叢集中的任何佇列及作業執行 `qmod` 指令。

選項

表 13 列出與 `qmod` 相關之選項。

表 13 qmod 選項

選項	說明
-c	清除指定佇列的錯誤狀態。
-d	停用佇列，換言之，沒有進一步的作業被分派至停用的佇列中，然而已經在該佇列執行的作業則允許其執行至結束。此選項是 Sun Grid Engine 第三版 <code>-soc</code> 選項的後繼選項。
-e	啓動佇列 此選項是 Sun Grid Engine 第三版 <code>-xsoc</code> 選項的後繼選項。

表 13 qmod 選項 (接上頁)

選項	說明
-f	針對佇列強制修改動作，不管佇列目前顯示的狀態為何。例如，若是一佇列顯示處於暫停狀態，但是其作業似乎在繼續執行，此時管理員 / 操作員可以強制暫停作業，此動作會送一個 SIGSTOP 訊息給作業。在任何狀況下，佇列或作業狀態都將被修改，即使是控制佇列 / 作業的 sge_execd(8) 所無法影響的項目。需要管理員 / 操作員的特權。
-help	列出所有選項清單。
-r	若是套用於佇列上，則重新排程所有目前在此佇列中執行的作業。若是套用在執行中的作業上，則重新排程該作業。
-s	若是套用在佇列上，則暫停佇列以及任何可能會被啟動的作業。若是套用在執行中的作業上，則暫停該作業。若是透過暫停其佇列以明確地暫停作業，則之後取消暫停佇列將不會釋放該作業的暫停狀態。
-us	若是套用在佇列上，則取消暫停佇列以及任何可能會被啟動的作業。若是套用在作業上，則取消暫停該作業。若是透過暫停其佇列以明確地暫停作業，則之後取消暫停佇列將不會釋放該作業的暫停狀態。
-verify	不執行任何修改，但若是沒有提出 -verify 時，則僅會列出需完成的項目。
作業 / 工作 id 清單 佇列清單	<p>qmod 應該要操作的作業或是佇列。作業 / 工作 id 清單藉由下列格式之一來指定：</p> <ul style="list-style-type: none"> • 作業 id [. 工作範圍] [, 作業 id [. 工作範圍] , ...] • 作業 id [. 工作範圍] [作業 id [. 工作範圍] ...] <p>若提出指定在作業 id 之後的工作範圍，就會限制了 qmod 對陣列作業作業範圍操作的效力（若要得知作業陣列進一步的詳細資訊，請參閱 qsub(1) 的 -t 選項）。</p> <p>工作範圍指定器的格式為：</p> <p><code>n[-m[:s]][,n[-m[:s]], ...]</code></p> <p>或：</p> <p><code>n[-m[:s]][n[-m[:s]] ...]</code></p> <p>因此，它構成了以逗號或空白分隔的範圍指定器 (range specifier) 清單 <code>n[-m[:s]]</code>。這些範圍串接成一個完整的工作 id 範圍。每個範圍可以是單一數字、簡單的範圍格式 <code>n-m</code>、或是一個步驟大小的範圍。</p>

環境變數

表 14 中說明與 `qmod` 相關之環境變數。

表 14 `qmod` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qmod</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">• 指定於環境中的儲存格名稱 如果已設定的話，變數 <code>SGE_CELL</code>。• 預設儲存格名稱，例如 <code>default</code>。
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，則指定特定 <code>sge_commd(8)</code> 所執行的主機，以使用於 <code>qmod</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設僅可使用本端主機。

檔案

- Sun Grid Engine 主要主機檔案 — `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

`sge_intro(1)`、`sge_ckpt(1)`、`qstat(1)`、`queue_conf(5)`、`sge_execd(8)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qmon (1)

名稱

qmon — Sun Grid Engine 的 X-Windows OSF/Motif 圖形使用者介面

語法

qmon [選項]

說明

qmon 允許管理員以及使用者在 X-Window 環境中操作 Sun Grid Engine 系統。qmon 以多種方式提供各種相互連結的對話程序。針對每一個使用者希望完成的工作，透過 qmon 可以提供相對應的對話程序。有多種方式可以針對特定工作提出適當的對話程序：

- 在啟動時首先開始的 qmon 主視窗中，包含所有主要管理與使用者工作的圖示按鈕。當指向不同的圖示時會顯示功能性的工具提示。
- 在 qmon 主視窗功能表列中會出現「工作 (Task)」下拉式功能表按鈕。在其上按一下會開啓可用工作的清單。在其中選取一個工作便會開啓相對應的對話程序。
- 「工作」下拉式功能表也包含按鍵加速器 (key accelerator)，其可以在主視窗中藉由依序按下鍵盤上的某些按鈕直接呼叫作業對話程序。
- 在瀏覽某些對話程序以及其子階層時，若是兩個對話程序之間的連接很明顯，則會發生連結至其他對話程序的情況。按下識別該連結的按鈕會開啓其他對話程序。

選項

支援的選項為標準 X Toolkit 選項，「選項 (Options)」，請見 x(1) 說明。此外，qmon 支援其中所列出的選項表 15。

表 15 qmon 選項

選項	說明
-cmap	安裝 qmon 的個人色彩對照表。若是其他應用程式已經配置大量的色彩，以及若 qmon 因此而列出相對應的錯誤訊息時，這項動作有時候很有用。然而使用個人色彩對照表將導致每次進入或離開 qmon 視窗時，色彩對照表會進行切換動作。
-fontFamily {big medium small}	通知 qmon 針對不同的解析度大小使用不同大小之字型系列。
-help	顯示用法資訊。
-nologo	在不使用商標的情況下啟動。

對話程序

下述章節將說明與 qmon 指令相關的對話程序。

作業控制

「作業控制 (Job Control)」對話程序提供仍擱置的作業、已經在執行的作業與最近完成作業的列表顯示清單資料夾。該對話程序允許顯示作業的詳細資訊，以及刪除與暫停所選取的作業。除此之外，作業控制對話程序也提供至「提交 (Submit)」對話程序的連結，以便提交新增的作業或變更擱置作業的屬性（「Qalter」按鈕）。在表格顯示中所顯示的欄位外觀以及作業可以藉由按下「自訂化 (Customize)」按鈕來進行自訂化的作業。此自訂化作業可以儲存至 `~/.qmon_preferences` 檔案中，並使用於下列「作業控制」對話程序的初始配置啟動上。

佇列控制

「佇列控制 (Queue Control)」對話程序以及伴隨的子對話程序階層，讓使用者能夠控制系統中實際配置之 Sun Grid Engine 佇列的狀態，並允許管理員新增佇列或修改、刪除已存在的佇列。在頂層「佇列控制」對話程序視窗中的每一個圖示按鈕代表一個已配置的 Sun Grid Engine 佇列。圖示符號、色彩以及按鈕上的文字，說明了關於佇列的架構、狀態與一些基本屬性。頂層對話程序也允許刪除之前所選取的佇列。藉由使用滑鼠的左邊按鈕在圖示上按一下，或按在按鈕四週的矩形區域內來選取佇列。

藉由按下「新增」或「修改」按鈕，或是使用當在頂層「佇列控制」對話程序圖示視窗中，按下滑鼠右鍵時所顯示的蹦現式功能表，便會開啓配置 Sun Grid Engine 佇列的子對話程序。若要使用修改作業，則需要選取一個佇列。配置子對話程序允許定義佇列與主機名稱，或是顯示修改時相對應的名稱。佇列配置參數（請參閱 `queue_conf(5)`）又再細分至不同的種類中（一般配置、執行方法、檢查點、載入 / 暫停臨限值、限制、複數、使用者存取、專案存取（只有針對 Sun Grid Engine，企業版），次級佇列、所有人），其可以藉由佇列配置子對話程序較下層區域中所顯示的 `tab widget` 區來進行選取。管理員可以從已經配置的佇列中選取預設值（「複製 (Clone)」按鈕）。藉由按下「確定」按鈕，該定義將使用 `sge_qmaster(8)` 進行登錄。「佇列控制」對話程序可以使用與「作業控制」對話程序類似的方法進行自訂化。在此處套用的設定也會儲存在 `~/qmon_preferences` 中。

提交

「作業提交 (Job Submission)」對話程序提供提交批次作業與互動式作業的服務，同時當從上述所說明的「作業控制」對話程序（「Alter」按鈕）處變更擱置作業的屬性時，也會啓動此對話程序。若要在批次與互動式作業之間進行切換，請使用「作業提交」畫面右邊，按鈕欄頂端的「批次 / 互動式 (Batch/Interactive)」按鈕。

該對話程序由包含兩個作業準備對話程序頁面的資料夾所組成。在「一般 (General)」頁面有提供作業提交過程中最常使用到的參數。必須要定義作業的 `script`，而所有其他欄位都是選擇性的。若是作業需要進階需求的規格，則「進階 (Advanced)」標籤可以用來切換至增強的參數顯示。

若是作業的資源需求為強制性的，則必須使用「要求資源 (Requested Resources)」圖示按鈕以蹦現「要求資源」子對話程序。這個子對話程序允許選取作業需求的資源，以及此資源欲提供的數量定義。「可用資源 (Available Resources)」是由那些被宣告為 *可要求的* (requestable) 聚合體屬性所組成（請參閱 `complex(5)` 以得知詳細資訊）。資源需求可以制定為「嚴格 (Hard)」，意即在啓動佇列中的作業之前必須符合其要求；或制定為「寬鬆 (Soft)」，意即在可用的基礎上即授與作業資源。

使用完成按鈕來關閉「要求資源」的子對話程序，便會登記作業的指定需求。按下「提交」對話程序頂層的「提交」按鈕以提交該作業。

聚合體配置

「聚合體配置」允許管理員新增聚合體、修改或刪除現有的聚合體（請參閱 `complex(5)`）。該對話程序提供現有聚合體的選取清單，並顯示已選取的聚合體配置。藉由按下「刪除」按鈕，便會從配置中刪除所選取的聚合體。按下「新增 / 修改」按鈕將會開啓聚合體配置對話程序，其允許建立新增聚合體，或提供變更現有聚合體的工具。若欲建立新增聚合體，則必須為其定義名稱。若是要進行修改作業的話，欲修改的聚合體名稱會顯示在相同的文字輸入欄位中。聚合體配置對話程序提供聚合體項目的列表顯示清單，以及一個宣告新增或修改項目的輸入區。「新增」按鈕會使用新增或變更的項目以更新列表顯示清單，並且「確定」按鈕會使用 `sge_qmaster(8)` 登錄額外或修改的聚合體。

主機配置

可以經由「主機配置 (Host Config)」對話程序維護三種類型的主機清單：

- 管理主機
- 提交主機
- 執行主機

按一下相對應命名的標籤之一來選取欲處理的主機清單。前兩個主機清單只用來提供新增或刪除項目的功能，其允許清單上的主機擁有管理或提交權限，除此之外則拒絕之。執行主機清單項目則有提供額外的功能，即定義載入感應器、可消耗聚合體屬性、以及存取屬性之比例倍數（存取、非存取、以及專案、非專案則只有針對 Sun Grid Engine，企業版模式），如同 `complex(5)` 中所說明的。在 Sun Grid Engine，企業版另外也能夠使用比例方式回報正在執行程式的系統 CPU、記憶體以及 I/O 用量，並且可以使用資源功能因數 (Resource Capability Factor) 來定義主機的相關效能（請參閱 `host_conf(5)`）。

叢集配置

此對話程序維護叢集的全域配置以及主機特定的衍生值（請參閱 `sge_conf(5)`）。當開啓時，該對話程序會顯示所有已指定配置主機的選取清單。特殊的名稱「全域」是參考叢集全域配置。按下「新增 / 修改」按鈕便會開啓子對話程序，允許修改叢集配置。針對主機特定的配置，「全域」主機特定配置欄位將設定為無回應，並且只能處理被允許的參數。

排程器配置

「排程器配置 (Scheduler Configuration)」對話程序會提供變更 Sun Grid Engine 常駐排程器運作方式的工具 `sge_schedd(8)` 的方法。該對話程序包含所有排程器配置參數的表示方法，如同 `sched_conf(5)` 中所說明的。其細分至兩個區段中，「一般參數 (General Parameters)」以及「載入調整 (Load Adjustment)」，其可以經由相對應的標籤進行選取。「確定」按鈕將使用 `sge_qmaster(8)` 登錄任何的變更。

行事曆配置

「行事曆配置 (Calendar Config)」允許管理員新增、修改或刪除現有的行事曆配置（請參閱 `calendar_conf(5)`）。該對話程序提供現有行事曆的選取清單，並顯示所選取行事曆的配置。按下「刪除」按鈕，將會從配置中刪除所選取的行事曆。按下「新增 / 修改」按鈕將會開啓行事曆配置對話程序，並可建立新增行事曆或提供變更現有行事曆的工具。「確定」按鈕會使用 `sge_qmaster(8)` 登錄額外或修改的行事曆。

使用者配置

使用者權限是經由「使用者配置 (User Config)」對話程序來控制。對話程序左邊區段中的標籤 widget 允許您在

- 管理員帳號配置
- 操作員帳號配置
- 使用者集的定義
- 使用者帳號的定義（只針對 Sun Grid Engine，企業版模式）之間進行選取

當新增至管理員或操作員帳號清單中的使用者帳號在其所屬帳號下存取 Sun Grid Engine 時，將各自授予扮演管理員或操作員角色的權限。

使用者集清單是伴隨*使用者清單*以及*非使用者清單*之主機、佇列、專案以及叢集參數一起使用（請參閱 `queue_conf(5)`, `project(5)` 以及 `sge_conf(5)`）以控制使用者對主機、佇列、專案、以及整個叢集的存取（只有在 Sun Grid Engine，企業版系統中可以使用）。使用者集 (User set) 只是一個使用者名稱以及 UNIX 群組名稱的集合。群組名稱是藉由在其字首使用一個 "@" 符號來加以識別。在選取清單中會顯示已定義的使用者集。這些清單可以被修改，並且可以使用「使用者集 (User set)」定義對話程序來建立新增的清單。

在 Sun Grid Engine，企業版系統中使用者集可以當做「存取清單」（與其在 Sun Grid Engine 系統中的用法相同）並且 / 或是當做所謂的功能策略以及置換策略之必需「部門 (Department)」（請參閱以下的票卷配置）。

Sun Grid Engine，企業版系統也需要新增擁有系統存取權的帳號，如同 Sun Grid Engine，企業版使用者資料庫中的項目（請參閱 `user(5)`）。這項動作可以使用「使用者 (User)」子對話程序來完成。

對話程序右邊的按鈕清單中之「票卷 (Ticket)」按鈕會開啓票卷配置對話程序（請見下述說明）。這個選項也只有有在 Sun Grid Engine，企業版系統中才可以使用。

平行環境配置

平行環境 (PE) 介面可以使用此對話程序進行配置。平行環境介面是有必要的，以說明平行程式設計環境像是 PVM（平行虛擬機器）、MPI（訊息傳遞介面）或是共享記憶體的平行系統如何舉例說明其運作方法，以及如何在平行環境中加上存取限制。當開啓對話程序時，會顯示已配置的平行環境清單，以及所選取之平行環境介面目前所使用的配置（請參閱 `pe_conf(5)`）。若要新增平行環境介面或是修改現有的平行環境介面，可以使用「新增」或「修改」按鈕，其會開啓平行環境介面配置子對話程序。在使用「確定」按鈕以套用變更並退出這個子對話程序後，將使用 `sge_qmaster(8)` 登錄新增或修改的平行環境介面。

檢查點配置

可以使用此對話程序配置檢查點環境介面。檢查點環境是有必要的，以說明不同的檢查點方法之屬性，與其在 Sun Grid Engine 所支援的不同作業系統平台之衍生值。當開啓此對話程序時，會顯示已配置的檢查點環境清單，以及所選取之檢查點環境目前所使用的配置（請參閱 `checkpoint(5)`）。若要新增檢查點環境或是修改現有的檢查點環境，可以使用「新增」或「修改」按鈕，其會開啓檢查點環境配置子對話程序。在使用「確定」按鈕以套用變更並退出這個子對話程序後，將使用 `sge_qmaster(8)` 登錄新增或修改的檢查點環境。

票卷配置

此對話程序提供將票卷配置到共享、功能與置換排程策略的概要與編輯畫面。只有 Sun Grid Engine，企業版系統才支援此選項，Sun Grid Engine 系統並沒有這個選項。

「截止日期作業 (Deadline Job)」按鈕會開啓「使用者配置 (User Conf)」對話程序方塊。請變更至「使用者集」子對話程序，並選取命名為「截止日期使用者 (`deadlineusers`)」的使用者集。只有在此使用者集內的使用者可以提交截止日期作業。

「共享樹策略 (Share Tree Policy)」按鈕會開啓對話程序以建立與編輯 Sun Grid Engine，企業版共享樹（請參閱 `share_tree(5)` 以及 `schedd_conf(5)` 以得知配置參數的說明）。

「功能性策略 (Functional Policy)」按鈕會開啓對話程序以建立與編輯功能共享的配置（請參閱 `sched_conf(5)`、`access_list(5)`、`project(5)`、`queue_conf(5)` 以及 `user(5)` 以得知不同類型之功能共享與可配置加權參數的說明）。

「置換策略 (Override Policy)」按鈕會開啓對話程序以建立與編輯置換票卷的配置（請參閱 `access_list(5)`、`project(5)`、`queue_conf(5)` 以及 `user(5)` 以得知不同類型之置換票卷的說明）。

專案配置

此按鈕會開啓對話程序以建立專案。只有 Sun Grid Engine，企業版系統才支援此選項，Sun Grid Engine 系統並沒有這個選項。

該對話程序提供現有專案的選取清單，並顯示所選取專案的配置。藉由按下「刪除」按鈕，便會從配置中刪除所選取的專案。按下「新增 / 修改」按鈕將會開啓專案配置對話程序，便可建立新增專案，或提供變更現有專案的方法。專案配置在本質上表示授予或拒絕使用者集對專案的存取（請參閱上述使用者配置說明以及 `project(5)`）。

「Ok」鈕將使用 `sge_qmaster(8)` 來登錄額外或修改的專案。

瀏覽器

「物件瀏覽器 (Object Browser)」對話程序可達到下述多種目的：首先，關於先前所執行之 Sun Grid Engine 以及 qmon 動作的錯誤或成功通知訊息，便會顯示在對話程序的輸出視窗中。同樣地，qmon 的標準輸出以及標準錯誤輸出都可以轉移到「物件瀏覽器」輸出視窗中。

除此之外，當滑鼠指標在其他 qmon 對話程序中移動至其表示方式，像是圖示或表格項目時，「物件瀏覽器」就可顯示關於 Sun Grid Engine 物件的持續資訊。目前，只支援在兩個分開的對話程序中顯示兩個 Sun Grid Engine 物件的配置：

- 當滑鼠指標在佇列控制對話程序上層輸入佇列圖示時，會馬上顯示佇列的配置（請參閱上述說明）。藉由按下「物件瀏覽器」對話程序中的「佇列」按鈕可以啟動此工具。
- 當使用者將滑鼠指標移動到被指定至正在執行，或擱置作業之「作業控制」對話程序（請參閱上述說明）中的資料行時，會列出詳細的作業資訊。
- 若是按下「作業控制」對話程序中的「為什麼？(Why?)」按鈕，則在瀏覽器中會顯示額外的作業排程資訊。在此案例中會間接的開啓「瀏覽器 (Browser)」對話程序，並且會顯示任何與排程相關的資訊。

退出

「退出」圖示按鈕沒有與對話程序連結。其唯一的目的就是關閉所有作用中的 qmon 對話程序，並退出該應用程式。

資源

在預設的 qmon 資源檔案中會說明可用的資源、其意義以及需跟隨其後以便修改之的語法（請參閱下述的「檔案」章節以得知資源檔案的位置）。

環境變數

表 16 說明與 qmon 相關之環境變數。

表 16 qmon 環境變數

變數名稱	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位置。
SGE_CELL	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格，qmon 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 SGE_CELL 中指定的儲存格名稱，若是有設定的話預設儲存格的名稱；例如，default
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 sge_commd(8) 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
COMMD_HOST	若有設定，則指定特定 sge_commd(8) 所執行的主機，以使用於 qmon 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

限制

若是在編輯視窗中輸入的資料行長度超過視窗寬度，則該文字將直接超出視窗結尾。

檔案

- qmon 資源檔案範本 — `<sge_root>/qmon/Qmon`
- qmon 系統資源檔案 — `/usr/lib/X11/defaults/Qmon`
- qmon 使用者資源檔案 — `$HOME/Qmon`
- qmon 作業 / 佇列自訂化檔案 — `$HOME/.qmon_preferences`

同時參閱

sge_intro(1)、sge_conf(5)、access_list(5)、sge_pe(5)、calendar_conf(5)、complex(5)、project(5)、queue_conf(5)、sched_conf(5)、user(5)、sge_qmaster(8)

版權

請參閱 `sge_intro(1)` 以及 `<sge_root>/3rd_party/qmon` 中所提供的資訊，以得知權利與權限以及欲授予公開網域，與免付費軟體 widget 開發人員信用的進一步說明。

qrls(1)

名稱

`qrls` — 從之前的暫停狀態恢復 Sun Grid Engine 作業

語法

```
qrls [ -h {u|o|s},... ] [ -help ] [ 作業 / 工作id 清單 ]
```

```
qrls [ -h {u|o|s},... ] [ -help ] -u 使用者清單 | -uall
```

說明

`qrls` 提供使用者、操作員或管理員從一個或多個正在擱置中，等待排程以便執行的作業內恢復所謂的暫停的方法。只要對作業指定任何類型的暫停，則暫停期間該作業沒有資格進入排程。

可以使用 `qhold(1)`、`qsub(1)` 或是 `qalter(1)` 指令以便對作業指定暫停。

有三種類型的暫停。

- 使用者 — 管理員、操作員以及作業的所有人可以指定與移除使用者暫停。
- 操作員 — 管理員以及操作員可以指定與移除操作員暫停。
- 系統 — 只有管理員可以指定與移除系統暫停。

若是沒有暫停類型是使用 `-h` 選項進行指定（請參閱表 17），預設值會假定為使用者暫停。

一個釋放暫停的替代方法為 `qalter(1)` 指令（請參閱表 17 中的 `-h` 選項）。

選項

表 17 列出與 `qrls` 相關之選項。

表 17 `qrls` 選項

選項	說明
<code>-h {u o s}...</code>	從一個或多個作業中釋放 <code>u</code> （使用者）、 <code>o</code> （操作員）或 <code>s</code> （系統）暫停或其組合。
<code>-help</code>	列出所有選項清單。
<code>-u 使用者名稱,... -uall</code>	修改這些在使用者名稱清單中所指定之使用者提交作業的暫停狀態。針對管理員，其可以使用 <code>qrls -uall</code> 指令以修改所有使用者作業的暫停狀態。 若是使用 <code>-u</code> 或 <code>-uall</code> 開關，您可以指定一個額外的 <i>作業 / 工作 id 清單</i> 。
<i>作業 / 工作 id 清單</i>	以下列格式來指定： <i>作業 id</i> [. <i>工作範圍</i>] [<i>作業 id</i> [. <i>工作範圍</i>] ,...] 若提出指定在作業識別碼之後的 <i>工作範圍</i> ，就會限制了被指定為作業識別碼字尾的陣列作業之工作範圍操作的效力（若要得知作業陣列進一步的詳細資訊，請參閱 <code>qsub(1)</code> 的 <code>-t</code> 選項）。 工作範圍指定器的格式為 <code>n[-m[:s]]</code> 。範圍可以是單一數字、簡單的範圍格式 <code>n-m</code> 或是一個步驟大小的範圍。有可能會使用關鍵字 <code>all</code> 替代 <i>作業 / 工作 id 清單</i> 來修改目前使用者的所有作業。

環境變數

表 18 說明與 `qrls` 相關之環境變數。

表 18 `qrls` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qrls</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中指定的儲存格名稱，若有設定的話預設儲存格名稱，例如 <code>default</code>。

表 18 qrls 環境變數 (接上頁)

變數名稱	說明
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 <code>tcp</code> 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
COMMD_HOST	若有設定，請指定特定 <code>sge_commd(8)</code> 所執行的主機，以便用於 <code>qrls</code> 客戶端常駐的 Sun Grid Engine 通訊。根據預設值，僅可使用本端主機。

同時參閱

`sge_intro(1)`、`qalter(1)`、`qhold(1)`、`qsub(1)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qselect(1)

名稱

`qselect` – 選取佇列。

語法

```
qselect [ -help ] [ -l resource=val,... ] [ -pe 平行環境名稱,... ]  
[ -q 佇列,... ] [ -U 使用者,... ]
```

說明

`qselect` 會列出對應於選取標準的 Sun Grid Engine 佇列名稱清單，其標準在表 19 中所說明的 `qselect` 引數內指定。`qselect` 的輸出常提供給其他 Sun Grid Engine 指令使用，以套用動作在所選取之佇列組上。例如，伴隨 `qconf(1)` 的 `-mqattr` 選項一起使用，`qselect` 可以用來修改佇列集 (set of queues) 的屬性。

選項

表 19 列出與 `qselect` 相關之選項。

表 19 `qselect` 選項

選項	說明
<code>-help</code>	列出所有選項清單。
<code>-l 資源 [= 數值]...</code>	定義欲給予佇列的資源，其應該包括在佇列清單輸出中。
<code>-pe 平行環境名稱 ,...</code>	將佇列包括至輸出中，且輸出將隨附至少一個列於以逗號分隔的選項引數清單中之平行環境上。
<code>-q 佇列 ,...</code>	直接指定欲包括在輸出中的佇列。此選項通常只有在與另一個 <code>qselect</code> 選項結合使用，以便從 <code>-q</code> 所指定的清單中取出佇列名稱之子集時才有意義。
<code>-U 使用者 ,...</code>	將指定使用者有存取權之佇列包含至 <code>qselect</code> 輸出中。

範例

下列範例示範 `qselect` 指令的使用。

```
qselect -l arch=linux
```

程式範例 6

程式範例 6 中的指令將列出那些常駐於 Linux 機器上的佇列名稱。

```
qselect -l arch=linux -U andreas,shannon
```

程式範例 7

在程式範例 6 所建立的程式範例 7 中，第二個指令針對 andreas 以及 shannon 使用者運用存取權限，另外限制對那些佇列的輸出。

```
qconf -mqattr h_vmem=1GB 'qselect -l arch=linux
```

程式範例 8

程式範例 8 將常駐在 Linux 機器上之佇列的 h_vmem 屬性變更為 1 GB（十億位元組）。（請參閱 qconf(1) 線上說明手冊以得知 -mqattr 選項的詳細資訊，以及 queue_conf(5) 線上手冊以得知佇列配置項目的詳細資訊）。

環境變數

表 20 說明與 qselect 相關之環境變數。

表 20 qselect 環境變數

變數名稱	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位置。
SGE_CELL	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格，qselect 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 SGE_CELL 中指定的儲存格名稱，若是有設定的話預設儲存格的名稱；例如，default
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 sge_commd(8) 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
COMMD_HOST	若有設定，請指定特定 sge_commd(8) 所執行的主機，以便用於 qselect 客戶端常駐的 Sun Grid Engine 通訊。根據預設值，僅可使用本端主機。

檔案

Sun Grid Engine 主要主機檔案 — <sg_e_root>/<儲存格>/common/act_qmaster

同時參閱

sge_intro(1)、qconf(1)、qmod(1)、qstat(1)、queue_conf(5)、
sge_commd(8)

版權

請見 sge_intro(1) 以得知權利與權限的完整聲明。

qstat(1)

名稱

qstat – 顯示 Sun Grid Engine 作業與佇列的狀態

語法

```
qstat [ -ext ] [ -f ] [ -F [ 資源名稱,... ] ] [ -g d ] [ -help ]  
[ -j [ 作業清單 ] ] [ -l 資源 = 數值,... ] [ -ne ]  
[ -pe 平行環境名稱,... ] [ -q 佇列,... ] [ -r ]  
[ -s {r|p|s|z|hu|ho|hs|hj|ha|h}[+] ] [ -t ] [ -U 使用者,... ]  
[ -u 使用者,... ]
```

說明

qstat 顯示與佇列相關之可用 Sun Grid Engine 佇列與作業目前的狀態。「選取 (Selection)」選項允許您取得關於特定作業、佇列或使用者的資訊。在不使用任何選項的情況下，qstat 將只會顯示不包含佇列狀態資訊的作業清單。

選項

表 21 列出與 qstat 相關之選項。

表 21 qstat 選項

選項	說明
-alarm	顯示佇列警報敘述的原因。一個原因會輸出一敘述行，內含資源值與臨界值資訊。若要得知關於資源值的詳細資訊，請參考下述「輸出格式」章節中的「完整格式」說明。
-ext	只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統並沒有這個選項。顯示每個作業額外的 Sun Grid Engine，企業版相關資訊（請參閱下述「輸出格式」）。
-f	指定「完整」格式顯示資訊。該 -f 選項會顯示所有佇列的摘要資訊與位於佇列中的作業清單。
-F [資源名稱,...]	如同 -f 案例中所示，會顯示所有作業與佇列的資訊。除此之外，qstat 將根據佇列提出目前與所有資源有關的資源可用性詳細資訊清單（若是忽略選項引數），或是與資源名稱清單中所包含的那些資源有關的資源可用性詳細資訊清單。請參考下述「輸出格式」章節中的「完整格式」說明。
-g d	根據單一作業工作的方式，在一行資料中冗長顯示工作陣列。依據預設值，作業陣列會群組化，並且所有擁有相同狀態的工作（只針對擱置的工作）會顯示在單一資料行中。輸出中的陣列工作 id 範圍欄位（請參閱「輸出格式」章節）會指定相對應的工作集(set of tasks)。 該 -g 開關目前只有單一選項引數 d。其他選項引數將保留以待未來擴充。
-help	列出所有選項清單。
-j [作業清單]	列出所有擱置的作業，或是作業清單中所包含的作業沒有進入排程的原因。
-l 資源 [= 數值],...	依據所要求的資訊來定義作業所需的資源或佇列所給予的資源。在佇列上執行配對。可能會在上述佇列之一執行的作業即限制為擱置作業。

表 21 qstat 選項 (接上頁)

選項	說明
-ne	與 -f 組合，該選項將抑制顯示空白佇列。這表示將不會顯示所有目前沒有作業正在執行的佇列。
-pe 平行環境名稱,...	顯示有關佇列的狀態資訊，此佇列最少需附接於以逗號分隔之選項引數中所列的平行環境的其中之一。將顯示在選取佇列其中之一執行的那些作業資訊，或是原則上可能會進入那些佇列排程的擱置作業狀態資訊。
-q 佇列,...	指定欲顯示作業資訊的佇列。
-r	列出關於顯示作業資源的需求延伸資訊。請參考下述「延伸格式」章節的「輸出格式」子章節以得知詳細資訊。
-s {p r s z hu ho hs hj ha h}[+]	只列出在指定狀態的作業 — 任何狀態組合都有可能。-s prs 對應於一般 qstat 將完全不輸出 -s。若要顯示最近完成的作業，請使用 -s z。若要顯示使用者 / 操作員 / 系統暫停中的作業，請使用 -s hu/ho/hs 選項。該 -s ha 選項將顯示使用 qsub -a 指令提交的作業。qstat -s hj 顯示所有不符合執行資格的作業，除非該作業在作業相依清單 (job dependency list) 中擁有資料項目。(請參閱 -a 以及 qsub(1) 的 -hold_jid 選項)。
-t	列出顯示平行作業之受控制的子工作延伸資訊。請參考下述「延伸格式」章節的「輸出格式」子章節以得知詳細資訊。平行作業的子工作不應該與陣列作業工作混淆 (請參閱上述的 -g 選項，以及 qsub(1) 的 -t 選項)。
-U 使用者,...	顯示指定使用者擁有存取權之佇列的狀態資訊。將顯示在選取佇列其中之一執行的那些作業資訊，或是原則上可能會進入那些佇列排程的擱置作業狀態資訊。
-u 使用者,...	只顯示與指定使用者清單中的使用者相關聯之作業以及佇列的資訊。若是額外指定 -f 或 -F 選項，並且若是使用者在那些佇列中執行作業，就會顯示佇列狀態資訊。

輸出格式

根據 `-alarm` 存在與否，就需要區別 `-f` 或是 `-F`，以及 `-r` 與 `-t` 選項的三種輸出格式。PP 若是 Sun Grid Engine，企業版系統，`-ext` 選項可以用來顯示每個作業的額外資訊。

精簡格式（不使用 `-f` 以及 `-F`）

跟隨在標題資料行後，將針對每一個作業列出資料行，由下列資訊組成。

- 作業 id
- 經由 `qsub(1)` 或 `qalter(1)` 的 `-p` 選項指定給作業的優先順序將判定擱置作業清單的順序
- 作業名稱
- 作業所有人的使用者名稱
- 作業的狀態 — `d`（刪除）、`t`（傳輸）、`r`（執行）、`R`（重新啓動）、`s`（暫停）、`S`（暫停）、`T`（臨界值）、`w`（等待）、或 `h`（暫停）其中之一

該 `d`（刪除）敘述指出 `qdel(1)` 已用來初始作業的刪除。狀態 `t`（傳輸）以及 `r`（執行）會指出將要執行該作業，或是已經執行作業；而狀態 `s`（暫停）、`S`（暫停）以及 `T`（臨界值）則會顯示已執行暫停的作業。該 `s`（暫停）狀態是經由 `qmod(1)` 指令來暫停作業所造成；`S`（暫停）則指出已暫停包含該作業的佇列，因此該作業也被暫停；`T`（臨界值）狀態顯示至少超過一個相對應佇列的暫停臨界值（請參閱 `queue_conf(5)`），結果造成該作業被暫停。狀態 `R`（重新啓動）指出該作業已經重新啓動。其可能是由作業移轉所造成，或是因為 `qsub(1)` 指令的 `-r` 區段中所說明的原因之一。

`w`（等待）以及 `h`（暫停）狀態只會出現在擱置作業中。`h`（暫停）狀態指出該作業目前不符合執行的資格，因為已經由 `qhold(1)`、`qalter(1)`、或是 `qsub(1)` `-h` 選項指定其一個暫停狀態，或是該作業正在等待其它作業的完成，因為已經由 `qsub(1)` 或 `qalter(1)` 的 `-hold_jid` 選項對其指定作業相依性。

- 作業的提交或啓動之時間與日期
- 該作業所指定的佇列（只針對正在執行或暫停的作業）
- 正在執行作業的功能（「主控者」或「從屬者」— 後者只針對平行作業）
- 陣列作業工作 id。若為非陣列作業，其將會是空白的。請參閱 `qsub(1)` 的 `-t` 選項，以及上述的 `-g` 選項以得知額外的資訊

若是提供 `-t` 選項，每個作業狀態資料行也包含下列資訊。

- 平行工作 ID（不要混淆平行工作與陣列作業工作）
- 平行工作的狀態 — 為 `r`（執行）、`R`（重新啓動）、`s`（暫停）、`S`（暫停）、`T`（臨界值）、`w`（等待）、`h`（暫停）、或是 `x`（退出）其中之一
- CPU、記憶體、與 I/O 用量（只針對 Sun Grid Engine，企業版）
- 平行工作的退出狀態

- 平行工作的錯誤代碼與訊息

使用 `-f` 以及 `-F` 的完整格式

跟隨在標頭資料行之後，提供每個以水平資料行分隔的佇列一個區段。針對每個佇列所列出的訊息由下列資訊所組成。

- 佇列名稱
- 佇列類型 — 是 **B** (批次)、**I** (互動式)、**C** (檢查點)、**P** (平行)、與 **T** (傳輸) 其中之一，或是其它組合
- 已使用以及可使用的作業插槽數量
- 佇列主機的平均負載
- 佇列主機的架構
- 佇列的狀態 — 是 **u** (未知的，若是無法連絡相對應的 `sge_execd(8)`)、**a** (警報)、**A** (警報)、**C** (暫停的行事曆)、**s** (暫停)、**S** (次級)、**d** (停用)、**D** (停用)、**E** (錯誤) 其中之一，或是其它組合

若是狀態為 **a** (警報)，則目前最少超過一種佇列配置之載入臨界值清單中所定義的載入臨界值 (請參閱 `queue_conf(5)`)，其將會制止其它作業排程至該佇列中。

與其相反的是，狀態 **A** (警報) 會指出目前最少超過一種佇列的暫停臨界值 (請參閱 `queue_conf(5)`)。這會導致在該佇列執行的作業會依序地暫停，直到沒有臨界值違反設定。

狀態 **s** (暫停) 以及 **d** (停用) 可以經由 `qmod(1)` 指令來指定與釋放佇列。暫停一個佇列會導致在該佇列執行的所有作業皆暫停。

狀態 **D** (停用) 以及 **C** (暫停的行事曆) 指出該佇列已經由 Sun Grid Engine (請參閱 `calendar_conf(5)`) 的行事曆工具自動停用或暫停，同時 **S** (次級) 狀態指出該佇列已因等級次於另一個佇列而被暫停。(請參閱 `queue_conf(5)` 以得知詳細資訊)。當佇列暫停時 (無論原因為何)，所有在該佇列執行的作業也將一併暫停。

若是針對佇列顯示 **E** (錯誤) 狀態，則在該主機上的 `sge_execd(8)` 將無法定位主機的 `sge_shepherd(8)` 執行檔案位置以啓動作業。檢查 `sge_execd(8)` 的錯誤日誌檔案以導引如何解決該問題。之後以手動方式經由 `qmod(1)` 指令的 `-c` 選項啓動佇列。

若是使用 `-F` 選項，資源可用性資訊將列出在佇列狀態資料行之後。針對每個資源 (在 `-F` 選項引數中所選取的，若是忽略該選項引數的話則針對所有的資源)，將使用下列格式顯示單一資料行。

- 一個單一字母之指定器 (specifier) 將指出目前資源可用性的值是否由下列之一所支配：
 - 'g' — 叢集全域
 - 'h' — 主機總和
 - 'q' — 佇列相關的資源消耗量
- 第二個單一字母之指定器指出目前資源可用性值的的來源，為下列其中之一：

- 'l' – 針對資源回報的載入值
- 'L' – 在套用管理員所定義的負載比例後的資源的載入值
- 'c' – 從可消耗資源工具所衍生的可用性（請參閱 `complexes(5)`），'v' – 永遠不會被載入報告或可消耗更新來覆寫之預設聚合體配置值
- 'f' – 從不可消耗的聚合體屬性，或固定的資源限制處衍生之固定的可用性定義
- 在冒號之後，會顯示資源名稱的資訊
- 在等號之後則是目前資源可用性數值

顯示的可用性數值以及其所衍生之來源，永遠都是所有可能組合裡的最小值。因此，舉例來說，一行的格式 "`qf:h_vmem=4G`" 意指一個佇列目前在虛擬記憶體中有 4 GB（十億位元組）的最大可用性，而此值是一個固定值（例如，在佇列配置中的資源限制），且是由佇列掌控，換言之，此主機的總數或許有比此值更多的虛擬記憶體可使用，但是佇列不允許使用更多的資源。相反的，一行 "`hl:h_vmem=4G`" 也是意指上限為 4 GB（十億位元組）的虛擬記憶體可用性上限，但是限制會是從目前主機報告的載入值所衍生出來。所以當佇列可能會允許有較高虛擬記憶體需求的作業，此特定佇列所常駐的主機目前也只有 4 GB 可用。

若是使用 `-alarm` 選項，就會顯示關於違反載入或暫停臨界值的資源資訊。

如同 `-F` 選項的相同格式，是使用下列的副檔名。

- 資料行是以關鍵字 `alarm` 當做起始。
- 附加於資源值的為適當臨界值的類型與其值。

在佇列狀態行（如果使用 `-f`）或是資源可用性資訊（如果使用 `-F`）之後，會針對每個目前在此佇列中執行的作業列出單一資料行。每個作業狀態資料行包含

- 作業 id
- 作業名稱
- 作業所有人名稱
- 作業狀態 – 為 `t`（傳輸）、`r`（執行）、`R`（重新啟動）、`s`（暫停）、`S`（暫停）或 `T`（臨界值）其中之一（請參閱「精簡格式」區段以得知詳細資訊）
- 作業的啟動日期、時間以及功能（主控者或從屬者，其只有在平行作業的情況下才有意義）
- 作業的優先順序

若是提供 `-t` 選項，每個作業狀態行也包含下列資訊。

- 工作 ID
- 工作的狀態 – 為 `r`（執行）、`R`（重新啟動）、`s`（暫停）、`S`（暫停）、`T`（臨界值）、`w`（等待）、`h`（暫停）或是 `x`（退出）其中之一（請參閱「精簡格式」區段以得知詳細資訊）
- CPU、記憶體、與 I/O 用量（只針對 Sun Grid Engine，企業版）
- 工作的退出狀態
- 工作的錯誤代碼與訊息

當發生作業等待被指定至佇列的情況時，在佇列區段的清單後可能會列出**擱置作業**清單。所顯示之每個等待作業的狀態行與執行作業所顯示的方式類似。其中的差異為作業的狀態是 **w**（等待）或 **h**（暫停），其會顯示提交時間與日期來取代啟動時間，而不會顯示該作業的功能。

在某些非常罕見的案例中，例如，若是 `sge_qmaster(8)` 從不一致的作業或佇列排存檔案狀態中啟動，或是使用 `qconf(1)` 的清除佇列 (`-cq`) 選項，則 `qstat` 無法將作業指定至輸出的執行或擱置作業區段內。在此案例中，由於已偵測到作業狀態不一致（例如，作業擁有執行中的狀態，但是未被指定至佇列）。像這樣的作業，在輸出的最後結尾處會列出至**錯誤作業**區段中。該 `ERROR JOBS` 區段應該在重新啟動 `sge_qmaster(8)` 之後消失。若是您不確定像這種作業的因果關係，請聯絡您的 Sun Grid Engine 技術支援代表。

延伸格式（使用 `-r`）

若是使用 `qstat` 並指定 `-r` 選項，則會列出每個顯示作業的下列資訊（下列每個作業特性皆有單一資料行）。

- 使用 `qsub(1)` `-l` 選項所指定之嚴格 (`hard`) 與寬鬆 (`soft`) 的作業資源需求
- 所要求的平行環境包括需要的佇列插槽範圍（請參閱 `qsub(1)` 的 `-pe` 選項）
- 所要求的作業檢查點環境（請參閱 `qsub(1)` `-ckpt` 選項）
- 在執行中作業的情況下，授予的平行環境會伴隨授予的佇列插槽數量

增強的 Sun Grid Engine，企業版輸出（使用 `-ext`）

針對每個作業會顯示下列額外的項目。

- `project` — 在 `qsub(1)` `-P` 選項中作業所指定的專案。
- `department` — 使用者所屬部門（請使用 `qconf(1)` 的 `-sul` 以及 `-su` 選項以顯示目前的部門定義）。
- `deadline` — 使用 `qsub(1)` `-dl` 選項所指定之作業的截止日期初始時間
- `cpu` — 作業目前累積的 CPU 用量
- `mem` — 作業目前累積的記憶體用量
- `io` — 作業目前累積的 IO 用量
- `tckts` — 目前指定給作業的票卷總數
- `ovrts` — 藉由 `qalter(1)` 的 `-ot` 選項所指定的置換票卷
- `otckt` — 目前指定給作業之票卷總數的置換部份
- `dtckt` — 目前指定給作業之票卷總數的截止日期部份
- `ftckt` — 目前指定給作業之票卷總數的功能部份
- `stckt` — 目前指定給作業之票卷總數的共享置換部份
- `share` — 整個系統中，目前作業擁有的共享

環境變數

表 22 說明與 `qstat` 相關之環境變數。

表 22 `qstat` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若有設定，則指定預設之 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qstat</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中指定的儲存格名稱，若是有設定的話。預設儲存格名稱，例如 <code>default</code>。
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，請指定特定 <code>sge_commd(8)</code> 所執行的主機，以使用於 <code>qstat</code> 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

檔案

Sun Grid Engine 主要主機檔案 — `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

`sge_intro(1)`、`qalter(1)`、`qconf(1)`、`qhold(1)`、`qhost(1)`、`qmod(1)`、`qsub(1)`、`queue_conf(5)`、`sge_commd(8)`、`sge_execd(8)`、`sge_qmaster(8)`、`sge_shepherd(8)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

qtcsch(1)

名稱

藉由使用 `qrsh` 達到 `qtcsch - tcsh v6.09` 透通性的遠端執行。

語法

```
qtcsch [ tcsh 選項 | -ABLR ]
```

說明

`qtcsch` 為普遍使用 `csch(1)` 之衍生物的延伸，`tcsh`。其允許在 `qtcsch` 控制中輸入透通性的遠端執行指令，經由 Sun Grid Engine。 `qtcsch` 可以用來當做互動式指令解釋器 (command interpreter) 以及處理 `tcsh` shell script。

當啟動時，`qtcsch` 會識別指令是否想要在遠端執行。針對此目的，會處理檔案 `<sgc_root>/<儲存格>/common/qttask` 以及 `~/.qttask`。在這些檔案中每個資料行會定義想要在遠端執行的指令（請參閱 `qttask(5)` 以得知檔案格式的定義）。在使用者起始目錄中的 `.qttask` 檔案有包含使用者的遠端工作規格，同時共用目錄中的檔案是由管理員維護，並定義叢集範圍內預設之運作方式。若是在使用者的 `qttask(5)` 檔案中有適當的項目，則會完全置換由管理員所提供之 `qttask(5)` 檔案內容。這是為了避免管理員的 `qttask` 檔案中，指令名稱的字首被加上驚嘆號。

`qtcsch` 會一直經由 `qrsh(1)` 來嘗試啟動指定的遠端工作。下列為例外狀況。

- 即使用者經由相對或絕對路徑名稱輸入這樣的指令用以替代單獨的指令名稱。請參閱 `qttask(5)` 以得知更多資訊。
- 若是設定環境變數 `JOB_ID`，`qtcsch` 會假設在遠端的 Sun Grid Engine 作業內已經產生執行動作，並開始在本端執行工作。這可避免不想要的遞迴 (recursions)，但是可以藉由指令行選項 `ñR`，以及內建指令 `qrshmode -R` 來置換（請參閱下述對應的說明）。
- 若是 `qtcsch` 無法在啟動時建立 Sun Grid Engine 連線。當沒有可用的 Sun Grid Engine 服務時，其允許使用 `qtcsch` 當做登入 shell，且不會有被阻擋的危險。

`qtcsch` 可以在三種不同的模式中操作，以判定是否：

- 工作為遠端執行
- 要求立即或批次執行

- 為冗長的狀態輸出，或是在發生任何錯誤時才產生冗長狀態輸出

這些模式可以在 `qtcsh` 啓動期間藉由下述說明的指令行開關控制，或是在執行 `qtcsh` 期間經由在第 69 頁的「內建指令」章節中所說明的內建指令 `qrshmode` 來控制。

選項

在表 23 中所列的選項為 `qtcsh` 的特殊選項。請參考 `tcsh(1)` 文件以得知進一步選項的說明。

表 23 `qtcsh` 選項

選項	說明
-A	如果在遠端執行時，於冗長模式 (verbose mode) 中切換 <code>qtcsh</code> 會造成偵錯輸出。
-B	將遠端作業執行切換至批次模式。提交至 Sun Grid Engine 的工作若是無法立即啓動，將會置於佇列中。結果 <code>qtcsh</code> 可能會阻擋之，直到 Sun Grid Engine 可以啓動放置在佇列中的作業。由於這種運作方式在互動式階段作業期間可能不是我們所想要的，所以透過 <code>qtcsh</code> 來執行 shell script 是非常有用的，其避免 script 因為特定作業暫時無法得知資源而發生錯誤。
-L	關閉指令遠端執行的預設運作方式。這會造成所有指令都會在本端執行，就算是其包含在 <code>qtask(5)</code> 檔案其中之一也一樣。
-R	這會強制指令遠端執行，即使將 <code>JOB_ID</code> 設定為環境變數也一樣。

內建指令

本章節說明只有額外的 shell 內建指令才不可使用在標準 `tcsh(1)` 中。

`qrshmode [-ANBILR]`

在不使用選項的情況下，會顯示 `qtcsh` 目前的操作模式。該選項有下列效果。

- -A — 切換至冗長輸出模式。
- -N — 切換至非冗長輸出模式。
- -B — 切換至批次執行模式。
- -I — 切換至立即執行模式。
- -L — 永遠在本端執行指令。
- -R — 遠端執行配置的指令。

環境變數

表 24 列出與 `qtcsch` 相關之環境變數。

表 24 `qtcsch` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位置。
<code>SGE_CELL</code>	若設定的話，請指定預設的 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格， <code>qtcsch</code> 會使用（以優先順序列出）： <ul style="list-style-type: none">在環境變數 <code>SGE_CELL</code> 中指定的儲存格名稱，若是有設定的話。預設儲存格的名稱；例如，<code>default</code>。
<code>SGE_DEBUG_LEVEL</code>	如果要設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
<code>COMMD_PORT</code>	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
<code>COMMD_HOST</code>	若有設定，請指定特定 <code>sge_commd(8)</code> 所執行的主機，以使用於 <code>qtcsch</code> 客戶端常駐的 Sun Grid Engine 通訊。根據預設值，僅可使用本端主機。

檔案

- 使用者 `qtask` 檔案 — `~/ .qtask`
- 叢集 `qtask` 檔案 — `<sge_root>/<儲存格>/common/qtask`

同時參閱

`sge_intro(1)`、`qrsh(1)`、`qtask(5)`、以及 `<sge 儲存格>/3rd_party/qtcsch` 中的 `tcsh(1)`

版權

`qtcsch` 包含一部份 `tcsh`，其是由美國加州大學董事會提供版權。因此須套用下列注意事項：此產品包含加州大學、柏克萊大學以及其貢獻者所開發的軟體。

請參閱 `sge_intro(1)` 以及 `<sge_root>/3rd_party/qtcsch` 中所提供的資訊，以得知權利與權限的進一步聲明。

submit(1)

名稱

- qsub — 將批次作業提交至 Sun Grid Engine。
- qsh — 將互動式的 X- 視窗作業階段提交至 Sun Grid Engine。
- qlogin — 將互動式的登入作業階段提交至 Sun Grid Engine。
- qrsh — 將一個互動式的 rsh 作業階段提交至 Sun Grid Engine。
- qalter — 修改 Sun Grid Engine 擱置的批次作業。
- qresub — 提交現有的 Sun Grid Engine 作業副本。

語法

- qsub [選項] [*script* 檔案 | -- [*script* 引數]]
- qsh [選項] [-- *xterm* 引數]
- qlogin [選項]
- qrsh [選項] [指令 [指令引數]]
- qalter [選項] 作業 / 作業 id 清單 [-- [*script* 引數]]
- qalter [選項] -u 使用者清單 | -uall [-- [*script* 引數]]
- qresub [選項] 作業 id 清單

說明

qsub 將批次作業提交至 Sun Grid Engine 佇列系統。Sun Grid Engine 支援單一與多重節點作業。*script* 檔案包含作業欲使用 shell 執行的指令（例如：`sh(1)` 或 `cs(1)`）。作業 *script* 的引數是由 *script* 引數所指定。若是 *script* 資料行中的頭兩個字元符合 '#\$'，或與下述說明的 -C 選項所定義的字首字串相同，則 Sun Grid Engine 旗號可以當成 qsub 的選項來輸入，或是當成 *script* 檔案中的內嵌式旗號 (embedded flags)。

qsh 將互動式 X- 視窗階段作業提交至 Sun Grid Engine。xterm(1) 是從執行機器上啟動，並將顯示導向 DISPLAY 環境變數所指出的 X- 伺服器，或是使用 -display qsh 選項所指定的位置。若是沒有可用的資源來執行互動式作業，則其不會進入排存。其將被派送至適合的機器上以便立即執行，或是藉由 qsh 通知提交該作業的使用者，沒有可用的適當資源以執該作業。xterm_args 會傳遞至 xterm(1) 執行檔案中。

qlogin 與 qsh 類似，都是將互動式作業提交至佇列系統。在 X 顯示上不會開啓 xterm(1) 視窗，而使用者的 I/O 會使用目前的終端機。通常，qlogin 會使用標準客戶端與伺服器端的指令，來建立一個與遠端主機之間的 telnet(1) 連線。這些指令可以使用 sge_conf(5) 之全域與本端配置設定中的 qlogin_daemon（在伺服器端，若尚未設定 Sun Grid Engine telnetd，則會以類似 /usr/sbin/in.telnetd 的方式顯示）以及 qlogin_command（在伺服器端，若尚未設定 Sun Grid Engine telnet，則會以類似 /usr/bin/telnet 的方式顯示）參數來配置。客戶端指令會自動的參數化 (parameterized) 遠端主機名稱以及連接埠號來連接（換言之，會產生如 /usr/bin/telnet my_exec_host 2442 的啟動）。qlogin 之啟動完全與 qsh 一樣，且它的作業只能執行在 INTERACTIVE 的佇列中。若是 sge_execd(8) 在 root 帳號下執行時，才能使用 qlogin 作業。

qrsh 與 qlogin 相似之處即為它提交互動式作業至佇列系統。它使用目前的終端機給使用者 I/O。通常，qrsh 會建立一個 rsh(1) 連接至遠端主機。若是沒有提供指令給 qrsh，則會建立一個 rlogin(1) 階段作業。伺服器端指令的使用可以藉由 sge_conf(5) 之全域與本端配置設定中的 rsh_daemon 與 rlogin_daemon 參數來配置。若是未設定參數，則會使用 Sun Grid Engine rshd 或 rlogind，或是其他類似需要配置的 /usr/sbin/in.rshd 或 /usr/sbin/in.rlogind。在客戶端，可以設定 rsh_command 以及 rlogin_command 參數至 sge_conf(5) 的全域與本端配置設定中。若是前二項都沒有設定，則使用 Sun Grid Engine 所提供的 rsh(1) 與 rlogin(1) 二進位程式。使用叢集配置來整合作業系統所提供的工具，像是 ssh 或 rsh(1) 與 rlogin(1) 機制。

qrsh 作業只能在「互動式」佇列中執行，除非使用 -now no 選項（請參閱下列說明）。這些作業只有 sge_execd(8) 在 root 帳號下執行時才能使用。

qrsh 提供一個有用的額外功能與互動式工具整合，以提供特定的指令 shell。當 qrsh 被啟動時，若有設定環境變數 QRSH_WRAPPER，則由 QRSH_WRAPPER 指向的指令解讀器將會執行 qrsh 指令，以替代使用者的登入 shell，或是在 qrsh 指令行中所指定的任何 shell。

qalter 可以用來變更擱置作業的屬性。一旦作業執行之後，就無法再進行變更。針對作業陣列，其工作的一部分可以擱置而其它部分可以繼續執行（請參閱下列的 -t 選項），qalter 的修改只作用於擱置作業。qalter 可以變更大部分的作業特性（請參閱下列「選項」章節所對應的敘述），包括那些在 script 中定義為內嵌旗號的特性（請參閱上述說明）。

qresub 允許您由複製現存的擱置或執行作業來建立作業複本。除了一個新的作業識別碼之外，複製的作業將會擁有與所複製的作業完全相同的屬性。qresub 唯一支援的修改複製作業功能，是以 -h 選項來指定暫停狀態。此指令可以用來先複製一個作業，然後再透過 qalter 變更該作業的屬性。

針對 `qsub`、`qsh`、`qrsh`、以及 `qlogin`，管理員與使用者可定義預設要求檔（請參閱 `sge_request(5)`），其可以包含說明於下的任何選項。若是在預設要求檔中的一個選項是為 `qsub` 與 `qlogin` 所理解而 `qsh` 則否，那麼若是啟動 `qsh` 時，會默然忽略該選項。因此，您可以同時為 `qsub` 以及 `qsh` 維護一個共享的預設要求檔案。

全叢集的預設要求檔案可能置於 `$sge_root/$sge_cell/common/sge_request` 的路徑之下。使用者個人的預設要求檔案是在 `$HOME/.sge_request` 和 `$cwd/.sge_request` 的位置下處理。作業目錄的本端預設要求檔案擁有最高之優先權，再來是位於起始目錄的檔案，然後才是叢集全域檔案。選項引數、內嵌的 `script` 旗號以及在預設要求檔內的選項，會依照下列的順序來處理。

1. 在 `script` 行中，由左到右處理
2. 在預設要求檔案中，由左到右處理
3. 在 `script` 中，由上到下處理（只用於 `qsub`）
4. 在預設要求檔案中，由上到下處理
5. 在指令行中，由左到右處理

換言之，指令行可以用來置換內嵌旗號以及預設要求設定。然而，內嵌旗號會置換預設的設定。

注意 - `-clear` 選項可以用來在任何時間刪除預設要求檔案、內嵌的 `script` 旗號或指令行選項中任何先前的設定。然而，`qalter` 無法使用此選項。

表 25 中說明的選項可以使用強制或是非強制的要求。根據預設值，所有的要求都視為強制的要求，直到遇見 `-soft` 選項（請參閱下列說明）。強制 / 非強制的狀態會保持其效用直到再次遇見對應的另一個選項為止。若是無法達到一個作業所有的強制要求，則該作業不會被排程執行。目前無法執行的作業會繼續存於多工緩衝區內。

選項

表 25 列出並說明與各式提交指令相關之選項。

表 25 `submit` 指令選項

選項	說明
<code>-@ 選項檔案</code>	強制 <code>qsub</code> 、 <code>qrsh</code> 、 <code>qsh</code> 、或是 <code>qlogin</code> 使用包含在 <code>選項檔案</code> 中的選項。該指定的檔案可包含所有有效的選項。以 <code>#</code> 號來啓始註釋行。

表 25 submit 指令選項 (接上頁)

選項	說明
-a 日期時間	<p>只能用於 qsub、qrsh、qsh、qlogin、以及 qalter 指令中。</p> <p>這會針對有執行資格的作業來定義或重新定義其時間以及日期。日期時間遵照 <code>[[CC]]YY]MMDDhhmm.[ss]</code> 的格式：</p> <p>CC 以 2 個數字表示世紀。</p> <p>YY 以 2 個數字表示西元年份。</p> <p>MM 以 2 個數字表示月份。</p> <p>DD 以 2 個數字表示日。</p> <p>hh 以 2 個數字表示小時。</p> <p>mm 以 2 個數字表示分鐘。</p> <p>ss 以 2 個數字表示秒數 (預設值是 00)。</p> <p>若是省略任何一個選擇性的日期欄位，其對應的值會假設為目前的日期。</p> <p>若是 Sun Grid Engine 儲存區的主機時鐘沒有同步，使用此選項可能會導致無法預期的結果。</p> <p>再者，當啟動 Sun Grid Engine 常駐程式 <code>sge_qmaster(8)</code> 以及 <code>sge_execd(8)</code> 時，此選項的適當運作方式很依靠正確且合適的時區設定，例如，TZ 環境變數 (若要得知詳細資訊，請參閱 <code>date(1)</code>)。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>
-ac 變數 [= 數值],...	<p>只能用於 qsub、qrsh、qsh、qlogin、以及 qalter 指令。</p> <p>新增所提供的名稱 / 數值配對至作業的上下文中。數值可以省略。Sun Grid Engine 會將所提供的引數附加至作業的內文變數清單中。可以提供多個 -ac、-dc、以及 -sc 選項。在此處，順序是重要的。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>
-A 計算字串	<p>只能用於 qsub、qrsh、qsh、qlogin、以及 qalter 指令。</p> <p>識別作業之資源消耗應記入的計算名稱。計算字串 (<i>account string</i>) 可以是任何隨意的 ASCII 字母與數字組成的字串，但不可包含空白或是分隔字元。底線 '_' 被視為非分隔字元。若是沒有提供此參數，Sun Grid Engine 則會放置預設的計算字串 <code>sge</code> 至作業的計算記錄中。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-c 狀態指定器 (<i>occasion_specifier</i>)	<p>只能用於 <code>qsub</code> 以及 <code>qalter</code>。</p> <p>定義或重新定義作業是否需要檢查點，以及若是需要的話，則是在何種情況之下。使用此選項的檢查點狀態規格，則會覆寫 <code>qsub -ckpt</code> 開關所參照的檢查點環境中 <i>when</i> 參數的定義 (請參閱 <code>checkpoint(5)</code>)。狀態指定器可能出現的值为：</p> <ul style="list-style-type: none"> n - 不執行檢查點。 s - 當批次伺服器關機時，執行檢查點。 m - 在最低的 CPU 間隔時，執行檢查點。 x - 當作業變成暫停狀態時，執行檢查點。 <間隔> - 在指定的時間間隔中執行檢查點。最低的 CPU 間隔時間定義於佇列配置中 (若要得知詳細資訊，請參閱 <code>queue_conf(5)</code>)。<interval> 必須使用 <code>hh:mm:ss</code> 格式來指定。若是指定 <interval> 時，則使用最大的 <interval> 以及佇列之最低的 CPU 間隔時間來執行檢查點。完成以確認機器不會因為太常產生檢查點而產生超載的現象。
-ckpt 檢查點名稱	<p>只能用於 <code>qsub</code> 以及 <code>qalter</code>。</p> <p>選擇檢查點環境 (請參閱 <code>checkpoint(5)</code>) 來針對作業使用檢查點。也同時宣告該作業為檢查點作業。</p>
-clear	<p>只能用於 <code>qsub</code>、<code>qrsh</code>、<code>qsh</code>、<code>qlogin</code>、以及 <code>qalter</code> 指令。</p> <p>在套用任何出現在此特定指令中的修改 (若有的話) 前，會導致作業的所有元素重設成為初始化的預設值狀態。</p>
-cwd	<p>只能用於 <code>qsub</code>、<code>qrsh</code>、<code>qsh</code>、<code>qlogin</code>、以及 <code>qalter</code>。</p> <p>自目前的工作目錄中執行作業。若是其對應的配置檔案存在的話，此開關會啟動 Sun Grid Engine 的路徑更名工具。(請參閱 <code>sge_aliases(5)</code>)。</p> <p>以 <code>qalter</code> 為例，若是 <code>qalter</code> 與先前的 <code>qsub</code> 或 <code>qalter</code> 執行在不同的目錄中，則會覆寫目前工作目錄先前的定義。</p> <p><code>qalter</code> 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-c 字首字串	<p>只能用於 qsub。</p> <p>字首字串 (<i>prefix_string</i>) 在作業的 script 中定義字首，其宣告一個指令給 qsub。字首不是一個作業屬性，但是會影響 qsub 的運作方式。若是 -C 選項使用空白字串作為指令字首的值，則 qsub 不會掃描該 script。</p> <p>指令字首以二個 ASCII 字元所組成，其使用顯示於 script 資料行的前二個位元來指定接於其後的是一個 Sun Grid Engine 指令 (預設值是 # \$)。</p> <p>使用者必須知道的是，變更第一個定義符號字元可能會產生無法預料的副作用。若是 script 包含任何敘述除了位於行的第一個位元位置之 # 字元，該作業的 shell 處理器將會拒絕該行並且可能會草率地結束該作業。</p> <p>若是 -C 選項存在於 script 中，則會被忽略。</p>
-dc 變數....	<p>只能用於 qsub、qcrsh、qsh、qlogin、以及 qalter。</p> <p>自作業的上下文中移除所提供的變數。可以提供多個 -ac、-dc、以及 -sc 選項。在此處，順序是重要的。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>
-display 顯示指定器	<p>只能用於 qsh。</p> <p>指示 xterm(1) 使用顯示指定器 (<i>display_specifier</i>) 來聯絡 X 伺服器。</p>
-dl 日期時間	<p>只能用於 qsub、qcrsh、qsh、qlogin、以及 qalter。只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統並沒有這個選項。</p> <p>以 <i>[[CC]YY]DDhhmm[.SS]</i> 的格式來指定截止日期起始時間 (請參閱說明於上的 -a 選項)。截止日期起始時間是指一個截止日期作業必須在給與的截止日期內，取得最高的優先順序來完成。在截止日期起始時間之前，有截止日期之作業的優先順序會穩定地提高，直到該作業已到達 Sun Grid Engine 管理員所配置的最高權限。</p> <p>此選項只適用於允許提交截止日期作業的使用者。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-e [主機名稱:] 路徑,...	<p>只能用於 qsub 以及 qalter。</p> <p>定義或重新定義作業的標準錯誤串流所使用的路徑。若是路徑指定一個絕對路徑的名稱，作業的錯誤路徑屬性則會設定為包含主機名稱以及此路徑的值。若為相對路徑名稱，Sun Grid Engine 會使用二種方式展開路徑，一是使用 -cwd (請參閱上述說明) 開關也有指定目前的工作目錄，否則就會使用起始目錄路徑。若是主機名稱存在時，如果在指定的主機上執行作業，則標準錯誤串流將會置於其對應的位置。</p> <p>針對陣列作業工作，標準錯誤檔案名稱有預設的格式：作業名稱.e 作業 id 以及作業名稱.e 作業 id. 工作 id (請參閱說明於下的 -t 選項)。</p> <p>若是路徑為一個目錄，則作業的標準錯誤串流將會使用預設的檔案名稱置於此目錄中。若是路徑名稱包含某些虛擬的環境變數，在作業執行時會展開這些值，並且用來組成標準錯誤串流的路徑名稱。以下列出目前所支援的虛擬環境變數：</p> <p><i>\$HOME</i> — 位於執行機器上的啓始目錄 <i>\$USER</i> — 作業所有人的使用者識別碼 <i>\$ 作業 ID</i> — 目前的作業識別碼 <i>\$ 作業名稱</i> — 目前的作業名稱 (請參閱 -N 選項)</p> <p><i>\$HOSTNAME</i> — 執行主機的名稱 <i>\$TASK_ID</i> — 陣列作業的作業索引號碼</p> <p>在 <i>csh(1)</i> 或是 <i>ksh(1)</i> 中，可以常見到使用代字號「~」來替代 <i>\$HOME</i>。</p> <p>此字號「~」也用在與使用者名稱的組合，以至於「~<使用者>」展開成為 <user> 的起始目錄。當然，若要使用作業所有人之外的另一個使用者識別碼，則需要對應的權限。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啓動或是移轉作業之後才會生效。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-hard	<p>只能用於 qsub、qrsh、qsh、qlogin、以及 qalter。</p> <p>表示在指令行之後的所有資源需求都是強制需求，而且全部的需求必須先被滿足之後，作業才可以被排程。</p> <p>當 Sun Grid Engine 掃描指令行以及 script 中的 Sun Grid Engine 選項與參數時，會建立作業所需求的資源清單。所有資源要求都被視為是啓動作業絕對不可或缺的要素。若是在掃描時遇見 -soft 選項 (請參閱下列說明)，則所有接下來的資源都會指定為「非強制需求 (soft requirements)」來執行，亦或是「有當然很好，但不是必要的 (nice-to-have, but not essential)」。</p> <p>若是在之後的掃描階段遇見 -hard 旗號，則所有接下來的資源要求又再度變回「不可或缺的要素 (essential)」。</p> <p>於掃描時，-hard 以及 -soft 選項實際上在掃描期間是扮演「開關」的角色。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-h -h {u s o n U O S}...	<p>只能用於 qsub、qrsh、qsh、qlogin、qalter、以及 qresub。</p> <p>列出置於作業中的暫停種類。</p> <p>u – 表示使用者暫停。</p> <p>s – 表示系統暫停。</p> <p>o – 表示操作員暫停。</p> <p>n – 表示沒有暫停。</p> <p>只要除了「n」以外的任何暫停指定給作業，該作業即沒有執行的資格。可以透過 qalter 以及 qrls(1) 來釋放暫停。以 qalter 為例，其使用下列額外的選項指定器來支援 -h 開關：</p> <p>U – 移除使用者暫停。</p> <p>S – 移除系統暫停。</p> <p>O – 移除操作員暫停。</p> <p>Sun Grid Engine 管理員可以指定及移除所有類型的暫停、Sun Grid Engine 操作員可以指定及移除使用者與操作員暫停、而使用者只能指定或移除使用者暫停。</p> <p>對 qsub 而言，只能放置使用者暫停於作業中，因此也只允許選項的第一種格式與 -h 開關一起使用。與 qsub 不同的是，qalter 則需要上述說明的第二種格式。</p> <p>qhold(1) 工具則提供一個可供選擇的方式來指定暫停。</p> <p>若是作業為陣列作業（請參閱下述 -t 選項說明），所有透過 -t 指定的作業都會同時被 -h 操作所影響。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>
-help	列出所有選項清單。
-hold_jid [作業 id 作業名稱],...	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>定義或重新定義提交作業的作業附屬清單。作業名稱所指定的參照，只有在相同的使用者擁有參照作業與所參考的作業時，才會被接受。除非在以逗號分隔的 作業 id 及 / 或 作業名稱 清單中的所有參照之作業都已成功地完成，否則提交的作業沒有執行的資格。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-inherit	<p>只能用於 q_rsh 以及 q_make(1)。</p> <p>q_rsh 允許您在已排程之平行作業中啟動一項作業。-inherit 選項命令 q_rsh 自環境變數 <i>作業 ID</i> 中讀取作業識別碼，並且在此作業中啟動指定的指令而成爲一項作業。請注意於此例中，應執行指令之主機的主機名稱必須在指令之前優先執行，其語法變更為 <code>q_rsh-inherit [其他選項] 主機名稱 指令 [指令引數]</code></p> <p>也請同時注意，在與 -inherit 組合時，其他大部分的指令行選項都會被忽略。只有 -verbose、-v 以及 -V 選項會被解讀。-v PWD 可用來替代 -cwd 選項。</p> <p>通常作業會與其對應的作業擁有相同的環境 (包含目前作業目錄)，因此指定 -v 選項應該會適用於大部分的應用程式。</p> <p>若在您的系統中尚未將指令埠設定爲服務，而是透過 COMMD_PORT 環境變數，那麼當您使用 -inherit 選項啟動 q_rsh 或 q_make 指令時，請確定此變數已設定於環境中。若是您在作業的 script 中使用 -inherit 選項啟動 q_rsh 或 q_make 指令，則使用 submit 選項或是特殊註解來匯出 COMMD_PORT: -v COMMD_PORT</p>
-j y n	<p>只能用於 q_sub 以及 q_alter。</p> <p>指定是否作業的標準錯誤串流已合併至標準輸出串流。</p> <p>若是 -j y 以及 -e 兩個選項都存在時，Sun Grid Engine 會設定錯誤一路徑的屬性，但是卻忽略之。</p> <p>q_alter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-l 資源 = 數值 ,...	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>在 Sun Grid Engine 的佇列中啟動符合所提供之資源要求清單的作業。以 qalter 為例，其使用指定的定義來置換之前的定義。</p> <p>complex(5) 中說明如何取得可用資源之清單以及與其相關之有效值指定器。</p> <p>在單一指令中可以有多個 -l 開關。您可以在相同的指令行中要求多個 -l 選項同時為非強制或是強制要求。以序列作業為例，多個 -l 開關則修正其尋找佇列的定義。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>
-m b e a s n,...	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>定義或重新定義在何種情形之下郵件會傳送至作業的所有人，或是傳送至藉由 -M 選項所定義之使用者。此選項的引數有下列意義：</p> <ul style="list-style-type: none"> b — 在作業開始時傳送郵件。 e — 在作業結束時傳送郵件。 a — 在作業中斷或是重新排程時傳送郵件。 s — 在作業暫停時傳送郵件。 n — 不傳送郵件。 <p>一般來說，當作業暫停時不會傳送郵件。</p> <p>針對 qsh 以及 qlogin，當其位於預設的要求檔時，則禁止在作業開始或是結束時傳送郵件。</p> <p>qalter 允許變更 b、e 以及 a 選項引數，即使當作業正在執行時。然而，修改 b 選項引數只有在重新啟動或是移轉作業之後才會生效。</p>
-M 使用者 [@ 主機] ,...	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>若是該伺服器傳送與作業相關的郵件，則針對執行作業的伺服器就會定義或重新定義其必須傳送郵件的使用者清單。預設值是來源主機上的作業所有人。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-masterq 佇列,...	<p>只能用於 qsub、qsh、qysh、qlogin 以及 qalter。此選項只對平行作業有意義，換言之，需與 -pe 選項一起使用。</p> <p>定義或是重新定義成爲此平行作業中所謂的主要佇列 (master queue) 之佇列清單。主要佇列的定義是啓動平行作業的佇列。平行作業所複製作業的其他佇列，稱之爲次要佇列 (slave queue)。一個平行作業只有一個主要佇列。</p> <p>此參數擁有資源要求的全部屬性，並且將與衍生自上述說明 -l 選項的需求合併。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啓動或是移轉作業之後才會生效。</p>
-notify	<p>只能用於 qsub、qysh、qsh、qlogin 以及 qalter。</p> <p>當設定此旗號時，會導致 Sun Grid Engine 傳送「警告」信號至執行中的作業，其優先於作業自己所傳送的信號。若是一個 SIGSTOP 擱置，則在執行 SIGSTOP 幾秒鐘前，作業會接收一個 SIGUSR1。若是一個 SIGKILL 擱置，則在執行 SIGKILL 幾秒鐘前，作業會接收一個 SIGUSR2。延遲時間的總量是由每個佇列配置中的 notify 參數來控制 (請參閱 queue_conf(5))。</p> <p>Linux 作業系統在其目前的 Posix 執行緒 (thread) 實作中，「誤用」了使用者信號 SIGUSR1 以及 SIGUSR2。若是您在 Linux 環境下的作業中執行執行緒應用程式，您或許不會想要使用 -notify 選項。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>
-now y[es] n[o]	<p>可用於 qsub、qysh、qsh 以及 qlogin。</p> <p>-now y 嘗試立即啓動作業，不然則不作任何動作。此指令成功則傳回 0，或是失敗則傳回 1 (若是作業不能立即排程，也傳回 1)。</p> <p>-now y 是 qsh、qlogin 以及 qysh 的預設值。</p> <p>若是使用 -now n 選項，假如該作業不能立即執行，則會將作業放入擱置的佇列。-now n 是 qsub 的預設值。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-N 名稱	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>作業的名稱。名稱可以是任何可列出的、且以字母字元起始的字元組。</p> <p>若是 -N 選項不存在時，Sun Grid Engine 會將任何目錄路徑的名稱自 script 名稱中移除之後，再指定作業 script 的名稱給該作業。若是從標準輸入中讀取 script，則作業名稱的預設值為 STDIN。</p> <p>以 qsh 或是 qlogin 為例 — 且若是 -N 選項不存在時 — INTERACT 字串會指定給作業。qalter 允許變更此選項，即使當作業正在執行時。</p>
-noshell	<p>只能與指令行一起用於 qrsh。請勿在使用者的登入 shell 中啟動提供給 qrsh 的指令行，而是要在沒有包裝 shell 之情況下執行它。</p> <p>此選項可以用來加快執行的速度，因為避免掉了一些零碎空間，如 shell 啟動和取得 shell 資源檔。</p> <p>若是沒有 shell 特定指令行分析的需求時，才能使用此指令。若是指令行包含 shell 語法，像是環境變數替換或是向後 (back) 引用時，則必須啟動 shell。在這種情況之下，既不要使用 -noshell 選項，也不要指令行中包括 shell 呼叫。</p> <p>範例：</p> <pre>qrsh echo '\$HOSTNAME'</pre> <p>另一個使用 -noshell 選項時可選擇的呼叫：</p> <pre>qrsh -noshell /bin/tcsh -f -c 'echo \$HOSTNAME'</pre>
-nostdin	<p>只能用於 qrsh。</p> <p>禁止輸入串流 STDIN — qrsh 會傳送 -n 選項至 rsh(1) 指令。若是多個作業使用 qrsh 來平行執行，此選項特別有用；例如，在一個 make(1) 處理程序中。不會定義能輸入哪一個處理程序。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-o [主機名稱:] 路徑,...	<p>只能用於 qsub 以及 qalter。</p> <p>使用於作業的標準輸出串流之路徑。路徑之操作就如同在 -e 選項中，針對標準錯誤串流的說明。</p> <p>針對陣列作業工作，標準輸出檔案名稱有預設的格式 <i>作業名稱.o 作業 id</i> 與 <i>作業名稱.o 作業 id. 工作 id</i> (請參閱下列的 -t 選項)。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啓動或是移轉作業之後才會生效。</p>
-ot 置換票卷	<p>只能用於 qalter。只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統並沒有這個選項。</p> <p>針對指定的作業變更置換票卷的數目。需要管理員 / 操作員的特權。</p>
-P 專案名稱	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。只有 Sun Grid Engine，企業版系統支援此選項，Sun Grid Engine 系統並沒有這個選項。</p> <p>指定作業所指派的專案。管理員必須授與權限給個別的使用者，以提交作業給指定的專案。(請參閱 qconf(1) 的 -aprj 選項)。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-p 優先順序	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>定義或重新定義與其他作業相對之作業的優先順序。優先順序是一個位在 -1023 到 1024 的範圍之間的整數。作業的預設優先順序值為 0。</p> <p>在 Sun Grid Engine 系統中，使用者只能降低其作業的優先順序。Sun Grid Engine 管理員以及管理員還可以增加與作業相關聯的優先順序。若是一項擱置的作業有較高的優先順序，該作業會較早有資格被 Sun Grid Engine 排程器分派。在 Sun Grid Engine 中，作業優先順序對執行中的作業則沒有影響。</p> <p>在 Sun Grid Engine，企業版中，作業優先順序會影響「共享樹策略」以及「功能策略」。其對於「截止期限策略」以及「置換策略」則沒有影響（若要得知 Sun Grid Engine，企業版所支援的資源管理策略之進一步資訊，請參閱 <code>share_tree(5)</code>、<code>sched_conf(5)</code> 以及《Sun Grid Engine，企業版 5.3 Sun Grid Engine 管理和使用者指南》）。</p> <p>以「共享樹策略」為例，使用者可以在他們的作業中，使用透過 -p 所指定之不同的優先順序，來分散使用者目前所擁有之票券。若是所有的作業擁有相同的優先順序值，則票券會平均地分散。否則，作業會接收相對不同優先順序的票券。在後面的例子中，共享樹中的優先順序被視為一個額外的層級。</p> <p>關於「功能策略」，優先順序可用來在功能作業類別範圍中衡量作業的份量。再次，在功能作業類別下，票券分散至相對任何不平均的優先順序分配中，而被視為是一個虛擬的分散層級。</p> <p>若是「共享樹策略」及「功能策略」兩者同時作用中，則作業優先順序會在兩個策略中產生影響，而且單獨自兩者其中之一所衍生出來的票券則會針對每個作業加起來形成票券的總數。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-pe 平行環境 <i>n</i> [- <i>m</i>]/[- <i>m</i>],...	<p>只能用於 qsub、qsh、qlogin 以及 qalter。</p> <p>具體化平行程式設計環境 (PE)。在平行環境名稱之後的範圍敘述元 (descriptor)，指定欲執行之平行處理程序的數目。Sun Grid Engine 會配置適當的可用資源。sge_pe(5) 線上說明手冊包含關於平行環境的定義，以及關於如何得知目前有效平行環境的清單。</p> <p>您可以使用萬用字元 * 來指定平行環境名稱，因此要求 pvm* 將會符合任何平行環境中以字串 pvm 開頭的名稱。</p> <p>範圍規格是一範圍表示式的清單，其格式為 <i>n-m</i> (<i>n</i> 與 <i>m</i> 一樣都是非零的正整數)，而 <i>m</i> 是 <i>m-m</i> 的縮寫、-<i>m</i> 是 1-<i>m</i> 的簡寫、以及 <i>n-</i> 是 <i>n-infinity</i> 的縮寫。範圍規格以下列方式處理：第一個檢查的是佇列最大數量之要求。若是有可用而足夠的佇列符合指定屬性清單，則會配置所有的佇列。再來是檢查佇列的次要數量，然後以此類推。</p> <p>若另外的 -l 選項存在時，就會限制了平行作業所能設定的佇列。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>
-q 佇列,...	<p>只能用於 qsub、qsh、qlogin 以及 qalter。</p> <p>定義或是重新定義可能用於執行此作業的佇列清單。此參數擁有資源要求的全部屬性，並且將與衍生自上述說明 -l 選項的需求合併。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>
-r <i>y</i> <i>n</i>	<p>只能用於 qsub 以及 qalter。</p> <p>這會識別作業是否具有重新執行的能力。若是 -r 的值為 <i>y</i> 時，當該作業沒有留下一致性的結束狀態即中斷 (這是典型的案例，若是作業所執行的節點故障時)，就重新執行該作業。若是 -r 是 <i>n</i> 時，在任何情況下都不會重新執行該作業。</p> <p>使用 qsh 或 qlogin 提交的互動式作業是無法重新執行的作業。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-sc 變數 [= 數值],...	<p>只能用於 qsub、qsh、qrsh、qlogin、以及 qalter。</p> <p>設定提供的名稱/數值配對為作業的上下文。數值可以省略。Sun Grid Engine 使用提供的引數來置換作業前一次定義的上下文。可以提供多個 -ac、-dc 以及 -sc 選項。在此處，順序是重要的。</p> <p>上下文是可動態附接於作業以及自作業移除超資訊 (meta-information) 之一種方式。在其環境中，上下文變數是不會傳送至作業的執行上下文。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。</p>
-soft	<p>只能用於 qsub、qrsh、qsh、qlogin 以及 qalter。</p> <p>表示在指令行之後的所有資源需求都是非強制需求，且是以一個「可用的」基礎來滿足其需求。</p> <p>當 Sun Grid Engine 掃描指令行以及 script 中的 Sun Grid Engine 選項與參數時，會建立作業要求之資源清單。所有資源要求都被視為是啟動作業絕對不可或缺的要素。若是在掃描時遇見 -soft 選項，則所有接下來的資源都會指定為「非強制需求 (soft requirements)」來執行，亦或是「有當然很好，但不是必要的 (nice-to-have, but not essential)」。若在之後的掃描階段遇見 -hard 旗號 (請參閱上述說明)，則所有接下來的資源要求又再度變回「不可或缺的要素 (essential)」。於掃描時，-hard 以及 -soft 選項實際上是扮演「開關」的角色。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-s [主機:] 路徑名稱...	<p>只能用於 qsub、qsh、qlogin 以及 qalter。</p> <p>指定作業的解讀 shell。只有一個路徑名稱元件而沒有主機指定器是有效的，而且只允許一個路徑名稱一個指定的主機。若是主機為執行主機，則主機指定的 Shell 路徑會定義作業的解讀 shell。若是執行主機沒有符合主機清單上的任一台機器，則使用沒有主機規格的 shell 路徑。</p> <p>此外，路徑名稱可以由虛擬的環境變數來構成，如同說明於上的 -e 選項。</p> <p>以 qsh 為例，在 xterm(1) 中，指定的 shell 路徑用來執行對應的指令解讀器（透過 qsh 的 -e 選項），其以互動式作業的方式來啟動。qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啟動或是移轉作業之後才會生效。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-t n[-m[:s]]	<p>只能用於 qsub 以及 qalter。</p> <p>提交所謂的陣列作業；換言之，相同作業的陣列只有藉由不同的索引數字來區別，並且被 Sun Grid Engine 視為幾乎是一系列的作業。-t 選項引數指定陣列作業工作的數目，以及會與作業相關聯的索引數字。透過環境變數 SGE_工作 ID，索引數字會被匯出至作業工作中。</p> <p>指定在選項引數中的工作 id 範圍，可以是單一數字、簡單的範圍格式 n-m 或是一個步驟大小的範圍。因此，以 2-10:2 指定的工作 id 範圍會產生工作識別碼索引 2、4、6、8 以及 10，換言之，總計為 5 個的相同作業藉由環境變數 SGE 工作 ID 來包含其每個作業的索引數字。所有的陣列作業工作會繼承相同的資源要求以及屬性定義，與指定在 qsub 或 qalter 指令行中的一樣，除了 -t 選項之外。作業會被個別的排程、並且提供其足夠的資源，此同時存在的情形就好像是獨立的作業一樣。然而，一個佇列作業或是其子佇列可以使用指令，像是 qmod(1) 或 qdel(1) 來一次存取。若要得知進一步的詳細資訊，請參閱對應的線上說明手冊。</p> <p>作業陣列通常用於執行相同類型的操作在不同的輸入資料組上，且以作業索引來互相關聯。一個陣列作業的作業數量是沒有限制的。</p> <p>陣列作業的 STDOUT 與 STDERR，會使用預設的位置寫入不同的檔案：</p> <p>< 作業名稱>.['e' 'o']< 作業 id>'. '< 工作 id></p> <p>為了變更此預設值，-e 以及 -o 選項（請參閱上述說明）藉由虛擬的環境變數 \$HOME、\$USER、\$JOB_ID、\$JOB_NAME、\$HOSTNAME 以及 \$SGE_TASK_ID，可以在一起使用。</p> <p>您可以使用輸出重新導向來轉移所有作業的輸出至相同的檔案中，但是此結果尚未被定義。</p>
-u 使用者名稱,... -uall	<p>只能用於 qalter。只有使用者在 使用者名稱清單 中指定提交的那些作業會進行變更。對管理員而言，可能會使用 qalter -uall 指令修改所有使用者的所有作業。</p> <p>若是您使用 -u 或 -uall 開關，則不允許指定一個額外的 作業 / 工作 id 清單。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-v 變數 [= 數值],...	<p>只能用於 qsub、qrsh、qsh、qlogin、qresub、以及 qalter。</p> <p>定義或重新定義欲匯出至作業之執行上下文的環境變數。若是有使用 -v 選項，Sun Grid Engine 會將定義為引數的環境變數新增至開關處，並且將指定變數的值新增至作業之執行上下文中，不過其是選擇性的。</p> <p>qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啓動或是移轉作業之後才會生效。</p>
-verbose	<p>只能用於 qrsh 以及 qmake(1)。</p> <p>不像 qsh 與 qlogin，當建立與標準 rsh(1) 和 rlogin(1) 系統呼叫相容的階段作業時，qrsh 不會輸出任何資訊訊息。若是有設定 -verbose 選項，qrsh 會以 qsh 與 qlogin 指令同樣的冗長方式運作，並輸出關於建立 rsh(1) 或 rlogin(1) 階段作業的處理程序的資訊。</p>
-verify	<p>只能用於 qsub、qrsh、qsh、qlogin、qresub 以及 qalter。</p> <p>這不會提交作業，但是會列出目前指令行所要求的作業資訊，以及所有有關的外部影響。</p>
-V	<p>只能用於 qresub、qsub、qrsh、qsh、qlogin、以及 qalter。</p> <p>指定所有在 qsub 公用程式內作用中的環境變數都要匯出至作業上下文。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
-w e w n v	<p>只能用於 qsub、qrsh、qsh、qlogin、qresub、以及 qalter。</p> <p>指定套用至欲提交作業 (qsub、qlogin、以及 qsh)，或是指定之佇列作業 (qalter) 的驗證層級。顯示的資訊指出，當有一個沒有其他作業的空間系統出現時，作業是否有可能進入排程。資源要求超過配置的最大臨界值，或是要求無法使用的資源屬性可能就是造成作業驗證失敗的原因。</p> <p>指定器 e、w、n、以及 v 可定義下列的驗證模式：</p> <p>e - 錯誤：使用無效要求的作業將會被拒絕；這是 qrsh、qsh、以及 qlogin 的預設值。</p> <p>w - 警告：只會針對無效的要求顯示警告。</p> <p>n - 無：關閉驗證；為 qalter 以及 qsub 的預設值。</p> <p>v - 檢驗：不會提交作業，但是會列出大量的驗證報告。</p> <p>必要的檢查將消耗效能，因此預設值為關閉檢查。</p> <p>有關佇列資源可用性之作業需求無效的原因顯示在 -w v 案例中，其使用針對 qstat(1)-F 選項所說明的格式 (請參閱 qstat(1) 線上說明手冊之「輸出格式」章節中的「完整格式」說明)。</p>
作業 / 工作 id 清單	<p>以下列格式來指定：</p> <p>作業 id[. 工作範圍][, 作業 id [. 工作範圍] ,...]</p> <p>若提出指定在作業識別碼之後的工作範圍，就會限制了被指定為作業識別碼字尾的陣列作業之工作範圍操作的效力 (若要得知作業陣列進一步的詳細資訊，請參閱 qsub(1) 的 -t 選項)。</p> <p>工作範圍指定器的格式為 n[-m[:s]]。範圍可以是單一數字、簡單的範圍格式 n-m 或是一個步驟大小的範圍。</p> <p>有可能會使用關鍵字 all 來替代作業 / 工作 id 清單以修改目前使用者的所有作業。</p>
script 檔案	<p>只能用於 qsub。</p> <p>該作業的 script 檔案。若不存在，或是運算域為單一字元字串，則 -，qsub 會從標準輸入讀取 script。</p>

表 25 submit 指令選項 (接上頁)

選項	說明
script 引數	只能用於 qsub 以及 qalter。 作業的引數。若 script 是從標準輸入取得則無效。 qalter 允許變更此選項，即使當作業正在執行時。然而，修改的參數只有在重新啓動或是移轉作業之後才會生效。
xterm 引數	只能用於 qsh。 如同在配置中所定義的 xterm(1) 執行檔案引數。若需詳細資訊，請參考 sge_conf(5)。

環境變數

表 26 列出與 submit 相關之環境變數

表 26 submit 環境變數

變數名稱	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位置。
SGE_CELL	若設定的話，請指定預設的 Sun Grid Engine 儲存格。若要定址 Sun Grid Engine 儲存格，qtcsd 會使用 (以優先順序列出)： <ul style="list-style-type: none"> 在環境變數 SGE_CELL 中指定的儲存格名稱，若是有設定的話。 預設儲存格的名稱；例如，default。
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 sge_commd(8) 的 TCP 埠。大部分的安裝將使用服務對映項目，而非定義該埠。
COMMD_HOST	若有設定，則指定特定 sge_commd(8) 所執行的主機，以使用於 qsub、qsh、qlogin 或 qalter 客戶端常駐的 Sun Grid Engine 通訊。單一預設值僅可使用本端主機。

除了這些經由 `-v` 或是 `-V` 選項（請參閱上述說明）以指定要匯出至作業的環境變數之外，`qsub`、`qsh`、以及 `qlogin` 會使用表 27 中所顯示的指定值，將下列變數新增至變數清單中。

表 27 額外的 `qsub`、`qsh`、以及 `qlogin` 環境變數

變數名稱	說明
<code>SGE_O_HOME</code>	提交客戶端的起始目錄。
<code>SGE_O_HOST</code>	提交客戶端正在執行的主機名稱。
<code>SGE_O_LOGName</code>	提交客戶端的 <i>LOGNAME</i> 。
<code>SGE_O_MAIL</code>	提交客戶端的郵件；這是提交客戶端的的郵件目錄。
<code>SGE_O_PATH</code>	提交客戶端的執行檔案搜尋路徑。
<code>SGE_O_SHELL</code>	提交客戶端的 <i>SHELL</i> 。
<code>SGE_O_TZ</code>	提交客戶端的時區。
<code>SGE_O_WORKDIR</code>	提交客戶端目前工作目錄的絕對路徑。

Sun Grid Engine 軟體也會設定額外的變數至作業環境中，如表 28 中所列出的。

表 28 藉由 `submit` 指令設定作業環境中的變數

變數名稱	說明
<code>ARC</code>	作業正在其中執行之 Sun Grid Engine 節點的架構名稱。名稱將編譯至 <code>sge_execd(8)</code> 二進位檔案內。
<code>SGE_CKPT_ENV</code>	指定檢查點作業執行在其下之檢查點環境（如同使用 <code>-ckpt</code> 選項所選取的）。只針對檢查點作業設定。
<code>SGE_CKPT_DIR</code>	只針對檢查點作業設定。包含路徑 <code>ckpt_dir</code> （請參閱檢查點介面的 <code>checkpoint(5)</code> ）。
<code>SGE_STDERR_PATH</code>	作業的標準錯誤串流轉移之檔案路徑名稱。通常都使用來自 <i>prolog</i> 、 <i>epilog</i> 、平行環境啟動 / 停止或檢查點 script 的錯誤訊息以增強輸出。

表 28 藉由 submit 指令設定作業環境中的變數 (接上頁)

變數名稱	說明
SGE_STDOUT_PATH	作業的標準輸出串流轉移之檔案路徑名稱。通常都使用來自 <i>prolog</i> 、 <i>epilog</i> 、平行環境啟動 / 停止或檢查點 script 的錯誤訊息以增強輸出。
SGE_JOB_SPOOL_DIR	<i>sge_shepherd</i> (8) 用來儲存在作業執行期間，與作業相關資料的目錄。root 或是 Sun Grid Engine 管理帳號擁有此目錄，並且通常不會開放讀取或寫入的存取權給一般使用者。
SGE_TASK_ID	目前陣列作業工作的索引編號 (請參閱上述的 -t 選項)。在每個陣列作業中這是唯一的編號，並且可以用來，例如，參考不同的輸入資料記錄。針對非陣列的作業，此環境變數設定為 <i>undefined</i> 。
ENVIRONMENT	該 ENVIRONMENT 變數會設定為 BATCH，以識別該作業是在 Sun Grid Engine 控制下執行。
HOME	來自 <i>passwd</i> (5) 檔案的使用者起始目錄路徑。
HOSTNAME	作業正在其中執行之節點的主機名稱。
JOB_ID	當提交作業時，藉由 <i>sge_qmaster</i> (8) 所指定之唯一的辨識器 (<i>identifier</i>)。作業的識別碼是一個十進位的整數，其範圍從 1 到 99999。
JOB_NAME	作業名稱；針對互動式作業其為 INTERACT，或是從 <i>qsub</i> script 名稱、一個期間以及作業識別碼的數字來建立。此預設值可以被 -N 選項所覆蓋。
LOGNAME	來自 <i>passwd</i> (5) 檔案的使用者登入名稱。
NHOSTS	平行作業正在使用之主機數量。
NQUEUES	配置給作業的佇列數量 (針對序列作業，其永遠是 1)。
NSLOTS	平行作業正在使用之佇列插槽的數量。
PATH	預設的 shell 搜尋路徑： /usr/local/bin:/usr/ucb:/bin:/usr/bin
PE	執行作業之下的平行環境 (只針對平行作業)。
PE_HOSTFILE	藉由 Sun Grid Engine 所指定至平行作業之虛擬平行機器，包含其定義之檔案路徑。請參閱 <i>sge_pe</i> (5) 中的 <i>Spe_hostfile</i> 參數說明以得知此檔案格式的詳細資訊。此環境變數僅供平行作業使用。

表 28 藉由 submit 指令設定作業環境中的變數 (接上頁)

變數名稱	說明
QUEUE	作業執行的佇列名稱。
REQUEST	只有批次作業可以使用。使用 -N 開關所指定之作業的要求名稱 (請參閱上述說明)，或當做是作業 script 的名稱。
RESTARTED	在檢查點作業的案例中，若是作業在系統當機或移轉之後重新啟動，此變數會設定為 1。否則該變數的值會是 0。
SHELL	來自 passwd(5) 檔案的使用者登入 shell。注意：這不見得就是作業使用中的 shell。
TMPDIR	作業暫時工作目錄的絕對路徑。
TMP	與 TMPDIR 相同；提供與 NQS 的相容性。
TZ	從 sge_execd(8) 匯入的時區變數，若有設定的話。
USER	來自 passwd(5) 檔案的使用者登入名稱。

限制

在 Sun Grid Engine 之下沒有批次作業的控制終端機，而且在控制終端機上的任何測試或動作將會失敗。若是在 .login 或 .cshrc 檔案中執行上述操作，就可能造成您的作業中斷。

在任何與您 .login 檔案中之批次作業無關的指令之前插入下列測試。

```
if ( $? 作業名稱 ) then
    echo "Sun Grid Engine spooled job"
    exit 0
endif
```

不要忘記在此程式碼之前，於 shell 啟動中設定您 shell 的搜尋路徑。

退出狀態

會傳回下列的退出值。

- 0 – 成功執行操作
- 25 – 根據配置的 *max_u_jobs* 或最大作業量限制是無法登錄一新增作業。在 `sgc_conf(5)` 中可以找到額外的資訊。
- >0 – 發生錯誤。

範例

下列為 Sun Grid Engine script 的最簡單格式。

```
#!/bin/csh
a.out
```

下一個範例為更複雜的 Sun Grid Engine script。

```
#!/bin/csh

# Which account to be charged cpu time
#$ -A santa_claus

# date-time to run, format [[CC]yy]MMDDhhmm[.SS]
#$ -a 12241200

# to run I want 6 or more parallel processes
# under the PE pvm. the processes require
# 128M of memory
#$ -pe pvm 6- -l mem=128

# If I run on dec_x put stderr in /tmp/foo, if I
# run on sun_y, put stderr in /usr/me/foo
#$ -e dec_x:/tmp/foo,sun_y:/usr/me/foo

# Send mail to these users
#$ -M santa@heaven,claus@heaven

# Mail at beginning/end/on suspension
#$ -m bes

# Export these environmental variables
#$ -v PVM_ROOT,FOOBAR=BAR

# The job is located in the current
# working directory.
#$ -cwd

a.out
```

檔案

- 作業 #JID 的 STDOUT – \$REQUEST.oJID[. 工作id]
- 作業的 STDERR – \$REQUEST.eJID[. 工作id]
- 作業參數環境的 STDOUT – \$REQUEST.poJID[. 工作id]
- 作業參數環境的 STDERR – \$REQUEST.peJID[. 工作id]
- 作業參數環境的主機檔案 – \$REQUEST.hostsJID[. 工作id]
- cwd 路徑更名 – \$cwd/.sge_aliases
- cwd 預設要求 – \$cwd/.sge_request
- 使用者路徑更名 – \$HOME/.sge_aliases
- 使用者預設要求 – \$HOME/.sge_request
- 叢集路徑更名 – <sge_root>/<儲存格>/common/.sge_aliases
- 叢集預設要求 – <sge_root>/<儲存格>/common/.sge_request
- Sun Grid Engine 主要主機檔案 – <sge_root>/<儲存格>/common/act_qmaster

同時參閱

sge_intro(1)、qconf(1)、qdel(1)、qhold(1)、qmod(1)、qrls(1)、
qstat(1)、accounting(5)、sge_aliases(5)、sge_conf(5)、
sge_request(5)、sge_pe(5)、complex(5)

版權

若是相對應的進行配置，qrsh 以及 qlogin 會包含 rsh、rshd、telnet 與 telnetd 程式碼的一部份，其版權是由美國加州大學董事會所提供。因此，qrsh 以及 qlogin 須套用下列的注意事項：此產品包含加州大學、柏克萊大學以及其貢獻者所開發的軟體。

請參閱 sge_intro(1) 以及 <sge_root>/3rd_party/qrsh 與 <sge_root>/3rd_party/qlogin 中所提供的資訊，以得知進一步的權利與權限聲明。

access_list(5)

名稱

access_list — Sun Grid Engine 存取清單檔案格式

說明

存取清單是使用在 Sun Grid Engine 產品中，以定義使用者對佇列（請參閱 `queue_conf(5)`）或平行環境（請參閱 `sge_pe(5)`）的存取權限。目前配置的存取清單之列表可以經由 `qconf(1) -sul` 選項來顯示。每個列表之存取清單的內容可以經由 `-su` 開關來顯示。輸出資訊會遵循 `access_list` 的格式說明。可以建立新增的存取清單，並且可以經由 `qconf(1)` 的 `-au` 以及 `-du` 選項來修改現有的存取清單。

格式

每個使用者或 UNIX 使用者群組會出現在單一資料行中。只允許符號名稱。可以在群組名稱之前使用「@」符號，以便區分使用者名稱與群組名稱。

同時參閱

`sge_intro(1)`、`qconf(1)`、`sge_pe(5)`、`queue_conf(5)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

accounting(5)

名稱

accounting — Sun Grid Engine 計算檔案格式

說明

針對每個已完成的作業，會在 Sun Grid Engine accounting 檔案寫入計算記錄。qacct(1) 會處理 accounting 檔案，以導出計算統計資料。

格式

每個作業是藉由 accounting 檔案中的一個資料行來代表。會忽略空白的資料行，以及包含一個或更少字元的資料行。計算記錄項目是藉由冒號 (:) 符號來做區隔。在下列說明中，會依出現的順序來顯示各項目。

- qname — 這是作業在其中執行的佇列名稱。
- hostname — 這是執行主機的名稱。
- group — 此為執行作業時，作業所有人的有效群組識別碼。
- owner — 此為 Sun Grid Engine 作業的所有人。
- job_name — 此為作業名稱。
- job_number — 此為作業識別碼或作業名稱。
- account — 此為藉由 qsub(1) 或 qalter(1) -A 選項所指定的計算字串。
- priority — 這是對應於佇列配置中的 priority 參數，所指定給作業的優先順序值（請參閱 queue_conf(5)）。
- submission_time — 這是以秒為單位的提交時間（起始於 epoch 格式）。
- start_time — 這是以秒為單位的啟動時間（起始於 epoch 格式）。
- end_time — 這是以秒為單位的結束時間（起始於 epoch 格式）。
- failed — 指出若是作業無法在執行主機上啟動所發生的問題（例如，因為作業的所有人在該機器上沒有有效帳號）。若是 Sun Grid Engine 軟體多次嘗試啟動作業，其可能會導致在計算檔案中，有多重項目相對應於相同的作業識別碼。
- exit_status — 這是作業 script 的退出狀態（或是在某些錯誤狀況發生時，Sun Grid Engine 的特定狀態）。

- `ru_wallclock` — 這是 `end_time` 與 `start_time` 之間的差額（請參閱上述說明）。

在此類別中，剩餘的計算項目遵循 `getrusage(2)` 中所說明之標準 UNIX `rusage` 結構的內容。提供下列項目。

- `ru_utime`
- `ru_stime`
- `ru_maxrss`
- `ru_ixrss`
- `ru_ismrss`
- `ru_idrss`
- `ru_isrss`
- `ru_minflt`
- `ru_majflt`
- `ru_nswap`
- `ru_inblock`
- `ru_oublock`
- `ru_msgsnd`
- `ru_msgrcv`
- `ru_nsignals`
- `ru_nvcsw`
- `ru_nivcsw`
- `project` — 這是指定至作業的專案。只有在 Sun Grid Engine，企業版系統中才支援專案。
- `department` — 這是指定至作業的部門。只有在 Sun Grid Engine，企業版系統中才支援部門。
- `granted_pe` — 這是針對該作業所選取的平行環境。
- `slots` — 這是排程器分派給作業的插槽數。
- `task_number` — 這是陣列作業工作的索引編號。
- `cpu` — 這是以秒為單位的 `cpu` 時間用量。
- `mem` — 這是以 GB 為單位的整體記憶體用量。
- `io` — 這是在輸入 / 輸出操作中所傳輸的資料量。
- `category` — 這是指定作業類別的字串。
- `iow` — 這是以秒為單位的 I/O 等待時間。
- `pe_taskid` — 若是設定此辨識器，則該作業為平行作業的一部份，並經由 `qrsh -inherit` 介面傳送至 Sun Grid Engine，企業版系統中。
- `maxvmem` — 這是以位元組為單位的最大 `vmem` 大小。

同時參閱

`sge_intro(1)`、`qacct(1)`、`qalter(1)`、`qsub(1)`、`getrusage(2)`、`queue_conf(5)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

calendar_conf(5)

名稱

`calendar_conf` — Sun Grid Engine 行事曆配置檔案格式

說明

`calendar_conf` 會反映 Sun Grid Engine 行事曆配置的格式。行事曆的定義是用來指定 Sun Grid Engine 佇列的「作業中 (on duty)」與「結束作業 (off duty)」時間期間，以當日的時數，當週的天數或該年的天數為計算基準。可以實施不同的行事曆，並且某些作業類別之適當的行事曆定義可以隨附至佇列中。

`calendar_conf` 可以使用 `qconf(1)` 的 `-Acal`、`-acal`、`-Mcal`、`-mcal`、`-scal` 以及 `-scall` 選項，或是圖形使用者介面 `qmon(1)` 的行事曆配置對話方塊，來新增、修改與顯示項目。行事曆配置項目的格式如下述所定義：

格式

- `calendar_name` — 這是當隨附至佇列，或管理行事曆定義時欲使用的行事曆名稱。
- `year` — 這是以該年的天數為基準之佇列狀態定義。此欄位一般將會指定行事曆所附接的佇列，在一年的那幾天（以及可選擇性地，那些指定日的那些時間）將會變更至某些狀態。年份欄位的語法如下述所定義：

```
year :=
```

```
{ 年_日期範圍清單 [= 當日範圍清單] [= 狀態]
```

[[年_日期範圍清單 =] 當日範圍清單 [= 狀態]

[[年_日期範圍清單 =] [當日範圍清單 =] 狀態 } ...

其中：

- 至少年_日期範圍清單、當日範圍清單以及狀態 其中一個必須永遠存在。
- 若忽略年_日期範圍清單，則假定為該年的每一天。
- 若忽略當日範圍清單，則假定為整天。
- 若忽略狀態，則假定將佇列切換至「關閉」（例如，停用之）。
- 不論是針對不明確（藉由忽略年_日期範圍清單）或明確的參照天數，佇列會假定為啟動。

並且按照句法的元件如下述所定義：

年_日期範圍清單 := { 年_日期 - 年_日期 | 年_日期 } ...

當日範圍清單 := 時[:分][:秒] - 時[:分][:秒] ...

狀態 := { on | off | suspended }

年_日期 := 月日.月.日

月_日 := { 1 | 2 | ... | 31 }

月份 := { jan | feb | ... | dec | 1 | 2 | ... | 12 }

年份 := { 1970 | 1971 | ... | 2037 }

- week — 此為以一週的天數為基準之佇列狀態定義。此欄位一般將會指定行事曆所隨附的佇列，在一週的哪幾天（及選擇日的某特定時間）將會變更至某特定狀態。

week 欄位的語法如下述所說明：

week :=

{ 週_日範圍清單 [= 當日範圍清單] [= 狀態]

[[週_日範圍清單 =] 當日範圍清單 [= 狀態]

[[週_日範圍清單 =] [當日範圍清單 =] 狀態 } ...

其中

- 至少週_日範圍清單、當日範圍清單以及狀態 其中一個必須永遠存在。
- 若忽略週_日範圍清單，則假定為該週的每一天。
- 當日範圍清單以及狀態的語法與語義，與上述的年份欄位所指定之定義相同。
- 不論是針對不明確（藉由忽略週_日範圍清單）或明確的參照天數，佇列會假定為啟動。

並且其中週_日範圍清單 會定義為

週_日範圍清單 := { 平日 - 平日 | 平日 },...

週_日 := { mon | tue | wed | thu | fri | sat | sun }

語義

依照下列規則組合年份與週欄位的依序項目（以空白分隔）：

- off — 藉由重疊 on 以及 suspended 區域來覆蓋區域。

因此，使用下列格式的項目，week 12-18 tue=13-17=on 表示參照相對應行事曆的佇列在整週內都已停用，除了星期二的 13.00-17.00，當時該佇列是可以使用的。

範例

（下列範例包含在以下目錄中，`$sge_root/util/resources/calendars`）。

- 夜間，週末以及國定假日行事曆 — 在國定假日中，「夜間」佇列是確定啟動的。在工作日中，佇列於 6.00 至 20.00 之間是停用的。星期六與星期日將間接地處理為啟動時間。

```
calendar_namenight
year
1.1.1999,6.1.1999,28.3.1999,30.3.1999-31.3.1999,18.5.1999-19.5.1999,
3.10.1999,25.12.1999,26.12.1999=on
weekmon-fri=6-20
```

- 白天行事曆 (day calendar) — 在國定假日中，「白天」佇列是停用的。在工作日中，於夜間的 20.00 至 6.00 之間，佇列是關閉的。換言之，在星期一的 0.00 到 6.00，以及星期五的 20.00 到 24.00，該佇列也是關閉的。在星期六與星期日該佇列是停用的。

```
calendar_nameday
year
1.1.1999,6.1.1999,28.3.1999,30.3.1999-31.3.1999,18.5.1999-19.5.1999,
3.10.1999,25.12.1999,26.12.1999
weekmon-fri=20-6 sat-sun
```

- 在暫停使用的夜間、週末以及國定假日行事曆 — 基本上與第一個範例的情況相同，但是會以暫停佇列來替代將其「關閉」。

```
calendar_namenight_s
year
1.1.1999,6.1.1999,28.3.1999,30.3.1999-31.3.1999,18.5.1999-19.5.1999,
3.10.1999,25.12.1999,26.12.1999=on
weekmon-fri=6-20=suspended
```

- 暫停使用的白天行事曆 — 基本上與第二個範例的情況相同，但是會以暫停佇列來替代將其「關閉」。

```
calendar_nameday_s
year
1.1.1999,6.1.1999,28.3.1999,30.3.1999-31.3.1999,18.5.1999-19.5.1999,
3.10.1999,25.12.1999,26.12.1999=suspended
week          mon-fri=206=suspended sat-sun=suspended
```

同時參閱

`sge_intro(1)`、`qconf(1)`、`queue_conf(5)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

checkpoint(5)

名稱

檢查點 — Sun Grid Engine 檢查點環境配置檔案格式

說明

檢查點是儲存執行中程式或作業之完整狀態的工具，若是原始程式由於某種原因停止，例如系統當機，則其可以稍後從所謂的檢查點處復原與重新啓動。

Sun Grid Engine 提供各種層級的檢查點支援（請參閱 `sge_ckpt(1)`）。在此說明的檢查點環境，是一種針對您的 Sun Grid Engine 叢集或其零件，配置使用中之不同類型檢查點的工具。針對該目的，您可以定義在初始化產生檢查點、將檢查點移轉至另一個主機、重新啓動已完成檢查點的應用程式時所必須執行的操作、且您也可以定義符合檢查點方法資格的佇列清單。

支援不同的作業系統可能會輕易促使 Sun Grid Engine 針對檢查點配置檔案的配置，引進作業系統相依性；並且更新支援的作業系統版本，可能會導致經常性的變更執行詳細資訊。請參考此檔案，`<sge_root>/doc/checkpointing.asc` 以得知更多資訊。

使用 `qconf(1)` 指令的 `-ackpt`、`-dckpt`、`-mckpt` 或 `-sckpt` 選項，以便從指令行操作檢查點環境；或使用相對應之 X- 視窗架構的互動式配置 `qmon(1)` 對話程序來操作檢查點環境。

格式

checkpoint 檔案的格式如下述所定義：

- `ckpt_name` — 這是檢查點環境的名稱。使用在 `qsub(1) -ckpt` 開關中，或針對上述所提及的 `qconf(1)` 選項。
- `interface` — 這是欲使用之檢查點類型。目前，下列的類型是有效的：
 - `hibernator` — 這是檢查點當做介面的 Hibernate 核心層次。
 - `cpr` — 這是檢查點所使用的 SGI 核心層次。
 - `cray-ckpt` — 這是檢查點所假定的 Cray 核心層次。
 - `transparent` — Sun Grid Engine 系統假定與此檢查點介面有關之提交作業，使用像是公用網域套裝軟體 *Condor* 所提供之檢查點程式庫。
 - `userdefined` — Sun Grid Engine 系統假定與此檢查點介面有關之提交作業會執行其個別的檢查點方法。
 - `application-level` — 其會使用在檢查點物件中所配置的所有介面指令，像是核心層級檢查點介面之一（`cpr`、`cray-ckpt`，等等），除了 *重新啓動指令*（請參閱下述說明）之外，其不會被使用（就算已配置），但是一旦發生重新啓動時會呼叫作業 `script` 來替代。
- `queue_list` — 這是一個以逗號分隔、屬於此平行環境的平行作業擁有存取權的佇列清單。
- `ckpt_command` — 這是 Sun Grid Engine 所執行之指令行類型的指令字串，以便初始檢查點。
- `migr_command` — 這是 Sun Grid Engine 在檢查點作業從一個主機移轉至另一個主機的期間，所執行之指令行類型的指令字串。

- `restart_command` — 這是當重新啓動之前已完成檢查點的應用程式時，Sun Grid Engine 所執行之指令行類型的指令字串。
- `clean_command` — 這是 Sun Grid Engine 執行之指令行類型的指令字串，以便在已完成檢查點的應用程式結束時進行清除。
- `ckpt_dir` — 這是潛在需注意大小的檢查點應該儲存之檔案系統位置。
- `queue_list` — 若是在提交作業時已指定檢查點環境，其包含一個針對作業爲合格的、以逗號或空白來做分隔之佇列名稱清單。
- `ckpt_signal` — 這是欲藉由 Sun Grid Engine 傳送給作業的 UNIX 信號，以初始產生檢查點。此欄位的值可以是來自 `kill(1)` 指令的 `-l` 選項所產生清單的符號名稱，或是一個整數，其在用來執行檢查點的系統上必須是有效信號。
- `when` — 當預期產生檢查點時，其會建立時間點。此參數的有效值是由字母 `s`、`m`、`x` 與 `r` 所組成，而且任何組合在字母之間都沒有分隔字元。`qsub(1)` 指令的 `-c` 選項也允許使用相同的字母，其將會覆蓋所使用之檢查點環境中的定義。字母的意義如下述所定義：
 - `s` — 若是在作業的機器上關閉對應的 `sge_execd(8)`，則此字母會對作業進行檢查點、中斷以及如果可能的話，移轉的動作。
 - `m` — 使用此字母，會依據作業在其中執行之佇列所定義的 *最小 cpu 間隔時間* 間隔以週期性的產生檢查點（請參閱 `queue_conf(5)`）。
 - `x` — 使用此字母，只要作業一經暫停（不論是手動與自動）便會對作業進行檢查點、中斷以及如果可能的話，移轉的動作。
 - `r` — 使用此字母，當作業目前執行的主機狀態進入未知，並且超過在全域 / 本端叢集配置中所定義之時間間隔時間 `reschedule unknown`（請參閱 `sge_conf(5)`）時，作業將會被重新排程（而非進行檢查點動作）。

限制

注意 — 不應該變更任何伴隨 Sun Grid Engine 銷售所提供之預設檢查點、移轉或重新啓動程序的功能，以及其在任何預設檢查點環境的 `ckpt_command`、`migr_command`、或 `restart_command` 參數中的啓動方法，否則配置檢查點環境的管理員需負起該功能性的完全責任。Sun Grid Engine 軟體將只會啓動這些程序並評估其退出狀態。若是該程序沒有正確地執行作業，或是無法以正確的方法啓動，則檢查點機制可能會有無法預期的運作方式。Sun Grid Engine 系統沒有偵測此錯誤的方法。

同時參閱

`sge_intro(1)`、`sge_ckpt(1)`、`qconf(1)`、`qmod(1)`、`qsub(1)`、`sge_execd(8)`

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

sge_request(5)

名稱

`sge_request` – Sun Grid Engine 預設要求定義檔格式

說明

`sge_request` 反映檔案的格式以定義預設的要求設定檔。若是可用的話，在考量任何內嵌於作業 `script` 中的提交選項，以及任何 `qsub(1)` 或 `qsh(1)` 指令行中的選項之前，於作業提交期間會讀取並處理預設的要求檔案。因此，指令行與內嵌的 `script` 選項會覆蓋預設要求檔案中的設定（請參閱 `qsub(1)` 或 `qsh(1)` 以得知詳細資訊）。

有叢集全域的、使用者個人的以及工作目錄本端預設的要求定義檔案。工作目錄本端預設的要求定義檔案擁有最高的優先權，跟隨其後的是使用者個人 (`user private`)，然後才是叢集全域的預設要求檔案。

注意 – `qsub(1)` 或 `qsh(1)` 的 `-clear` 選項可以用來在任何時間刪除預設要求檔案、內嵌的 `script` 旗號或 `qsub(1)` 或 `qsh(1)` 指令行選項中任何先前的設定。

預設要求定義檔案的格式如下所述。

- 預設要求檔案會包含任意數量的資料行。會略過空白的資料行以及在第一個欄位中使用「#」符號的資料行。
- 每個不被忽略的資料行會包含任何《*Sun Grid Engine 5.3 和 Sun Grid Engine，企業版 5.3 參考手冊*》中所說明的 `qsub(1)` 選項。每行可以允許一個以上的選項。批次 `script` 檔案以及批次 `script` 的引數選項將不會被視為 `qsub(1)` 選項，因此並不允許在預設要求檔案中使用。

範例

下列是預設要求定義檔案的簡單範例：

```
# Default Requests File
# request arch to be sun4 and a CPU-time of 5hr
-l arch=sun4,s_cpu=5:0:0
# don't restart the job in case of system crashes
-r n
```

已定義像這樣的預設要求定義檔案並提交如下所述的作業：

```
qsub test.sh
```

若是作業使用下述方式提交的話，將會擁有正好相同的效果：

```
sub -l arch=sun4,s_cpu=5:0:0 -r n test.sh
```

檔案

- 全域預設檔案 — `<sgc_root>/<儲存格>/common/sgc_request`
- 使用者個人預設檔案 — `$HOME/.sgc_request`
- cwd 目錄預設檔案 — `$cwd/.sgc_request`

同時參閱

`sgc_intro(1)`、`qsh(1)`、`qsub(1)`、《*Sun Grid Engine*，企業版 5.3 管理和使用者指南》、《*Sun Grid Engine 5.3 管理與使用者指南*》

版權

請見 `sgc_intro(1)` 以得知權利與權限的完整聲明。

sge_aliases(5)

名稱

sge_aliases — Sun Grid Engine 路徑更名檔案格式

說明

Sun Grid Engine 路徑更名工具提供管理員與使用者方法，以反映分散式環境中的複雜與異質檔案系統結構（像是裝載於不同主機上之不同路徑的使用者初始目錄），並確定 Sun Grid Engine 能夠定位適當的工作目錄以執行批次作業。

有系統全域路徑更名檔案以及使用者本端檔案。sge_aliases 定義兩者的格式：

- 會略過空白的資料行以及在第一個欄位中使用「#」符號的資料行。
- 每個不是空白的資料行，或是並非以「#」當成起始字元的資料行，必須包含以任何數量的空白或標籤來分隔的四個字串。
- 第一個字串指定資源路徑、第二個是提交主機、第三個為執行主機、而第四個為資源替代路徑。
- 提交主機與執行主機項目兩者可能都只是以符合任何主機的「*」符號所組成。

若是指定 qsub(1) 的 -cwd 旗號（並且是唯一在指定此旗號時 — 否則會選取執行主機上的使用者起始目錄來執行作業），便會啟動路徑更名機制，並且檔案將如下述處理：

- 在 qsub(1) 擷取目前工作目錄的實體路徑後，若存在於叢集全域路徑更名檔案，則讀取之。之後便會讀取使用者路徑更名檔案，如同附加在全域檔案之後。
- 將從檔案的頂端一個一個讀取不被忽略的資料行，若是需要的話，將儲存這些資料行所指定的轉換翻譯。
- 只有在提交主機項目符合 qsub(1) 所執行的主機、來源路徑構成目前作業路徑或是已儲存之來源路徑取代方案的初始部份時，才會儲存轉換翻譯。
- 只要兩個檔案皆被讀取，儲存之路徑更名資訊就會與提交作業一同傳送出去。
- 在執行主機中會評估更名資訊。如果更名路徑的執行主機記錄符合執行主機，最新工作目錄之第一部分將會由資源替代路徑取代。

注意 – 在此案例中將會變更目前作業目錄字串，並且之後的路徑更名必須符合欲套用之置換的工作目錄路徑。

範例

若是使用 automount(8)，下列是一個伴隨異質 (in-homogeneous) 路徑之路徑更名檔案解析問題的簡單範例：

```
# Path Aliasing File
# src-paths sub-host exec-host replacement
/tmp_mnt/ ** /
# replaces any occurrence of /tmp_mnt/ by /
# if submitting or executing on any host.
# Thus paths on nfs server and clients are the same
```

檔案

- 全域別名檔案 — `<sg_e_root>/<儲存格>/common/sg_e_aliases`
- 使用者本端更名檔案 — `$HOME/.sg_e_aliases`

同時參閱

`sg_e_intro(1)`、`qsub(1)`、《*Sun Grid Engine，企業 5.3 管理和使用者指南*》以及《*Sun Grid Engine 5.3 管理和使用者指南*》

版權

請見 `sg_e_intro(1)` 以得知權利與權限的完整聲明。

`sg_e_conf(5)`

名稱

`sg_e_conf` — Sun Grid Engine 配置檔案

說明

`sge_conf` 可定義全域與本端 Sun Grid Engine 配置，還可藉由使用 `-sconf` 與 `-mconf` 選項的 `qconf(1)` 來顯示與修改配置。

於初始啓動時，`sge_qmaster(8)` 會檢視有效的 Sun Grid Engine 的配置是否存在於已知之 Sun Grid Engine 內部目錄階層。若如此，它將載入配置資訊然後進行。若否，`sge_qmaster(8)` 會撰寫含有預設值一般性的配置檔案至相同的位置中。Sun Grid Engine 執行常駐程式 `sge_execd(8)` 會於啓動時自 `sge_qmaster(8)` 擷取其配置。

`sge_qmaster(8)` 及 `sge_execd(8)` 的實際配置即為所謂的全域配置及本端配置的重疊，適合於主要及執行常駐程式所在的主機。如果本端配置是有效的，它的項目會改寫對應的全域配置之項目。

注意 – 本端配置不需要包括每一項有效的配置項目，只需要那些須針對全域項目作修改的項目。

格式

下述段落會簡述組成 Sun Grid Engine 叢集的全域及本端配置的個別參數。

- `qmaster_spool_dir` – 這裡是主要多工緩衝處理目錄 (spool directory) 所在的位置。只有 `sge_qmaster(8)` 及 `sge_shadowd(8)` 需要存取此目錄。但是它需要 `root` 的讀 / 寫權限。主要多功能緩衝目錄 – 特別是作業目錄及訊息日誌檔案 – 根據叢集的大小及作業的數目可能會變的非常大。確定要配置足夠的硬碟空間，並經常清除日誌檔案，例如透過 `cron(8)` 作業。

由於它是安裝時的參數組，所以並不支援於系統運作時變更 `qmaster_spool_dir`。

主要多工緩衝目錄預設的位置是：`<sge_root>/<儲存格>/spool/qmaster`

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- `execd_spool_dir` – 此為執行常駐程式多工緩衝目錄路徑。同樣地，一個可用的多工緩衝目錄也需要 `root` 的讀 / 寫存取權限。此參數於全域配置中的項目可由執行主機本端配置改寫，意即每一 `sge_execd(8)` 可能有不同路徑的個別多工緩衝目錄，並只需要提供對應執行主機的 `root` 帳號讀 / 寫權限。

於 `execd_spool_dir` 之下會開啓一個以對應執行主機之不完整主機名稱 (unqualified hostname) 而命名的目錄，及包含多工緩衝至硬碟的所有資訊。因此，所有執行主機的 `execd_spool_dirs` 仍可以具體的參照相同的目錄路徑（但仍需遵照上述的 `root` 存取限制）。

由於它是安裝時的參數組，所以並不支援於系統運作時變更 `execd_spool_dir`。

執行常駐程式多工緩衝目錄之預設路徑為 `<sge_root>/<儲存格>/spool`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **二進位路徑** — 此為 Sun Grid Engine 二進位檔案所在的目錄路徑，用以於 Sun Grid Engine 元件中尋找及啟動其他 Sun Grid Engine 程式。

這裡出現的路徑名稱會用以尋找二進位檔案，且下述有目錄名稱之任何目錄名稱皆目前的作業系統架構相同。因此，如果對應的二進位檔案位於名稱為 aix43、cray、glinux、hp10、irix6、osf4、solaris、及其它子目錄，/usr/Sun Grid Engine/bin 將適用於所有的架構。

每一個 sge_execd(8) 可以有它個別的二進位路徑。變更二進位路徑可使 sge_execd(8) 立即生效。

二進位檔案路徑的預設位址為：<sge_root>/bin

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **郵件程式** — 此為系統上電子郵件傳遞代理程式的絕對路徑名稱。它必須接受下列語法：

```
mailer -s <郵件主題> <收信人>
```

每一個 sge_execd(8) 皆可以使用個別特定的郵件代理程式。變更郵件程式可以立即生效。

郵件程式的預設值會依據執行 Sun Grid Engine 主要安裝之主機作業系統而定。常用的值為 /bin/mail 或 /usr/bin/Mail。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **xterm** — 此為 X Window 系統終端模擬器 xterm(1) 之絕對路徑名稱。

每一個 sge_execd(8) 可使用個別郵件代理程式。變更 xterm 可立即生效。

xterm 的預設值為：/usr/bin/X11/xterm

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **載入感應器** — 此為可由 sge_execd(8) 所啟動，由逗點分隔之可執行檔案 shell script 路徑或程式清單，及可用來擷取站點可配置之負載資訊（例如，特定硬碟分割區之可用空間）。

每一個 sge_execd(8) 可能會使用一組個別載入感應器程式或 script。變更載入感應器會在二個載入報告間隔時間之後生效。（請見 *load_report_time*）如果載入感應器的可執行檔案修改時間變更，載入感應器便會自動重新啟動。

此數值於全域配置中的項目可由執行主機本端配置改寫。

除了透過載入感應器配置的載入感應器，sge_execd(8) 會尋找位於執行主機 Sun Grid Engine 二進位目錄路徑中，名為 qloadsensor 的執行檔案。如果找到此檔案，便會將其視為如定義於載入感應器中的可配置載入感應器。這個功能主要用意在於預先安裝一個預設載入感應器。

- **prolog** — 為執行 Sun Grid Engine 作業前先啟動的 shell script 執行檔案路徑，其環境設定與之後要執行的 Sun Grid Engine 作業設定是相同的。字首「user@」（非必要項目）會在欲啟動之程序底下指定使用者。此程序被視為 Sun Grid Engine 管理員用以自動執行一般站點特定作業（如：依據與作業相同內容資訊來準備暫停檔案

系統)的方法。每一個 `sge_execd(8)` 可使用個別的 *prolog* script。相對地，執行主機之本端配置可由佇列之配置改寫 (請見 `queue_conf(5)`)。變更 *prolog* 將立即生效。

注意 – *prolog* 會完全被視為作業 script 來執行。因此，所有說明於參數 *shell 啟動模式* 與登入 *shell* 底下的含義都適用。

prolog 的預設值為特殊值 NONE，可避免執行 *prologue* script。

下述特殊變數會在執行期間被擴充，並可用來 (除了須由程序解讀之其它字串之外) 建構一指令行：

- *\$host* – 此為啟動 *prolog* 或 *epilog* 程序上的主機名稱。
- *\$job_owner* – 此為作業所有人的使用者名稱。
- *\$job_id* – 此為 Sun Grid Engine 系統的獨特作業識別碼。
- *\$job_name* – 此為作業名稱。
- *\$processors* – 處理器字串正如包含在主要佇列 (即啟動 *prolog* 及 *epilog* 程序之佇列) 中的佇列配置 (請見 `queue_conf(5)`)。
- *\$queue* – 主要佇列，即啟動 *prolog* 及 *epilog* 程序的佇列。此數值於全域配置中的項目可由執行主機本端配置改寫。
- *epilog* – 在執行 Sun Grid Engine 作業與和完成 Sun Grid Engine 作業相同之環境設定後，此為啟動之 *shell* script 的可執行路徑。字首 `user@` (非必要項目) 會在欲啟動之程序底下指定使用者。此程序被視為 Sun Grid Engine 管理員用以自動執行一般站點特定作業 (如：依據與作業相同內容資訊來清除暫停檔案系統) 的方法。每一個 `sge_execd(8)` 可使用個別的 *epilog* script。相對地，執行主機之本端配置可由佇列之配置改寫 (請見 `queue_conf(5)`)。變更 *epilog* 將立即生效。

注意 – *epilog* 會完全被視為作業 script 來執行。因此，所有說明於參數 *shell 啟動模式* 與登入 *shell* 底下的含義都適用。

epilog 的預設值為特殊值 NONE，可避免執行 *epilogue* script。*prolog* 中相同之特別變數可用來組成一指令行。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *shell 啟動模式* – 此參數會定義用於執行主機上實際啓動作業 scripts 的機制。下述為可辨識的值：
 - *unix_behavior* – 如果使用者於 UNIX 底下以 script 名稱來互動地啓動作業 script，此作業系統的執行載入器 (executable loader) 會使用註釋中的資訊，如 script 第一行中的「`#!/bin/csh`」，來偵測用哪個指令解讀器啓動解讀 script。
 - *posix_compliant* – POSIX 並不會將 script 的第一行註釋，如「`#!/bin/csh`」，視為重要訊息。因此，批次佇列系統的 POSIX 標準 (P1003.2d) 需要一相容之佇列系統以忽略第一行註釋，並轉而利用使用者指定或配置之預設解讀器。因此，如

果 *shell 啟動模式* 設定為 `posix_compliant`，Sun Grid Engine 就會使用指令解讀器，由 `qsub(1)` 指令的 `-S` 選項，或使用佇列之 *shell* 參數指定（請見 `queue_conf(5)` 得知詳細資訊）。

- `script_from_stdin` — 要設定 *shell 啟動模式* 參數至 `posix_compliant` 或 `unix_behavior` 時，需要先設定 `sge_execd(8)` 執行 `umask`，如此一來所有使用者才可讀到存取至對應執行常駐程式之多工緩衝處理目錄中的 `active_jobs` 目錄。如果您有配置 *prolog* 與 *epilog* script，這兩個 script 也必須讓可能執行作業的使用者讀到。

如果這侵犯了您的站點的安全策略，您也許可設定 *shell 啟動模式* 至 `script_from_stdin`。這會強制 Sun Grid Engine 在變更至作業擁有人的使用者帳號前，開啓作業 script 與 *pilogue* 和 *prologue* scripts，以 `root` 讀入 STDIN（如果 `sge_execd(8)` 是以 `root` 啟動）。然後，此 script 就會供入指令解讀器的 STDIN 串流之中。指令解讀器是由 `qsub(1)` 指令之 `-S` 選項，或由使用佇列之 *shell* 參數指定（請見 `queue_conf(5)` 得知詳細資訊）。

因此，要設定 *shell 啟動模式* 至 `script_from_stdin` 也必須包含 `posix_compliant` 的運作方式。

注意 — 如果諸如 `rsh(1)` 的指令啟動於作業 script 中，而你又執行指令解讀器的 STDIN 串流，將 scripts 供入指令解讀器的 STDIN 中也許會產生麻煩。解決方法為，將這些指令的 STDIN 通道重新導向給 `/dev/null`（如，`rsh 主機 日期 < /dev/null`）。

注意 — 任何與作業相關之指令行選項會通過至執行中的 shell。如果這些指令行選項無法被辨視為有效 shell 選項，此 shell 就只能將其轉寄至作業。

變更 *shell 啟動模式* 將立即生效。*shell 啟動模式* 的預設值為 `posix_compliant`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *登入 shell* — UNIX 指令解譯器如 Bourne-Shell（請見 `sh(1)`）或者 C-Shell（請見 `csh(1)`）可由 Sun Grid Engine 用以啓動作業 script。指令解譯器會以登入 shell 啓動（即當指令解譯器啓動後，所有系統及使用者預設資源檔案如 `.login` 或 `.profile` 會被執行，且如果使用者剛剛登入，將會設定作業環境。），或者以指令執行（即只有 shell 特定資源檔案 `.shrc` 會被執行及 Sun Grid Engine — 請見 `qsub(1)`）。此參數登入 shell 包含一由逗點分隔之指令解讀器的執行名稱清單，會以登入 shell 啓動。

變更登入 shell 將立即生效。預設之登入 shell 為 `sh`、`csh`、`tcsh`、`ksh`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- `min_uid` — 此參數可設定使用叢集使用者 ID 的下限。使用者 ID（由 `getpwnam(3)`）少於 `min_uid` 者將不被允許於叢集上執行作業。

變更 `min_uid` 將立即生效。`min_uid` 之預設值為 0。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *min_gid* — 此參數設定可使用叢集群組 ID 的下限。使用者 ID (由 `getpwnam(3)`) 少於 *min_uid* 者將不被允許於叢集上執行作業。群組 ID (由 `getpwnam(3)`) 少於 *min_gid* 之使用者將不被允許於叢集上執行作業。

變更 *min_gid* 將立即生效。*min_uid* 之預設值為 0。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *使用者清單* — 此參數包含由逗點分隔之所謂的使用者存取清單，請見 `access_list(5)` 中的說明。每個至少列於存取清單之一的使用者皆可存取至叢集。如果 *使用者清單* 參數設定為 `NONE` (預設值)，任何未明確包含於 *非使用者清單* 參數 (見下述說明) 中的使用者皆可存取。如果一使用者同時列於存取清單中的 *非使用者清單* 與 *使用者清單* 中，則此使用者會被拒絕存取至叢集。

變更 *使用者清單* 將立即生效。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *非使用者清單* — 此參數包含由逗點分隔之所謂的使用者存取清單，請見 `access_list(5)` 中的說明。每個至少列於存取清單之一的使用者會被拒絕存取至叢集。如果 *非使用者清單* 參數設為 `NONE` (預設值)，則任何使用者皆可存取。如果一使用者同時列於存取清單中的 *非使用者清單* 與 *使用者清單* 中，則此使用者會被拒絕存取至叢集。如果一使用者同時列於存取清單中的 *非使用者清單* 與 *使用者清單* (請見上述說明) 中，則此使用者會被拒絕存取至叢集。

變更 *非使用者清單* 將立即生效。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *管理員郵件* — 此參數指定一由逗點相隔之叢集管理員電子郵件地址清單，可用以傳送內部問題報告。郵件地址格式根據電子郵件系統及其如何配置而定；欲知更多資訊請參閱系統配置指南。

變更 *管理員郵件* 將立即生效。*管理員郵件* 之預設值為空郵件清單。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *專案* — 此參數只適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並沒有此參數。

專案 清單包含所有已被授權可存取 Sun Grid Engine，企業版系統的專案。若使用者未包含至任一專案中，就無法使用 Sun Grid Engine，企業版系統。若使用者同時位於 *專案* 清單及 *非專案* 清單 (請見下述說明) 中，仍然無法使用此系統。

變更 *專案* 將立即生效。*專案* 之預設值為 0。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *x 專案* — 此參數只適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並沒有此參數。

x 專案 清單包含所有已未被授權可存取 Sun Grid Engine，企業版系統的專案。若使用者包含至任一專案中，就無法使用 Sun Grid Engine，企業版系統。若使用者同時位於 *專案* 清單 (請見上述說明) 及 *x 專案* 清單中，仍然無法使用此系統。

變 *x 專案* 將立即生效。*x 專案* 之預設值為 0。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **載入報告時間** – 系統負載值會由執行常駐程式定期報告給 `sge_qmaster(8)`。**載入報告時間** 參數會定義二個載入報告間的間隔時間。

每一個 `sge_execd(8)` 可使用不同的載入報告時間。變更**載入報告時間**將立即生效。

注意 – 修改**載入報告時間**要格外注意。報告太頻繁會阻斷 `sge_qmaster(8)`，尤其是當執行主機數量大時。而且由於系統載入通常會平順的增加或減少，頻繁的負載報告不會有任何益處。

載入報告時間之預設值為 40 秒鐘。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **重新排程 unknown** – 決定主機上之作業是否為 `unknown` 狀態後，再重新排程並送至其它主機。如果 `sge_master(8)` 無法於主機上的 `sge_execd(8)` (請見 `max_unheard`) 建立連繫，那些主機就會被記錄為 `unknown`。可能的原因有主機當機、二者間之網路連接斷線、或者主機上之 `sge_execd(8)` 沒有在執行。

上述任何狀況中，Sun Grid Engine 皆可以將上述主機上執行之作業重新排程至其它主機。**排程 unknown** 可在當主機變為 `unknown` 後，於作業重新排程前控制 Sun Grid Engine 的等候時間。時間格式之規格為 `hh:mm:ss`。如果設定特殊值 `00:00:00`，作業將不會從主機上重新排程。

惟有啟動「重新執行 (rerun)」旗號才會起始重新排程作業 (請見 `qsub(1)` 的 `-ry` 選項，與 `queue_conf(5)` 的 `rerun` 選項)。如果平行作業的主要工作執行的主機處於 `unknown` 狀態，平行作業才可被重新排程。檢核點作業只有當對應檢查點環境中包含適當旗號時才會被重新排程 (請見 `checkpoint(5)`)。互動式作業 (請見 `qsh(1)`, `qrsh(1)`, `qtcsh(1)`) 無法被重新排程。

重新排程 unknown id 的預設值：00:00:00

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **stat_log_time** – Sun Grid Engine 軟體會定期紀錄目前叢集配置之快照 (snapshot) 至硬碟。二個連續快照間的秒數是由 `stat_log_time` 所定義。

變更 `stat_log_time` 將立即生效。`stat_log_time` 之預設值為 2 小時 30 分鐘。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **max_unheard** – 如果 `sge_qmaster(8)` 無法連接或未與主機之執行常駐程式相連接長達 `max_unheard` 秒數，所有該主機上的佇列狀態將設定為 `unknown`。至少應由執行常駐程式連接 `sge_qmaster(8)`，才可得到負載報告。因此，`max_unheard` 應大於**載入報告時間** (請見上述說明)。

變更 `max_unheard` 將立即生效。`max_unheard` 之預設值為 2 分鐘 30 秒鐘。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **日誌等級** — 此參數會指定用以產生資訊、警告、或錯誤訊息之 Sun Grid Engine 組件，如 `sge_qmaster(8)` 或 `sge_execd(8)`，的詳細訊息等級。這些訊息會被記錄至主要與執行常駐程式多工緩衝目錄中（請見上述 `qmaster_spool_dir` 與 `execd_spool_dir` 的說明）。`messages` 檔案中。可用之訊息等級為：
 - `log_err` — 會記錄所有可辨識之錯誤事件。
 - `log_warning` — 記錄所有可辨識之錯誤事件及所偵測到的潛在錯誤行為訊息。
 - `log_info` — 記錄所有可辨識之錯誤事件、所偵測到的潛在錯誤行為訊息、及不同種類的訊息。

變更日誌等級將立即生效。

日誌等級的預設值為 `log_info`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **`max_aj_instances`** — 此參數可定義每個陣列作業中被排程以同時執行之陣列工作最大量。當自排程器取得啟動命令後，將於主要常駐程式內建立一個陣列工作的範例。當陣列作業完成時，將會銷毀此範例。因此，參數主要可控制主要常駐程式及排程器常駐程式內的陣列作業記憶體用量。對於大量叢集及大量陣列作業是非常有用的。此參數之預設值為 2000。此數值 0 會廢除此限制，並允許排程器在叢集中依據適當的資源數量來啟動適合數量的陣列工作。

變更 `max_aj_instances` 會立即生效。此值為全域配置參數，不可以由執行主機之本端配置改寫。

- **`max_aj_tasks`** — 此參數會定義一個陣列作業內陣列作業工作之最大量。如果請求超出 `max_aj_tasks` 陣列作業工作件數時，`sge_qmaster(8)` 就會拒絕所有提交的陣列作業。此參數之預設值為 75000。此值 0 將取消此限制。

變更 `max_aj_tasks` 會立即生效。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **`max_u_jobs`** — 此參數可控制 Sun Grid Engine 使用者可以在系統內同時擁有之執行中（未完成）的作業量。大於 0 的值可用以定義此限制。預設值 0 代表「無限制」。如果因提交作業而超出 `max_u_jobs` 限制，則提交指令會以狀態 25 及適合之錯誤訊息退出。

變更 `max_u_jobs` 會立即生效。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **`max_jobs`** — 此參數可控制 Sun Grid Engine 系統中同時可被允許之執行中（未完成）作業。大於 0 的數值可用以定義此限制。預設值 0 代表「沒有限制」。如果因提交作業而超出 `max_jobs` 限制，則提交指令會以狀態 25 及適合之錯誤訊息退出。

變更 `max_jobs` 會立即生效。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **執行專案** — 此參數只適用於 Sun Grid Engine，企業版系統。

如果設定為 `true`，使用者就必須於提交作業時提出專案要求。請見 `-P` 選項至 `qsub(1)` 以得知詳細資訊。

變更 `enforce_project` 會立即生效。`enforce_project` 之預設值為 `false`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **執行使用者** – 此參數只適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並沒有此參數。

如果設定為 `true`，必須要執行 Sun Grid Engine 企業版 `user(5)` 才可允許作業提交。如果沒有對應使用者存在，作業將遭拒絕。

變更 `enforce_user` 會立即生效。`enforce_user` 的預設值為 `false`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **設定記號指令** – 如果你的 Sun Grid Engine 系統有經授權支援 AFS 才會存在此參數。

設定記號指令 指向為 Sun Grid Engine 作業設定及延伸 AFS 記號 (token) 的指令。於標準的 Sun Grid Engine AFS 版本中，其會以二指令行參數之 `script` 形式出現。其會自 STDIN 讀入記號、延伸記號之期限、並設定下述記號：

```
< 設定記號指令 > < 使用者 > < 數秒後記號延伸 >
```

作為 shell script，此指令會呼叫下述程式：

- `SetToken`
- `forge`

由你的廠商提供原始程式碼。`script` 看起來如下：

```
#!/bin/sh
# 設定記號指令
forge -u $1 -t $2 | SetToken
```

由於 `forge` 必須讀取機密 AFS 伺服器鑰匙，站點可能期望由連接至 AFS 伺服器上自行撰寫之常駐程式的指令來取代 **設定記號指令 script**。記號必須在 AFS 伺服器中制訂，然後傳回至執行 `SetToken` 之本端電腦。

變更 **設定記號指令** 將立即生效。**設定記號指令** 之預設值為 `none`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **pag_cmd** – 如果你的 Sun Grid Engine 系統有經授權支援 AFS 才會存在此參數。

透過此參數來指定至 `pagsh` 之路徑。`sge_shepherd(8)` 工作與作業會執行於 `pagsh` 中。欲知詳細資訊請洽詢您的 AFS 管理員。

變更 `pag_cmd` 將立即生效。`pag_cmd` 之預設值為 `none`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *記號延伸時間* — 如果您的 Sun Grid Engine 系統有經授權支援 AFS 才會存在此參數。

此參數設定 AFS 記號定期延伸的時段。Sun Grid Engine 在記號終了之前會呼叫記號延期 30 分鐘，直至作業完成以及不再需要對應之記號為止。

變更 *記號延伸時間* 將立即生效。*記號延伸時間* 之預設值為 24:0:0，即 24 小時。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *gid_range* — 此參數只適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並沒有此參數。

gid_range 為一以逗點相隔之範圍表示式 *n-m* (*n* 和 *m* 為非零的正整數) 之清單，而 *m* 為 *m-m* 的縮寫。在 *sge_execd(8)* 中使用這些數字來辨識屬於相同作業之工作。

因此目的，每一個 *sge_execd(8)* 可以使用個別的群組 *id* 組來達到上述目的。所有於群組 *id* 範圍中的數字必須是以 *sge_execd(8)* 啟動之系統上未使用之附屬群組 *id*。

變更 *gid_range* 將立即生效。*記號延伸時間* 之預設值為 24:0:0，即 24 小時。

gid_range 沒有預設值。管理員必須於安裝 Sun Grid Engine，企業版時指定 *gid_range* 的值。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *qmaster_params* — 可以傳遞至 Sun Grid Engine *qmaster* 中的其餘參數清單。下述為可辨識的值：
 - *ENABLE_FORCED_QDEL* — 如果設定此參數，非管理員之使用者可以透過 *qdel(1)* 的 *-f* 選項來強制刪除他們自己的作業。沒有此參數，只允許 Sun Grid Engine 管理員或是操作員來強制刪除作業。

注意 — 因使用者是否為 Sun Grid Engine 管理員而有不同執行強制刪除作業之方式。如果是管理使用者，作業會立即自 Sun Grid Engine 內部資料庫中移除。對一般使用者而言，會首先執行同等之一般 *qdel(1)*，然後若一般刪除失敗，才會強制執行刪除。

更改 *qmaster_params* 將立即生效。*qmaster_params* 之預設值為 *none*。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *FORBID_RESCHEDULE* — 如果設定此參數，作業重新排入佇列便不可由使用者控制之作業 *script* 來啟動。如無此參數，傳回 99 數值之作業將會被重新排程。這可以用來於不同機器上重新啟動作業，例如當目前所在之主機上無足夠資源時。
- *DISABLE_AUTO_RESCHEDULING* — 如果設定為 *true* 或 1，*重新排程 unknown* 參數將不予考慮。

變更 *qmaster_params* 將立即生效。*qmaster_params* 之預設值為 *none*。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *schedd_params* — 此參數會預留傳遞至 Sun Grid Engine 排程器的其餘參數。下述為目前可供辨識的值：

- `FLUSH_SUBMIT_SEC`、`FLUSH_FINISH_SEC` — 提供此參數以調整系統排程的運作方式。根據預設值，排程器之執行會於排程器間隔時間內啟動，而此時間是定義於排程器配置 `sched_conf(5)` 中的 *排程間隔時間* 參數。

參數 `FLUSH_SUBMIT_SEC/FLUSH_FINISH_SEC` 會定義啟動排程器之執行，與送出 / 完成作業之間的時間。

設定二者之值為將兩個數值同時設為 0 即可立即啟動排程器運作。預設之排程運作方式可由移除參數，或設定其值為 -1 來執行。

- `CLASSIC_SGEEE_SCHEDULING` — 如果設定為 `true` 或 1，就使用 Sun Grid Engine，企業版原先的擱置作業排程之演算。Sun Grid Engine，企業版原先的擱置作業排程之演算會藉由票卷均分至每一個排程策略內的所有進行中或擱置的作業，來計算擱置作業之票卷。然後擱置作業清單是根據指定給擱置作業之票卷來排序。預設值為 `false`。此參數只有於 Sun Grid Engine，企業版系統中有效。
- `POLICY_HIERARCHY` — 此參數設定策略的從屬關係鏈。從屬關係中每一個策略會受到前置策略所影響，並影響跟隨之後的策略。一個典型的情形為指定置換策略優先於共享基礎策略。置換策略會決定於此狀況下如何指定共享基礎票卷給相同使用者或專案的作業。請注意：不論策略階層之定義如何，所有的策略都將貢獻票卷給指定的特定作業。雖然根據 `POLICY_HIERARCHY`，每種策略計算票卷方式皆不同。

`POLICY_HIERARCHY` 參數最多可由四個字母（由每種策略之第一個字母）作組合，字母分別為 S（共享）、F（功能）、D（截止日期）、及 O（置換）。因此，OFSD 的值代表置換策略優先於功能策略，並會影響共享策略及最後的截止日期策略。少於四個字母代表某些策略並不會影響其它策略，同時也不受其它策略影響。因此 FS 代表功能策略會影響共享策略，但並不會影響其他策略。

特殊值 `NONE` 會關閉策略階層。

- `SHARE_OVERRIDE_TICKETS` — 如果設定為 `true` 或 1，任何置換物件之實例置換票卷會和所有與物件相關之作業均分。如果 `false` 或 0，每個作業會得到與物件相關之全數置換票卷。預設值為 `true`。此參數只有於 Sun Grid Engine，企業版系統中有效。
- `SHARE_DEADLINE_TICKETS` — 若設定為 `true` 或 1，全部截止日期作業會共享全部的截止日期票卷。若設定為 `false` 或 0，當作業之截止日期到達時，每一截止日期作業就會接收所有的期限票卷。預設值為 `true`。此參數只有於 Sun Grid Engine，企業版系統中有效。
- `MAX_FUNCTIONAL_JOBS_TO_SCHEDULE` — 功能策略中要進入排程的最大擱置作業數量。預設值為 200。此參數只有於 Sun Grid Engine，企業版系統中有效。
- `MAX_PENDING_TASKS_PER_JOB` — 等候進入排程之陣列作業之子工作的最大數量。此參數存在的目的在於減低排程的負載量。預設值為 50。此參數只有於 Sun Grid Engine，企業版系統中有效。
- `PROFILE` — 若設定，排程器將紀錄定義資訊至日誌檔案，並於每次執行排程時作摘要。

變改 `schedd_params` 將立即生效。`schedd_params` 之預設值為 `none`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *execd_params* — 此參數會預留傳遞至 Sun Grid Engine 執行常駐程式的其餘參數。下述為可辨識的值：
 - ACCT_RESERVED_USAGE — 若此參數設定為 true，會保留使用 cpu、mem 及 io 的計算項目而取代測量使用。
 - KEEP_ACTIVE — 此值只能設定作為偵錯用途。若設定為 true，執行常駐程式將不會移除由 sge_shepherd(8) 作業所保留的多工緩衝處理目錄。
 - NO_REPRIORITIZATION — 若設定為 true 或 1，作業將不會取得動態優先權。未設定此項屬性，sched_conf(5) 中的 Sun Grid Engine，企業版系統沒有永久性的效果，因為 sge_execd(8) 中的 PTF 模組會執行以取得動態優先權。
 - PTF_MIN_PRIORITY, PTF_MAX_PRIORITY — 此參數只有於 Sun Grid Engine，企業版中有效。

指定給作業之 Sun Grid Engine，企業版最高 / 最低的優先權。對於系統允許使用 nice(2) 系統呼叫作優先權設定，典型值介於 -20（最大值）至 19（最小值）範圍的正 / 負值。其他系統可提供不同的範圍。

預設的優先權範圍（系統各有不同）會由移除參數或設定 -999 值來安裝。

請查看常駐程式的 messages 檔案可以得知主機中事先定義的預設值。此值將於啟動執行常駐程式時被記錄起來。

- NOTIFY_KILL — 此參數允許你變更訊號 SIGKILL（請見 qsub(1) 的 -notify 選項）之通知訊號。此參數可以接受訊號名稱（使用 kill(1) -l 選項），或是特殊值 none。如果設定為 none，沒有任何通知訊號會被送出。若設定為 TERM，舉例來說，或其它訊號名稱，此訊號便會以通知訊號被送出。
- NOTIFY_SUSP — 以此訊號可以修改 SIGSTOP 的通知訊號（請見 qsub(1) 的 -notify 選項）。此參數可以接受訊號名稱（使用 kill(1) -l 選項），或是特殊值 none。如果設定為 none，沒有任何通知訊號會被送出。若設定為 TSTP，舉例來說，或其它訊號名稱，此訊號便會以通知訊號被送出。
- SET_SGE_ENV, SET_COD_ENV, SET_GRD_ENV — 於 qsub(1) 中你可以發現匯出至每一個 Sun Grid Engine，企業版作業執行環境的環境變數清單。某些變數名稱字首為 SGE_。在 Sun Grid Engine，企業版 (Codine, GRD) 之前的版本中，變數的字首則為 COD_ 及 GRD_。根據這些環境變數之一就足以更新所有的作業 script。SET_SGE_ENV、SET_COD_ENV 及 SET_GRD_ENV 參數可以用來平順的自 Codine/GRD 轉換成 Sun Grid Engine，企業版。

上述每個參數可確認 qsub(1) 中提及的環境參數組將以對應字首開頭匯出至作業環境中。每一個變數都可被設定為 true、1、或 false、0。預設值為 SET_SGE_ENV=1、SET_COD_ENV=0、及 SET_GRD_ENV=0。

若上述三個變數都被設定為 false 或 0，作業將因沒有設定任何環境變數無法成功執行。因此，在此情況下會設定字首為 SGE_ 的環境變數。

- SHARETREE_RESERVED_USAGE — 如此參數設定為 true，就會保留 Sun Grid Engine，企業版之使用分享樹之消耗量，而非用來測量。

- **USE_QSUB_GID** — 如參數設定為 `true`，當作業送件時，執行中的主群組 `id` 就會被設定為作業執行之主要群組 `id`。如參數沒有設定，就會使用於執行主機中 `passwd(5)` 檔案為作業所有人所定義的主要群組 `id`。

本功能只適用於透過 `qsub(1)`、`qrsh(1)`、`qmake(1)` 及 `qtcs(1)` 提交的作業。而如果使用由 Sun Grid Engine（即 `rsh_daemon` 與 `rsh_command` 參數的預設值沒有被變更）提供的 `rsh` 與 `rshd` 元件，本功能就僅可用於 `qrsh(1)` 作業中（`qtcs(1)` 與 `qmake(1)` 同理可證）。

變更 `execd_params` 將立即生效。`execd_params` 之預設值為 `none`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **admin_user** — 此參數為 Sun Grid Engine 用以處理所有內部檔案的管理使用者帳號（狀態多工緩衝處理，訊息紀錄等等）。當 `root` 帳戶沒有對應檔案存取權限時便可使用（例如，於共享檔案系統中沒有全域 `root` 讀取 / 寫入存取權）。

由於此參數於安裝時設定，所以不支援於系統執行時變更 *管理使用者*。可於叢集關機時變更參數值，但是若存取於 Sun Grid Engine 多工緩衝處理區時被打斷，將有不可預期之後果。

管理使用者 參數沒有預設值，但是會於主安裝程序時定義。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **已完成作業** — Sun Grid Engine 軟體會儲存特定數量之 *剛完成* 的作業以提供後 `mortem` 狀態資訊。`已完成作業` 參數會定義要儲存的完成作業數量。如果已達最大數目，當新完成的作業加入已完成作業清單時，最老的已完成作業將被丟棄。

變更 `已完成作業` 將立即生效。`已完成作業` 之預設值為 `0`。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- **qlogin_daemon** — 此參數會指定於 `qlogin(1)` 要求的伺服器端上啟動之執行檔案。通常為完全合格的系統 `telnet` 常駐程式路徑名稱。若未賦值，將使用 Sun Grid Engine 特定之元件。

變更 `qlogin_daemon` 將立即生效。`qlogin_daemon` 之預設值為 `none`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **qlogin_command** — 此為於 `qlogin(1)` 要求的客戶端上執行之指令。通常為完全合格的系統 `telnet` 客戶程式之路徑名稱。若未賦值，將使用 Sun Grid Engine 特定之元件。它將自動以目標主機及連接埠數量作為參數。

變更 `qlogin_command` 將立即生效。`qlogin_daemon` 之預設值為 `none`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- **rlogin_daemon** — 此參數會指定於 `qrsh(1)` 要求的伺服器端上啟動之執行檔案，不包含遠端執行之指令引數。通常為完全合格的系統 `rlogin` 常駐程式路徑名稱。若未賦值，將使用 Sun Grid Engine 特定之元件。

變更 `rlogin_daemon` 將立即生效。`qlogin_command` 之預設值為 `none`。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *rlogin_command* — 此為執行於 *qrsh(1)* 要求的客戶端上之指令，不包含遠端執行之指令引數。通常為完全合格的系統 *rlogin* 常駐程式路徑名稱。若未賦值，將使用 Sun Grid Engine 特定之元件。它將自動以目標主機及連接埠數量來做為參數，如同 *telnet(1)* 要求的參數一樣。你能使用也懂得此語法之客戶端，如 *ssh*。

變更 *rlogin_command* 將立即生效。*rlogin_command* 之預設值為 *none*。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *rsh_daemon* — 此參數會指定於 *qrsh(1)* 要求的伺服器端上啓動之執行檔案，包含遠端執行之指令引數。通常為完全合格的系統 *rsh* 常駐程式路徑名稱。若未賦值，將使用 Sun Grid Engine 特定之元件。

變更 *rsh_daemon* 會立即生效。*rsh_daemon* 之預設值為 *none*。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *rsh_command* — 此為執行於 *qrsh(1)* 要求的客戶端上之指令，包含遠端執行之指令引數。通常為完全合格的系統 *rlogin* 客戶程式之路徑名稱。若未賦值，將使用 Sun Grid Engine 特定之元件。它將自動以目標主機及連接埠數量來做為參數，如同 *telnet(1)* 要求的參數一樣。Sun Grid Engine *rlogin* 客戶端經延伸便可接收並使用連接埠號碼引數。你能使用也懂得此語法之客戶端，如 *ssh*。

變更 *rsh_command* 會立即生效。

此數值於全域配置中的項目可由執行主機本端配置改寫。

- *ignore_fqdn* — 此參數導致系統忽略主機名稱之完全合乎規定的領域名稱元件。如果 Sun Grid Engine 叢集所屬的主機都是單一 DNS 領域的一部分，就該設定此參數。如果設定為 *true* 或 *1*，其便會啓動。開啓它或許可以解決因橫跨叢集中不同主機名稱解讀方式所導致之負載報告問題。

由於此參數於安裝時設定，所以不支援於系統執行時變更 *ignore_fqdn*。*ignore_fqdn* 的預設值為 *true*。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

- *預設領域* — 只有您的 Sun Grid Engine 叢集涵括一個以上 DNS 網域之主機時，才需要此參數。此情況下，如果主機名稱之解讀 DNS 領域之一可用以產生合格及不合格的主機名稱就可使用。*預設領域* 之值會接在不合格主機名稱之後來定義成爲一個完全合格的主機名稱。如果 *ignore_fqdn* 設為 *true*，*預設領域* 參數將無效。

由於此參數於安裝時設定，所以不支援於系統執行時變更 *預設領域*。*預設領域* 預設值為 *none*，將不用於此例中。

這些值只是全域配置參數，不可以由執行主機之本端配置改寫。

同時參閱

`sge_intro(1)`、`csh(1)`、`qconf(1)`、`qsub(1)`、`rsh(1)`、`sh(1)`、`getpwnam(3)`、`queue_conf(5)`、`sched_conf(5)`、`sge_execd(8)`、`sge_qmaster(8)`、`sge_shepherd(8)`、`cron(8)`、《*Sun Grid Engine 企業版 5.3 管理和使用者指南*》、《*Sun Grid Engine 5.3 管理和使用者指南*》

版權

請見 `sge_intro(1)` 以得知權利與權限的完整聲明。

sge_h_aliases(5)

名稱

`sge_h_aliases` — Sun Grid Engine 主機別名檔案格式

說明

所有 Sun Grid Engine 組件會使用一獨特的主機名稱來辨識主機，而此獨特的主機名稱是由 `sge_commd(8)` 提供以決定服務的主機名稱。`sge_commd(8)` 本身可用來當作標準 UNIX 目錄服務的參考，例如 DNS、NIS、與 `/etc/hosts` 來決定主機的名稱。在少數情況中會無法乾淨地設定這些標準服務，而且在不同主機上執行的 Sun Grid Engine 通訊常駐程式也無法自動確認一個或多個主機上的獨特主機名稱以供所有主機使用。在此情況下，就會使用 Sun Grid Engine 主機別名檔案來提供通訊常駐程式，與一個隱密且一致的主機名稱解析資料庫。

主機別名檔案的預設位址為：`<sge 根目錄>/<儲存格>/common/host_aliases`

要使用不同的主機別名檔案，就必須透過 `-a` 指令行選項來明確指定至 `sge_commd(8)`。

格式

在每個主機上，單一行中皆必須有一以空格、逗點、或分號分隔之主機名稱別名清單。第一個更改的別名會定義為獨特的主機名稱，並將為所有使用 `sge_commd(8)` 主機名稱別名服務的 Sun Grid Engine 組件所用。

同時參閱

`sge_intro(1)`、`sge_commd(8)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

sge_pe(5)

名稱

`sge_pe` – Sun Grid Engine 平行環境配置檔案格式

說明

平行環境為一平行設計之程式，而且其執行時期環境可允許執行共享記憶體或分派記憶體的平行應用程式。在啟動平行應用程式前，通常需要先做好平行環境的一些設定才可操作。一般平行環境的範例為共享記憶體平行操作系統與分發記憶體環境平行虛擬器 (Parallel Virtual Machine, PVM) 或訊息傳遞介面 (Message Passing Interface, MPI)。

`sge_pe` 允許定義介面至仲裁平行環境中。一旦平行環境經 `-ap` 或 `-mp` 選項至 `qconf(1)` 定義或修改過後，作業就可使用 `-pe` 開關至 `qsub(1)` 對此環境提出要求，並對由作業分派之平行作業數量範圍提出要求。額外的 `-l` 選項可用來指定更詳細的作業要求。

格式

sge_pe 檔案的格式定義如下：

- *pe* 名稱 – 此為平行環境的名稱。用於 `qsub(1) -pe` 開關 (switch) 中。
- *佇列清單* – 此為一以逗點分隔之佇列清單，屬於此平行環境的平行作業具有其存取權。
- *插槽* – 此為允許可同時執行於平行環境底下的總平行作業數量。
- *使用者清單* – 此為一以逗點分隔之使用者存取名稱清單（請參閱 `access_list(5)`）。每個至少列於存取清單之一的使用者皆可存取至平行環境。如果 *使用者清單* 參數設定為 `NONE`（預設值），任何未明確包含於 *x 使用者清單* 參數（見下述說明）中的使用者皆可存取。如果一使用者同時列於存取清單中的 *x 使用者清單* 與 *使用者清單* 中，則此使用者會被拒絕存取至平行環境中。
- *x 使用者清單* – *x 使用者清單* 參數為一逗點分隔之使用者存取清單，請參考 `access_list(5)` 的說明。每個至少列於存取清單之一的使用者皆無法存取至平行環境中。如果 *x 使用者清單* 參數設為 `NONE`（預設值），則任何使用者皆可存取。如果一使用者同時列於存取清單中的 *x 使用者清單* 與 *使用者清單* 中，則此使用者會被拒絕存取至平行環境中。
- *start_proc_args* – 此為平行環境啟動程序的呼叫指令行。在執行作業 `script` 前會先以 `sge_shepherd(8)` 來開啓呼叫程序，目的是為了對應平行環境的需要來做相關設定。字首「`user@`」（非必要項目）會在欲啟動之程序底下指定使用者。啟動程序的標準輸出資訊會重新導至作業工作目錄中的 `REQName.po.JID`（請參閱 `qsub(1)`），和由作業識別碼 `qstat(1)` 與 `JID` 顯示的作業名稱 *要求名稱* 中。同樣的，標準錯誤輸出資訊也會重新導至 *要求名稱.pe.JID* 中。

下述特殊變數會在執行期間擴充，並可用來（除了須由啟動與中止程序解讀之其它字串之外）建構一指令行：

- *\$pe_hostfile* – 此為檔案路徑名稱，內含由啟動程序設定之詳盡的平行環境配置說明。檔案每行代表平行作業欲執行之主機。每行的第一個項目為主機名稱、第二個項目為主機上欲執行之平行作業量、而第三個則為多重處理器要使用的處理器範圍。
- *\$host* – 此為執行啟動或中止程序的主機名稱。
- *\$job_owner* – 此為作業所有人的使用者名稱。
- *\$job_id* – 此為 Sun Grid Engine 軟體的獨特作業識別碼。
- *\$job_name* – 此為作業名稱。
- *\$pe* – 此為使用中的平行環境名稱。
- *\$pe_slots* – 此為作業得到的插槽數量。
- *\$processors* – 此為包含在主要佇列（即開啓啟動和中止程序的佇列）之佇列配置中的 *processors* 字串（請參閱 `queue_conf(5)`）。
- *\$queue* – 此為主要佇列，即開啓啟動和中止程序的佇列。

- **停止程序引數** — 此為平行環境關閉程序的啓動指令行。在作業 script 結束後，才可使用 `sge_shepherd(8)` 呼叫關閉程序，其目的是為了中止，或從所有執行的系統中移除此平行環境。字首「`user@`」（非必要項目）會在欲啓動之程序底下指定使用者。中止程序的標準輸出資訊會被重新導至作業工作目錄中的**要求名稱**.`poJID`（請參閱 `qsub(1)`），和由作業識別碼 `qstat(1)` 與 `JID` 顯示的作業名稱**要求名稱**中。同樣的，標準錯誤輸出資訊也會重新導至**要求名稱**.`peJID` 中。

可使用與**啓動程序引數**相同的特殊變數來建立一指令行。

- **訊號程序引數** — 此為平行環境發訊程序 (signalling procedure) 的呼叫指令行。在透過 `qmod(1)`、`qdel(1)`、或遷移要求 (migration request) 將訊息傳送至平行作業後，就可使用 `sge_shepherd(8)` 來呼叫發訊程序。其目的是為了對平行環境與其對應相關應用程式作業的所有組件發出訊號。發訊程序的標準輸出資訊會重新導至作業工作目錄中的**要求名稱**.`poJID`（請參閱 `qsub(1)`），和由作業識別碼 `qstat(1)` 與 `JID` 顯示的作業名稱**要求名稱**中。同樣的，標準錯誤輸出資訊也會重新導至**要求名稱**.`peJID` 中。

可使用與**啓動程序引數**相同的特殊變數來建立一指令行。

- **配置規則** — 配置規則是由 `sge_schedd(8)` 來解譯，可幫助排程器決定該如何於可用的機器中分派平行作業。假若一平行環境僅可建立於共享記憶體應用程式中，所有的平行作業就必須指定至單一機器中，不論其它機器有多適用。但假如平行環境是尾隨分派記憶體範例 (distributed memory paradigm)，就傾向平均分派作業至各機器中。

目前的排程器版本僅可瞭解下述分配規則：

- `<int>` — 此為一整數，用以固定單一主機上的處理器數量。如果數量為 1，所有的作業都必須位於不同的主機上。如果使用此特殊名稱 `$pe_slots`，以 `qsub(1)` `-pe` 開關 (switch) 指定的作業範圍就必須分配至單一主機上（不論最後選擇何種範圍值分派至作業上）。
- `$fill_up` — 所有可用插槽會從最佳適用的主機/佇列開始配置。當一作業為平行作業要求更多插槽時，就會改「填滿」其餘主機與佇列。
- `$round_robin` — 直到平行作業要求的所有作業都分派出去後，才會開始從所有適合的主機上配置單一插槽。如果要求的作業量超過可找到的適合的主機，就會從第一個主機重新開始配置。此配置計劃會依「最適合優先 (best-suitable-first)」的順序來分配作業至各合適主機上。
- **控制/slave** — 此參數可以設定為 TRUE 或 FALSE（預設值）。這可顯示 Sun Grid Engine 是否可透過 `sge_execd(8)` 與 `sge_shepherd(8)` 來建立平行應用程式的 slave 作業，並可全權控制平行應用程式中的所有作業，即啓動如資源限制與修正計算之類的功能。但是，如果想控制平行應用程式的 slave 作業，就必須要有一精密的 PE 介面以與 Sun Grid Engine 功能緊密操作。此類 PE 介面請洽詢您的當地 Sun Grid Engine 支援辦公室。

請將所有其餘 PE 介面的 `control_slaves` 參數設定至 `false`。

- **作業為首要工作** – 當 `control_slaves` (請參閱上述說明) 設定為 `TRUE`，且 Sun Grid Engine 可使用 `sge_execd(8)` 和 `sge_shepherd(8)` 來建立平行應用程式的 `slave` 作業時，才會檢查此參數。在此情況中，就必須有一精密的 PE 介面，才可與平行環境與 Sun Grid Engine 緊密操作。PE 介面的相關文件內含 `job_is_first_task` 的建議設定。

`job_is_first_task` 參數可以被設為 `TRUE` 或 `FALSE`。TRUE 值代表 Sun Grid Engine 作業 script 已包含平行應用程式的作業之一，而 `FALSE` 值則代表作業 script (與其子作業) 並不屬於平行程式的一部分。

限制

注意 – 啟動、關閉、與發訊程序有完整的管理員職責以配置平行環境。Sun Grid Engine 僅會啟動這些程序，然後評估其退出狀態。如果這些程序無法正確地執行其作業，或者平行環境或平行應用程式的表現不如預期，Sun Grid Engine 也無法提供偵測的方法。

同時參閱

`sge_intro(1)`、`qconf(1)`、`qdel(1)`、`qmod(1)`、`qsub(1)`、
`access_list(5)`、`sge_qmaster(8)`、`sge_schedd(8)`、`sge_shepherd(8)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

complex(5)

名稱

`complex` – Sun Grid Engine 聚合體配置檔案格式

說明

聚合體會反映 Sun Grid Engine 聚合體配置的格式。聚合體的定義可透過 `qsub(1) -l` 選項，提供所有使用者可能要求的 Sun Grid Engine 作業資源屬性相關資訊，與 Sun Grid Engine 系統內的相關參數解譯資訊。

聚合體配置檔案不應直接被存取。爲了新增或修改聚合體，應使用 `qconf(1)` 選項 `-Ac`、`-ac`、`-Mc` 與 `-mc` 來代替。`-Ac` 與 `-Mc` 選項將聚合體配置檔案視爲引數，而 `-ac` 與 `-mc` 選項則會叫出以範例聚合體配置，或現有聚合體配置寫入之編輯器 (`editor`)。

Sun Grid Engine 聚合體物件會整合下述四種聚合體：

- 佇列聚合體 — 此名是參考此特殊名稱佇列而來的。

於其預設格式中，可選擇定義於 `queue_conf(5)` 裡的佇列配置參數。使用者基本上會要求之作業佇列配置參數爲：

- 佇列名稱
- 主機名稱
- 通知
- 行事曆
- 最少 `cpu` 間隔時間
- 暫存目錄
- 序列編號
- `s_rt`
- `h_rt`
- `s_cpu`
- `h_cpu`
- `s` 資料
- `h` 資料
- `s` 堆疊
- `h` 堆疊
- `s` 核心
- `h` 核心
- `s_rss`
- `h_rss`

如果每個佇列上可用的屬性增加，就會擴展佇列聚合體。佇列聚合體會定義其包含屬性的特性（例如資料種類）。佇列聚合體屬性的設定值會由佇列配置定義，以避免遭遇上述列出之標準參數，但如果新增一參數至預設佇列聚合體中，就會由佇列配置（請參閱 `queue_conf(5)` 得知詳細資訊）中的聚合體值項目來定義。如果在上述之後方假設有提供佇列配置的聚合體值項目，就會依下述說明之 `value` 欄位定義來設定此數值。

- **主機聚合體** — 是參考此特殊名稱**主機**而來。此聚合體包含於主機上受管理的所有屬性之特徵定義。與主機相關之屬性的標準組包含兩種類型，但也有可能會增加上述之佇列聚合體。第一種類型是由許多佇列配置屬性建立而成，特別適用於以主機為主 (host basis) 之管理。這些屬性包含：

- *插槽*
- *s_vmem*
- *h_vmem*
- *s_fsize*
- *h_fsize*

(請參閱 `queue_conf(5)` 以得知詳細資訊)。

注意 — 同時於主機聚合體與佇列配置中來定義這些屬性並不會造成互相矛盾的現象，還可同時允許維持主機等級 (host level) 和佇列等級 (queue level) 中的對應資源。舉例來說，總虛擬空白記憶體 (`h_vmem`) 可供主機管理使用，而總數量的一個子集則與此主機上的一個佇列有關。

第二種標準主機聚合體中的屬性類型為預設載入數值。每個 `sge_execd(8)` 皆會定期回報載入值至 `sge_qmaster(8)` 中。回報的載入數值如不是標準的 Sun Grid Engine 載入數值，例如 CPU 載入平均值 (請參閱 `uptime(1)`)，就是由 Sun Grid Engine 管理所定義之載入數值 (請參閱叢集配置 `sge_conf(5)`、《*Sun Grid Engine Installation and Administration Guide*》、與《*Sun Grid Engine 5.3 管理和使用者指南*》的 `load_sensor` 參數以得知詳細資訊)。標準載入數值之特性定義為預設主機聚合體的一部份，而管理員定義之載入數值則需要延申主機聚合體。請參閱此檔案，`<sge_root>/doc/load_parameters.asc`，以得知標準載入數值組的詳細資訊。

主機聚合體不僅可被延申來包含非標準載入參數，還可管理主機相關資源，例如指定至主機之軟體版權數量，或者主機本端檔案系統上的可用磁碟空間。

一旦對應主機配置 `complex_values` 項目中的屬性、回報載入數值、或數值明確定義是由佇列配置而定 (請參閱 `host_conf(5)`)，就會由相關佇列配置來確認特定主機聚合體屬性上的具體數值。如果上述無一可行 (例如：此數值應為一載入參數，但是 `sge_execd(8)` 卻無法回報載入數值)，就會使用下述提及之 `value` 欄位。

- **全域聚合體** — 參考此特殊名稱**全域**而來。

全域聚合體中配置的項目可用來參照各式叢集資源屬性，例如特定軟體可用的「浮動」版權 ("floating" license) 量，或者網路可用檔案系統之空白磁碟空間。如果對應載入回報值包含 GLOBAL 辨識器 (identifier) (請參閱《*Sun Grid Engine Installation and Administration Guide*》或《*Sun Grid Engine 5.3 管理和使用者指南*》以得知詳細資訊)，全域資源屬性也可與載入回報相連結。全域載入數值可由叢集上任何主機回報得知。預設之 Sun Grid Engine 不會報告全域載入數值，所以就沒有預設全域聚合體配置。

全域聚合體屬性的具體數值若不是由全域載入回報值確認，便是由「全域」主機配置 `complex_values` 參數中的明確定義來確認（請參閱 `host_conf(5)`）。如果找不到上述兩種確認方式（例如：還沒回報載入數值），就會使用下述提及之 `value` 欄位。

- **使用者定義聚合體** — 藉由設定使用者定義聚合體，Sun Grid Engine 管理員就可延伸由 Sun Grid Engine 管理之屬性，並加以限制這些屬性於特定佇列與 / 或主機上的影響力。使用者聚合體只是一群屬性與對應定義的總稱，內含如何以 Sun Grid Engine 軟體管理這些屬性等諸如此類的資訊。只要使用 `complex_list` 佇列與主機配置參數（請參閱 `queue_conf(5)` 與 `host_conf(5)`），即可隨附一個或多個使用者定義聚合體至一個佇列與 / 或主機中。所有指定之聚合體內定義的屬性可分別用於佇列、主機、與預設聚合體屬性中。

使用者定義的具體數值必須由佇列與主機配置中的 `complex_values` 參數來設定，或者是使用下述的 `value` 欄位來設定。

格式

聚合體配置的主要格式為一圖表清單。以「#」字元開頭的每行都是一個註釋行。除了註釋行以外的每行都會定義一個聚合體要素，而要素的每行定義行皆由下述六欄項目組成（依出現順序排列）：

- **名稱** — 此為用以要求 `qsub(1) -l` 選項中作業屬性的聚合體項目名稱。屬性名稱在所有聚合體中也許僅會出現一次，此為聚合體屬性定義的獨特性。
- **捷徑** — 此為名稱的捷徑，也可用以要求 `qsub(1) -l` 選項中的作業屬性。此屬性捷徑在所有聚合體中也許僅會出現一次，以避免出現模糊不清的聚合體屬性參考。
- **種類** — 如果出現比較的情況，或載入聚合體項目中開始排列載入比例，此設定可確認 Sun Grid Engine 軟體內部要如何對待對應數值。
 - 有了 INT，則僅允許原始整數 (raw integers)。
 - 有了 DOUBLE，則能以十進位或科學計數法來指定倍準度浮點數 (floating point numbers in double precision)。
 - 有了 TIME，則僅允許時間指定器 (time specifier)。請參閱 `queue_conf(5)` 以得知格式說明。
 - 有了 MEMORY，則僅允許記憶體容量指定器 (memory size specifier)。請參閱 `queue_conf(5)` 以得知格式說明。
 - 有了 BOOL，則允許 TRUE 與 FALSE 兩個字串。若要在載入公式中使用（請參閱 `sched_conf(5)`），TRUE 與 FALSE 就必須對映於「1」與「0」中。
 - 有了 STRING，則可允許所有字串，且 `strcmp(3)` 可做為對照用。
 - CSTRING 就像 STRING，除了在進行對照時是不分大小寫之外。
 - HOST 就像 CSTRING，但是此字串必須是一有效主機名稱。
- **數值** — 數值欄位為一事先定義的屬性數值設定。此數值僅會在試圖確認佇列、主機、或 Sun Grid Engine 叢集中的屬性具體值時，在未發生覆寫的情況下才會產生影響。數值欄位可由下述數值覆寫。

- 參考佇列的佇列配置數值
- 主機特定與叢集相關之載入數值
- 利用佇列或主機配置中的聚合體值參數得到的明確數值規格（請參閱 `queue_conf(5)` 與 `host_conf(5)` 以得知詳細資訊）

如果上述無一可行，就得設定屬性數值。

- *relop* – 此為關係操作員 (*relation operator*)。當使用者要求的參數數值要與為選擇佇列所配置的對應數值相對照時，就會使用關係操作員。如果對照的結果為 `false`，此作業就無法執行於佇列中。可能的關係操作員有 `==`、`<`、`>`、`<=` 與 `>=`。唯一適用於字串種類屬性的有效操作員為 `==`。
- *可要求* – 如果此欄位設定為 `y` 或 `yes`，此項目就可用來要求 `qsub(1)` 資源。如果設定為 `n` 或 `no`，使用者就無法使用此項目來要求佇列或佇列等級。如果此項目設為 `forced` 或 `f`，此屬性就必須由作業提出要求，否則就會遭到退回。
- *可消耗* – *可消耗* 參數可設定為 `yes` (`y` 為其縮寫) 或 `no` (`n`)。僅有碰到數字型屬性 (`INT`、`MEMORY`、`TIME` – 請參閱上述提及之種類) 時才可設定為 `yes`。如果設定為 `yes`，就會由 Sun Grid Engine 內部簿記 (*internal bookkeeping*) 來管理對應資源的消耗量。在此情況中，Sun Grid Engine 會說明所有作業的資源消耗量，並確認作業僅會在 Sun Grid Engine 內部簿記顯示有足夠可用可消耗資源時才會被分派出。可消耗資源可有效管理有限的資源，例如可用的記憶體、檔案系統上的空白空間、網路頻寬、或浮動軟體版權 (*floating software license*)。

可消耗資源可與預設值，或使用者定義之載入參數相連結（請參閱 `sge_conf(5)` 與 `host_conf(5)`）。例如，為可消耗屬性回報載入數值，或者為載入屬性設定可消耗旗號 (*consumable flag*)。Sun Grid Engine 可消耗資源管理會同時考量載入（衡量資源可用性）與內部簿記，並確認此兩者皆沒有超出既定的限制。

要啟動可消耗資源管理，就必須定義基本的資源可用性。您能以叢集全域、主機、與佇列為主來完成定義的工作，因為這些種類可以依序互相替換（即主機可以限制叢集資源可用性，佇列可以限制主機與叢集資源）。資源可用性的定義會以 `host_conf(5)` 與 `queue_conf(5)` 的聚合體值項目執行。「全域」主機的聚合體值定義會指定叢集全域的可消耗資源設定。對每個聚合體值清單中的可消耗聚合體屬性而言，指定的數值皆代表資源的最大可用量，而內部簿記將自所有作業的總預計消耗資源（與作業要求的資源符合）中扣除。

注意 – 利用可要求參數的「強制 (*force*)」值即可強迫作業提出資源要求，並進而指定其預計資源消耗量（請參閱上述說明）。

注意 – 管理員可以事先替尚未明確由作業提出要求的可消耗屬性來定義預設資源的消耗值（請參閱以下的預設參數）。這僅在尚未強制要求屬性時才有意義，如上述說明。

請參閱《Sun Grid Engine 5.3 管理和使用者指南》或《Sun Grid Engine，企業版 5.3 管理和使用者指南》以得知可消耗資源功能的使用範例。

- 預設值 — 僅用於可消耗聚合體屬性時才有意義（請參閱上述 *可消耗* 參數）。Sun Grid Engine 假設分派至管理可消耗屬性的主機或佇列之作業並未明確耗損參數中的資源量。明確提出屬性要求的作業可利用 `-l` 選項至 `qsub(1)` 來置換此預設值。

同時參閱

`sge_intro(1)`、`qconf(1)`、`qsub(1)`、`uptime(1)`、`host_conf(5)`、`queue_conf(5)`、`sge_execd(8)`、`sge_qmaster(8)`、`sge_schedd(8)`、《*Sun Grid Engine Installation and Administration Guide*》、《*Sun Grid Engine 5.3 管理與使用者指南*》

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

host_conf(5)

名稱

`host_conf` — Sun Grid Engine 執行主機配置檔案格式

說明

`Host_conf` 會反映執行主機配置的範本檔案 (template file)。透過 `qconf(1)` 指令的 `-ae` 與 `-me` 選項，你就可以新增執行主機，與修改叢集中執行主機的任意配置。只要 `sge_execd(8)` 第一次由特定主機記錄至 `sge_qmaster(8)`，就會自動新增預設執行主機記錄。`qconf(1) -sel` 開關 (switch) 可以用來顯示您的 Sun Grid Engine 系統中最新配置的執行主機清單。透過 `-se` 選項您就可以列出指定主機的執行主機配置。

此特殊主機名稱，`global`，可以用來定義叢集全域的特性。

格式

host_conf 檔案的格式定義如下。

- **主機名稱** – 此為執行主機名稱。
- **載入比例** – 此為一以逗點分隔之比例數值清單，以供主機上的 sge_execd(8) 回報之單一或部分載入數值使用，而此載入數值會定義於叢集全域的「主機」聚合體中（請參閱 complex(5)）。載入比例因數會將執行主機間的硬體或作業系統之特定差異做等級欄位分。舉例來說，如果多重處理器的載入平均值（主機聚合體中的 load_avg；請參閱 uptime(1)）須與單一處理器比較，由單一 CPU 主機回報的數值必須先加權，才可與多重處理器數值（相同的 CPU 硬體）相比較。載入比例因數為整數，且會與回報之載入量相乘以建立加權的載入數值。因此，依上述範例所示，單一處理器的載入數值需要乘以單一處理器內的處理器數量後才可與多重處理器比較。

載入因數規格的語法如下：首先為載入數值的名稱（與「主機」聚合體定義相同），並以等號相接載入比例數值。在載入比例數值字串間不可留任何空白。

在「全域」主機中，此參數**載入比例**是沒有任何意義的。

- **聚合體清單** – 此為逗號分隔之管理員定義聚合體清單（請參閱 complex(5) 以得知詳細資訊），且此清單與主機相關。只有包含在列出聚合體中，與「全域」和「主機」聚合體中的聚合體屬性（未明確隨附至每個主機）才可用於下述**聚合體值**清單中。以「全域」主機為例，在「全域」主機的預設**聚合體值**清單中並沒有隨附「主機」聚合體，而且僅允許「全域」聚合體屬性。

此參數的預設值為 NONE，意即無任何管理員定義之聚合體與主機有關。

- **聚合體值** – 聚合體值會透過主機管理來定義資源屬性的配額。「=」記號後方即為聚合體屬性，並符合聚合體屬性種類的數值規格（請參閱 complex(5)）。配額的規格是由逗點分隔的。僅可使用**聚合體清單**中（請參閱上述說明）定義的屬性。

就可消耗資源而言，配額與主機上的所有作業之資源消耗有關（請參閱 complex(5) 以得知可消耗資源的詳細資訊）；就非可消耗資源而言，配額是解讀於單一作業插槽上。可消耗資源屬性一般是用以管理空白記憶體、空白磁碟空間、或可用的浮動軟體版權，而非可消耗資源屬性則可定義特殊屬性，例如安裝硬體的種類。

就可消耗資源屬性而言，可用的資源量會由**聚合體值**清單中的配額，減去所有執行於主機上的作業的最新消耗資源來確認。僅有當資源要求皆不超出由此方案得到的對應可用資源時，才會將作業分派至主機上。如果此資源會監控載入數值，而且回報的載入數值比配額更可靠的話，在**聚合體值**清單中的配額定義便會自動由此屬性回報之最新載入數值取代。此作法可有效避免資源過度耗損。

注意 – 取代配額特性的載入數值也許會變得更可靠，因這些數值皆已被排列過（請參閱上述之載入排列），而且 / 或者被載入調整過（請參閱 sched_conf(5)）。qstat(1) 中的 -F 選項與 qmon(1) 佇列控制對話程序（同時按下 Shift 鍵與佇列圖示啓動）中的載入顯示會提供可消耗資源的實際可用量，與最近考量的原始數值之詳細資料。

注意 – 執行中作業的資源消耗量（用以計算可用性）與等待分派的作業資源要求也許會經由作業提交期間由明確的使用者提出要求（請參閱 `-l` 選項至 `qsub(1)`），或者是由管理員配置的屬性「預設」值（請參閱 `complex(5)`）而得到。`-r` 選項至 `qstat(1)` 可用來獲得系統中所有作業實際資源要求的完整詳細資訊。

就非可耗損資源而言，Sun Grid Engine 僅會比較將聚合體屬性列入考量之聚合體值中的作業屬性要求與對應規格（請參閱 `complex(5)`）。如果比較的結果為「true」，此主機就適用於與此特定屬性相關之作業。就平行作業而言，每個由平行作業佔據之作業插槽應該會提供相同的資源屬性數值。

注意 – 僅有數字的聚合體屬性才可定義為可消耗資源，所以非數字的屬性總是於單一作業插槽上受到管理。

此參數的預設值為 NONE，意即無任何管理員定義之資源屬性配額與主機有關。

- **載入數值** – 此項目無法被配置，且只有在執行 `qconf(1) -se` 指令時才會顯示。所有載入數值的顯示值皆會符合主機上 `sge_execd(8)` 的回報值。載入數值會列於一張以逗點分隔之清單中。每個載入數值會以其名稱開頭，等號後即為回報的數值。
- **處理器** – 此項目無法配置，且只有在執行 `qconf(1) -se` 指令時才會顯示出來。其數值即為處理器的數量，且會對應主機上的 `sge_execd(8)` 偵測到。
- **使用比例** – 此項目僅會顯示於 Sun Grid Engine，企業版系統中，且並不適用於 Sun Grid Engine 中。

使用格式與**載入比例**相同（請參閱上述說明），但是唯一依比例的有效屬性則為 CPU 時間消耗量的 `cpu`、整合作業總生命期之記憶體消耗量的 `mem`、與使用 I/O 裝置傳送資料的 `io`。預設值 NONE 代表「無比例 (no scaling)」，即所有比例因數為 1。

- **資源功能因數 (resource_capability_factor)** – 此項目僅會顯示於 Sun Grid Engine，企業版系統中，且並不適用於 Sun Grid Engine 中。

Sun Grid Engine，企業版使用資源功能因數來指定作業至執行主機上。資源功能因數會告知 Sun Grid Engine，企業版執行主機上的資源（如 CPU、記憶體、I/O 等）會如何與其它執行主機的資源相比較。這將有助於確認需要大量資源比例（例如：許多票卷）的作業是被置於包含大量可用資源比例的執行主機上，還會將執行主機上的載入情況列入考量，以保證選取的主機功能不僅夠強大，還可適用低負載。

舉例來說，您也許可以考慮以 CPU 數量、CPU 速度、與安裝的主要記憶體為主，在執行主機上設定您的資源可用性因素：

```
#_of_CPUs * (MHz/200) + GB_of_memory
```

執行主機會得到 32 200 MHz 的 CPU 與 10 GB 的記憶體，即資源可用性因數 42，而設定資源功能因數為 64 的執行主機則會得到 24 200 MHz 的 CPU 與 40 GB 的記憶體，即在此範例中記憶體會受到重大的影響。

至於其他您會考慮如何設定資源功能因數的要素為：說明如下。

- **混合作業** – CPU 或記憶體混合作業

- CPU 基準 — 由 CPU 廠商做比較
- Mmegaflops (MFLOPS) — 數字運算用
- I/O 功能 — 磁碟 / 網路速度
- 可用磁碟空間 — 位於執行主機上

資源功能因數會以浮點雙數值被儲存。使用的數值範圍並不重要。Sun Grid Engine，企業版僅會注意不同主機數值之間的關係。

同時參閱

`sg_intro(1)`、`qconf(1)`、`uptime(1)`、`complex(5)`、`sg_execd(8)`、`sg_qmater(8)`

版權

請參閱 `sg_intro(1)` 以得知權利與權限的完整聲明。

hostgroup(5)

名稱

`hostgroup` — 主機群組項目檔案格式

說明

`hostgroup` 項目可用以合併主機名稱至群組中。每個 `hostgroup` 項目檔案皆會定義一個群組。名稱的第一個字元有 @ 記號者，即代表此為群組。在此情況下，您可以使用 `usermapping(5)` 配置中的 `hostgroup`。在群組定義檔案中，您同時也可參考其他群組；這些群組就叫做子群組。

`hostgroup` 項目檔案中的每行都會指定一屬於此群組的主機名稱或群組。

您可透過 `qconf(1) -shgrp1` 選項來顯示最新配置的 `hostgroup` 項目清單，也可透過 `-shgrp` 開關 (switch) 來顯示每項列出的 `hostgroup` 項目內容。輸出資訊會尾隨 `hostgroup` 格式說明而出。使用 `-ahgrp`、`-mhgrp`、與 `-dhgrp` 選項至 `qconf(1)`，您就可以建立或修改 `hostgroup` 項目。

格式

hostgroup 項目至少包含兩種參數。

- *群組名稱* 關鍵字 – *群組名稱* 關鍵字會定義 hostgroup 名稱。在關鍵字 *群組名稱* 之後的其餘文字行將會被視為 hostgroup 名稱數值。
- *主機名稱* – 此為主機名稱，是以 *群組名稱* 指定之現有群組的成員之一。如果 *主機名稱* 的第一個字元為 @ 記號，此名稱即可用以參考 hostgroup(5)，即為組群中的子群組。

範例

以下為典型的 hostgroup 項目。

```
group_name bigMachines
@calculate
speedhost
```

此項目將定義新的 hostgroup，名為 bigMachines。其中，hostgroup 為主機 speedhost，與 hostgroup、calculate 的所有成員。

同時參閱

qconf(1)、usermapping(5)

版權

請參閱 sge_intro(1) 以得知權利與權限的完整聲明。

project(5)

名稱

project — Sun Grid Engine，企業版專案項目檔案格式

說明

專案物件僅適用於 Sun Grid Engine，企業版系統。Sun Grid Engine 沒有專案物件。

在 Sun Grid Engine，企業版中可以提交作業至專案，也可使用功能或置換策略來指定重要性的特定等級。而重要性的等級則是由執行於專案底下的作業傳承下來的。

透過 `qconf(1) -sprjl` 選項即可顯示目前配置的專案清單，而每個列出的專案定義內容則可使用 `-sprj` 開關 (switch) 來顯示。輸出資訊會尾隨 `project` 格式說明而出。透過 `-aprj`、`-mprj`、與 `-dprj` 選項至 `qconf(1)` 就可以新增專案，與修改現有專案。

格式

專案定義會包含下述參數：

- `name` — 此為專案名稱。
- `oticket` — 此為目前指定給專案的置換票卷量。
- `fshare` — 此為目前的專案功能共享。
- `facl` — 此為使用者存取清單一覽表 (ACL — 請參閱 `access_list(5)`)，內含允許提交作業至專案的使用者。
- `fxacl` — 此為使用者存取清單一覽表 (ACL — 請參閱 `access_list(5)`)，內含不被允許提交作業至專案的使用者。

同時參閱

`sge_intro(1)`、`qconf(1)`、`access_list(5)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

qtask(5)

名稱

qtask – qtask 檔案的檔案格式。

說明

qtask 檔案會定義該提交哪個指令至 Sun Grid Engine，以供 `qtcs(1)` 做遠端執行。qtask 檔案也許會選擇性地包含 `qrsh(1)` 指令行參數。這些參數會傳遞至 `qtcs(1)` 所使用的 `qrsh(1)` 指令來提交指令。

定義叢集廣域預設的叢集全域 qtask 檔案，與最終置換和增加定義的使用者特定 qtask 檔案有受到支援。叢集全域檔案位於 `<sge_root>/<儲存格>/common/qtask`，而使用者特定檔案則位於 `~/.qtask`。管理員可於叢集全域中將驚嘆號置於指令定義之前，以拒絕使用者置換指令。

格式

qtask 檔的主要格式為圖表清單。以「#」字元開頭的每行都是一個註釋行。除了註釋行之外的每行都會定義要遠端啟動的指令。

定義必須以輸入於 `qtcs(1)` 指令行的名稱完全符合的指令名稱開頭。在 qtask 檔案中不允許有路徑名稱。因此，在 `qtcs(1)` 指令行中的絕對或相對路徑名稱會導致本端執行 (local execution)，即使指令本身與 qtask 檔案的定義相同。

`qrsh(1)` 選項引數的仲裁數量會尾隨一指令名稱，這些選項引數會由 `qtcs(1)` 傳遞至 `qrsh(1)`。在叢集全域 qtask 檔案中，指令前的驚嘆號可預防受到使用者提供的 qtask 檔案置換。

範例

```
netscape -l a=solaris64 -v DISPLAY=myhost:0
grep -l h=filesurfer
verilog -l veri_lic=1
```

上述 `qtask` 檔案會指定應用程式 `netscape`、`grep` 與 `verilog`，以透過 Sun Grid Engine 軟體做互動遠端執行。`Netscape` 僅可執行於 Solaris64 架構中且 `DISPLAY` 環境變數需設為 `myhost:0`、`grep` 僅執行於名為 `filesurfer` 的主機中、而 `verilog` 則會要求 `verilog` 版權的可用性以達成遠端執行的目的。

同時參閱

`sge_intro(1)`、`qtcsh(1)`、`qrsh(1)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

queue_conf(5)

名稱

`queue_conf` — Sun Grid Engine 佇列配置檔案格式

說明

`queue_conf` 會顯示佇列配置的平台檔案格式。透過 `qconf(1)` 指令的 `-aq` 與 `-mq` 選項，您可新增佇列，並修改叢集中任何佇列配置。

`queue_conf` 參數為數值字串 (value strings)、整十進位數值或布林值 (boolean)、時間與記憶體指定器 (specifier)、與逗點分隔之清單。時間指定器是由正十進位、十六進位、或八進位整數組成 (於幾秒內將數值整合)，或者是由冒號分隔之三個十進位整數建立 (第一個數字計算小時、第二個為分、而第三個為秒)。如果數值為零就可省去，但仍必須保留分隔的冒號 (例：`1:0:1 = 1::1` 代表一小時又一秒)。

記憶體指定器為正十進位、十六進位、或八進位整數常數，並尾隨於乘號字母之後。有效的乘號字母為 `k`、`K`、`m`、與 `M`；其中 `k` 代表乘以 `1000`、`K` 代表乘以 `1024`、`m` 代表乘以 `1000*1000`、`M` 代表乘以 `1024*1024`。如果沒有出現乘號字母，就以位元組計算數值。

格式

下述的 `queue_conf` 參數清單會說明 `queue_conf` 內容：

- *佇列名稱* – 此為節點上的佇列名稱 (種類為字串；平台預設：平台)。
- *主機名稱* – 此為節點的完整主機名稱 (種類為字串；平台預設：`host.dom.dom.dom`)。
- *seq_no* – 在主機載入情況中，此參數會一次於排程順序中指定佇列位置與適合分派作業 (在 *佇列分類法* 考慮下) 的佇列 (請參閱 `sched_conf(5)`)。

不管 `queue_sort_method` 設定，`qstat(1)` 都會依 `seq_no` 數值定義的順序回報佇列資訊。設定此參數至單調遞增順序 (monotonically increasing sequence)。種類為數量，而預設為 0。

- *載入臨界值* – 此為載入臨界值清單。如果已超過一臨界值，則此節點不會再做作業排程至佇列，而且 `qmon(1)` 將會顯示節點的超載情況。可以使用「主機」與「全域」聚合體中定義的仲裁載入數值 (請參閱 `complex(5)` 以得知詳細資訊)。

此語法為由逗號分隔之清單，且每個清單皆由載入數值的名稱、等號、與引發超載情況的臨界數值組成 (例：`load.avg=175, users_logged_in=5`)。

注意 – 如果在對應執行主機定義中有指定載入數值，載入數值與可耗損資源也許會由不同主機分別測量 (請參閱 `host_conf(5)` 得知更多資訊)。載入臨界值會與受測之載入與可耗損數值相比較。

- *暫停臨界值* – 此為載入臨界值清單。除了超出指定臨界值時就會暫停佇列中的多項作業之一不同以外，其與 *載入臨界值* 參數的語義是相同的 (請參閱上述說明)。請參閱下述 `nsuspend` 參數以得知暫停作業的數量詳細資訊。
- *nsuspend* – 當 *暫停臨界值* 清單中至少有超出一項載入臨界值，或者完全沒有超出 *暫停臨界值* 時，其分別於間隔期間一次暫停 / 啟動之作業數量。`nsuspend` 作業會於每個間隔期間中被暫停，直到沒有超出任何 *暫停臨界值*，或者乾脆直接暫停佇列上的所有作業。如果不再超出 *暫停臨界值*，作業就會對應啟動。暫停作業發生的間隔期間會定義於下述 *暫停間隔時間* 中。

- **暫停間隔期間** — 當佇列所在主機上的現有數值超出其中一個**暫停臨界值**（請參閱上述說明）時，此為更多 *nsuspend* 作業暫停的間隔期間。啟動作業時也會用到間隔期間。
- **優先權** — **優先權** 參數會指定佇列中將執行之作業的 *nice*(2) 值。種類為數量，且預設值為零（意即沒有明確設定 *nice* 值）。負數值（最多到 -20）會對應至更高的排程優先權，而正數值（最多到 +20）則會對應至更低的排程優先權。在 Sun Grid Engine，企業版系統中，**優先權** 的數值不會造成任何影響，因為 Sun Grid Engine，企業版系統會動態調整優先權以執行 Sun Grid Engine，企業版的應達策略目標。但是動態調整優先權可以藉由設定全域或執行常駐程式本端叢集配置中的**執行參數** *NO_REPRIORITIZATION* 至 *true* 以關閉此功能（請參閱 *sge_conf*(5)）。在此情形中，**優先權** 的設定也將對 Sun Grid Engine，企業版系統造成影響。
- **最小 *cpu* 間隔時間** — 如果發生明顯的檢查點作業時，此為兩個自動檢查點之間的時間。使用者透過 *qsub*(1) 要求的最多時間，與佇列配置定義的時間可用來當作檢查點的間隔期間。既然檢查點檔案也許會很大，而且將其寫入檔案系統也許會浪費時間，所以建議使用者與管理員有效選擇時間較長之間隔期間。**最小 *cpu* 間隔時間** 為時間種類的一種，其預設值為五分鐘（通常僅供測試目的用）。
- **處理器** — 當多重處理器執行主機定義於佇列作業時，此為處理器組。此參數的數值種類會說明一範圍，就像 *qsub*(1) 的 *-pe* 選項（例：1-4、8、10），顯示處理器群組使用的處理器數量。當然這些數值的解讀就得仰賴特定作業系統，也就會顯示於執行在佇列主機的 *sge_execd*(8) 之中。因此，這些參數的語法分析就必須由執行常駐程式提供，而且參數也僅可以字串的形式通過 *sge_qmaster*(8)。

目前僅有執行 IRIX 6.2 與 Digital UNIX 多重處理器的 SGI 多重處理器才有提供支援。在 Digital UNIX 的情況中，同時間僅允許單一處理器組單一作業；即佇列中的**插槽**（見下述）應設為 1。

- **佇列種類** — 此為佇列種類。目前在逗點分隔的清單中只有**批次**、**互動**、**平行**或**檢查點**或其他組合。此種類為字串；預設值為 *batch interactive parallel*。
- **重新執行** — 因佇列主機上的系統當機，或因手動「暴力」（使用 *kill*(1)）關閉整個 Sun Grid Engine 系統（包括作業與其執行階層之 *sge_shepherd*(8)）而導致作業失敗時，此參數會定義一預設運作模式。只要 *sge_execd*(8) 再次被啟動，並偵測到一作業因上述原因而執行失敗時，此作業就可再被啟動 — 如果作業是可以被重新啟動的話。舉例來說，如果一作業更新其資料庫，就有可能無法重新啟動此作業（首先讀取然後寫到資料庫 / 檔案相同的紀錄），因為執行失敗的作業會使其資料庫處於不一致的狀態。如果作業的所有人想自設佇列作業的預設行為，就可以使用 *qsub*(1) 中的 *-r* 選項。

此參數的種類為布林值 (boolean)，所以可以指定 *TRUE* 或 *FALSE*。預設值為 *FALSE*，即為不自動重新啟動作業。

- **插槽** — 此為同時允許佇列上執行之作業最大量。種類為數量。
- ***tmpdir*** — ***tmpdir*** 參數會指定絕對路徑至暫時的目錄檔案系統中。當 *sge_execd*(8) 執行作業時，就會於此檔案系統中建立獨特命名之目錄，以利於作業執行期間存放暫存檔案 (scratch files)。當作業結束時，此目錄及其內容就會被自動移除。環境變數 *TMPDIR* 與 *TMP* 會設為每項作業暫存目錄之路徑（種類為字串；預設：/tmp）。

- *shell* – 如果 *posix_compliant* 或 *script_from_stdin* 其中之一被指定為 `sge_conf(5)` 之中的 *shell 啟動模式* 參數，*shell* 參數就會指定指令解讀器的可執行路徑（例如：使用 `sh(1)` 或 `csh(1)` 來執行位於佇列中的作業 *script*）。*shell* 的定義可由作業所有人透過 `qsub(1) -S` 來控制。

參數種類為字串，預設值為 `/bin/csh`。

- *shell 啟動模式* – 此參數會定義用於執行主機上實際啟動作業 *script* 的機制。下述值可供辨識：
 - *unix 運作方式* – 如果使用者於 UNIX 底下以 *script* 名稱來互動地啟動作業 *script*，此作業系統的可執行載入器會使用註釋中的資訊，如 *script* 第一行中的 `#!/bin/csh`，來偵測用哪個指令解讀器啟動解讀 *script*。如果 *unix 運作方式* 定義為 *shell 啟動模式*，則啟動作業時，Sun Grid Engine 就會使用此機制。
 - *posix_compliant* – POSIX 並不會將 *script* 的第一行註釋，如 `#!/bin/csh`，視為重要訊息。因此，批次佇列系統 (P1003.2d) 的 POSIX 標準需要一相容之佇列系統以忽略第一行註釋，並轉而利用使用者指定或配置之預設解讀器。因此，如果 *shell 啟動模式* 設定為 *posix_compliant*，Sun Grid Engine 就會使用指令解譯器，由 `qsub(1)` 指令的 `-S` 選項，或使用佇列之 *shell* 參數指定（請參閱上述說明）。
 - *script_from_stdin* – 要設定 *shell 啟動模式* 參數至 *posix_compliant* 或 *unix 運作方式* 時，需要先設定 `sge_execd(8)` 執行 `umask`，如此一來所有使用者才可讀到存取至對應執行常駐程式之多工緩衝處理目錄 (pool directory) 中的 `active_jobs` 目錄。如果您有配置 *prolog* 與 *epilog* *script*，這兩個 *script* 也必須讓可能執行作業的使用者讀到。

如果這侵犯了您的站點的安全策略，您也許可設定 *shell 啟動模式* 至 *script_from_stdin*。這會強制 Sun Grid Engine 在變更至作業所有人的使用者帳號前，開啓作業 *script* 以及 *pilogue* 和 *prologue* *script*，以 `root` 讀入 STDIN（如果 `sge_execd(8)` 是以 `root` 啟動）。然後，此 *script* 就會被輸入指令解讀器的 STDIN 串流之中。指令解讀器是由 `qsub(1)` 指令之 `-S` 選項，或由使用佇列之 *shell* 參數指定（見上述說明）。

因此，設定 *shell 啟動模式* 至 *script_from_stdin* 也包含 *posix_compliant* 運作方式。

注意 – 如果諸如 `rsh(1)` 的指令啟動於作業 *script* 中，而其又執行指令解譯器的 STDIN 串流，將 *scripts* 輸入指令解譯器的 STDIN 中也許會產生麻煩。但只要將這些指令的 STDIN 通道重新由 `/dev/null`（例如：`rsh 主機日期 < /dev/null`）導入即可解決此類問題。

注意 – 任何與作業相關之指令行選項會通過至執行中的 *shell*。如果這些指令行選項無法被辨視為有效 *shell* 選項，此 *shell* 就只能將其轉寄至作業。

shell 啟動模式 的預設值為 *posix_compliant*。

- *prolog* — 在執行 Sun Grid Engine 作業與和 Sun Grid Engine 作業啟動之後的相同環境設定前，此為已啟動之 shell script 的可執行路徑。字首 user@（非必要項目）會於欲啟動的程序之下指定使用者。此程序被視為 Sun Grid Engine 管理員用以自動執行一般站點特定作業（如：依據與作業相同內容資訊來準備暫時檔案系統）的方法。此佇列配置項目會覆寫叢集全域或執行主機之特定 *prolog* 定義（請參閱 sge_conf(5)）。

注意 — *prolog* 會完全被視為作業 script 來執行。因此，所有說明於參數 *shell* 啟動模式與登入 shells 底下的含義都適用。

prolog 的預設值為特殊值 NONE，可避免執行 prologue script。建構指令行的特殊變數與叢集配置的 *prolog* 定義中相同（請參閱 sge_conf(5)）。

- *epilog* — 在執行 Sun Grid Engine 作業與和完成 Sun Grid Engine 作業相同之環境設定後，此為啟動之 shell script 的可執行路徑。字首 user@（非必要項目）會於欲啟動的程序之下指定使用者。此程序被視為 Sun Grid Engine 管理員用以自動執行一般站點特定作業（如：依據與作業相同內容資訊來清除暫時檔案系統）的方法。此佇列配置項目會覆寫叢集全域或執行主機之特定 *epilog* 定義（請參閱 sge_conf(5)）。

注意 — *epilog* 會完全被視為作業 script 來執行。因此，所有說明於參數 *shell* 啟動模式與登入 shell 底下的含義都適用。

epilog 的預設值為特殊值 NONE，可避免執行 epilogue script。建構指令行的特殊變數與叢集配置 *prolog* 定義中的相同（請參閱 sge_conf(5)）。

- **啟動器法** — 指定的可執行路徑將用為作業啟動器工具，用以負責啟動批次作業。此可執行路徑將會被執行，而非配置 shell 來啟動作業。作業引數會以引數之姿傳遞至作業啟動器。下述環境變數會考量被配置或要求以啟動作業的 shell 環境，將傳遞資訊至作業啟動器。
 - **SGE 啟動器 SHELL 路徑** — 用以啟動作業的要求 shell 之名稱。
 - **SGE 啟動器 SHELL 啟動模式** — 此為配置的 shell 啟動模式。
 - **SGE 啟動器使用登入 SHELL** — 如果 shell 做為登入 shell（請參閱 sge_conf(5) 中的登入 shell），就將此變數設為 true。

啟動器法將不會用以 rlogin 之姿出現的 qsh、qlogin、或 qrsh 來啟動。

- **暫停法、恢復法、中止法** — 這三個參數可以用來覆寫由 Sun Grid Engine 用來暫停、暫停的釋出、或中止作業的預設法。每項預設值、訊號 SIGSTOP、SIGCONT 與 SIGKILL 都會傳遞至作業來執行這些動作，但卻不適用於一些應用程式。

如果沒有給予任何可執行路徑，Sun Grid Engine 就會以指定的參數項目做為傳遞的訊號，而非預設訊號。訊號必須為正數，或以 SIG 為首之訊號名稱，而訊號名稱即為由 kill -l 列出之名稱（例如：SIGTERM）。

如果有給予可執行路徑，（必須為以「/」開始之絕對路徑），就可由 Sun Grid Engine 啟動本指令與其引數來執行合適之動作。以下的特殊變數會在執行期間擴充，並可用來（除了須由程序解釋之其他字串之外）建構一指令行：

- *\$host* — 此為啟動程序的主機名稱。

- *\$job_owner* — 此為作業所有人的使用者名稱。
 - *\$job_id* — 此為 Sun Grid Engine 系統的獨特作業辨認號碼。
 - *\$job_name* — 此為作業名稱。
 - *\$queue* — 此為佇列名稱。
 - *\$job_pid* — 此為作業的 pid。
 - **通知** — 如果作業使用 `qsub(1) -notify` 選項提交，此為傳遞 SIGUSR1/SIGUSR2 通知訊號與暫停 / 中止訊號的等待時間。
 - **所有人清單** — (於以逗號分隔之清單中) 被授權可暫停此佇列之使用者的 *所有人清單* 名稱與登入名稱 (Sun Grid Engine 操作員與管理員可預設暫停佇列)。通常會於互動工作站上為佇列設定此區。此作業站為互動階段與 Sun Grid Engine 作業之間共享計算資源的地方，讓工作站的所有人擁有優先存取 (鍵入字串；預設：NONE)。
 - **使用者清單** — *使用者清單* 參數包含 `access_list(5)` 中所說明由逗點分隔之 *使用者存取清單*。每個至少列於存取清單之一的使用者皆可存取佇列。如果 *使用者清單* 參數設定為 NONE (預設值)，任何未明確包含於 *x 使用者清單* 參數 (見下述說明) 中的使用者皆可存取。如果使用者同時列於存取清單中的 *x 使用者清單* 與 *使用者清單* 中，則此使用者會被拒絕存取佇列。
 - **x 使用者清單** — *x 使用者清單* 參數為一逗號分隔如 `access_list(5)` 說明之 *使用者存取清單*。每個至少列於存取清單之一的使用者皆無法存取佇列。如果 *x 使用者清單* 參數設為 NONE (預設值)，則任何使用者皆可存取。如果使用者同時列於存取清單中的 *x 使用者清單* 與 *使用者清單* 中，則此使用者會被拒絕存取至佇列。
 - **專案** — *專案* 參數為一可存取佇列的逗點分隔之專案清單。未列入清單中的專案就無法存取佇列。如果設定為 NONE (預設值)，則未特別由 *x 專案* 參數 (請參考下述說明) 排除之任何專案皆可存取。如果一專案同時位於 *專案* 與 *x 專案* 參數中，則此專案將無法存取佇列。此參數僅適用於 Sun Grid Engine，企業版系統。
 - **x 專案** — *x 專案* 參數包含被拒絕存取至佇列的由逗點分隔之專案清單。如果設定為 NONE (預設值)，就不會拒絕任何專案存取，除了那些以 *專案* 參數為主被拒絕存取的專案。如果一專案同時位於 *專案* 與 *x 專案* 參數中，則此專案將無法存取佇列。此參數僅適用於 Sun Grid Engine，企業版系統。
 - **附屬清單** — 當一特定數量之作業執行於佇列中時，此為欲暫停之位於相同主機上的已配置 Sun Grid Engine 佇列清單。此清單規格與上述之 *載入臨界值* 相同；即 `low_pri_q=5, small_q`。顯示數字的數字為必須啟動次級佇列暫停功能之佇列作業插槽量。如果沒有指定數值，當填滿所有佇列插槽時才會啟動暫停功能。
- 在管理一個佇列以上的節點上，您也許會想使用與特定作業等級中相同的較佳服務 (例如：用以執行平行作業之佇列也許會需要較高之優先生產佇列；預設值為 NONE)。
- **聚合體清單** — 此為與佇列相關之管理員定義聚合體清單 (請參閱 `complex(5)` 得知詳細資訊)，內以逗點分隔。僅有位於列出聚合體中，與隨附於佇列中的來自 *全域、主機、與佇列* 聚合體的那些聚合體中的聚合體屬性才可用於下述 *聚合體值* 清單中。

本參數的預設值為 NONE；即無任何管理員定義聚合體與佇列有關。

- **聚合體值** — **聚合體值** 會定義由佇列管理資源屬性的配額。此出現於 **聚合體值** 之中的被允許 **complex** 屬性會由 **聚合體清單** 定義（請參閱上述說明）。使用語法與 **載入臨界值** 相同（請參閱上述說明）。就可耗損資源而言（請參閱 **complex(5)** 得知可耗損資源的詳細資訊），配額多寡會與佇列中的所有作業資源耗損量有關；就非可耗損資源而言，就會依據一佇列插槽（請參閱上述 **插槽**）來解釋配額量。可耗損資源屬性通常用以管理空白記憶體、空白磁碟空間、或可用浮動軟體版權，而非可耗損資源屬性則可定義特殊特性，例如安裝硬體的種類。

就可消耗資源屬性而言，可用的資源量會由 **聚合體值** 清單中的配額，減去所有執行於佇列上的作業的最新消耗資源來確認。僅有當資源要求不超出由此方案得到的對應可用資源時，才會將作業分派至佇列中。如果此資源會監控載入數值，而且回報的載入數值比配額更可靠的話，在 **聚合體值** 清單中的配額定義會自動由此屬性回報之最新載入數值取代。此作法可有效避免資源過度耗損。

注意 — 取代配額規格的載入數值也許會變得更可靠，因這些數值皆已被測量過（請參閱 **host_conf(5)**），而且 / 或者被載入調整過（請參閱 **sched_conf(5)**）。**qstat(1)** 中的 **-F** 選項與 **qmon(1)** 佇列控制對話（同時按下 **Shift** 鍵與佇列圖示啟動）中的載入顯示會提供可耗損資源的實際可用量，與最近考量的原始數值之詳細資料。

注意 — 執行作業的耗損資源量（用於計算可用量）與等待分派之作業資源要求也許會源於作業提交期間的明確使用者要求（請參閱 **-l** 選項至 **qsub(1)**），或者源於由管理員配置屬性之「預設」值（請參閱 **complex(5)**）。**-r** 選項至 **qstat(1)** 可用以獲得系統中所有作業實際資源要求的完整詳細資訊。

就非可耗損資源而言，**Sun Grid Engine** 僅會比較將聚合體屬性列入考量之 **聚合體值** 中的作業屬性要求與對應規格（請參閱 **complex(5)**）。如果比較的結果為 **true**，此佇列就適用於與此特定屬性相關之作業。就平行作業而言，每個由平行作業佔據之佇列插槽應該會提供相同的資源屬性數值。

注意 — 僅有數字的聚合體屬性才可定義為可耗損資源，所以非數字的屬性總是於單一佇列插槽上受到管理。

本參數的預設值為 **NONE**；即無任何管理員定義的資源屬性配額與佇列有關。

- **行事曆** — 這會指定 **行事曆** 對此佇列有效，或者包含 **NONE**（預設值）。行事曆會依據日期、星期、與年的時間來定義佇列的可用性。請參閱 **calendar_conf(5)** 以得知 **Sun Grid Engine** 行事曆工具的詳細資訊。

注意 — 作業可透過 **-l c=<cal_name>** 選項至 **qsub(1)** 以特定行事曆模式來要求佇列。

- **初始狀態** — 當第一次新增佇列至系統，或者在佇列所在的主機上啟動 **sge_execd(8)** 時，此參數會定義佇列的初始狀態。可能的數值為：

- **預設值** — 當新增佇列時，或當 `sge_execd(8)` 啟動而重設至之前的設定時（此與先前未支援初始狀態的 Sun Grid Engine 發行運作模式相對應），就會啟動佇列。
- **啟動** — 在任一情況下皆可啟動佇列。這與手動且明確之 `qmod -e` 指令相同（請參閱 `qmod(1)`）。
- **關閉** — 在任一情況下皆會關閉佇列。這與手動且明確之 `qmod -d` 指令相同（請參閱 `qmod(1)`）。
- **功能共享** — 本參數僅適用於 Sun Grid Engine，企業版系統之中，Sun Grid Engine 並不支援此參數。
佇列的功能共享（例如：作業等級）。如果有啟動功能策略的話，執行於佇列的作業也許會自佇列相對重要性得到功能票卷。
- **置換票卷** — 本參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
佇列的置換票卷（例如：作業等級）。Sun Grid Engine，企業版會從所有執行於佇列中的作業來分配配置的置換票卷量。

資源限制

先兩個資源限制參數，`s_rt` 與 `h_rt`，會由 Sun Grid Engine 啟動執行。在作業啟動時，此兩參數會定義已傳遞之 "real time"，或又稱為 "elapsed" 或 "wall clock" 時間。如果執行於佇列中的作業已超出 `h_rt`，此作業就會藉由 SIGKILL 訊號而宣告失敗（請參閱 `kill(1)`）。如果已超出 `s_rt`，SIGUSR1 訊號（作業可收到此訊號）就會第一次「警告」此作業，而在佇列配置參數通知（請參閱上述）中定義的通知時間過了之後，最後就會宣告作業失敗。

資源限制參數 `s_cpu` 與 `h_cpu` 會由 Sun Grid Engine 啟動執行，以做為一作業限制。它們會在結合的 CPU 時間量（由作業中所有過程消耗）上使用此限制。如果執行於佇列中的作業超出 `h_cpu`，SIGKILL 訊號（請參閱 `kill(1)`）就會中斷此作業。如果超出 `s_cpu`，作業就會接收到一個可由作業抓到的 SIGXCPU 訊號。如果您想允許作業得到「警告」，好讓其在被中斷前優雅地退出，您就該設定 `s_cpu` 限制至比 `h_cpu` 更低的數值。對平行作業來說，此限制只能得到單一插槽，意即在使用此限制前，此限制是乘以作業使用的插槽量。

資源限制參數 `s_vmem` 與 `h_vmem` 會由 Sun Grid Engine 啟動執行，以做為一作業限制。它們會在結合的虛擬記憶體量（由作業中所有過程消耗）上使用此限制。如果執行於佇列中的作業超出 `h_vmem`，SIGKILL 訊號（請參閱 `kill(1)`）就會中斷此作業。如果超出 `s_vmem`，作業就會接收到一個可由作業抓到的 SIGXCPU 訊號。如果您希望允許作業得到「警告」，好讓其在被中斷前優雅地退出，您就該設定 `s_vmem` 限制至比 `h_vmem` 更低的數值。對平行作業來說，此限制只能得到單一插槽，意即在使用此限制前，此限制是乘以作業使用的插槽量。

其他佇列配置平台中的參數則會指定單一作業的軟 (soft) 與硬 (hard) 資源限制，並由 `setrlimit(2)` 系統呼叫啟動執行。請參閱您系統中的線上說明手冊來得知更多資訊。在預設模式中，每個限制區都會設為無限（意即 `RLIM_INFINITY`，請參閱線上說明手冊中的 `setrlimit(2)` 說明）。CPU 時間限制 `s_cpu` 與 `h_cpu` 的數值種類為時間，而其他限制的數值種類則為記憶體。

注意 – 並非所有系統皆支援 `setrlimit(2)`。

注意 – `s_vmem` 與 `h_vmem`（虛擬記憶體）供可用於支援 `RLIMIT_VMEM` 的系統（請參閱您作業系統中的 `setrlimit(2)`）。

由 SGI/Cray 提供的 UNICOS 作業系統並不支援 `setrlimit(2)` 系統呼叫，而改用其本身擁有之資源限制設定系統呼叫。下述意義僅適用於 UNICOS 系統：

- `s_cpu` – 此為以秒為單位的單一作業 CPU 時間限制。
- `s_core` – 此為單一作業最大核心檔案大小（以位元組為單位）。
- `s_data` – 此為單一作業最大記憶體限制（以位元組為單位）。
- `s_vmem` – 與 `s_data` 相同（如果兩者皆有設定最小量）。
- `h_cpu` – 此為以秒為單位的單一作業 CPU 時間限制。
- `h_data` – 此為單一作業最大記憶體限制（以位元組為單位）。
- `h_vmem` – 與 `h_data` 相同（如果兩者皆有設定最小量）。
- `h_fsize` – 此為作業可建立之總磁碟區塊數量。

同時參閱

`sge_intro(1)`、`csch(1)`、`qconf(1)`、`qmon(1)`、`qrestart(1)`、`qstat(1)`、`qsub(1)`、`sh(1)`、`nice(2)`、`setrlimit(2)`、`access_list(5)`、`calendar_conf(5)`、`sge_conf(5)`、`complex(5)`、`host_conf(5)`、`sched_conf(5)`、`sge_execd(8)`、`sge_qmaster(8)`、`sge_shepherd(8)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

sched_conf(5)

名稱

sched_conf — Sun Grid Engine 預設排程器配置檔案。

說明

sched_conf 會定義由 sge_schedd(8) 提供的 Sun Grid Engine 系統預設排程器之配置檔案格式。為了修改配置，請使用圖形使用者介面 qmon(1) 或 qconf(1) 指令的 -msconf 選項。Sun Grid Engine 分配套裝軟體會提供一預設配置。

格式

如果下述參數出現在 sched_conf 中，就會由 Sun Grid Engine 排程器辨識：

- *演算法* — 此參數允許選擇替代排程演算法。目前為止，default 是唯一允許之設定。
- *載入公式* — 此為一簡單的代號表達方式，用以自每個主機上的所有或部分載入參數（由 sge_execd(8) 回報），與每個主機保留之部分或所有可耗損資源中（請參閱 complex(5)），取得單一加權載入數值。載入公式表達語法為加權載入數值的總和，即：

```
load_val1[*w1][{+|-}load_val2[*w2][{+|-}...]]
```

注意 — 載入公式中不可留任何空間。

載入數值與可耗損資源 (load_val1, ...) 會由聚合體（請參閱 complex(5)）中定義的名稱指定。

注意 — 也許可以使用管理員定義的載入數值（請參閱 sge_conf(5) 中的載入感應器參數得知詳細資訊）與 Sun Grid Engine 預設載入參數。

加權因素 (w1, ...) 為正整數。在評量每個主機的表達方式後，評量結果就會指定至主機，並用以分類主機（對應加權值）。之後，被分類的主機清單就會用來分類佇列。

預設載入公式為 load_avg。

- **作業載入調整** – 位於執行在系統上的 Sun Grid Engine 作業載入數值會隨著時間不同而改變，而且通常 – 以 CPU 載入為例 – 都需要一些時間等待作業系統回報合適的數量。因此，如果最新執行一作業，回報的載入數值就無法提供此作業已於主機上的完整數值。此回報值會隨著時間適應真正的數值，但在一段時間內若回報值若過低，就會導致主機過度耗損。Sun Grid Engine 允許管理員指定用 Sun Grid Engine 排程器之**作業載入調整**要素來補償這個問題。

作業載入調整 會被指定為逗點分隔之仲裁載入參數清單，或可耗損資源與（以等號分開之）相關載入修正值。不論 `sge_schedd(8)` 何時將作業分派至主機上，載入參數與可耗損值皆會設定由**作業載入調整**清單中提供的數值來增加主機。這些修正值會隨著時間而線性 (linearly) 延遲，直到一開始的**載入調整延遲時間**修正值為 0 之後。如果**作業載入調整**清單被分配至特殊名稱 NONE，就不會執行任何載入修正。

調整的載入值與可耗損值可用**載入公式**來計算主機的結合與加權載入值（請參閱上述說明），並以佇列配置（請參閱 `queue_conf(5)`）中定義的載入臨界值清單來比較載入值與可耗損值。如果您的**載入公式**僅只由 CPU 載入平均參數**載入平均值**組成，而您的作業計算又十分密集，您也許會想要設定**作業載入調整**清單至**載入平均值**=100，即代表每個分派至主機的新作業都將需要 100% 的 CPU 容量，所以機器的載入值也必須立即提升 100。

- **載入調整延遲時間** – 位於上述「**作業載入調整**」清單中的載入修正會隨時間，由啓動作業當下呈線性延遲，且對應的載入或可耗損參數也會因完整修正值而增加，直到「**載入調整延遲時間**」結束為止，即修正值為 0。「**載入調整延遲時間**」的數值是否適當與使用的載入或可耗損參數，和特定作業系統有很密切的關係。因此，這些數值僅可經由現場 (on-site) 確認或實驗 (experimentally) 確認。藉由預設值**載入平均值**載入參數，七分鐘的「**載入調整延遲時間**」已經證實可產生最佳的結果。
- **最大u作業** – 此為使用者可同時於 Sun Grid Engine 叢集上執行之作業最大量。如果設定值為 0（預設值），使用者也許會執行一作業仲裁數量 (arbitrary number)。
- **排程間隔期間** – 此為 `sge_schedd(8)` 最初記錄至 `sge_qmaster(8)` 的時間。**排程間隔期間**可用以設定間隔期間，`sge_qmaster(8)` 會傳遞更新之排程事件至 `sge_schedd(8)` 中。排程事件意指 `sge_qmaster(8)` 狀態的變更，並有可能啓動或影響排程器的下一步驟（例如：一作業結束後，則分配至此作業的資源又可再度被利用）。

在 Sun Grid Engine 預設排程器中，回報排程事件到達後便會啓動排程器作業，或者排程器會等待事件回報。

排程間隔期間為一時間值（請參閱 `queue_conf(5)` 得知時間值語法的定義）。

- **使用者分類** – Sun Grid Engine 系統通常會對應「**先來先處理 (first-come-first-served)**」的策略來排程使用者作業。如果一使用者於短時間內提交大量作業，就可能造成不公平的情況發生，因為之後提交作業的使用者就會被迫停擺，直到第一位使用者的大部份作業結束為止。因此，Sun Grid Engine 允許變更此策略此所謂的**平等共享分類 (equal share sort)**：只要一使用者有作業正在執行，他的其餘作業便會被歸類至擱置作業清單的尾端。如此一來，其他使用者的第一項作業便擁有相對的機會可尋找佇列。

注意 – 平等共享分類僅適用於相同作業優先權類型（請參閱 `qsub(1)` 中的 `mp` 選項與 `galter(1)` 指令以得知更多資訊）。

使用者分類的預設值為 FALSE。

- **佇列分類法** – 此參數可確認在考量多種標準後，分類出的佇列清單順序。兩種設定值為有效值：`seq_no` 與 `load`。在上述兩種設定值中，Sun Grid Engine 軟體想最大化軟體要求的數量（請參閱 `qsub(1) -s` 選項），以滿足一特定作業佇列的要求為首要標準。
 - 如果 **佇列分類法** 參數設定為 `seq_no`，Sun Grid Engine 軟體將會使用配置於最新佇列配置中（請參閱 `queue_conf(5)`）的 **序列編號** 參數做為首要標準來分類佇列清單。**載入公式**（見上述說明）僅當兩個佇列擁有相同順序號碼時才有意義。
 - 如果 **佇列分類法** 設定為 `load`，記錄 **載入公式** 的載入值則為最大化作業之軟要求 (`soft request`) 後的首要標準。在此設定中，僅當兩主機有相同載入值時才會使用順序號碼。

如果您想要定義一填滿佇列的固定順序（例如：最便宜的資源優先），順序號碼分類就是最有用的方法。

此參數的預設值為 `load`。

- **sgc 企業版排程間隔時間** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。

此為 Sun Grid Engine，企業版動態排程器中作業優先權調整之間的時期。有效值會被指定為 **時間** 種類，如同 `queue_conf(5)` 中指定的一樣。

- **半衰期** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。

在共享策略底下執行時，Sun Grid Engine，企業版會「老化」（即減少）可達到共享應得資源（如共享樹定義之）的滑動視窗的使用。**半衰期** 可定義時間間隔期間，而累積的使用量將會衰退至其原有值的一半。有效值會被指定為 **時間** 種類，如同 `queue_conf(5)` 中指定的一樣。

- **使用加權清單** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。

Sun Grid Engine，企業版會計算 CPU 時間、記憶體、與 IO 資源的耗損以確認一作業於系統中的使用量。相乘此三個輸入參數之加權個別值後再相加個別值，便可計算出單一使用值。加權定義於 **使用加權清單** 中。清單格式如下：

```
cpu=wcpu、mem=wmem、io=wio
```

其中 `wcpu`、`wmem`、與 `wio` 為可配置之加權。加權為實際數值。所有樹的加權總和應為 1。

- **補償因素** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。

確認 Sun Grid Engine，企業版應多快補償過去之使用量，要補償比共享樹中定義之共享應得資源更多或更少。建議值為介於 2 和 10 之間，10 代表最快補償。

- **加權使用者** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。

在功能策略中此使用者共享之相對重要性。數值為種類 **真實**。

- **加權專案** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
在功能策略中此專案共享之相對重要性。數值為種類真實。
- **加權作業等級** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
在功能策略中此使用者等級（即佇列）共享之相對重要性。數值為種類真實。
- **加權部門** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
在功能策略中此部門共享之相對重要性。數值為種類真實。
- **加權作業** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
在功能策略中此作業共享之相對重要性。數值為種類真實。
- **加權票卷功能** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
Sun Grid Engine，企業版可分配之最大功能票卷量。確認功能策略之相對重要性。
- **加權票卷共享** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
Sun Grid Engine，企業版可分配之最大共享票卷量。確認共享樹策略之相對重要性。
- **加權截止日期** – 此參數僅適用於 Sun Grid Engine，企業版系統，Sun Grid Engine 並不支援此參數。
Sun Grid Engine，企業版可分配之最大截止日期票卷量。確認截止日期策略之相對重要性。
- **排程作業資訊** – 預設排程器會持續追蹤作業無法於最後排程器作業期間排程的原因。此參數會啟動或關閉這項觀察功能。true 值會啟動監控功能；false 則關閉此功能。
也有可能僅啟動特定作業之觀察功能，只要在逗點分隔之作業 id 清單後將參數設為作業清單即可。
使用者可利用 `qstat -j` 指令獲得收集到的資訊。

檔案

- `sge_schedd` 配置 – `<sge_root>/<儲存格>/common/sched_configuration`

同時參閱

sge_intro(1)、qalter(1)、qconf(1)、qstat(1)、qsub(1)、complex(5)、queue_conf(5)、sge_execd(8)、sge_qmaster(8)、sge_schedd(8)、《*Sun Grid Engine 5.3 管理和使用者指南*》、與《*Sun Grid Engine 企業版 5.3 管理和使用者指南*》

版權

請參閱 sge_intro(1) 以得知權利與權限的完整聲明。

share_tree(5)

名稱

共享樹 — Sun Grid Engine，企業版共享樹檔案格式

說明

共享樹物件僅適用於 Sun Grid Engine，企業版系統。Sun Grid Engine 沒有共享樹物件。

共享樹定義使用者 / 專案和仲裁群組階層的長期應得資源。

可透過 qconf(1) -sstree 選項顯示目前的共享樹。輸出位在 share_tree 格式說明之後。透過 -astree 與 -mstree 選項至 qconf(1) 可以建立新的共享樹，或者修改現有共享樹。而透過 -astnode、-dstnode、-mstnode 與 -sstnode 選項至 qconf(1) 則可建立、修改、刪除、或顯示個別共享樹節點。

格式

共享樹檔案格式定義如下：

- 新的節點以屬性 id 為開頭，然後接著等號和節點的數字辨識碼。之後便會顯示此節點的其餘屬性，並止於另一個 id 關鍵字。
- 子節點 (childnodes) 屬性包含此節點之逗點分隔的子節點清單。

- 如果此節點為共享樹之子節點，此參數 *名稱* 則代表節點的仲裁名稱，或者對應使用者（請參閱 `user(5)`）或專案（請參閱 `project(5)`）。共享樹 `root` 節點的名稱依慣例命名為 "Root"。
- 參數 *共享* 會為同一母節點中的節點定義節點共享。
- 一名為 *預設* 的使用者子節點可被定義為共享樹中 `project(5)` 節點的子節點。預設節點會為執行於專案中，或未於專案中定義使用者節點之使用者來定義共享數量。預設使用者節點可輕鬆為所有使用者指定單一節點，並應得到專案資源的平等共享。*預設* 節點可能會自行指定，或由專案底下相同等級之 `user(5)` 節點指定。所有使用者，不論是有明確指定為使用者節點，或者對映至 *預設* 使用者節點，皆必須定義對應 `user(5)` 物件以得到共享。請勿配置名稱為 `default` 之 `user(5)` 物件。

同時參閱

`sge_intro(1)`、`qconf(1)`、`project(5)`、`user(5)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

user(5)

名稱

使用者 — Sun Grid Engine，企業版使用者項目檔案格式

說明

此使用者物件僅適用於 Sun Grid Engine，企業版系統中，Sun Grid Engine 系統沒有使用者物件。

每個使用者會使用一個 Sun Grid Engine，企業版的使用者項目儲存票卷與使用資訊。如果 Sun Grid Engine，企業版是執行於使用者共享樹策略底下，就必須維護所有 Sun Grid Engine，企業版系統使用者的使用者項目。

使用 `qconf(1) -suserl` 選項即可顯示最新配置使用者項目的清單。透過 `-suser` 開關來顯示每個列出的使用者項目內容，輸出位在*使用者*格式說明之後。透過 `-auser`、`-muser` 與 `-duser` 選項至 `qconf(1)` 就可建立新的使用者項目，並修改現有使用者項目。

格式

一個使用者項目包含四個參數。

- *名稱* — 此為使用者名稱。
- *置換票卷* — 此為最新分配至使用者之置換票卷量。
- *功能共享* — 此為目前使用者功能共享。
- *預設專案* — 此為使用者預設專案。

同時參閱

`sge_intro(1)`、`qconf(1)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

usermapping(5)

名稱

`usermapping` — 使用者對映項目檔案格式

說明

`usermapping` 項目可用以定義叢集使用者之別名。使用者名稱 — 即為排程系統 — 命名為*叢集使用者*。如果叢集名稱不符合執行主機上的使用者帳號，`usermapping` 功能便可解決此問題。

usermapping 項目檔案的每一行皆會列出使用者名稱與其擁有帳號之主機。

您可使用 `qconf(1) -sumapl` 選項來顯示最新配置 usermapping 項目的清單。利用 `-sumap` 開關來顯示每個列出的 usermapping 項目內容，與 usermapping 格式說明。使用 `-aumap`、`-mumap`、與 `-dumap` 選項至 `qconf(1)` 來建立或修改使用者項目。

格式

usermapping 項目包含至少三種參數。

- **關鍵字叢集使用者** – 關鍵字 *叢集使用者* 可定義叢集使用者名稱。在關鍵字 *叢集使用者* 後的其餘文字行會被視為叢集使用者值。
- **名稱** – 此為執行主機上的使用者名稱。
- **主機名稱** – 此為指定於 *名稱* 中的使用者擁有帳號之主機名稱。如果主機名稱的第一個字元有 @ 記號，則此名即用來參考 `hostgroup(5)`。

範例

下例為一典型之叢集使用者對映 (usermapping) usermapping 項目。

```
cluster_user office
alice wonderland, realworld, @office2
```

此項目會對映使用者，`office`，用以於叢集中定義系統至 `wonderland` 與 `realworld` 主機上之使用者，`alice`。@ 記號是用以參考主機群組。因此使用者，`office`，會對映至指定於主機群組，`office2`，中的所有主機上之 `alice`。請參閱 `hostgroup(5)` 以得知更多相關資訊。

同時參閱

`qconf(1)`、`hostgroup(5)`

版權

請參閱 `sgc_intro(1)` 以得知權利與權限的完整聲明。

sge_commd(8)

名稱

sge_commd — Sun Grid Engine 通訊代理程式

摘要

```
sge_commd [ -S ] [ -a 別名檔案 ] [ -dhr ] [ -help ]  
[ -ll 記錄等級 ] [ -ml 檔案名稱 ] [ -nd ] [ -p 連接埠 ]  
[ -s 服務 ]
```

說明

利用通訊常駐程式 `sge_commd` 就可顯示 Sun Grid Engine 叢集中所有網路通訊。諸如 `qsub(1)` 或 `qstat(1)` 的客戶端程式與 `sge_qmaster(8)` 或 `sge_execd(8)` 的 Sun Grid Engine 常駐程式會使用 `sge_commd` 提供的服務來傳送訊息至其他 Sun Grid Engine 組件，或接收來自其他 Sun Grid Engine 組件的訊息。

`sge_commd` 會處理同步或非同步通訊之仲裁數量。通常會自動啟動主機上的 `sge_commd`，即啟動 `sge_qmaster(8)`、`sge_execd(8)` 或 / 與 `sge_schedd(8)`。但是也有可能使用 `sge_commd` 來連接多重主機，或使用提交 Sun Grid Engine 主機或管理 Sun Grid Engine 主機上的 `sge_commd`（而非執行其餘 Sun Grid Engine 常駐程式）做為由主機啟動之 Sun Grid Engine 客戶端程式的通訊代理程式。

選項

表 29 列出並說明 `sge_commd` 相關之選項。

表 29 `sge_commd` 選項

選項	說明
<code>-S</code>	強迫使用安全埠以做為 <code>sge_commd</code> 之間，與其他 Sun Grid Engine 組件和 <code>sge_commds</code> 之間的通訊。這需要啟動所有 Sun Grid Engine 常駐程式，與配置 <code>set-uid root</code> 之 <code>root</code> 權限和客戶端程式。這可確保限制非 <code>root</code> 帳號未經授權之通訊。
<code>-a 別名檔案</code>	包含 <code>sge_commd</code> 用以解決叢集中所有主機之 Sun Grid Engine 獨特主機名稱的 Sun Grid Engine 主機別名檔案。置換位於 <code><sge 根目錄>/<儲存格>/common/host_aliases</code> 下之預設主機別名檔案的使用。解決問題之 <code>sge_commd</code> 的主機名稱也可被 Sun Grid Engine 組件使用。請參閱 <code>sge_h_aliases(5)</code> 中的說明以得知檔案格式與其使用之意義。
<code>-dhr</code>	解決 C-library 功能的主機名稱（例如 <code>gethostent(3)</code> 、 <code>gethostbyname(3)</code> 、與 <code>gethostbyaddr(3)</code> ）會於 OS 架構上執行某種快取。由 DNS (Domain Name Service) 與 NIS 等服務分配之網路主機名稱資料庫會於幾分鐘內更新完畢。這僅會影響反覆解析主機名稱的應用程式（例如 <code>sge_commd</code> ）。在啟動程式初，會先存取最新的資訊，這樣才不會影響諸如 <code>telnet(1)</code> 或 <code>nslookup(1)</code> 的指令。 但是對於 <code>sge_commd</code> 來說，隨時解析主機名稱則大可不必（反正返回的資訊也許是過期的），而且如果網路已超載，和 / 或 NIS 或 DNS 伺服器非常忙碌時，解析的作業可能會非常耗損。因此， <code>sge_commd</code> 僅可一次一次解析主機名稱資訊。 但是如果主機名稱解析仍然因網路載入造成問題，例如可使用 <code>ndhr</code> 開關關機。當主機名稱解析功能關閉時，管理員必需知道要在主機名稱資料庫有重大變更當下立即啟動 <code>sge_commd</code> 。
<code>-help</code>	列出所有選項清單。
<code>-ll 記錄等級</code>	設定錯誤追蹤登入等級。錯誤追蹤會寫入 <code>/tmp/commd/err.<pid></code> 檔案中。但是必須要有 <code>/tmp/commd</code> 目錄，否則就會捨棄追蹤的輸出資訊。目前僅有 255 是唯一的有效登入等級。
<code>-ml fname</code>	啟動訊息登入至檔案。
<code>-nd</code>	無法常駐。如果以 <code>-nd</code> 開頭， <code>sge_commd</code> 便會維持與控制終端機間的連結，而追蹤輸出資訊也可自啟動的 <code>sge_commd</code> 直接傳送至終端機中。追蹤資訊也可被存取，與使用 <code>-ll</code> 選項相同（請參閱上述說明）。
<code>-p 連接埠編號</code>	使用此 TCP 埠維持與其他 <code>commd</code> 的通訊。
<code>-s 服務名稱</code>	使用本服務名稱與相關 TCP 埠以維持與其他 <code>commds</code> 的通訊。

環境變數

表 30 列出與 `sge_cmmd` 相關的環境變數。

表 30 `sge_cmmd` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位址。
<code>COMMD_PORT</code>	如果已設定，請指定可聽取通訊要求之 <code>sge_cmmd</code> 上的 tcp 埠。大部分的安裝將使用服務對映項目，而非定義此埠。
<code>COMMD_HOST</code>	(不會影響 <code>sge_cmmd</code> 的運作方式，但會影響其他連繫 <code>sge_cmmd</code> 的 Sun Grid Engine 組件。) 如果已設定，請指定仲裁 Sun Grid Engine 客戶端程式或常駐程式的 Sun Grid Engine 通訊所使用之特定 <code>sge_cmmd</code> 上的主機。單一預設僅可使用本端主機。

限制

`sge_cmmd` 通常是藉由 `sge_qmaster(8)` 與 `sge_execd(8)` 而啟動的，所以會於 `root` 權限下執行。如果是由普通使用者啟動，當安全模式要求 `root` 權限時（請參閱上述說明），也許就不會使用 `-S` 開關。

同時參閱

`sge_intro(1)`、`sge_h_aliases(5)`、`sge_execd(8)`、`sge_qmaster(8)`、`commdctl(8)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

sge_execd(8)

名稱

sge_execd — Sun Grid Engine 作業執行代理程式

指令概要

sge_execd [-help] [-lj 記錄檔案] [-nostart-commd]

說明

sge_execd 會控制 Sun Grid Engine 佇列本端至機器，sge_execd 正在作業，並執行 / 控制由 sge_qmaster(8) 傳送之作業，以供於佇列上執行。

與 sge_execd 一同，sge_commd(8) 也會於相同機器上被自動叫出（如果原本就沒有出現的話）。

選項

表 31 列出與 sge_execd 相關之選項。

表 31 sge_execd 選項

選項	說明
-help	列出所有選項清單。
-lj 日誌檔案	啓動作業登入。由 sge_execd 執行之所有動作，即從接收作業直到將其傳送回 sge_qmaster(8)，都會被記錄至日誌檔案。此功能同時也適用於 sge_qmaster(8) 常駐程式。
-nostart-commd	請勿自動開啓 sge_commd(8) 與 sge_execd 並評估 COMMD_HOST 環境變數以尋找相對 sge_commd(8)。

載入感應器 (Load Sensors)

如果 `sge_execd` 的載入感應器是由特定叢集配置之全域或執行主機配置（請參閱 `sge_conf(5)`），則載入感應器的可執行路徑就會由一般 `sge_execd` 啟動，並傳送一個或多個執行主機（例如使用者目前登入）或完整叢集（例如：網路暫存檔案系統上的空白磁碟空間）上的載入數值。此載入感應器可以為一 `script` 或二進位可執行檔案。在任一情況中，STDIN 與 STDOUT 串流，和其控制流的處理皆必須符合下述規則：

載入感應器必須無限迴路寫入，從 STDIN 於一特定點等待輸入。如果由 STDIN 讀出此 `quit` 字串，就該退出載入感應器。只要由 STDIN 讀到 `end-of-line`，就可以再度啟動載入資料循環。接著，載入感應器須執行所需之操作以計算理想載入數值。在循環結束前，載入感應器會將結果寫入 `stdout` 中。格式如下。

- 載入數值回報起始於一行僅有 `start` 字樣的文字行。
- 個別之載入數值會被新行分開。
- 每個載入數值報告都由冒號 (`:`) 分成三部份，中間沒有空格。
- 載入數值資訊第一部分為載入回報之主機名稱或特定的名稱，`global`。
- 第二部份為載入數值的象徵名稱，如主機或全域 `complex` 清單中定義一般（請參閱 `complex(5)` 以得知詳細資訊）。如果回報的載入數值於主機或全域聚合體清單中並無任何記錄，就不會使用此回報的載入數值。
- 第三個部分為測量的載入數值。
- 載入數值回報會以 `end` 字樣做結束。

環境變數

表 32 列出並說明 `sge_execd` 相關之環境變數。

表 32 `sge_execd` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位址。
<code>SGE_CELL</code>	如果已設定，請指定預設 Sun Grid Engine 儲存格。若要提出 Sun Grid Engine 儲存格， <code>sge_execd</code> 會使用（依先後順序排列）： <ul style="list-style-type: none">• 指定於環境中的儲存格名稱如果已設定的話，變數 <code>SGE_CELL</code>。• 預設儲存格名稱，例如 <code>default</code>。

表 32 sge_execd 環境變數 (接上頁)

變數名稱	說明
SGE_DEBUG_LEVEL	如果已設定的話，請指定應寫入 <code>stderr</code> 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 <code>sge_commd(8)</code> 。大部分的安裝將使用服務對映項目，而非定義此埠。
COMMD_HOST	如果已設定，請指定 <code>sge_execd</code> 客戶端所在的 Sun Grid Engine 通訊所使用之特定 <code>sge_commd(8)</code> 上的主機。僅當 <code>-nostart-command</code> 選項指定於 <code>sge_execd</code> 指令行時才進行評估。單一預設僅可使用本端主機。

限制

`sge_execd` 通常是由 Sun Grid Engine 儲存區 (pool) 每個機器上的 `root` 啟動。如果是由普通使用者啟動，就必須於使用者讀取 / 寫入存取的地方使用多工緩衝處理目錄 (spool directory)。在此情況中，僅有由相同使用者提交的作業才會被系統視為正確。

檔案

- Sun Grid Engine 全域配置 — `<sge_root>/<儲存格>/common/configuration`
- Sun Grid Engine 主機特定配置 (host-specific configuration) — `<sge_root>/<儲存格>/common/local_conf/<主機>`
- 預設執行主機多工緩衝處理目錄 — `<sge_root>/<儲存格>/spool/<主機>`
- Sun Grid Engine 主要主機檔案 — `<sge_root>/<儲存格>/common/act_qmaster`

同時參閱

`sge_intro(1)`、`sge_conf(5)`、`complex(5)`、`sge_commd(8)`、`sge_qmaster(8)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

sge_qmaster(8)

名稱

sge_qmaster — Sun Grid Engine 主控制常駐程式

指令概要

```
sge_qmaster [ -help ] [ -lj 日誌檔案 ] [ -nohist ]  
[ -nostart-commd ] [ -s ]
```

```
sge_qmaster -show-license
```

說明

sge_qmaster 控制叢集中的整體 Sun Grid Engine 運作方式。爲了排程作業，sge_qmaster 會與 sge_schedd(8) 合作。在啓動 sge_qmaster 時，通常會從相同機器上自動叫出 sge_commd(8)（如果原本就沒有出現的話）。

選項

表 33 列出並說明 sge_qmaster 相關之選項。

表 33 sge_qmaster 選項

選項	說明
-help	列出所有選項清單。
-lj 日誌檔案	啓動作業記錄。由 sge_qmaster 所執行的所有動作，包括自提交作業到退出作業，都會被記錄至日誌檔案中。此功能同時也適用於 sge_execd(8) 常駐程式。

表 33 sge_qmaster 選項 (接上頁)

選項	說明
-nohist	在一般作業期間，sge_qmaster 會將佇列、聚合體、與主機配置的變更記錄轉儲至歷史資料庫中。此資料庫主要使用 qacct(1) 指令來允許 qacct(1) 指令行中的 qsub(1)，如 -l 資源要求。此開關可阻止將資料寫入資料庫中。
-nostart-commd	請勿自動啟動 sge_commd(8) 與 sge_qmaster。
-s	開啓 sge_qmaster 靜音模式 (silent mode)。通常 sge_qmaster 會於啟動時顯示版權資訊，稍候便可返回繼續。使用 -s 開關來安靜 (silently) 啟動 sge_qmaster。

環境變數

表 34 列出並說明 sge_qmaster 相關之環境變數。

表 34 sge_qmaster 環境變數

變數名稱	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位址。
SGE_CELL	如果要設定，請指定預設 Sun Grid Engine 儲存格。要提出 Sun Grid Engine 儲存格，sge_qmaster 使用 (依先後順序排列)： <ul style="list-style-type: none"> 指定於環境中的儲存格名稱如果已設定的話，variable SGE_CELL。 預設儲存格名稱，例如 default。
SGE_DEBUG_LEVEL	如果要設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果要設定，請指定預期將聽取通訊需求之 sge_commd(8)。大部分的安裝將使用服務對映項目，而非定義此埠。

限制

sge_qmaster 通常會從叢集主要或備份主要機器上的 root 來啟動 (請參閱 《Sun Grid Engine Installation and Administration Guide》以得知更多備用主要主機的資訊)。如果是由一般使用者啟動，就必須於主要讀取 / 寫入存取的地方使用多工緩衝處理目錄。在此情況中，僅有由相同使用者提交的作業才會被系統視為正確。

檔案

- Sun Grid Engine 全域配置 — `<sge_root>/<儲存格>/common/configuration`
- Sun Grid Engine 主機特定配置 — `<sge_root>/<儲存格>/common/local_conf/<主機>`
- 歷史資料庫 — `<sge_root>/<儲存格>/common/history`
- `sge_qmaster` 引數檔案 — `<sge_root>/<儲存格>/common/qmaster_args`
- 預設主要多工緩衝處理目錄 — `<sge_root>/<儲存格>/spool/<主機>`

同時參閱

`sge_intro(1)`、`sge_conf(5)`、`sge_commd(8)`、`sge_execd(8)`、`sge_schedd(8)`、`sge_shadowd(8)`、《*Sun Grid Engine 5.3 管理和使用者指南*》、與《*Sun Grid Engine Enterprise Edition 5.3 管理和使用者指南*》

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

sge_schedd(8)

名稱

`sge_schedd` — Sun Grid Engine 作業排程代理程式

指令概要

```
sge_schedd [ -help ]
```

說明

`sge_schedd` 可於 Sun Grid Engine 叢集中計算排程決定。做決定所需的資訊會透過一事件介面，從 `sge_qmaster(8)` 取得。在使用排程演算法後，`sge_schedd` 將透過 Sun Grid Engine GDI 再次與 `sge_qmaster(8)` 溝通排程決定。爲了啓動執行 `sge_schedd`，叢集狀態中的 `sge_qmaster(8)` 樣本會變更，並於定期間隔期間通知 `sge_schedd`。

與 `sge_schedd` 一同，在相同機器上會自動叫出 `sge_commd(8)`（如果原先就沒有出現的話）。

只要使用 `qconf(1)` 指令之 `-tsm` 選項，就會強制 `sge_schedd` 列出其下個排程作業至 `<sge_root>/<儲存格>/common/schedd_runlog` 的追蹤訊息。訊息會透露出作業與佇列並未被選入執行作業的原因。

選項

表 35 列出與 `sge_schedd` 相關之選項。

表 35 `sge_schedd` 選項

選項	說明
<code>-help</code>	列出排程器的版本編號。

環境變數

表 36 列出與 `sge_schedd` 相關的環境變數。

表 36 `sge_schedd` 環境變數

變數名稱	說明
<code>SGE_ROOT</code>	指定 Sun Grid Engine 標準配置檔案的位址。

表 36 sge_schedd 環境變數

變數名稱	說明
SGE_CELL	如果要設定，請指定預設 Sun Grid Engine 儲存格。要提出 Sun Grid Engine 儲存格，sge_schedd 使用（依先後順序排列）： <ul style="list-style-type: none">• 指定於環境中的儲存格名稱如果已設定的話，變數 SGE_CELL。• 預設儲存格名稱，例如 default。
SGE_DEBUG_LEVEL	如果已設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果已設定，請指定預期將聽取通訊需求之 sge_commd(8)。大部分的安裝將使用服務對映項目，而非定義此埠。

檔案

- sge_schedd 多工緩衝處理目錄 — `<sge_root>/<儲存格>/spool/qmaster/schedd`
- sge_schedd 追蹤資訊 — `<sge_root>/<儲存格>/common/sched_runlog`
- sge_schedd 配置 — `<sge_root>/<儲存格>/common/sched_configuration`
- 請參閱 sched_conf(5) 得知排程器配置檔案的詳細資訊。

同時參閱

sge_intro(1)、sge_gdi(3)、sched_conf(5)、sge_commd(8)、sge_qmaster(8)

版權

請參閱 sge_intro(1) 以得知權利與權限的完整聲明。

sge_shadowd(8)

名稱

sge_shadowd — Sun Grid Engine 備用主要常駐程式

指令概要

sge_shadowd

說明

sge_shadowd 為一「輕量級」作業，可於 Sun Grid Engine 叢集中所謂的備用主要主機上執行，用以偵測目前 Sun Grid Engine 主要常駐程式 sge_qmaster(8) 的故障，並於 sge_shadowd 執行之主機上啟動新的 sge_qmaster(8)。如果多種備用常駐程式正執行於叢集中，它們還會執行一通訊協定，以確保僅有一個常駐程式將會啟動新的主要常駐程式。

適用於備用主要主機的主機必須擁有共享 root 讀寫存取至目錄 `<sge_root>/<儲存格>/common` 與主要常駐程式多工緩衝處理目錄（預設 `<sge_root>/<儲存格>/spool/qmaster`）中。`<sge_root>/<儲存格>/common/shadow_masters` 檔案中必須包含備用主要主機。

限制

sge_shadowd 也可以僅由 root 啟動。

環境變數

表 37 列出並說明 sge_shadowd 相關之環境變數。

表 37 sge_shadowd 環境變數

變數名稱	說明
SGE_ROOT	指定 Sun Grid Engine 標準配置檔案的位址。
SGE_CELL	如果已設定，請指定預設 Sun Grid Engine 儲存格。要提出 Sun Grid Engine 儲存格，sge_shadowd 使用（依先後順序排列）： <ul style="list-style-type: none">指定於環境中的儲存格名稱如果已設定的話，變數 SGE_CELL。預設儲存格名稱，例如 default。

表 37 sge_shadowd 環境變數

變數名稱	說明
SGE_DEBUG_LEVEL	如果已設定的話，請指定應寫入 stderr 的除錯資訊。除此之外，也要定義除錯資訊的詳細等級。
COMMD_PORT	如果已設定，請指定 sge_commd(8) 預期將聽取通訊需求之 tcp 埠。大部分的安裝將使用服務對映項目，而非定義此埠。
COMMD_HOST	如果已設定，請指定 sge_shadowd 客戶端所在的 Sun Grid Engine 通訊所使用之特定 sge_commd(8) 上的主機。單一預設僅可使用本端主機。

檔案

- 預設配置目錄 — `<sge_root>/<儲存格>/common`
- 備用主要主機名稱檔案 — `<sge_root>/<儲存格>/common/shadow_masters`
- 預設主要常駐程式之多工緩衝處理目錄 — `<sge_root>/<儲存格>/spool/qmaster`

同時參閱

sge_intro(1)、sge_conf(5)、sge_commd(8)、sge_qmaster(8)、*Sun Grid Engine Installation and Administration Guide*、《*Sun Grid Engine 5.3 管理員和使用者指南*》

版權

請參閱 sge_intro(1) 以得知權利與權限的完整聲明。

sge_shepherd(8)

名稱

sge_shepherd — Sun Grid Engine 單一作業控制代理程式

指令概要

sge_shepherd

說明

sge_shepherd 提供單一 Sun Grid Engine 作業的母過程功能。UNIX 系統上的母功能 (parent functionality) 是必要的，它可以在作業結束後接收到資源的使用資訊 (請參閱 `getrusage(2)`)。除此之外，sge_shepherd 會轉寄訊號至作業，如暫停、啟動、結束訊號，與 Sun Grid Engine 檢查點訊號 (請參閱 `sge_ckpt(1)` 得知詳細資訊)。

sge_shepherd 會從 `sge_execd(8)` 接收到欲啓動作業之資訊。在作業執行期間，會啓動五個子作業。首先，如果 *prolog* 參數在叢集配置中啓動此功能的話，就會啓動 *prolog script* (請參閱 `sge_conf(5)`)。接著，如果此作業為一平行作業，就會啓動平行環境啓動 *script* (請參閱 `sge_pe(5)` 得知更多資訊)。接著，在關閉平行作業的平行環境後，就會啓動作業本身。最後，如果叢集配置中的 *epilog* 參數有要求的話，就會開啓 *epilog script*。*prolog*、*epilog scripts*、與平行環境啓動和關閉程序皆由 Sun Grid Engine 管理員提供，用於執行實際使用者作業之前與之後的站點特定行動。

一旦作業結束，也已處理 *epilog script* 後，sge_shepherd 就會獲得作業資源使用數據、將其置於多工緩衝處理目錄 `sge_execd(8)` 之作業特定子目錄中以透過 `sge_execd(8)` 回報、然後結束。

限制

sge_shepherd 不應手動呼叫，應由 `sge_execd(8)` 呼叫。

檔案

作業特定目錄 - < 執行多工緩衝處理 > / 作業目錄 / < 作業 id >

同時參閱

`sge_intro(1)`、`sge_conf(5)`、`sge_execd(8)`

版權

請參閱 `sge_intro(1)` 以得知權利與權限的完整聲明。

sgecommdctl(8)

名稱

sgecommdctl — Sun Grid Engine 通訊代理程式控制指令

摘要

```
sgecommdctl [ -d | -k | -t 等級 ]  
[ -gid commprocname ] [ -h[elp] ] [ -p port ] [ -S ]  
[ -unreg commprocname id ]
```

說明

sgecommdctl 可用以控制 sge_commd(8) 的運作模式，或者自執行中的 sge_commd(8) 獲得資訊。

選項

表 38 列出並說明 `sgecommdcntl` 相關之選項。

表 38 `sgecommdcntl` 選項

選項	說明
<code>-d</code>	轉儲執行中 <code>sge_commd(8)</code> 過程的內部架構至 <code>/tmp/commd/commd.dump</code> 。 <code>/tmp/commd</code> 目錄必須存在，且 <code>sge_commd(8)</code> 也必須有寫入存取至此目錄才行，否則就忽略此要求。 此選項主要用於除錯，並不會影響提及的 <code>sge_commd(8)</code> 功能。
<code>-k</code>	刪除提及的 <code>sge_commd(8)</code> 。在下達刪除命令當下，擱置中的通訊會立即被刪除，但是 <code>sge_commd(8)</code> 的關閉將不會讓其餘作業與中斷作業在不一致的狀態下繼續連結。
<code>-t 等級</code>	<code>sgecommdcntl</code> 建立與 <code>sge_commd(8)</code> 的連結，並對應等級指定之追蹤等級，持續顯示追蹤輸出資訊。如果 <code>sge_commd(8)</code> 是以 <code>-ll</code> 選項呼叫，則輸出資訊便是由顯示追蹤輸出的子集組成。 目前僅支援 255 的追蹤等級。
<code>-gid commprocname</code>	取得通訊作業的 <code>commprocname</code> 識別號碼。Sun Grid Engine 組件會記錄至 <code>sge_commd(8)</code> ，以維繫與 Sun Grid Engine 作業之間的通訊。這些組件是由 <code>sge_commd(8)</code> 與一獨特之識別方式（由名稱與一識別號碼組成）記錄下來。識別名稱與 Sun Grid Engine 組件的名稱是相同的（例如： <code>sge_qmaster</code> 與 <code>sge_qmaster(8)</code> ）。而識別號碼可以使用 <code>-gid</code> 選項而取得。
<code>-help</code>	列出所有選項清單。

表 38 sgecommdcntl 選項 (接上頁)

選項	說明
-p <i>commdport</i>	用以討論 <i>sge_commd</i> (8) 的連接埠編號。
-S	強迫使用安全埠以做為 <i>sge_commds</i> 之間，與其他 Sun Grid Engine 組件和 <i>sge_commds</i> 之間的通訊。這需要啓動所有 Sun Grid Engine 常駐程式，與配置 <i>set-uid root</i> 之 <i>root</i> 權限和客戶端程式。這可確保限制非 <i>root</i> 帳號未經授權之通訊。
-unreg <i>commprocname id</i>	取消 Sun Grid Engine 組件 <i>commprocname</i> 以此 <i>Id id</i> 記錄至 <i>sge_commd</i> (8) 中的記錄。(請參閱上述的 <i>-gid</i> 以得知 <i>commprocname</i> 與 <i>id</i> 的說明)。 從 <i>sge_commd</i> (8) 移除 Sun Grid Engine 組件的記錄是必要的 - 如果諸如 <i>sge_qmaster</i> (8)、 <i>sge_execd</i> (8)、或 <i>sge_schedd</i> (8) 的常駐程式因不可抗拒之因素而中斷(例如因透過 <i>kill</i> (1) 傳送 <i>SIGKILL</i> 訊號而中斷)，而且 <i>sge_commd</i> (8) 拒絕重新啓動組件，因遭遇下述訊息 <i>error enrolling to commd: COMMPROC ALREADY REGISTERED</i> 。 <i>sge_commd</i> (8) 的記錄功能可用以預防執行於相同主機上的容錯 Sun Grid Engine 常駐程式。如果 Sun Grid Engine 組件已中斷，但卻無法由 <i>sge_commd</i> (8) 移除記錄，此記錄就會被保留下來，直到數分鐘過後，或者直到通訊作業記錄已被手動移除。

環境變數

表 39 列出並說明 *sgecommdcntl* 相關之環境變數。

表 39 sgecommdcntl 環境變數

變數名稱	說明
COMMD_PORT	如果已設定，請指定 <i>sgecommdcntl</i> 預期將聽取通訊需求之 <i>tcp</i> 埠。大部分的安裝將使用服務對映項目，而非定義此埠。
COMMD_HOST	如果已設定，請指定 <i>sgecommdcntl</i> 所在的 Sun Grid Engine 通訊所使用之特定 <i>sge_commd</i> 上的主機。單一預設僅可使用本端主機。

同時參閱

`sgc_intro(1)`、`sgc_commd(8)`、`sgc_execd(8)`、`sgc_qmaster(8)`、`sgc_schedd(8)`

版權

請參閱 `sgc_intro(1)` 以得知權利與權限的完整聲明。

