



# Sun™ ONE Grid Engine / Sun™ ONE Grid Engine, Enterprise Edition リファレンスマニュアル

---

Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

Part No. 816-7476-10  
2002 年 9 月, Revision A

コメントの宛先: [docfeedback@sun.com](mailto:docfeedback@sun.com)

Copyright 2002 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします)は、本書に記述されている製品に採用されている技術に関する知的所有権を有しています。これら知的所有権には、<http://www.sun.com/patents>に掲載されているひとつまたは複数の米国特許、および米国ならびにその他の国におけるひとつまたは複数の特許または出願中の特許が含まれています。

本書およびそれに付属する製品は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社の書面による事前の許可なく、本製品および本書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品のフォント技術を含む第三者のソフトウェアは、著作権法により保護されており、提供者からライセンスを受けているものです。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company Limited が独占的にライセンスしている米国ならびに他の国における登録商標です。

本製品は、株式会社モリサワからライセンス供与されたリュウミン L-KL (Ryumin-Light) および中ゴシック BBB (GothicBBB-Medium) のフォント・データを含んでいます。

本製品に含まれる HG 明朝 L と HG ゴシック B は、株式会社リコーがリョービマジクス株式会社からライセンス供与されたタイプフェースマスタをもとに作成されたものです。平成明朝体 W3 は、株式会社リコーが財団法人日本規格協会 文字フォント開発・普及センターからライセンス供与されたタイプフェースマスタをもとに作成されたものです。また、HG 明朝 L と HG ゴシック B の補助漢字部分は、平成明朝体 W3 の補助漢字を使用しています。なお、フォントとして無断複製することは禁止されています。

Sun, Sun Microsystems, AnswerBook2, docs.sun.com は、米国およびその他の国における米国 Sun Microsystems 社の商標もしくは登録商標です。サンのロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャーに基づくものです。Energy Star のロゴは、米国 EPA の登録商標です。

OPENLOOK、OpenBoot、JLE は、サン・マイクロシステムズ株式会社の登録商標です。

ATOK は、株式会社ジャストシステムの登録商標です。ATOK8 は、株式会社ジャストシステムの著作物であり、ATOK8 にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。ATOK Server/ATOK12 は、株式会社ジャストシステムの著作物であり、ATOK Server/ATOK12 にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPENLOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザーインターフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本書には、技術的な誤りまたは誤植の可能性があります。また、本書に記載された情報には、定期的に変更が行われ、かかる変更は本書の最新版に反映されます。さらに、米国サンまたは日本サンは、本書に記載された製品またはプログラムを、予告なく改良または変更することがあります。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典: Sun Grid Engine 5.3 and Sun Grid Engine, Enterprise Edition 5.3 Reference Manual  
Part No: 816-4767-11  
Revision A



Adobe PostScript

# 目次

---

Sun Grid Engine 5.3 および Sun Grid Engine, Enterprise Edition 5.3 のマニュアル  
ページ集 1

概要 1

sge\_intro(1) 1

sge\_ckpt(1) 5

qacct(1) 7

qconf(1) 12

qdel(1) 37

qhold(1) 40

qhost(1) 43

qmake(1) 48

qmod(1) 52

qmon(1) 55

qrls(1) 65

qselect(1) 68

qstat(1) 71

qtcsh(1) 80

submit(1) 84

access\_list(5) 115

accounting(5) 116

calendar\_conf(5) 118  
checkpoint(5) 121  
sge\_request(5) 124  
sge\_aliases(5) 126  
sge\_conf(5) 128  
sge\_h\_aliases(5) 147  
sge\_pe(5) 148  
complex(5) 152  
host\_conf(5) 157  
hostgroup(5) 161  
project(5) 163  
qtask(5) 164  
queue\_conf(5) 166  
sched\_conf(5) 176  
share\_tree(5) 181  
user(5) 183  
usermapping(5) 184  
sge\_commd(8) 185  
sge\_execd(8) 189  
sge\_qmaster(8) 192  
sge\_schedd(8) 195  
sge\_shadowd(8) 197  
sge\_sheperd(8) 199  
sgecommdctl(8) 201

# 表目次

---

表 1	ユーザーインタフェースプログラムとその説明	2
表 2	qacct のオプション	8
表 3	qacct 関係の環境変数	11
表 4	qconf のオプション	13
表 5	qconf 関係の環境変数	36
表 6	qdel のオプション	38
表 7	qdel 関係の環境変数	39
表 8	qhold のオプション	41
表 9	qhold 関係の環境変数	42
表 10	qhost のオプション	43
表 11	qhost 関係の環境変数	47
表 12	qmake 関係の環境変数	50
表 13	qmod のオプション	53
表 14	qmod 関係の環境変数	54
表 15	qmon のオプション	57
表 16	qmon 関係の環境変数	64
表 17	qrls のオプション	66
表 18	qrls 関係の環境変数	67
表 19	qselect のオプション	68
表 20	qselect 関係の環境変数	70

表 21	qstat のオプション	72
表 22	qstat 関係の環境変数	79
表 23	qtcssh のオプション	82
表 24	qtsch 関係の環境変数	83
表 25	実行依頼コマンドのオプション	88
表 26	submit 関係の環境変数	108
表 27	その他の qsub、qsh、qlogin 関係の環境変数	109
表 28	submit コマンドによってジョブ環境に組み込まれる変数	109
表 29	sgc_commd のオプション	187
表 30	sgc_commd 関係の環境変数	188
表 31	sgc_execd のオプション	190
表 32	sgc_execd 関係の環境変数	191
表 33	sgc_qmaster のオプション	193
表 34	sgc_qmaster 関係の環境変数	194
表 35	sgc_schedd のオプション	196
表 36	sgc_schedd 関係の環境変数	196
表 37	sgc_shadowd 関係の環境変数	198
表 38	sgecommdcntl のオプション	202
表 39	sgecommdcntl 関係の環境変数	204

# はじめに

---

このマニュアルは、2つの製品に付属するマニュアル (man) ページを編集して1つにまとめたものです。このマニュアルの大部分が両方の製品に共通の情報です。一方の製品にだけ固有の情報については、そのことを明記しています。

---

## 内容の紹介

このマニュアルには、このはじめにと目次の他に、1つの章があるだけです。

- Sun Grid Engine 5.3 および Sun Grid Engine, Enterprise Edition 5.3 のマニュアル ページ集

---

## UNIX コマンド

このマニュアルには、UNIX® の基本的なコマンド、およびシステムの停止、システムの起動、デバイスの構成などの基本的な手順の説明は記載されていません。

基本的なコマンドや手順についての説明は、次のマニュアルを参照してください。

- 『Sun 周辺機器 使用の手引き』
- Solaris™ オペレーティング環境についてのオンライン AnswerBook2™
- 本システムに付属している他のソフトウェアマニュアル

---

## 書体と記号について

書体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例。	.login ファイルを編集します。 ls -a を実行します。 % You have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して表します。	マシン名% su Password:
AaBbCc123 またはゴシック	コマンド行の可変部分。実際の名前や値と置き換えてください。	rm <i>filename</i> と入力します。 rm <b>ファイル名</b> と入力します。
『』	参照する書名を示します。	『Solaris ユーザーマニュアル』
「」	参照する章、節、または、強調する語を示します。	第 6 章「データの管理」を参照。 この操作ができるのは「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅をこえる場合に、継続を示します。	% grep `^#define \ XV_VERSION_STRING`

---



---

## シェルプロンプトについて

シェル	プロンプト
UNIX の C シェル	マシン名%
UNIX の Bourne シェルと Korn シェル	\$
スーパーユーザー (シェルの種類を問わない)	#

---

## 関連マニュアル

用途	タイトル	Part No.
管理および使用方法	『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』	816-7470-10
管理および使用方法	『Sun Grid Engine 5.3 管理およびユーザーマニュアル』	816-7463-10

---

## Sun のオンラインマニュアル

サンの各種システムマニュアルは下記 URL より参照できます。

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

Solaris およびその他のマニュアルは下記 URL より参照できます。

<http://docs.sun.com>

---

## コメントをお寄せください

弊社では、マニュアルの改善に努力しており、お客様からのコメントおよびご忠告をお受けしております。コメントは下記宛に電子メールでお送りください。

[docfeedback@sun.com](mailto:docfeedback@sun.com)

電子メールの表題にはマニュアルの Part No. (8xx-xxxx-xx) を記載してください。

なお、現在日本語によるコメントには対応できませんので、英語で記述してください。

# Sun Grid Engine 5.3 および Sun Grid Engine, Enterprise Edition 5.3 のマニュアルページ集

---

## 概要

このマニュアルには、Sun™ Grid Engine ディストリビューションに含まれているものと同じマニュアルページが含まれています。コマンドの説明中にあるように、いくつかのコマンドは、Sun Grid Engine, Enterprise Edition 5.3 にのみ該当します。

---

## sg\_e\_intro(1)

## 名前

Sun Grid Engine の概要 - リモートマシンで UNIX ジョブを実行するための機能

## 説明

Sun Grid Engine 5.3 および Sun Grid Engine, Enterprise Edition 5.3 ソフトウェア (特に明記していない場合、Sun Grid Engine の表記はこの両方を指す) は、協力関係にあるプールされたワークステーション上で UNIX バッチジョブ (シェルスクリプト) を実行することを可能にする製品です。ジョブはキューに入れられ、ワークステー

ションが休止しているか、負荷が軽い場合にのみワークステーション上で遠隔実行されます。作業負荷は、各マシンの負荷状況やジョブの資源要求に応じてクラスタを構成するワークステーション間で分散されます。

ユーザーレベルのチェックポイントプログラムがサポートされており、透過的なチェックポイント機構を利用することができます (sge\_ckpt(1) を参照)。チェックポイントジョブは、負荷に応じてユーザーの介入なしでワークステーション間を移動します。バッチジョブのほか、対話形式のジョブや並列ジョブを Sun Grid Engine に実行依頼することもできます。

## ユーザーインタフェース

Sun Grid Engine のユーザーインタフェースは、表 1 でまとめているようなプログラムで構成されています。

表 1 ユーザーインタフェースプログラムとその説明

プログラム名	説明
qacct(1)	クラスタログファイルから任意のアカウント情報抽出します。
qalter(1)	すでに実行依頼されているジョブの特性を変更します。
qconf(1)	キューの構成・変更・削除・照会およびクラスタ構成のためのユーザーインタフェースを提供します。
qdel(1)	このコマンドを使用して、ユーザー / オペレータ / マネージャーはジョブを取り消すことができます。
qhold(1)	実行依頼されたジョブの実行を保留します。
qhost(1)	Sun Grid Engine の実行ホストに関するステータス情報を表示します。
qlogin(1)	自動的に選択された、負荷の小さい適切なホストとの telnet または類似のログインセッションを開始します。
qmake(1)	UNIX 標準の make の代替機能です。make を機能拡張して、適切なマシンのクラスタに個々の make ステップを分散できるようにしています。
qmod(1)	このコマンドを使用し、キューの所有者は自身のマシンに関連づけられているあらゆるキューを一時停止または使用可能にしたり、あるいは所有しているキューで実行中のジョブを一時停止または使用可能にしたりできます。
qmon(1)	Sun Grid Engine のあらゆる機能に対する Motif コマンドインタフェースを提供します。構成済みキューのすべてまたは一部のステータスが、キューアイコンで色分けしながらオンラインで表示されます。

表 1 ユーザーインタフェースプログラムとその説明 (続き)

プログラム名	説明
qresub(1)	現在実行中または保留中のジョブをコピーすることによってジョブを新規作成します。
qrls(1)	たとえば上記の qhold(1) で設定されていたジョブのホールドを解除します。
qrsh(1)	UNIX 標準の rsh に相当する Sun Grid Engine の機能を使用した対話型のアプリケーションの遠隔実行、実行後ただちに端末入出力 (標準 / エラー出力と標準入力) と端末制御をサポートするバッチジョブの実行依頼、ジョブが完了するまで動作し続けるバッチジョブ実行依頼クライアントの提供、Sun Grid Engine 制御下での並列ジョブのタスクの遠隔実行など、さまざまな目的に利用することができます。
qselect(1)	指定された選択条件に一致するキュー名を一覧表示します。通常、このコマンドの出力は、選択されたキューにアクションを適用する目的で他の Sun Grid Engine コマンドに供給されます。
qsh(1)	負荷の小さいホストで対話形式のシェル (xterm (1) 内) を開きます。このシェルであらゆる種類の対話形式のジョブを実行することができます。
qstat(1)	クラスタに関連づけられているすべてのジョブとキューの状態を一覧表示します。
qsub(1)	Sun Grid Engine ジョブの実行依頼をするためのユーザーインタフェースです。
qtcsh(1)	広く知られ、かつ使用されている UNIX の C シェル (csh) の高機能版 tcsh と完全互換のコマンドです。コマンドシェルの機能を拡張し、Sun Grid Engine を使用して、負荷の小さい適切なホストに指定されたアプリケーションの実行を透過的に分散できるようにします。

## 関連項目

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 管理およびユーザーマニュアル』、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』

## 著作権

Copyright 2002 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) は、本書に記述されている製品に採用されている技術に関する知的所有権を有しています。これら知的所有権には、<http://www.sun.com/patents> に掲載されているひとつまたは複数の米国特許、および米国ならびにその他の国におけるひとつまたは複数の特許または出願中の特許が含まれています。

本書およびそれに付属する製品は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社の書面による事前の許可なく、本製品および本書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品のフォント技術を含む第三者のソフトウェアは、著作権法により保護されており、提供者からライセンスを受けているものです。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company Limited が独占的にライセンスしている米国ならびに他の国における登録商標です。

本製品は、株式会社モリサワからライセンス供与されたリュウミン L-KL (Ryumin-Light) および中ゴシック BBB (GothicBBB-Medium) のフォント・データを含んでいます。

本製品に含まれる HG 明朝 L と HG ゴシック B は、株式会社リコーがリョービマジクス株式会社からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。平成明朝体 W3 は、株式会社リコーが財団法人 日本規格協会 文字フォント開発・普及センターからライセンス供与されたタイプフェイスマスタをもとに作成されたものです。また、HG 明朝 L と HG ゴシック B の補助漢字部分は、平成明朝体 W3 の補助漢字を使用しています。なお、フォントとして無断複製することは禁止されています。

Sun、Sun Microsystems、AnswerBook2、docs.sun.com は、米国およびその他の国における米国 Sun Microsystems 社の商標もしくは登録商標です。サンロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャーに基づくものです。Energy Star のロゴは、米国 EPA の登録商標です。

OPENLOOK、OpenBoot、JLE は、サン・マイクロシステムズ株式会社の登録商標です。

ATOK は、株式会社ジャストシステムの登録商標です。ATOK8 は、株式会社ジャストシステムの著作物であり、ATOK8 にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。ATOK Server/ATOK12 は、株式会社ジャストシステムの著作物であり、ATOK Server/ATOK12 にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザーインターフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本書には、技術的な誤りまたは誤植のある可能性があります。また、本書に記載された情報には、定期的に変更が行われ、かかる変更は本書の最新版に反映されます。さらに、米国サンまたは日本サンは、本書に記載された製品またはプログラムを、予告なく改良または変更することがあります。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

---

## sgc\_ckpt (1)

### 名前

Sun Grid Engine チェックポイント - Sun Grid Engine のチェックポイント機構とチェックポイント機能のサポート

## 説明

Sun Grid Engine は、ユーザーレベルとオペレーティングシステムが提供する透過的レベルの 2 つのレベルのチェックポイントをサポートしています。ユーザーレベルのチェックポイント機能はアプリケーションを意味し、特定の時点またはアルゴリズムステップでチェックポイントファイル (再開ファイル) を作成し、再起動時にそれらのチェックポイントファイルを正しく処理することによって独自のチェックポイント機能を実現します。

透過的なチェックポイント機能はオペレーティングシステムによって提供される必要があり、通常、オペレーティングシステムのカーネルに組み込まれています。そうしたカーネル組み込みのチェックポイント機能としては、たとえば **Softway** から出ている SGI IRIX プラットフォーム用の **Hibernator** パッケージがあります。

チェックポイントジョブであることは、`qsub(1)` コマンドの `-ckpt` オプションを使用して **Sun Grid Engine** システムに認識させる必要があります。このフラグの引数は、使用するチェックポイント生成方法の属性を定義した、いわゆるチェックポイント環境を示します (詳細は `checkpoint(5)` を参照)。チェックポイント環境は、`qconf(1)` のオプションの `-ackpt`、`-dckpt`、`-mckpt`、`-sckpt` を使用して構成します。`qsub(1)` のオプション `-c` を使用して、参照先のチェックポイント環境に対する `when` 属性を書き換えることができます。

キューのタイプが **CHECKPOINTING** の場合、ジョブのチェックポイント属性はそうしたキューで実行できるように設定されている必要があります (`qsub(1)` の `-ckpt` オプションを参照)。通常のバッチジョブと異なり、チェックポイントジョブは、バッチまたは対話形式のジョブが一時停止されたり、影響を受けたりしない場合でも実行が中止されます。そうしたケースとしては、以下が挙げられます。

- ジョブに `x` イベント指定が行われていて (`qsub(1)` `-c` および `checkpoint(5)` を参照)、クラスタ管理者またはキュー所有者が `qmod(1)` を使用してキューまたはジョブを明示的に一時停止した場合
- 負荷平均値が対応するキューに設定されている移動しきい値を超えた場合 (`queue_conf(5)` を参照)
- チェックポイントジョブを受け持っている **Sun Grid Engine** 実行デーモン `sge_execd(8)` が停止した場合

明示的なユーザー要求によって他の特定のキューに実行依頼されたのでない限り、実行中止後、ジョブは他のキューに移動します。こうしたジョブの移動によって、動的な負荷均衡が図られます。

---

**注** - チェックポイントジョブの実行が中止されると、その時点でジョブが占有していたすべての資源 (メモリー、スワップ空間) が解放されます。この状態は、スワップ空間の占有が続く、通常のジョブの一時停止と異なります。

---



## 制限事項

現状では、ジョブが別のマシンのキューに移動した場合、自動的にはそのマシンにファイルは転送されません。このことは、ジョブ全体を通じて使用される、チェックポイントファイルや実行可能ファイル、スクラッチファイルなどのすべてのファイルを、たとえば、ジョブスクリプトの先頭で目的地的に見えるようにしておくか、転送しておく必要があることを意味します。

透過的なチェックポイントジョブの場合は、ディスク領域の使用についても、現実的な制約がいくつかあります。透過的なチェックポイントアプリケーションのチェックポイントは、通常、オペレーティングシステムによってチェックポイントファイルまたはディレクトリに保存されます。このファイルまたはディレクトリには、追加の制御情報とともに、プロセスのすべてのテキスト、データ、スタック空間が含まれます。このことは、非常に大きな仮想アドレス空間を使用するジョブは、非常に大きなチェックポイントファイルを生成することを意味します。ジョブが実際に実行されるワークステーションの空きディスク容量はわずかしらないこともあります。このため、マシンが休止中であつたとしても、いつでも、透過的なチェックポイントジョブを転送できるわけではありません。大きな仮想メモリージョブは休止していて、かつ十分な空きディスク領域があるマシンを待つため、ターンアラウンド時間が長くなる可能性があります。

## 関連項目

`sge_intro(1)`、`qconf(1)`、`qmod(1)`、`qsub(1)`、`checkpoint(5)`、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## qacct(1)

### 名前

`qacct` - Sun Grid Engine の使用量の報告とアカウントिंग

## 形式

```
qacct [ -A Account ] [ -b BeginTime ] [ -d Days ]  
[ -e EndTime ] [ -g [GroupId|GroupName] ]  
[ -h [HostName] ] [ -help ] [ -history HistoryPath ]  
[ -j [JobId|JobName] ] [ -l attr=val,... ] [ -nohist ]  
[ -o [Owner] ] [ -pe [PEname] ] [ -q [Q_name] ]  
[ -slots [SlotNumber] ] [ -t task_id_range_list ]  
[ -P [Project] ] [ -D [Department] ] [ -f Account FileName ]
```

## 説明

qacct ユーティリティはアカウントングデータファイルを読み取り (accounting(5) を参照)、ホスト名、キュー名、グループ名、所有者名、ジョブ名、ジョブ ID のカテゴリと -l スイッチを使用して指定された資源要求を満たすキューの時計時間、CPU 時間、システム時間に関する要約情報を生成します。カテゴリは組み合わせることができます。-j スイッチを使用してすべてまたは特定のジョブの情報を列挙することもできます。たとえば検索条件に、あるキューと所有者に関する要約を含めて、同じ要求内で 2 つのキューに関する要約は含めないように指定することができます。

## オプション

表 2 は qacct のオプションをまとめています。

表 2 qacct のオプション

オプション	説明
-A Account	要約を生成するジョブのアカウント
-b BeginTime	要約を生成するジョブの最も早い開始時間 ([CC]YY]MMDDhhmm].SS) の形式)。関連項目: -d オプション
-d Days	アカウントング情報を要約して表示する日数。-b BeginTime オプションと組み合わせられた場合は、上記の BeginTime と BeginTime + Days の範囲内に開始されたジョブが対象になります。下記の -e EndTime と組み合わせられた場合は、EndTime - Days からカウントが始まります。
-e EndTime	要約を生成するジョブの最も遅い開始時間 ([CC]YY]MMDDhhmm].SS) の形式)。関連項目: -d オプション

表 2 qacct のオプション (続き)

オプション	説明
<code>[-f AcctFileName]</code>	使用するアカウントリングファイル。省略された場合は、システムデフォルトのアカウントリングファイルが処理されます。
<code>-g [GroupId GroupName]</code>	アカウントリングに含めるジョブ所有者の数値のシステムグループ ID か英数字のグループ名。ID と名前のどちらも省略された場合は、すべてのグループが対象になります。
<code>-h [HostName]</code>	アカウントリング情報の要求先のホストの名前。大文字と小文字は区別されません。ホスト名が省略された場合は、すべてのホストの合計値がホストごとに一覧表示されま
<code>-help</code>	qacct コマンドのヘルプを表示します。
<code>-history HistoryPath</code>	キューおよび複合構成の履歴データが存在するディレクトリのパス。-l スイッチと組み合わせて、資源要求の一致検索に使用されます。-l スイッチが設定されていない場合、このオプションは無視されます。
<code>-j [[JobName JobId]]</code>	アカウントリング情報を表示する、実行中のジョブの名前か ID。名前と ID のどちらも省略された場合は、すべてのジョブが対象になります。このオプションは、qacct の出力形式を変更します。有効な場合、CPU 時間は集計されずに、書式付きの形式で「生の」アカウントリング情報が表示されます。表示される情報については、 <a href="#">accounting(5)</a> を参照してください。
<code>-l attr=val,...</code>	アカウントリング対象のジョブが実行されていたキューが満たす必要がある資源要求指定。履歴データを使用して一致するものが検索されます。すなわち、ジョブ開始時のキューおよび複合構成の状態が反映されます。資源要求は、 <a href="#">qsub(1)</a> で説明しているものと非常によく似ていますが、大きな違いは、使用する構成履歴データには負荷情報が含まれないため、たえず変化する負荷情報が要求できないことです。
<code>-nohist</code>	-l オプションとの組み合わせでのみ有効です。資源要求の一致検索でキューおよび複合構成の履歴データを強制的に使用させないようにして、 <a href="#">sge_qmaster(8)</a> から実際のキューおよび複合構成を読み取らせるようにします。現在のキューおよび複合構成が過去のジョブで有効であったときの状況のものとなりに異なることがあるため、紛らわしい統計結果が生成される可能性があることに注意してください。また、結果を得るには、アカウントリングファイルで参照されているすべてのホストが動作している必要があることにも注意してください。

表 2 qacct のオプション (続き)

オプション	説明
-o [Owner]	アカウントリング統計を収集するジョブの所有者名。 <b>Owner</b> 引数が省略された場合は、アカウントリングファイルに含まれているすべてのジョブ所有者のアカウントリング統計が一覧表示されます。
-pe [PEname]	使用統計の要約を生成する並列環境名。名前が省略された場合は、すべての並列環境のアカウントリングデータが環境ごとを一覧表示されます。
-q [Q_name]	使用統計の要約を生成するキュー名。名前が省略された場合は、すべてのキューのアカウントリングデータがキューごとを一覧表示されます。
-slots [SlotNumber]	使用統計の要約を生成するキュースロット数。スロット数が省略された場合は、すべてのキュースロットのアカウントリングデータがスロット番号ごとを一覧表示されます。
-t <i>task_id_range_list</i>	上記の -j オプションとの組み合わせでのみ使用できます。このスイッチは、アカウントリング情報を生成する配列ジョブのタスク範囲を指定します。 <i>task_id_range_list</i> の構文と形式は、qsub(1) の -t オプションで説明しているものと同じです。qsub(1) の配列ジョブについての詳細も参照してください。
-P [Project]	使用統計の要約を生成するプロジェクト名。名前が省略された場合は、すべてのプロジェクトのアカウントリングデータがプロジェクトごとを一覧表示されます。プロジェクトは、Sun Grid Engine, Enterprise Edition モードでのみ使用できます。
-D [Department]	使用統計の要約を生成する部署名。名前が省略された場合は、すべての部署のアカウントリングデータが部署ごとを一覧表示されます。部署は、Sun Grid Engine, Enterprise Edition モードでのみ使用できます。

## 環境変数

表 3 は、qacct に関する環境変数をまとめています。

表 3 qacct 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qacct は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qacct クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

- Sun Grid Engine のデフォルトのアカウントングファイル -  
<sge\_root>/<cell>/common/accounting
- Sun Grid Engine のデフォルトの履歴データベース -  
<sge\_root>/<cell>/common/history
- Sun Grid Engine のマスターホストファイル -  
<sge\_root>/<cell>/common/act\_qmaster

## 関連項目

sge\_intro(1)、qsub(1)、accounting(5)、sge\_qmaster(8)、sge\_commd(8)

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## qconf (1)

### 名前

`qconf` - Sun Grid Engine のキュー構成

### 説明

`qconf` を使用して、システム管理者は、キュー管理やホスト管理、複合管理、ユーザー管理などに関する現在の **Sun Grid Engine** 構成の追加や削除、変更を行うことができます。また、`qconf` を使用して、既存のキューの現在のキュー構成を確認することもできます。

# オプション

特に明記していない限り、表 4 にまとめたオプションと対応する操作は、正当なアカウントを持つすべてのユーザーが行うことができます。

表 4 qconf のオプション

オプション	説明
-Aattr <i>obj_spec fname obj_instance,...</i>	オブジェクト属性への追加 - 下記の -aattr に似ていますが、 <i>fname</i> という名前のファイルから、追加するオブジェクト属性の指定を受け取ります。 -aattr と異なり、複数の属性に指定を追加できます。この指定は、対応するオブジェクトのファイル形式 (たとえばキューの場合は <code>queue_conf(5)</code> ) に従った <i>fname</i> に列挙する必要があります。 このオプションには、root/ マネージャー特権が必要です。
-Ac <i>complex_name fname</i>	複合の追加 - Sun Grid Engine クラスタに <i>fname</i> に定義されている <i>complex_name</i> という名前の複合を追加します。複合指定の形式については、 <code>complex(5)</code> に説明があります。root またはマネージャー特権が必要です。
-Acal <i>fname</i>	カレンダーの追加 - Sun Grid Engine 環境に新規カレンダー定義を追加します。Sun Grid Engine では、カレンダーを使用してキューが使用可能な時間と使用不可能な時間を定義します。カレンダー定義の形式については、 <code>calendar_conf(5)</code> に説明があります。カレンダー定義は、 <i>fname</i> というファイルから受け取られます。root またはマネージャー特権が必要です。
-Ackpt <i>fname</i>	チェックポイント環境の追加 - サポートされているチェックポイント環境リストに、 <i>fname</i> に定義されているチェックポイント環境を追加します ( <code>checkpoint(5)</code> を参照)。root またはマネージャー特権が必要です。

表 4 qconf のオプション (続き)

オプション	説明
-Aconf <i>file_list</i>	構成の追加 - コンマで区切られた <i>file_list</i> に列挙されたファイルに指定されているクラスタ構成を追加します (sge_conf (5) を参照)。root またはマネージャー特権が必要です。
-Ae <i>fname</i>	実行ホストの追加 - Sun Grid Engine クラスタに、 <i>fname</i> に定義されている実行ホストを追加します。実行ホスト指定の形式については、host_conf (5) に説明があります。root またはマネージャー特権が必要です。
-Ap <i>fname</i>	並列環境の追加 - Sun Grid Engine クラスタに、 <i>fname</i> に定義されている並列環境を追加します。root またはマネージャー特権が必要です。
-Aprj <i>fname</i>	プロジェクトの追加 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。登録されているプロジェクトのリストに、 <i>fname</i> に定義されているプロジェクトを追加します (project (5) を参照)。root またはマネージャー特権が必要です。
-Aq <i>fname</i>	キューの追加 - Sun Grid Engine クラスタに、 <i>fname</i> に定義されているキューを追加します。root またはマネージャー特権が必要です。
-Au <i>fname</i>	ACL の追加 - Sun Grid Engine にユーザーアクセスリスト (ACL) を追加します。ユーザーリストは、キュー使用の認証に使用されます。root / マネージャー / オペレータ特権が必要です。
-Dattr <i>obj_spec fname obj_instance,...</i>	オブジェクト属性からの削除 - 下記の -dattr に似ていますが、 <i>fname</i> というファイルから、削除するエントリの存在するリスト属性の定義を受け取ります。-dattr と異なり、複数の属性を変更できます。この指定は、対応するオブジェクトのファイル形式 (たとえばキューの場合は queue_conf (5)) に従った <i>fname</i> に列挙する必要があります。root またはマネージャー特権が必要です。



表 4 qconf のオプション (続き)

オプション	説明
-Mattr <i>obj_spec fname obj_instance,...</i>	<p>オブジェクト属性の変更 - 下記の -mattr に似ていますが、<i>fname</i> という名前のファイルから、変更するオブジェクト属性の指定を受け取ります。</p> <p>-mattr と異なり、複数の属性を変更できます。この指定は、対応するオブジェクトのファイル形式 (たとえばキューの場合は <code>queue_conf(5)</code>) に従った <i>fname</i> に列挙する必要があります。</p> <p>root またはマネージャー特権が必要です。</p>
-Mc <i>complex_name fname</i>	<p>複合の変更 - 指定された複合を <i>fname</i> の内容で書き換えます。引数ファイルは、<code>complex(5)</code> に指定されている形式に準じている必要があります。root またはマネージャー特権が必要です。</p>
-Mcal <i>fname</i>	<p>カレンダーの変更 - <i>fname</i> に指定されているとおりにカレンダーの定義を書き換えます。引数ファイルは、<code>calendar_conf(5)</code> に指定されている形式に準じている必要があります。root またはマネージャー特権が必要です。</p>
-Mckpt <i>fname</i>	<p>チェックポイント環境の変更 - 既存のチェックポイント環境を <i>fname</i> 内の定義内容で書き換えます (<code>checkpoint(5)</code> を参照)。<i>fname</i> 内の名前属性が既存のチェックポイント環境に一致する必要があります。root またはマネージャー特権が必要です。</p>
-Me <i>fname</i>	<p>実行ホストの変更 - <code>host_conf(5)</code> に定義されている形式に準じた <i>fname</i> の内容で、指定されたホストの実行ホスト構成を書き換えます。root またはマネージャー特権が必要です。</p>
-Mp <i>fname</i>	<p>並列環境構成の変更 - 下記の -mp と同じですが、エディタを使用して並列環境の構成を変更するのではなく、<i>fname</i> というファイルに変更した構成が含まれているとみなされます。並列環境構成の形式についての詳細は、<code>sge_pe(5)</code> を参照してください。root またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-Mprj <i>fname</i>	プロジェクト構成の変更 - 下記の -mprj と同じですが、エディタを使用してプロジェクトの構成を変更するのではなく、 <i>fname</i> というファイルに変更した構成が含まれているとみなされます。プロジェクト構成の形式についての詳細は、project (5) を参照してください。root またはマネージャー特権が必要です。
-Mq <i>fname</i>	キュー構成の変更 - 下記の -mq と同じですが、エディタを使用してキューの構成を変更するのではなく、 <i>fname</i> というファイルに変更した構成が含まれているとみなされます。キュー構成の形式についての詳細は、queue_conf (5) を参照してください。root またはマネージャー特権が必要です。
-Mqattr <i>fname q_name</i> ,... このコマンドの代わりに、-Mattr を使用のこと。	キュー属性の変更 - 複数のキューの複数のキュー構成属性を 1 つのコマンドで変更できます。コマンド区切りのキュー名リストに含まれるすべてのキューに、 <i>fname</i> に含まれるキュー属性の定義が適用されます。 <i>fname</i> に含まれていないキュー属性は変更されません。 <i>qname</i> と <i>ghostname</i> 以外のすべてのキュー属性を変更できます。キュー構成の形式についての詳細は、queue_conf (5) を参照してください。root またはマネージャー特権が必要です。
-Mu <i>fname</i>	ACL の変更 - <i>fname</i> に定義されているユーザーアクセスリスト (ACL) を受け取って、同じ名前の既存の ACL を書き換えます。ACL 構成の形式については、access_list (5) を参照してください。root またはマネージャー特権が必要です。
-Rattr <i>obj_spec fname obj_instance</i> ,...	オブジェクト属性の置換 - 下記の -rattr に似ていますが、 <i>fname</i> というファイルから、置換するエントリの存在するリスト属性の定義を受け取ります。-rattr と異なり、複数の属性を変更できます。この指定は、対応するオブジェクトのファイル形式 (たとえばキューの場合は queue_conf (5)) に従った <i>fname</i> に列挙する必要があります。root またはマネージャー特権が必要です。

表 4 qconf のオプション (続き)

オプション	説明
-aattr <i>obj_spec attr_name val obj_instance,...</i>	<p>オブジェクト属性への追加 - オブジェクトの複数インスタンスの構成リスト属性の 1 つに 1 つのコマンドで指定を追加できます。現在サポートされているオブジェクトは、<i>obj_spec</i> で <i>queue</i> または <i>host</i> として指定されているキューとホスト構成だけです。リスト属性には、たとえばキューの <i>load_thresholds</i> パラメータがあります。-aattr オプションが、そうしたリストにエントリを追加するのに対し、-datrr はエントリを削除、-matrr は変更、-ratrr は置換します。追加先の構成属性名は、<i>name=value</i> の形式で <i>attr_name</i> の後に <i>val</i> を続けることによって指定します。変更の適用先のオブジェクトのインスタンスの、コンマ区切りのリスト (たとえばキューのリスト) は、コマンドの最後に指定します。次の制限があります。<i>host</i> オブジェクトの場合、<i>load_values</i> は変更できません (<i>host_conf</i> (5) を参照)。<i>root</i> またはマネージャー特権が必要です。</p>
-ac <i>complex_name</i>	<p>複合の追加 - Sun Grid Engine 環境に複合を追加します。複合エントリには、システムに実行依頼するジョブが要求可能な資源が含まれます。複合定義の形式についての詳細は、<i>complex</i> (5) のマニュアルページを参照してください。このオプションの引数に複合名が指定されていると、qconf は一時ファイルを開き、環境変数 EDITOR が示すテキストエディタ (EDITOR が設定されていない場合は、デフォルトエディタの vi) を起動します。複合の定義を入力して、エディタを終了すると、新しい複合が検査され、<i>sge_qmaster</i> (8) に登録されます。<i>root</i> またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-acal <i>calendar_name</i>	<p>カレンダーの追加 - Sun Grid Engine 環境に新規カレンダー定義を追加します。Sun Grid Engine では、カレンダーを使用してキューが使用可能な時間と使用不可能な時間を定義します。カレンダー定義の形式については、calendar_conf(5) に説明があります。オプション引数にカレンダー名が指定されていると、qconf は一時ファイルを開き、環境変数 EDITOR が示すテキストエディタ (EDITOR が設定されていない場合は、デフォルトエディタの vi) を起動します。カレンダーの定義を入力して、エディタを終了すると、新しいカレンダーが検査され、sge_qmaster(8) に登録されます。root またはマネージャー特権が必要です。</p>
-ackpt <i>ckpt_name</i>	<p>チェックポイント環境の追加 - Sun Grid Engine が管理するチェックポイント環境リストに <i>ckpt_name</i> の名前でチェックポイント環境を追加し、チェックポイントジョブの実行依頼に使用できるようにします (チェックポイント環境定義の形式についての詳細は、checkpoint(5) を参照)。qconf はデフォルトのチェックポイント環境構成を読み取り、vi(1) (EDITOR 環境変数が設定されている場合は \$EDITOR) を実行して、チェックポイント環境の構成をカスタマイズできるようにします。エディタを終了すると、その新しいチェックポイント環境が sge_qmaster(8) に登録されます。root またはマネージャー特権が必要です。</p>
-aconf <i>host,...</i>	<p>構成の追加 - コンマ区切りの <i>file_list</i> に指定されたホストにクラスタ構成を続けて追加します (sge_conf(5) を参照)。すべてのホストについてエディタ (指定された \$EDITOR か vi(1)) が起動されホストの構成を入力できます。ファイルを保存し、エディタを終了すると、構成が sge_qmaster(8) に登録されます。root またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-ae [ <i>host_template</i> ]	<p>実行ホストの追加 - Sun Grid Engine 実行ホストリストにホストを追加します。ホストにキューが構成されている場合、そのホストは自動的に Sun Grid Engine 実行ホストリストに追加されます。明示的に実行ホストを追加することの利点は、実行ホストの登録で負荷スケール値などのパラメータを指定できることです。ただし、それらのパラメータは下記の -me オプションを使用して、いつでもデフォルト値から変更できます。</p> <p><i>host_template</i> 引数が指定されている場合、qconf は sge_qmaster(8) から指定された実行ホストの構成を読み取り、指定されていない場合は、一般的なテンプレートを読み取ります。読み取られたテンプレートはファイルに保存され、qconf は vi(1) (EDITOR 環境変数が設定されている場合は \$EDITOR が示すエディタ) を実行して、そのファイルのエントリの変更を可能にします。実行ホスト指定の形式については、host_conf(5) に説明があります。エディタで変更を保存し、エディタを終了すると、その新しい実行ホストが sge_qmaster(8) に登録されます。このオプションには、root/ マネージャー特権が必要です。</p>
-ah <i>hostname</i> ,...	<p>管理ホストの追加 - Sun Grid Engine によってトラストされたホストのリストにホスト <i>hostname</i> を追加します。Sun Grid Engine の管理コマンドは、このリストに登録されているホストから実行する必要があります。これに対する唯一の例外は、sge_qmaster(8) ノードで qconf を実行することだけです。通常、Sun Grid Engine のデフォルトのインストールでは、指定されたすべての実行ホストが、Sun Grid Engine によってトラストされたホストリストに自動的に追加されます (上記の -ae オプションを参照)。root またはマネージャー特権が必要です。</p>
-am <i>user</i> ,...	<p>マネージャーの追加 - 指定されたユーザーを Sun Grid Engine マネージャーリストに追加します。root またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-ao <i>user</i> ,...	オペレータの追加 - 指定されたユーザーを <b>Sun Grid Engine</b> オペレータリストに追加します。root またはマネージャー特権が必要です。
-ap <i>pe_name</i>	新規並列環境の追加 - <b>Sun Grid Engine</b> が管理する並列環境のリストに <i>pe_name</i> という名前で並列環境 (PE) の定義を追加して、並列ジョブの実行依頼に使用できるようにします (並列環境定義の形式についての詳細は、 <a href="#">sge_pe(5)</a> を参照)。qconf はデフォルトの並列環境構成を読み取り、vi(1) (EDITOR 環境変数が設定されている場合は \$EDITOR) を実行して、並列環境の構成をカスタマイズできるようにします。エディタを終了すると、その新しい並列環境が <a href="#">sge_qmaster(8)</a> に登録されます。root またはマネージャー特権が必要です。
-aprj	新規プロジェクトの追加 - このオプションは、 <b>Sun Grid Engine, Enterprise Edition</b> システムでのみ使用できます。 <b>Sun Grid Engine</b> システムでは使用できません。登録されているプロジェクトのリストに、プロジェクトの定義を追加します ( <a href="#">project(5)</a> を参照)。qconf はプロジェクト構成用のテンプレートを読み取り、vi(1) (EDITOR 環境変数が設定されている場合は \$EDITOR) を実行して、プロジェクトをカスタマイズできるようにします。エディタを終了すると、その新しいプロジェクトが <a href="#">sge_qmaster(8)</a> に登録されます。root またはマネージャー特権が必要です。

表 4 qconf のオプション (続き)

オプション	説明
-aq [ <i>q_template</i> ]	<p>新規キューの追加 - qconf はデフォルトのキュー構成 (queue_conf (5) を参照) または <i>q_template</i> キューの構成 (省略可能な引数が指定されている場合) を読み取り、vi (1) (EDITOR 環境変数が設定されている場合は \$EDITOR) を実行して、そのキューの構成をカスタマイズできるようにします。エディタを終了すると、その新しいキューが sge_qmaster (8) に登録されます。最小構成としては、キュー名とキューのホスト名のみ設定する必要があります。root またはマネージャー特権が必要です。</p>
-as <i>hostname</i> ,...	<p>実行依頼ホストの追加 - Sun Grid Engine ジョブの実行依頼とその動作の制御のみを許可されたホストのリストにホスト <i>hostname</i> を追加します。root またはマネージャー特権が必要です。</p>
-astnode <i>node_path=shares</i> ,...	<p>基本割当ツリーノードの追加 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定された基本割当ツリーノード (複数指定可能) を基本割当ツリーに追加します (share_tree (5) を参照)。<i>node_path</i> は基本割当ツリー内の新規ノードの位置を示す階層形式のパス (<i>[/node_name[/.]node_name...]</i>) です。<i>node_path</i> のベース名は、新規ノードの名前です。ノードには、指定された配分量が初期設定されます。root またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-astree	<p>基本割当ツリーの追加 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。システムに基本割当ツリーの定義を追加します (<code>share_tree(5)</code> を参照)。qconf は基本割当ツリーのテンプレートを読み取り、エディタ (デフォルトの <code>vi(1)</code> か <code>\$EDITOR</code> に指定されたエディタ) を起動して、基本割当ツリーの定義を変更できるようにします。エディタを終了すると、その基本割当ツリーの定義が <code>sgc_qmaster(8)</code> に登録されます。root またはマネージャー特権が必要です。</p>
-au <i>user,... acl_name,...</i>	<p>ACL へのユーザーの追加 - Sun Grid Engine ユーザーアクセスリスト (ACL) にユーザーを追加します。ユーザーリストは、キュー使用の認証に使用されません。root / マネージャー / オペレータ特権が必要です。</p>
-Auser <i>fname</i>	<p>ユーザーの追加 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。Sun Grid Engine, Enterprise Edition クラスタに <i>fname</i> に定義されたユーザーを追加します。ユーザー指定の形式については、<code>user(5)</code> に説明があります。root またはマネージャー特権が必要です。</p>
-auser	<p>ユーザーの追加 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。登録されているユーザーのリストに、ユーザーを追加します (<code>user(5)</code> を参照)。qconf はエディタ (デフォルトの <code>vi(1)</code> か <code>EDITOR</code> 環境変数が示すエディタ) を起動して、ユーザー定義用のテンプレートを開きます。エントリを変更して、エディタを終了すると、その新規ユーザーが登録されます。root またはマネージャー特権が必要です。</p>



表 4 qconf のオプション (続き)

オプション	説明
-clearusage	基本割当ツリー利用情報のクリア - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。基本割当ツリーからすべてのユーザーおよびプロジェクトの利用情報をクリアします。すべての利用情報が初期値のゼロに戻されます。
-cq <i>queue_name</i> ,...	キューの後処理 - キューから取得されていなかったジョブをクリアします。主として開発ツールです。root/ マネージャー / オペレータ特権が必要です。
-dattr <i>obj_spec attr_name val obj_instance</i> ,...	オブジェクト内の属性の削除 - オブジェクトの複数インスタンスの構成リスト属性の1つから、1つのコマンドで指定を削除できます。現在サポートされているオブジェクトは、 <i>obj_spec</i> で <i>queue</i> または <i>host</i> として指定されているキューとホスト構成だけです。リスト属性には、たとえばキューの <i>load_thresholds</i> パラメータがあります。-dattr オプションが、そうしたリストからエントリを削除するのに対し、-aattr はエントリを追加、-mattr は変更、-rattr は置換します。変更する構成属性名は、 <i>attr_name</i> の後に、削除する属性リストエントリ名を示す <i>val</i> を続けることによって指定します。変更の適用先のオブジェクトのインスタンスの、コンマ区切りのリスト (たとえばキューのリスト) は、コマンドの最後に指定します。次の制限があります。 <i>host</i> オブジェクトの場合、 <i>load_values</i> は変更できません ( <i>host_conf</i> (5) を参照)。root またはマネージャー特権が必要です。
-dc <i>complex_name</i> ,...	複合の削除 - Sun Grid Engine から複合を削除します。このオプションには、root/ マネージャー特権が必要です。
-dcal <i>calendar_name</i> ,...	カレンダーの削除 - Sun Grid Engine から指定されたカレンダーの定義を削除します。root またはマネージャー特権が必要です。

表 4 qconf のオプション (続き)

オプション	説明
-dckpt <i>ckpt_name</i>	チェックポイント環境の削除 - 指定されたチェックポイント環境を削除します。 <b>root</b> またはマネージャー特権が必要です。
-dconf <i>host,...</i>	構成の削除 - 指定されたホストの構成エントリを構成リストから削除します。 <b>root</b> またはマネージャー特権が必要です。
-de <i>host_name,...</i>	実行ホストの削除 - Sun Grid Engine の実行ホストリストからホストを削除します。 <b>root</b> またはマネージャー特権が必要です。
-dh <i>host_name,...</i>	管理ホストの削除 - Sun Grid Engine によってトラストされているホストリストからホストを削除します。現在 <code>sge_qmaster(8)</code> が動作しているホストを削除することはできません。このオプションには、 <b>root/</b> マネージャー特権が必要です。
-dm <i>user[,user,...]</i>	マネージャーの削除 - マネージャーリストからマネージャーを削除します。 <b>root</b> またはマネージャー特権が必要です。
-do <i>user[,user,...]</i>	オペレータの削除 - オペレータリストからオペレータを削除します。 <b>root</b> またはマネージャー特権が必要です。
-dp <i>pe_name</i>	並列環境の削除 - 指定された並列環境 (PE) を削除します。 <b>root</b> またはマネージャー特権が必要です。
-dprj <i>project,...</i>	プロジェクトの削除 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定されたプロジェクト (複数指定可能) を削除します。 <b>root</b> またはマネージャー特権が必要です。
-dq <i>queue_name,...</i>	キューの削除 - 指定されたキュー (複数指定可能) を削除します。アクティブなジョブは最後まで実行が許可されます。 <b>root</b> またはマネージャー特権が必要です。

表 4 qconf のオプション (続き)

オプション	説明
-ds <i>host_name</i> ,...	実行依頼ホストの削除 - Sun Grid Engine の実行依頼ホストリストからホストを削除します。root またはマネージャー特権が必要です。
-dstnode <i>node_path</i> ,...	基本割当ツリーノードの削除 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定された基本割当ツリーノード (複数指定可能) を削除します。 <i>node_path</i> は基本割当ツリーから削除するノードの位置を示す階層形式のパス ( <code>[/]<i>node_name</i>[/<i>node_name</i>...]</code> ) です。root またはマネージャー特権が必要です。
-dstree	基本割当ツリーの削除 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。現在の基本割当ツリーを削除します。root またはマネージャー特権が必要です。
-du <i>user</i> ,... <i>acl_name</i> ,...	ACL からのユーザーの削除 - Sun Grid Engine のユーザーアクセスリスト (ACL) からユーザーを削除します (どちらも複数指定可能)。root / マネージャー / オペレータ特権が必要です。
-dul <i>acl_name</i> ,...	ユーザーリストの削除 - システムからユーザーリスト (複数指定可能) を削除します。root / マネージャー / オペレータ特権が必要です。
-duser <i>user</i> ,...	ユーザーの削除 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。登録されているユーザーのリストから指定されたユーザー (複数指定可能) を削除します。root またはマネージャー特権が必要です。
-help	全オプションの一覧を表示します。

表 4 qconf のオプション (続き)

オプション	説明
-k{m s e[j] {host,... all}}	<p>Sun Grid Engine ソフトウェアの停止                      -km の場合、sge_qmaster(8) は制御された方法で強制的に終了します。同様に、-ks スイッチは sge_schedd(8) を強制的に終了します。現在登録されていて、実行中の sge_execd(8) プロセスの停止は、-ke オプションを使用して行います。-kej が指定されると、実行ホスト上のすべてのジョブの実行が、対応する sge_execd(8) の終了前に中止されます。コンマ区切りのホストリストには、-ke および ke j オプションの処理対象の実行ホストを指定します。ホストリストではなくキーワードの all が指定された場合は、実行中のすべて sge_execd(8) プロセスが停止させられます。root またはマネージャー特権が必要です。</p>
-kec {id,... all}	<p>イベントクライアントの終了 -                      sge_qmaster(8) に登録されているイベントクライアントの停止に使用します。コンマ区切りのイベントクライアントリストには、-kec オプションの処理対象となるイベントクライアントを指定します。イベントクライアントリストの代わりにキーワードの all が指定された場合は、sge_schedd(8) などの特殊なクライアントを除く実行中のすべて イベントクライアントが停止させられます。root またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-mattr <i>obj_spec attr_name val obj_instance,...</i>	<p>オブジェクト属性の変更 - オブジェクトの複数インスタンスの構成属性の1つを1つのコマンドで変更できます。現在サポートされているオブジェクトは、<i>obj_spec</i> で <i>queue</i> または <i>host</i> として指定されているキューとホスト構成だけです。-mattr <i>queue attr_name val q_name, ...</i> は -mqattr <i>attr_name val q_name,...</i> と同等であることに注意してください (下記を参照)。後者は、下位互換性を維持するために提供されているオプションです。変更する構成属性名は、<i>attr_name</i> の後に、属性に設定する値を続けることによって指定します。属性がキュー <i>load_thresholds</i> などのリストの場合、<i>val</i> は名前 = 値のペアで指定することができます。この場合、リスト内の対応するエントリだけが変更されます。特にこうしたリスト属性を変更するその他の方法については、-aattr、-dattrib、-rattr オプションの説明を参照してください。変更の適用先のオブジェクトのインスタンスの、コンマ区切りのリスト (たとえばキューのリスト) は、コマンドの最後に指定します。次の制限があります。キューオブジェクトの場合、<i>qname</i> と <i>ghostname</i> 属性は変更できません (<i>queue_conf</i> (5) を参照)。<i>host</i> オブジェクトの場合、<i>hostname</i> と <i>load_values</i>、<i>processors</i> 属性は変更できません (<i>host_conf</i> (5) を参照)。<i>root</i> またはマネージャー特権が必要です。</p>
-mc <i>complex_name</i>	<p>複合の変更 - 指定された複合の構成を読み取り (<i>complex</i> (5) を参照)、エディタ (デフォルトの <i>vi</i> (1) または <i>\$EDITOR</i> が示すエディタ) を実行します。エディタを終了すると、変更された複合構成が <i>sge_qmaster</i> (8) に登録されます。<i>root</i> またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-mcal <i>calendar_name</i>	<p>カレンダーの変更 - 指定された複合の定義を読み取り (<i>calendar_conf</i> (5) を参照)、エディタ (デフォルトの <i>vi</i> (1) または <i>\$EDITOR</i> が示すエディタ) を実行します。エディタを終了すると、変更されたカレンダーの定義が <i>sgc_qmaster</i> (8) に登録されます。<i>root</i> またはマネージャー特権が必要です。</p>
-mckpt <i>ckpt_name</i>	<p>チェックポイント環境の変更 - 指定されたチェックポイント環境の現在の構成を読み取り、エディタ (デフォルトの <i>vi</i> (1) か <i>\$EDITOR</i> が示すエディタ) を実行して、新しい構成を <i>sgc_qmaster</i> (8) に登録します。チェックポイント環境構成の形式についての詳細は、<i>checkpoint</i> (5) を参照してください。<i>root</i> またはマネージャー特権が必要です。</p>
-mconf [ <i>host,...</i>   <i>global</i> ]	<p>構成の変更 - 指定されたホストの構成を読み取り、エディタ (デフォルトの <i>vi</i> (1) または <i>\$EDITOR</i> が示すエディタ) を実行します。エディタを終了すると、変更された構成が <i>sgc_qmaster</i> (8) に登録されます。<i>host</i> 引数が省略されるか、特殊ホスト名の <i>global</i> が指定された場合は、セルのグローバル構成が変更されます。ホスト構成の形式については、<i>sgc_conf</i> (5) に説明があります。<i>root</i> またはマネージャー特権が必要です。</p>
-me <i>hostname</i>	<p>実行ホストの変更 - 指定された実行ホストの現在の構成を読み取り、エディタ (デフォルトの <i>vi</i> (1) か <i>\$EDITOR</i> が示すエディタ) を実行します。エディタを終了すると、変更された構成が <i>sgc_qmaster</i> (8) に登録されます。実行ホスト構成の形式については、<i>host_conf</i> (5) に説明があります。<i>root</i> またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-mp <i>pe_name</i>	<p>並列環境の構成の変更 - 指定された並列環境 (PE) の現在の構成を読み取り、エディタ (デフォルトの vi (1) か \$EDITOR が示すエディタ) を実行して、新しい構成を <code>sgc_qmaster (8)</code> に登録します。並列環境構成の形式についての詳細は、<code>sgc_pe (5)</code> を参照してください。root またはマネージャー特権が必要です。</p>
-mprj <i>project</i>	<p>プロジェクトの変更 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。qconf は指定されたプロジェクトのデータを読み取り、エディタ (デフォルトの vi (1) か \$EDITOR に指定されたエディタ) を起動して、プロジェクトの定義を変更できるようにします。エディタを終了すると、その変更後の定義が <code>sgc_qmaster (8)</code> に登録されます。root またはマネージャー特権が必要です。</p>
-mq <i>queuename</i>	<p>キュー構成の変更 - 指定されたキューの現在の構成を読み取り、エディタ (デフォルトの vi (1) か \$EDITOR が示すエディタ) を実行して、新しい構成を <code>sgc_qmaster (8)</code> に登録します。キュー構成の形式についての詳細は、<code>queue_conf (5)</code> を参照してください。root またはマネージャー特権が必要です。</p>
<p>-mqattr <i>attr_name val q_name...</i> このコマンドの代わりに、-mattr を使用のこと。</p>	<p>キュー属性の変更 - 複数のキューの構成属性の 1 つを 1 つのコマンドで変更できます。コンマ区切りのキュー名リストに含まれるすべてのキューについては、属性 <i>attr_name</i> の値が <i>val</i> で書き換えられます。<i>qname</i> と <i>qhostname</i> 以外のすべてのキュー属性を変更できます。キュー構成の形式についての詳細は、<code>queue_conf (5)</code> を参照してください。root またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-msconf	<p>スケジューラ構成の変更 - 指定されたスケジューラの構成を読み取り (sched_conf (5) を参照)、エディタ (デフォルトの vi (1) または \$EDITOR が示すエディタ) を実行します。エディタを終了すると、変更されたスケジューラの構成が sge_qmaster (8) に登録されます。root またはマネージャー特権が必要です。&lt;&gt;</p>
-mstnode <i>node_path=shares,...</i>	<p>基本割当ツリーノードの変更 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定された基本割当ツリーノード (複数指定可能) を基本割当ツリーから削除します (share_tree (5) を参照)。node_path は基本割当ツリー内のノードの位置を示す階層形式のパス ([/node_name[/./node_name...]) です。ノードには、指定された配分量が設定されます。root またはマネージャー特権が必要です。</p>
-mstree	<p>基本割当ツリーの変更 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。基本割当ツリーの定義を変更します (share_tree (5) を参照)。qconf は現在の基本割当ツリーを読み取り、エディタ (デフォルトの vi (1) か \$EDITOR が示すエディタ) を起動して、基本割当ツリーの定義を変更できるようにします。エディタを終了すると、その基本割当ツリーの定義が sge_qmaster (8) に登録されます。root またはマネージャー特権が必要です。</p>
-mu <i>acl_name</i>	<p>ユーザーアクセスリストの変更 - 指定されたユーザーアクセスリストの現在の構成を読み取り、エディタ (デフォルトの vi (1) か \$EDITOR が示すエディタ) を実行して、新しい構成を sge_qmaster (8) に登録します。root またはマネージャー特権が必要です。</p>



表 4 qconf のオプション (続き)

オプション	説明
-rattr <i>obj_spec attr_name val obj_instance,...</i>	<p>オブジェクト属性の置換 - オブジェクトの複数インスタンスの構成属性の1つを1つのコマンドで置換できます。現在サポートされているオブジェクトは、<i>obj_spec</i> で <i>queue</i> または <i>host</i> として指定されているキューとホスト構成だけです。リスト属性には、たとえばキューの <i>load_thresholds</i> パラメータがあります。-rattr オプションが、そうしたリストを置換するのに対し、-aattr はエントリを追加、-dattr は削除、-mattr は変更します。変更する構成属性名は、<i>attr_name</i> に続けて、削除する属性リストエントリ名の後に、属性の新しい設定を定義する <i>val</i> を続けることによって指定します。変更の適用先のオブジェクトのインスタンスの、コンマ区切りのリスト (たとえばキューのリスト) は、コマンドの最後に指定します。次の制限があります。<i>host</i> オブジェクトの場合、<i>load_values</i> 属性は変更できません (<i>host_conf</i> (5) を参照)。<i>root</i> またはマネージャー特権が必要です。</p>
-Muser <i>fname</i>	<p>ユーザーの変更 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。Sun Grid Engine, Enterprise Edition クラスタに <i>fname</i> に定義されたユーザーを追加します。ユーザー指定の形式については、<i>user</i> (5) に説明があります。<i>root</i> またはマネージャー特権が必要です。</p>
-muser <i>user</i>	<p>ユーザーの変更 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。qconf は指定されたユーザーのデータを読み取り、エディタ (デフォルトの <i>vi</i> (1) か <i>\$EDITOR</i> が示すエディタ) を起動して、ユーザーの定義を変更できるようにします。エディタを終了すると、その変更後の定義が <i>sge_qmaster</i> (8) に登録されます。<i>root</i> またはマネージャー特権が必要です。</p>

表 4 qconf のオプション (続き)

オプション	説明
-sc <i>complex_name</i> ,...	複合の表示 - 指定された複合 (複数指定可能) の構成を表示します。
-scal <i>calendar_name</i>	カレンダーの表示 - 指定されたカレンダーの構成を表示します。
-scall	カレンダーリストの表示 - 定義済みのすべてのカレンダーのリストを表示します。
-scl	複合リストの表示 - 構成済みのすべての複合のリストを表示します。
-sckpt <i>ckpt_name</i>	チェックポイント環境の表示 - 指定されたチェックポイント環境の構成を表示します。
-sckptl	チェックポイント環境リストの表示 - 構成済みのすべての並列環境名のリストを表示します。
-sconf [ <i>host</i> ,...  <i>global</i> ]	構成の表示 - グローバルまたは指定されたホスト (複数指定可能) のクラスタ構成を表示します。コンマ区切りのホストリスト引数が省略されるか、特殊文字列の <i>global</i> が指定された場合は、セルのグローバル構成が表示されます。リストが指定された場合は、グローバル構成と各ホストに固有の構成を 1 つにしたものが表示されます。ホスト構成の形式については、 <code>sge_conf(5)</code> に説明があります。
-sconf1	構成リストの表示 - 構成が定義されているホストのリストを表示します。特殊ホスト名 <i>global</i> は、セルのグローバル構成を意味します。
-se <i>hostname</i>	実行ホストの表示 - 指定された実行ホストの定義を表示します。
-secl	イベントクライアントの表示 - Sun Grid Engine のイベントクライアントのリストを表示します。
-sel	実行ホストの表示 - Sun Grid Engine の実行ホストのリストを表示します。

表 4 qconf のオプション (続き)

オプション	説明
-sep	ライセンス付きプロセッサの表示 - 各実行ホストの、ライセンス供与されているプロセッサ数とその合計数のリストを表示します。
-sh	管理ホストの表示 - Sun Grid Engine の管理ホストのリストを表示します。
-sm	マネージャーの表示 - マネージャーのリストを表示します。
-so	オペレータの表示 - オペレータのリストを表示します。
-sp <i>pe_name</i>	並列環境構成の表示 - 引数に指定された並列環境 (PE) の定義を表示します。
-spl	並列環境リストの表示 - 定義済みのすべての並列環境 (PE) のリストを表示します。
-sprj <i>project</i>	プロジェクトの表示 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定されたプロジェクトの定義を表示します ( <i>project</i> (5) を参照)。
-sprjl	プロジェクトリストの表示 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。定義済みのすべてのプロジェクトのリストを表示します。
-sq <i>queue_name[queue_name,...]</i>	キューの表示 - キュー (複数指定可能) の構成を表示します。
-sql	キューリストの表示 - 構成済みのすべてのキューのリストを表示します。
-ss	実行依頼ホストの表示 - Sun Grid Engine の実行依頼ホストのリストを表示します。
-ssconf	スケジューラ構成の表示 - <i>sched_conf</i> (5) で説明している形式で現在のスケジューラ構成を表示します。

表 4 qconf のオプション (続き)

オプション	説明
-sstnode <i>node_path</i> ,...	基本割当ツリーノードの表示 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定された基本割当ツリーノード (複数指定可能) の名前と配分を表示します ( <i>share_tree</i> (5) を参照)。 <i>node_path</i> は基本割当ツリー内のノードの位置を示す階層形式のパス ( <i>/[node_name[/.]node_name...]</i> ) です。
-sstree	基本割当ツリーの表示 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。基本割当ツリーの定義を表示します ( <i>share_tree</i> (5) を参照)。
-sss	スケジューラのステータスの表示 - Sun Grid Engine スケジューラがアクティブなホストを表示します。スケジューラが動作していない場合は、エラーメッセージが表示されます。
-su <i>acl_name</i>	ユーザー ACL の表示 - Sun Grid Engine のユーザーアクセス制御リスト (ACL) を表示します。
-sul	ユーザーリストの表示 - 定義済みのすべての Sun Grid Engine ユーザーアクセス制御リスト (ACL) のリストを表示します。

表 4 qconf のオプション (続き)

オプション	説明
-suser <i>user</i> ,...	ユーザーの表示 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定されたユーザー (複数指定可能) の定義を表示します ( <i>user</i> (5) を参照)。
-suserl	ユーザーの表示 - このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。定義済みのすべてのユーザーのリストを表示します。
-tsm	スケジューラ監視の開始 - このオプションを使用すると、Sun Grid Engine スケジューラの <i>sge_schedd</i> (8) が強制的に次のスケジューリングのトレースメッセージをファイル <code>&lt;sge_root&gt;/&lt;cell&gt;/common/schedd_r unlog</code> に出力します。メッセージには、そのスケジューリングでジョブおよびキューが選択されなかった理由が示されます。 <i>root</i> またはマネージャー特権が必要です。

注 - キューの資源可用性の面からジョブ要求が無効になった理由が、*qstat* (1) -F オプションで説明している形式で表示されます (*qstat* (1) のマニュアルページの「出力形式」の節の「完全形式」の説明を参照)。

## 環境変数

表 5 は、qconf に関する環境変数をまとめています。

表 5 qconf 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qconf は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することもできます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qconf クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 制限事項

キュー構成に対する変更がアクティブなキューに影響することはありません。次のそのキューを起動すると (すなわち、次のジョブから)、変更が有効になります。

## ファイル

- Sun Grid Engine のマスターホストファイル -  
`<sge_root>/<cell>/common/act_qmaster`

## 関連項目

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)

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## qdel(1)

### 名前

`qdel` - キューから Sun Grid Engine ジョブを削除します。

### 構文

```
qdel [ -f ] [ -help ] [ -verify ] [ job/task_id_list ]
```

```
qdel [ -f ] [ -help ] [ -verify ] -u user_list | -uall
```

### 説明

ユーザー、オペレータ、マネージャーは `qdel` を使用して、ジョブを削除することができます。`qdel` は、ジョブ識別子が指定された順にジョブを削除します。

# オプション

表 6 は qdel のオプションをまとめています

表 6 qdel のオプション

オプション	説明
-f	実行中のジョブに対して強制的な処理を行います。ジョブを制御している <code>sge_execd(8)</code> が <code>sge_qmaster(8)</code> からの削除要求に応答しなくても、 <code>sge_qmaster(8)</code> に登録されているジョブのリストからジョブを削除します。クラスタ構成エントリの <code>qmaster_params</code> にフラグ <code>ENABLE_FORCED_QDEL</code> が含まれている場合は、Sun Grid Engine マネージャあるいはオペレータ以外のユーザーでも自分のジョブに <code>-f</code> オプションを使用することができます。ただし、管理者であるかどうかによって動作は異なります。管理者の場合は、Sun Grid Engine データベースからただちにジョブが削除されます。管理者以外のユーザーの場合は、最初に通常の削除が試みられ、失敗した場合にのみ強制的な取り消しが実行されます。
-help	全オプションの一覧を表示します。
-u <i>username</i> ,...   -uall	<i>username</i> リストに指定されたユーザーが実行依頼したジョブだけ削除します。マネージャの場合は、 <code>qdel -uall</code> コマンドを使用してすべてのユーザーのすべてのジョブを削除できます。 <code>-u</code> または <code>-uall</code> スイッチを使用すると、 <i>job/task_id_list</i> も指定できます。
-verify	実際の変更を行わずに <code>-verify</code> が存在しない場合に行われることの表示だけを行います。
<i>job/task_id_list</i>	次の形式で指定します。 <code>job_id[.task_range][,job_id[.task_range],...]</code> この指定がある場合、 <i>task_range</i> は、 <code>qdel</code> の処理対象を、ジョブ ID の接尾辞として指定された配列ジョブのタスク範囲に限定します (配列ジョブについての詳細は、 <code>qsub(1)</code> の <code>-t</code> オプションを参照)。タスク範囲は、 <code>n[-m[:s]]</code> の形式で指定します。範囲として、単一の番号、 <code>n-m</code> の形式の簡単な範囲、ステップ付きの範囲を指定することができます。 <i>job/task_id_list</i> 代わりにキーワードの <code>all</code> を使用して、現在のユーザーのすべてのジョブを変更することができます。



## 環境変数

表 7 は、qdel に関する環境変数をまとめています。

表 7 qdel 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qdel は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qdel クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

- Sun Grid Engine のマスターホストファイル -  
`<sge_root>/<cell>/common/act_qmaster`

## 関連項目

sge\_intro(1)、qstat(1)、qsub(1)、sge\_qmaster(8)、sge\_execd(8)

## 著作権

著作権および許諾条件全文については、sge\_intro(1) を参照してください。

---

# qhold(1)

## 名前

qhold - Sun Grid Engine のジョブの実行を保留します。

## 構文

```
qhold [ -h {u|o|s},... ] [ -help ] [ job/task_id_list ]
```

```
qhold [ -h {u|o|s},... ] [ -help ] -u user_list | -uall
```

## 説明

qhold を使用して、ユーザー、オペレータ、マネージャーは、実行予定待ちのジョブをいわゆるホールドすることができます。その種類に関係なく、ジョブにホールドが設定されている限り、そのジョブが実行対象になることはありません。

ホールドは、qrs1(1) または qalter(1) コマンドで解除することができます。

ホールドは次の 3 種類あります。

- **ユーザー** - ユーザーホールドは、マネージャーやオペレータ、ジョブの所有者が割り当てたり、割り当て解除したりできます。
- **オペレータ** - オペレータホールドは、マネージャーとオペレータが割り当てたり、割り当て解除したりできます。
- **システム** - システムホールドは、マネージャーだけが割り当てたり、割り当て解除したりできます。

-h オプション (下記を参照) にホールドの種類が指定されなかった場合は、デフォルトでユーザーホールドとみなされます。

qsub(1) や qalter コマンドを使用してジョブにホールドを割り当てることもできます (-h オプションを参照)。

# オプション

表 8 は qhold のオプションをまとめています。

表 8 qhold のオプション

オプション	説明
-h {u o s},...	u (ユーザー) か o (オペレータ)、s (システム) ホールドまたはその組み合わせのホールドをジョブに割り当てます。
-help	全オプションの一覧を表示します。
-u <i>username</i> ,...   -uall	<i>username</i> リストに指定されたユーザーが実行依頼したジョブにだけ変更を加えます。マネージャーの場合は、qhold -uall コマンドを使用してすべてのユーザーのすべてのジョブにホールドを設定できます。-u または -uall スイッチを使用すると、 <i>job/task_id_list</i> も指定できます。
<i>job/task_id_list</i>	次の形式で指定します。 <i>job_id</i> [ <i>task_range</i> ] [ <i>job_id</i> [ <i>task_range</i> ],...] この指定がある場合、 <i>task_range</i> は、qhold の処理対象を、ジョブ ID の接尾辞として指定された配列ジョブのタスク範囲に限定します (配列ジョブについての詳細は、qsub(1) の -t オプションを参照)。 タスク範囲は、n[-m[:s]] の形式で指定します。範囲として、単一の番号、n-m の形式の簡単な範囲、ステップ付きの範囲を指定することができます。 <i>job/task_id_list</i> 代わりにキーワードの all を使用して、現在のユーザーのすべてのジョブのホールド状態を変更することができます。

## 環境変数

表 9 は、qhold に関する環境変数をまとめています。

表 9 qhold 関係の環境変数

オプション	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qhold は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qhold クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 関連項目

sge\_intro(1)、qalter(1)、qrls(1)、qsub(1)

## 著作権

著作権および許諾条件全文については、sge\_intro(1) を参照してください。

---

# qghost (1)

## 名前

qghost - Sun Grid Engine のホスト、キュー、ジョブのステータスを表示します。

## 構文

```
qghost [ -F [resource_name,...] [ -help ] [ -h host_list ]  
[ -j ] [ -l resource=val,... ] [ -u user,... ]
```

## 説明

qghost は、使用可能な Sun Grid Engine のホスト、キューとキューに関連づけられているジョブの現在のステータスを表示します。選択オプションを使用して、特定のホストかキュー、ジョブ、ユーザーに関する情報を取得することができます。オプションなしで qghost を使用すると、キューまたはジョブ情報なしですべてのホストの一覧が表示されます。

## オプション

表 10 は qghost のオプションをまとめています。

表 10 qghost のオプション

オプション	説明
-F [ resource_name,... ]	qghost は、オプション引数が省略された場合は、あらゆる資源、オプション引数が指定された場合は、その resource_name リストに含まれる資源の現在の可用性を示す、ホストごとの詳細なリストを表示します。詳細は、下記の「出力形式の」の節の「完全形式」の説明を参照してください。
-help	全オプションの一覧を表示します。
-h host_list	host_list に含まれる全ホストのリストを表示します。

表 10 qghost のオプション (続き)

オプション	説明
-j	示されたホストがホストになっているキューで実行中のすべてのジョブを表示します。このスイッチは <code>-q</code> を暗黙で呼び出します。
-l <i>resource[=value],...</i>	情報が要求されたホストの必要とする資源を定義します。ホストに対して一致検索が行われます。
-q	表示されたホストがホストになっているキューの情報を表示します。
-u <i>user,...</i>	指定されたユーザーリストに含まれるユーザーに関連づけられているジョブおよびキューに関する情報だけを表示します。

## 出力形式

-q、-F、-j オプションの有無によって、3通りの出力形式があります。

### デフォルトの形式

デフォルトの形式、すなわち、-q、-F、-j オプションのどれも指定されなかった場合は、ヘッダー行に続いて、1つのホストに1行の長さで以下の情報が表示されます。

- ホスト名
- アーキテクチャ
- プロセッサ数
- 負荷
- 総メモリー
- 使用メモリー
- 総スワップ空間
- 使用スワップ空間

### オプションが指定された場合の形式

-q オプションが指定された場合は、各ホストのステータスとして、上記の情報の他に、そのホストがホストになっているすべてのキューに関する情報からなる追加の行が表示されます。

- キュー名
- キューの種類 - B (バッチ)、I (対話形式)、C (チェックポイント)、P (並列)、T (転送) のいずれかか、その組み合わせ
- 使用ジョブスロット数と使用可能なジョブスロット数

- キューの状態 - u (不明)、a (アラーム)、A (アラーム)、C (カレンダーに基づく一時停止)、s (一時停止)、S (従属)、d (使用不可)、D (使用不可)、E (エラー) のいずれかか、その組み合わせ。状態が u (不明) の場合は、対応する `sge_execd(8)` にアクセスできないことを意味します。

a (アラーム) の場合は、キューが、その構成 (`queue_conf(5)` を参照) の `load_thresholds` リストに定義されている負荷しきい値の少なくとも 1 つを上回っていて、新たにジョブをスケジューリングできない状態であることを意味します。

これに対し A (アラーム) は、キューがその一時停止しきい値の少なくとも 1 つを上回った状態であることを示します (`queue_conf(5)` を参照)。この結果、そのキューで実行中のジョブは、しきい値違反がなくなるまで連続して一時停止されま

す。状態の s (一時停止) と d (使用不可) は、`qmod(1)` コマンドを使用してキューに割り当てたり、割り当て解除したりできます。キューを一時停止すると、そのキューで実行中のすべてのジョブが一時停止されます。

状態の D (使用不可) と C (カレンダーに基づく一時停止) は、それぞれキューが使用負荷状態であるか、Sun Grid Engine のカレンダー機能を使用して自動的に一時停止されていることを意味します (`calendar_conf(5)` を参照)。これに対して S (従属) の状態は、キューが、別のキューに従属しているために一時停止されていることを意味します (詳細は、`queue_conf(5)` を参照)。原因に関係なくキューが一時停止されると、そのキューで実行中のあらゆるジョブも一時停止されます。

キューに E (エラー) 状態が表示された場合は、そのキューのホスト上の `sge_execd(8)` が、ジョブを開始する際、同じホストにある `sge_shepherd(8)` 実行可能ファイルを検出できなかったことを意味します。この問題を解決する方法の手がかりを得るには、その `sge_execd(8)` のエラーログファイルを調べてください。問題を解決したら、`qmod(1)` コマンドの `-c` オプションを使用し手動でキューを使用可能にします。

`-F` オプションが使用された場合は、ホストのステータス行に続いて、資源が使用可能かどうかの情報が表示されます。1 つの資源 (`-F` のオプション引数で選択されている資源か、オプション引数が省略されている場合はすべての資源) について 1 行の長さで、次の情報が表示されます。

- 現在の資源可用性値が次のどちらの種類の値であるかを示す英字 1 字
  - g - グローバル (クラスタ全体)
  - h - ホスト全体
- 現在の資源可用性値の情報源を示す英字 1 字
  - l - 資源の負荷値報告
  - L - 管理者定義の負荷スケーリングの適用後の資源の負荷値
  - c - 消費可能資源機能から得られた可用性 (`complexes(5)` を参照)
  - v - 負荷報告または消費可能資源更新で書き換えられていないデフォルトの複合構成値
  - f - 非消費資源複合属性または固定資源制限から得られた固定可用性の定義

- 情報が表示されている資源名 (コロンの後)
- 現在の資源可用性値 (等号の後)

可用性値とその情報源として、可能なあらゆる組み合わせのうちの最小値がつねに表示されます。このため、たとえば「`qf:h_vmem=4G`」は、現在 4G バイトまでの仮想メモリーを使用可能なキューが存在し、この値は固定値で (たとえばキュー構成の資源制限値)、キュー主体の値であること、すなわち、ホスト全体ではもっと多くの仮想メモリーが使用可能である可能性があるが、キューが許可する上限は 4G バイトであることを示します。これに対し「`hl:h_vmem=4G`」という行も、4G バイトまで仮想メモリーが使用可能であることを示しますが、この上限値はホストに関する負荷報告から得られたものであることを示します。つまり、キューはもっと多くの仮想メモリーを要求しているジョブを受け付けられるかもしれないものの、そのキューが存在しているホストで現在使用可能なのは 4G バイトだけであることとなります。

キューのステータス行 (-j の場合) の後には、そのキューで実行中のジョブごとに 1 行の情報が表示されます。このジョブステータス行には、以下の情報が含まれます。

- ジョブ ID
- ジョブ名
- ジョブの所有者名
- ジョブのステータス - `t` (転送中)、`r` (実行中)、`R` (再開)、`s` (一時停止中)、`S` (一時停止中)、`T` (しきい値) のいずれか (詳細は、74 ページの「簡略形式 (-f および -F なし)」の節を参照)。
- ジョブの開始日時と役割 (MASTER または SLAVE で、並列ジョブの場合にのみ意味がある)
- ジョブの優先順位



## 環境変数

表 11 は、qghost に関する環境変数をまとめています。

表 11 qghost 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qstat は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qghost クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

- Sun Grid Engine のマスターホストファイル -  
<sge\_root>/<cell>/common/act\_qmaster

## 関連項目

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)

## 著作権

著作権および許諾条件全文については、sge\_intro(1) を参照してください。

---

# qmake (1)

## 名前

qmake - Sun Grid Engine, Enterprise Edition によってスケジューリングされる分散  
並列 make

## 構文

qmake [ *options* ] -- [ *gmake options* ]

## 説明

qmake は、分散並列版の make (1) ユーティリティです。並列 make のタスクは、Sun Grid Engine ソフトウェアによってスケジューリングされます。qmake は、gmake (GNU make) のバージョン 3.78.1 に基づいています。Sun Grid Engine と gmake コマンド行オプションの両方を使用することができます。両者のオプションは -- 文字で区切ります。

qmake では、qsub (1) または qrsh (1) で有効な、Sun Grid Engine のすべてのオプションを使用することができます。Sun Grid Engine のすべてのコマンド行オプションについては、submit (1) を参照してください。gmake コマンド行構文については、make (1) のマニュアルページに説明があります。

qmake のメイクファイルの構文は gmake に対応しており、『GNU Make Manual』に説明があります。

一般的な qmake の呼び出しでは、Sun Grid Engine のコマンド行オプションの -cwd を使用して、Sun Grid Engine 環境が自分の .cshrc または .profile シェルリソースファイルに設定されていない場合に実行ホスト -v *path* の現在の作業ディレクトリで、スケジューリングされている make を開始し、並列環境にあるスロットを要求します。

資源要求が指定されていない場合 (QS\_NAME *commandline\_option* -1)、qmake は ARCH 環境変数を使用することによって、タスク実行用のアーキテクチャとして実行依頼ホストと同じアーキテクチャを要求します。ARCH が設定されている場合は、コマンド行オプションに -l arch=\$ARCH オプションを挿入することによって、その変数に指定されているアーキテクチャが要求されます。ARCH が設定されていない場

合、make のタスクは使用可能な任意のアーキテクチャ上で実行することができます。一般的な make (コンパイル) ジョブではアーキテクチャが重大に意味をもつため、警告が出力されます。

## 使用例

次のコード例 1 では、実行依頼ホストと同じアーキテクチャ上の並列環境 `compiling` にあるスロットを 1 から 10 個使用するように要求しています。make タスクは呼び出し元のシェルと完全な環境を継承し、Sun Grid Engine ソフトウェアが付与したスロットと同じ数の並列タスクとして実行されます。

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

### コード例 1

次のコード例 2 では、実行依頼ホストと同じアーキテクチャ上の並列環境 `make` にあるスロットを 1 から 4 個使用するように要求しています。

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

### コード例 2

次のコード例 3 では、アーキテクチャ `solaris` のホストで 3 つの配列 make タスクを実行するように要求しています。実行依頼は、任意のアーキテクチャのホストから行うことができます。

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

### コード例 3

次のコード例 4 はシェルスクリプトです。

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

### コード例 4

コード例 4 のシェルスクリプトは、コード例 5 の形式を使用することによって実行依頼することができます。

```
qsub -cwd -v PATH -pe make 1-10 [further_options] <スクリプト>
```

#### コード例 5

コード例 5 では qmake は、並列環境 make で、実行依頼されたジョブに付与された資源を継承します。

## 環境変数

表 12 は、qmake に関する環境変数をまとめています。

表 12 qmake 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qmake は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
ARCH	実行依頼ホストのアーキテクチャ。実行依頼環境にこの変数が設定されている場合、qmake はジョブの実行用として指定されているアーキテクチャを要求します (上記の説明を参照)。
COMMD_PORT	設定されている場合は、sge_commd (8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qmake クライアントとの通信に使用する特定の sge_commd (8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 判明している問題

qmake コマンド関係で判明している問題は以下のとおりです。

## NFS サーバーの速度の問題

ファイルサーバーのパフォーマンスが非常に低いと、依存ファイルで問題が発生する可能性があります。

たとえば、ホスト a が a.c. をコンパイルして a.o. を生成し、ホスト b が b.c. をコンパイルして b.o. を生成して、ホスト c が a.o. と b.o. をリンクしてプログラム c を生成すると仮定します。NFS のパフォーマンスが非常に悪い場合、ホスト c がファイル a.o. および b.o. をすぐに得られない可能性があります。

### 1 つの規則で複数のコマンドを実行する際の問題

1 つの規則で複数のコマンドを実行する場合、メイクファイルは、それらのコマンドが 1 つのコマンド行で処理されるようにする必要があります。

たとえば以下を入力したと仮定します。

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

この例の場合、コマンドが並列 (場合によって異なる異なるホスト) で実行されると、libx.a の構築が失敗します。このような場合は、次のように入力します。

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

もう 1 つの有効な方法は、次のように入力することです。

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

## 関連項目

submit(1)、sge\_pe(5)、make(1) (GNU make のマニュアルページ)、『The GNU Make Manual』(<sge\_root>/3rd\_party/qmake にある)

## 著作権

qmake には、一部、Free Software Foundation, Inc. (米国マサチューセッツ州ボストン) が著作権を保有し、Gnu General Public License で保護されている GNU Make (gmake) のコードが含まれています。

他の権利および許諾条件文については、`sge_intro(1)` と `<sge_root>/3rd_party/qmake` に提供されている情報を参照してください。

---

## qmod (1)

### 名前

qmod - Sun Grid Engine のキューを変更します。

### 構文

qmod [ *options* ] [ *jobtask\_id\_list* | *queue\_list* ]

### 説明

ワークステーションの所有者として分類されるユーザーは (詳細は、`queue_conf(5)` を参照)、qmod を使用して、自分のジョブの状態ばかりでなく、使用しているマシンの Sun Grid Engine キューの状態も変更することができます。マネージャー / オペレータあるいは root ユーザーは、クラスタ内のあらゆるキューおよびジョブに対して qmod を使用することができます。

# オプション

表 13 は qmod のオプションをまとめています。

表 13 qmod のオプション

オプション	説明
-c	指定されたキューのエラー状態をクリアします。
-d	キューを使用不可に、すなわち、それ以上ジョブがキューにディスパッチされないようにします。キューですでに実行中のジョブは、最後まで実行することができます。このオプションは、Sun Grid Engine バージョン 3 の <code>-soc</code> オプションの後継です。
-e	キューを使用可能にします。このオプションは、Sun Grid Engine バージョン 3 の <code>-xsoc</code> オプションの後継です。
-f	外見上のキューの現在の状態に関係なく、キューに対して強制的な変更を行います。たとえばキューが一時停止したにもかかわらず、ジョブの実行が継続しているように見える場合、マネージャー / オペレータは強制的な一時停止 (ジョブに SIGSTOP が送信される) を行うことができます。この場合は、キューまたはジョブを制御している <code>sge_execd(8)</code> にアクセスできなくても、強制的にキューまたはジョブのステータスが設定されます。マネージャー / オペレータ特権が必要です。
-help	全オプションの一覧を表示します。
-r	キューに使用された場合は、そのキューで実行中のすべてのジョブがスケジューリングし直されます。実行中のジョブに使用された場合は、そのジョブがスケジューリングし直されます。
-s	キューに使用された場合は、キューおよびアクティブなすべてのジョブを一時停止します。実行中のジョブに使用された場合は、そのジョブを一時停止します。ジョブが明示的に、かつそのキューの一時停止によっても一時停止されている場合、その後、キューを停止解除しても、ジョブの一時停止状態は解除されません。

表 13 qmod のオプション (続き)

オプション	説明
-us	キューに使用された場合は、キューおよびアクティブなすべてのジョブを停止解除します。ジョブに使用された場合は、そのジョブを停止解除します。ジョブが明示的に、かつそのキューの一時停止によっても一時停止されている場合、その後、キューを停止解除しても、ジョブの一時停止状態は解除されません。
-verify	実際の変更を行わずに <code>-verify</code> が存在しない場合に行われることの表示だけを行います。
<code>job/task_id_list</code>   <code>queue_list</code>	<p>qmod の処理対象にするジョブまたはキューを指定します。 <code>job/task_id_list</code> は、以下のいずれかの形式で指定します。</p> <ul style="list-style-type: none"> <li>• <code>job_id[.task_range][.job_id[.task_range],...]</code></li> <li>• <code>job_id[.task_range][ job_id[.task_range] ...]</code></li> </ul> <p>この指定がある場合、<code>task_range</code> は、qmod の処理対象を、ジョブ ID の接尾辞として指定された配列ジョブのタスク範囲に限定します (配列ジョブについての詳細は、<code>qsub(1)</code> の <code>-t</code> オプションを参照)。タスク範囲は、次のいずれかの形式で指定します。<code>n[-m[:s]][n[-m[:s]], ...]</code> <code>n[-m[:s]][ n[-m[:s]] ...]</code> つまり、タスク範囲は、コンマまたは空白で <code>n[-m[:s]]</code> を区切って指定します。指定された範囲を連結することによってタスク ID 範囲全体が構成されます。範囲として、単一の番号、<code>n-m</code> の形式の簡単な範囲、ステップ付きの範囲を指定することができます。</p>

## 環境変数

表 14 は、qmod に関する環境変数をまとめています。

表 14 qmod 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	<p>設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qmod は以下を使用します (優先順)。</p> <ul style="list-style-type: none"> <li>• 環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li> <li>• デフォルトのセル名 (例: default)</li> </ul>



表 14 qmod 関係の環境変数 (続き)

変数名	説明
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を <code>stderr</code> に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、 <code>sge_commd(8)</code> が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qmod クライアントとの通信に使用する特定の <code>sge_commd(8)</code> が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

- Sun Grid Engine のマスターホストファイル -  
`<sge_root>/<cell>/common/act_qmaster`

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## qmon(1)

### 名前

`qmon` - Sun Grid Engine の X-Windows OSF/Motif グラフィカルユーザーインタフェース

# 構文

qmon [ *options* ]

## 説明

管理者およびユーザーは、qmon を使用し、X-Window 環境から Sun Grid Engine システムを操作することができます。qmon は、複数の方法で連携されたさまざまなダイアログボックスを提供します。qmon を使用してユーザーが行う作業ごとに対応するダイアログボックスが用意されています。ある 1 つの作業用の適切なダイアログボックスにアクセスする方法は複数あります。

- 起動時に最初に表示される qmon メインウィンドウに、主要な管理およびユーザー作業用のアイコンボタンが含まれています。アイコン上にマウスポインタを置くと、そのアイコンの機能に関するツールチップが表示されます。
- qmon メインウィンドウのメニューに「タスク」プルダウンメニューボタンが表示されます。このボタンをクリックすると、行うことができる作業の一覧が開きます。項目を選択すると、対応するダイアログボックスが開きます。
- 「タスク」プルダウンメニューにはまた、キーボードから特定のキーシーケンスを押すことによってメインウィンドウから直接、対応する作業のダイアログボックスを起動することが可能なキーアクセラレータが含まれています。
- 特定のダイアログボックスからそのサブダイアログ階層を辿ることができ、他のダイアログボックスとの間に明らかな関連性がある場合は、必ず、そのダイアログボックスへのリンクが用意されています。リンクがあることを示すボタンをクリックすると、別のダイアログボックスが開きます。

# オプション

サポートされているオプションは、x(1) の「オプション」の節で説明されている X Toolkit 標準のオプションです。また、qmon は表 15 に示すオプションもサポートしています。

表 15 qmon のオプション

オプション	説明
-cmap	qmon の個人用カラーマップをインストールします。このオプションは、他のアプリケーションがすでに多数の色を割り当てているため、qmon から対応するエラーメッセージが表示される場合に使用すると有用なことがあります。ただし、個人用のカラーマップを使用すると、qmon のウィンドウを起動または終了するたびにカラーマップの切り替えが発生します。
-fontFamily {big medium small}	解像度によってサイズの異なるフォントを使用するように qmon に指示します。
-help	使用方法を表示します。
-nologo	ロゴマークなしで起動します。

# ダイアログボックス

ここでは、qmon コマンド関係のダイアログボックスについて説明します。

## ジョブ制御

「ジョブ制御」ダイアログボックスは、保留中のジョブや実行中のジョブ、最近完了したジョブの表形式のリストからなるフォルダです。ジョブに関する詳細な情報を提供するばかりでなく、ジョブを選択して、削除したり、一時停止したりすることができます。また、「実行依頼」ダイアログボックスへのリンクが用意されており、新規ジョブの実行依頼をしたり、保留中のジョブの属性を変更（「Qalter」ボタン）したりできます。表に表示するフィールドおよび表示するジョブは、「カスタマイズ」ボタンをクリックすることによってカスタマイズすることができます。このカスタマイズ内容は `~/.qmon_preferences` ファイルに保存することができ、保存すると、「ジョブ制御」ダイアログボックスの初期構成として以降の起動から使用されます。

## キュー制御

「キュー制御」ダイアログボックスとそのサブダイアログ階層では、システムに実際に構成されている **Sun Grid Engine** キューの状態を制御したり、新しいキューを追加したり、既存のキューを変更または削除したりできます。最上位の「キュー制御」ダイアログボックスのアイコンボタンはそれぞれ、構成済みの **Sun Grid Engine** キューを表します。ボタンのアイコンの記号、色分け、テキストによって、アーキテクチャとキューのステータス、いくつかの基本的な属性の情報が得られます。また、選択されているキューを削除することもできます。キューは、マウスの左ボタンでキューのアイコンボタンをクリックするか、長方形でボタンを囲むことによって選択します。

「追加」または「変更」ボタンをクリックするか、最上位の「キュー制御」ダイアログボックスのアイコン表示区画でマウスの右ボタンをクリックしたときに表示されるポップアップメニューを使用すると、**Sun Grid Engine** のキューを構成するためのサブダイアログが表示されます。変更を行うには、まずキューを選択する必要があります。構成用のサブダイアログでは、キューとホスト名を定義したり、対応する名前を表示（変更の場合）したりできます。キュー構成パラメータ (`queue_conf(5)` を参照) はいくつかのカテゴリに分かれており、これらのカテゴリは、キュー構成用のサブダイアログの下部に表示されたタブウィジェットを使用して選択することができます。キュー構成パラメータのカテゴリとしては、一般構成、実行方法、チェックポイント、負荷 / 一時停止しきい値、制限、複合、ユーザーアクセス、プロジェクトアクセス (**Sun Grid Engine, Enterprise Edition** のみ)、従属キュー、所有者があります。管理者は、「クローン作成」ボタンを使用することによって、構成済みのキューからデフォルトの値を選択することができます。「了解」ボタンをクリックすると、定義が `sge_qmaster(8)` に登録されます。「キュー制御」ダイアログボックスは、「ジョブ制御」ダイアログボックスと同様の方法でカスタマイズすることができます。ここでの設定も `~/qmon_preferences` に保存されます。

## 実行依頼

「ジョブの実行依頼」ダイアログボックスは、バッチジョブや対話形式のジョブの実行依頼に使用します。上記の「ジョブ制御」ダイアログボックスから「**Qalter**」ボタンを使用して保留中のジョブの属性を変更する際にも呼び出されます。バッチジョブと対話形式のジョブの切り替えには、「ジョブの実行依頼」ダイアログボックスの右側にあるボタン欄の上にある「バッチ / 対話形式」ボタンを使用します。

「ジョブの実行依頼」ダイアログボックスは、ジョブの実行依頼準備用の 2 つのダイアログページからなるフォルダで構成されています。ジョブの実行依頼でよく使うパラメータは、「一般」ページにまとめられています。ジョブスクリプトの指定は必須で、それ以外のフィールドはすべて省略可能です。ジョブに高度な要求の指定が必要な場合は、「高度」タブを使用して、拡張パラメータの表示に切り替えることができます。

ジョブに必須の資源要求がある場合は、「資源要求」アイコンボタンをクリックして、「要求資源」サブダイアログを表示します。このサブダイアログでは、ジョブに必要な資源を選択したり、その資源の供給量を定義したりできます。「使用可能な資

源」は、要求可能と定義された複合属性によって構成されています (詳細は `complex(5)` を参照)。資源要求は、「ハード」、すなわち、キューでジョブを開始するには必ず満たされている必要がある要求と「ソフト」、すなわち、使用可能になったときに付与される要求があります。

「完了」ボタンをクリックして、「要求資源」サブダイアログを閉じると、指定されたジョブの要求が記録されます。最上位の「実行依頼」ダイアログボックスで「実行依頼」ボタンをクリックすると、ジョブの実行依頼が行われます。

## 複合構成

管理者は、「複合構成」ダイアログボックスを使用して新しい複合を追加したり、既存の複合を変更、削除したりできます (`complex(5)` を参照)。このダイアログボックスでは、既存の複合の選択リストが提供され、選択した複合の構成が表示されます。「削除」ボタンをクリックすると、選択されている複合が構成から削除されます。「追加 / 変更」ボタンをクリックすると、複合構成用のダイアログボックスが開き、このダイアログボックスから、新規複合を作成したり、既存の複合を変更したりできます。新しい複合を作成する場合は、その名前を定義する必要があります。変更の場合は、複合名が同じテキスト入力フィールドに表示されます。「複合構成」ダイアログボックスには、複合のエントリからなる表形式のリストと、新規または既存のエントリを定義するための入力フィールドがあります。「追加」ボタンは、新規または変更されたエントリで表形式のリストを更新し、「了解」ボタンは追加または変更された複合を `sge_qmaster(8)` に登録します。

## ホスト構成

「ホスト構成」ダイアログボックスでは、3つのホストリストを操作することができます。

- 管理ホスト
- 実行依頼ホスト
- 実行ホスト

操作するホストリストは、対応する名前の付けられたタブをクリックすることによって選択します。上記の最初の2つのホストリストでは、エントリの追加または削除を行えるだけです。リストに登録されているホストには、それぞれ管理または実行依頼権限が付与され、そうでないホストには付与されません。`complex(5)` で説明しているように、実行ホストリストのエントリには、負荷センサーに対するスケールリング係数や消費可能複合属性、アクセス属性 (アクセス、X アクセス、プロジェクト、X プロジェクトは Sun Grid Engine, Enterprise Edition のみ) を定義することもできます。Sun Grid Engine, Enterprise Edition システムでは、実行中のジョブに関して報告される CPU やメモリー、入出力の使用量をスケールリングすることもでき、資源能力係数を使用してホストの相対的なパフォーマンスを定義することもできます (`host_conf(5)` を参照)。

## クラスタ構成

「クラスタ構成」ダイアログボックスでは、クラスタ全体のグローバル構成とホスト別の構成を管理します (`sge_conf` (5) を参照)。このダイアログボックスを開くと、構成が割り当てられているすべてのホストの選択リストが表示されます。特殊ホスト名「`global`」は、クラスタ全体のグローバル構成を意味します。「追加 / 変更」ボタンをクリックすると、サブダイアログが開き、クラスタ構成を変更することができます。ホスト別の構成の場合は、「グローバル」関係の構成フィールドが使用不可になり、使用可能なパラメータだけを操作することができます。

## スケジューラ構成

「スケジューラ構成」ダイアログボックスでは、**Sun Grid Engine** スケジューラデーモンの `sge_schedd` (8) の動作を変更することができます。このダイアログボックスには、`sched_conf` (5) で説明しているすべてのスケジューラ構成パラメータが含まれています。ダイアログボックスは「一般パラメータ」と「負荷調整」の2つのセクションに分かれ、それらのセクションは対応するタブを使用して選択することができます。「了解」ボタンは、`sge_qmaster` (8) に変更内容を登録します。

## カレンダー構成

管理者は、「カレンダー構成」ダイアログボックスを使用して新しいカレンダーを追加したり、既存のカレンダーを変更、削除したりできます (`calendar_conf` (5) を参照)。このダイアログボックスでは、既存のカレンダーの選択リストが提供され、選択したカレンダーの構成が表示されます。「削除」ボタンをクリックすると、選択されているカレンダーが構成から削除されます。「追加 / 変更」ボタンをクリックすると、カレンダー構成用のダイアログボックスが開き、このダイアログボックスから、新規カレンダーを作成したり、既存のカレンダーを変更したりできます。「了解」ボタンは、新規または変更されたカレンダーを `sge_qmaster` (8) に登録します。

## ユーザー構成

ユーザーのアクセス権は、「ユーザー構成」ダイアログボックスを使用して制御します。このダイアログボックスには、タブウィジェットがあり、操作するアカウントのタイプを選択することができます。

- マネージャアカウントの構成
- オペレータアカウントの構成
- ユーザーセットの定義
- ユーザーアカウントの定義 (**Sun Grid Engine, Enterprise Edition** のみ)

マネージャまたはオペレータアカウントのリストに追加されたユーザーアカウントには、そのユーザーアカウントで **Sun Grid Engine** にアクセスしたときにそれぞれマネージャまたはオペレータとしての権限が付与されます。

ユーザーセットリストは、ホストやキュー、プロジェクト、クラスタ構成パラメータの `user_lists` および `xuser_lists` (`queue_conf`(5)、`project`(5)、`sgc_conf`(5) を参照) と組み合わせられて、ホストやキュー、プロジェクト (Sun Grid Engine, Enterprise Edition でのみ提供)、クラスタ全体に対するユーザーのアクセス権を制御します。ユーザーセットは、単にユーザー名と UNIX グループ名をまとめたものです。グループ名は、名前の先頭に @ 記号を付けることによって識別します。センタクルリストには、すでに定義されているユーザーセットが表示されます。これらのリストは、ユーザーセット定義ダイアログボックスを使用して変更したり、新規に作成したりできます。

Sun Grid Engine, Enterprise Edition システムでは、いわゆる業務優先ポリシーや一時優先ポリシーに必要なアクセスリスト (Sun Grid Engine システムの使用法と同じ) や部署、あるいはその両方としてユーザーセットを使用することができます (この後の「チケット構成」を参照)。

Sun Grid Engine, Enterprise Edition システムではまた、システムへのアクセス権を持つアカウントを、Sun Grid Engine, Enterprise Edition ユーザーデータベースのエントリとして追加する必要があります (`user`(5) を参照)。これは、「ユーザー」サブダイアログを使用して行うことができます。

ダイアログボックスの左側にあるボタン欄の「チケット」ボタンをクリックすると、「チケット構成」ダイアログボックスが開きます (この後で説明)。これもまた、Sun Grid Engine, Enterprise Edition システムでのみ使用することができます。

## 並列環境構成

「並列環境構成」ダイアログボックスでは、並列環境 (PE) インタフェースを構成することができます。PVM (Parallel Virtual Machine) や MPI (Message Passing Interface)、共有メモリー並列システムなどの並列プログラミング環境をインスタンス化したり、それら並列環境へのアクセスを制限したりする方法を定義するには、並列環境インタフェースが必要です。このダイアログボックスを開くと、構成済みの並列環境のリストとともに、選択状態の並列環境インタフェースの現在の構成が表示されます (`pe_conf`(5) を参照)。新しい並列環境インタフェースを追加したり、既存の並列環境インタフェースを変更したりするには、それぞれ「追加」または「変更」ボタンを使用して、並列環境インタフェース構成用のサブダイアログを開きます。

「了解」ボタンを使用して変更を適用し、サブダイアログを終了すると、その新しいまたは変更された並列環境インタフェースが `sgc_qmaster`(8) に登録されます。

## チェックポイント構成

「チェックポイント環境構成」ダイアログボックスでは、チェックポイント環境インタフェースを構成することができます。Sun Grid Engine がサポートするさまざまなオペレーティングシステムプラットフォームによって、チェックポイントの生成方法は異なります。そうしたチェックポイントの生成方法の属性を定義するには、チェックポイント環境が必要です。このダイアログボックスを開くと、構成済みのチェック

ポイント環境のリストとともに、選択状態のチェックポイント環境インタフェースの現在の構成が表示されます (checkpoint (5) を参照)。新しいチェックポイント環境インタフェースを追加したり、既存のチェックポイント環境インタフェースを変更したりするには、それぞれ「追加」または「変更」ボタンを使用して、チェックポイント環境インタフェース構成用のサブダイアログを開きます。「了解」ボタンを使用して変更を適用し、サブダイアログを終了すると、その新しいまたは変更されたチェックポイント環境インタフェースが `sge_qmaster` (8) に登録されます。

## チケット構成

「チケット構成」ダイアログボックスは、基本割当や業務優先、一時優先スケジューリングポリシーへのチケット割り当ての概要および編集画面を提供します。この機能は、**Sun Grid Engine, Enterprise Edition** システムでのみ使用できます。**Sun Grid Engine** システムでは使用できません。

「締め切り優先ジョブ」ボタンは、「ユーザー構成」ダイアログボックスを開きます。「ユーザーセット」サブダイアログに切り替えて、「`deadlineusers`」という名前のユーザーセットを選択してください。このユーザーセットのユーザーだけが、締め切り優先ジョブの実行依頼をすることができます。

「基本割当ポリシー」ボタンは、**Sun Grid Engine, Enterprise Edition** の基本割当ツリーを作成・編集するためのダイアログボックスを開きます (構成パラメータについては、`share_tree` (5) および `schedd_conf` (5) を参照)。

「業務優先ポリシー」ボタンは、業務優先配分の新規割り当てや割り当て変更のためのダイアログボックスを開きます (さまざまな種類の業務優先配分と設定可能な重みパラメータについては、「`sched_conf` (5)、`access_list` (5)、`project` (5)、`queue_conf` (5)、`user` (5) を参照)。

「一時優先ポリシー」ボタンは、一時優先チケットの新規割り当てや割り当て変更のためのダイアログボックスを開きます (さまざまな種類の一時優先チケットについては、「`access_list` (5)、`project` (5)、`queue_conf` (5)、`user` (5) を参照)。

## プロジェクト構成

「プロジェクト構成」ボタンは、プロジェクトを作成するためのダイアログボックスが開きます。この機能は、**Sun Grid Engine, Enterprise Edition** システムでのみ使用できます。**Sun Grid Engine** システムでは使用できません。

「プロジェクト構成」ダイアログボックスでは、既存のプロジェクトの選択リストが提供され、選択したプロジェクトの構成が表示されます。「削除」ボタンをクリックすると、選択されているプロジェクトが構成から削除されます。「追加 / 変更」ボタンをクリックすると、プロジェクト構成用のダイアログボックスが開き、このダイアログボックスから、新規プロジェクトを作成したり、既存のプロジェクトを変更し



たりできます。基本的にプロジェクト構成とは、ユーザーセットに対してプロジェクトへのアクセス権を付与または拒否することを意味します。「了解」ボタンは、新規または変更されたプロジェクトを `sgc_qmaster(8)` に登録します。

## ブラウザ

「オブジェクトブラウザ」ダイアログボックスの用途は、多岐にわたります。第一に、その出力区画に、行われた操作に関するエラーあるいは成功の通知などの、**Sun Grid Engine** および `qmon` のメッセージを表示することができます。また、`qmon` の標準出力や標準エラー出力を出力ウィンドウに送ることもできます。

また、他の `qmon` のダイアログボックスでアイコンまたは表のエントリとして表されているものの上にマウスポインタを置くことによって、オブジェクトブラウザで、**Sun Grid Engine** のオブジェクトに関する情報を連続的に表示することもできます。現在、オブジェクトブラウザでの構成の表示がサポートされているのは、それぞれ独立したダイアログボックスに表示される 2 種類の **Sun Grid Engine** オブジェクトだけです。

- 最上位の「キュー制御」ダイアログボックス内のキューのアイコン上にマウスポインタを置くと、キューの構成が表示されます。この機能は、「オブジェクトブラウザ」ダイアログボックスで「キュー」ボタンをクリックすると有効になります。
- 「ジョブ制御」ダイアログボックス内の実行中または保留中のジョブの行上にマウスポインタを置くと、そのジョブの詳細情報が表示されます。
- 「ジョブ制御」ダイアログボックスの「調査」ボタンをクリックすると、ジョブスケジューリング情報が表示されます。この場合は、暗黙で「ブラウザ」ダイアログボックスが開き、スケジューリング関連のすべての情報が表示されます。

## 終了

「終了」アイコンボタンはダイアログボックスに関連づけられていません。「終了」ボタンの目的は、アクティブな `qmon` のダイアログボックスをすべて閉じて、アプリケーションを終了することだけです。

## 資源

使用可能な資源とその意味、資源を変更する際の構文については、デフォルトの `qmon` リソースファイルに説明があります (リソースファイルがある場所については、下記の「ファイル」を参照)。

## 環境変数

表 16 は、qmon に関する環境変数をまとめています。

表 16 qmon 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qmon は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>• 環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>• デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qmon クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 制限事項

編集フィールドの幅を超える長さのテキストを入力した場合は、テキストがフィールドの終わりからはみ出た状態になります。

## ファイル

- qmon のリソースファイル例 - <sge\_root>/qmon/Qmon
- qmon のシステムリソースファイル - /usr/lib/X11/defaults/Qmon
- qmon のユーザーリソースファイル - \$HOME/Qmon
- qmon のジョブ / キューカスタマイズファイル - \$HOME/.qmon\_preferences

## 関連項目

sgc\_intro(1)、sgc\_conf(5)、access\_list(5)、sgc\_pe(5)、calendar\_conf(5)、complex(5)、project(5)、queue\_conf(5)、sched\_conf(5)、user(5)、sgc\_qmaster(8)

## 著作権

著作権および許諾条件全文、パブリックドメインおよびフリーウェアウィジェットの開発者へのクレジットについては、sgc\_intro(1) および、  
<sgc\_root>/3rd\_party/qmon に提供されている情報を参照してください。

---

## qrsls(1)

### 名前

qrsls - Sun Grid Engine, Enterprise Edition ジョブのホールド状態を解除します。

### 構文

```
qrsls [ -h {u|o|s},... ] [ -help ] [ job/task_id_list ]
```

```
qrsls [ -h {u|o|s},... ] [ -help ] -u user_list | -uall
```

### 説明

qrsls を使用して、ユーザー、オペレータ、マネージャーは、実行予定待ちのジョブのいわゆるホールド状態を解除することができます。その種類に関係なく、ジョブにホールドが設定されている限り、そのジョブが実行対象になることはありません。

ホールドは、qhold(1)、qsub(1)、qalter(1) コマンドを使用してジョブに割り当てることができます。

ホールドは次の 3 種類あります。

- ユーザー - ユーザーホールドは、マネージャーやオペレータ、ジョブの所有者が割り当てたり、割り当て解除したりできます。
- オペレータ - オペレータホールドは、マネージャーとオペレータが割り当てたり、割り当て解除したりできます。
- システム - システムホールドは、マネージャーだけが割り当てたり、割り当て解除したりできます。

-h オプション (表 17 を参照) にホールドの種類が指定されなかった場合は、デフォルトでユーザーホールドとみなされます。

qalter コマンドを使用してホールドを解除することもできます (表 17 の -h オプションを参照)。

## オプション

表 17 は qrls のオプションをまとめています。

表 17 qrls のオプション

オプション	説明
-h {u o s }...	ジョブの <b>u</b> (ユーザー) か <b>o</b> (オペレータ)、 <b>s</b> (システム) ホールドまたはその組み合わせのホールドを解除します。
-help	全オプションの一覧を表示します。
-u <i>username</i> ,...   -uall	ユーザー名リストに指定されているユーザーが実行依頼したジョブのホールド状態を変更します。マネージャーの場合は、 <b>qrls -uall</b> コマンドを使用してすべてのユーザーのジョブのホールド状態を解除できます。-u または -uall スイッチを使用した場合は、 <i>job/task_id_list</i> も指定できます。
<i>job/task_id_list</i>	次の形式で指定します。 <i>job_id[.task_range][,job_id[.task_range]]...</i> この指定がある場合、 <i>task_range</i> は、 <b>qrel</b> の処理対象を、ジョブ ID の接尾辞として指定された配列ジョブのタスク範囲に限定します (配列ジョブについての詳細は、 <b>qsub(1)</b> の -t オプションを参照)。タスク範囲は、 <i>n[-m[:s]]</i> の形式で指定します。範囲として、単一の番号、 <i>n-m</i> の形式の簡単な範囲、ステップ付きの範囲を指定することができます。 <i>job/task_id_list</i> 代わりにキーワードの <b>all</b> を使用して、現在のユーザーのすべてのジョブを変更することができます。

## 環境変数

表 18 は、`qrls` に関する環境変数をまとめています。

表 18 `qrls` 関係の環境変数

変数名	説明
<code>SGE_ROOT</code>	Sun Grid Engine の標準構成ファイルの場所を示します。
<code>SGE_CELL</code>	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、 <code>qrls</code> は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 <code>SGE_CELL</code> に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: <code>default</code>)</li></ul>
<code>SGE_DEBUG_LEVEL</code>	設定されている場合は、デバッグ情報を <code>stderr</code> に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
<code>COMMD_PORT</code>	設定されている場合は、 <code>sge_commd(8)</code> が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。
<code>COMMD_HOST</code>	設定されている場合は、Sun Grid Engine が <code>qrls</code> クライアントとの通信に使用する特定の <code>sge_commd(8)</code> が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# qselect (1)

## 名前

qselect - キューを選択します。

## 構文

```
qselect [ -help ] [ -l resource=val,... ] [ -pe pe_name,... ]  
[ -q queue,... ] [ -U user,... ]
```

## 説明

qselect は、表 19 で説明している qselect 引数に指定された選択条件に一致する Sun Grid Engine キュー名を一覧表示します。このコマンドの出力を他の Sun Grid Engine コマンドに供給することによって、特定のキューセットにアクションを適用することができます。たとえば qconf(1) の -mqattr オプションと qselect を組み合わせることによって、一群のキューのキュー属性を変更することができます。

## オプション

表 19 は qselect のオプションをまとめています。

表 19 qselect のオプション

オプション	説明
-help	全オプションの一覧を表示します。
-l resource [=value] ,...	キューリストの出力に含まれるキューが付与する資源を定義します。

表 19 qselect のオプション (続き)

オプション	説明
-pe <i>pe_name</i> ,...	コンマ区切りのオプション引数に列挙された並列環境の少なくとも 1 つに関連づけられているキューを出力に取り込みます。
-q <i>queue</i> ,...	出力に取り込むキューを直接指定します。このオプションが意味があるのは、-q で指定されたリストからキュー名の一部を抽出する、qselect のもう 1 つのオプションと組み合わせた場合だけです。
-U <i>user</i> ,...	指定したユーザーがアクセス権を持つキューを出力に取り込みます。

## 使用例

以下は、qselect コマンドの使用例です。

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

### コード例 6

コード例 6 のコマンドは、Linux マシンに存在するキュー名を表示します。

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

### コード例 7

コード例 6 を基にしたコード例 7 では、ユーザー *andreas* と *shannon* がアクセス権を持つキューだけが出力されます。

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

### コード例 8

コード例 8 では、Linux マシンに存在するキューのキュー属性 *h\_vmem* を 1G バイトに変更しています (-mqattr オプションについての詳細は [qconf \(1\)](#)、キュー構成エントリについての詳細は [queue\\_conf \(5\)](#) のマニュアルページをそれぞれ参照)。

## 環境変数

表 20 は、qselect に関する環境変数をまとめています。

表 20 qselect 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qselect は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>• 環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>• デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qselect クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

Sun Grid Engine のマスターホストファイル -  
<sge\_root>/<cell>/common/act\_qmaster

## 関連項目

sge\_intro(1)、qconf(1)、qmod(1)、qstat(1)、queue\_conf(5)、  
sge\_commd(8)

## 著作権

著作権および許諾条件全文については、sge\_intro(1) を参照してください。



---

# qstat (1)

## 名前

qstat - Sun Grid Engine のジョブおよびキューのステータスを表示します。

## 構文

```
qstat [ -ext ] [ -f ] [ -F [resource_name,...] ] [ -g d ] [ -help ]  
[ -j [job_list] ] [ -l resource=val,... ] [ -ne ]  
[ -pe pe_name,... ] [ -q queue,... ] [ -r ]  
[ -s {r|p|s|z|hu|ho|hs|hj|ha|h} [ + ] ] [ -t ] [ -U user,... ]  
[ -u user,... ]
```

## 説明

qstat は、使用可能な Sun Grid Engine のキューとキューに関連づけられているジョブの現在のステータスを表示します。選択オプションを使用して、特定のジョブ、キュー、またはユーザーに関する情報を取得することができます。オプションなしの場合、qstat は、キューのステータス情報を表示しないで、ジョブのリストだけを表示します。

# オプション

表 21 は qstat のオプションをまとめています。

表 21 qstat のオプション

オプション	説明
-alarm	キューのアラーム状態の理由を表示します。資源値としきい値を含む理由が 1 行に 1 つ出力されます。資源値についての詳細は、下記の「出力形式」の節の「完全形式」の説明を参照してください。
-ext	このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。各ジョブに関する Sun Grid Engine, Enterprise Edition 関連の追加情報を表示します (下記の「出力形式」を参照)。
-f	「完全」な形式で表示するよう指示します。すべてのキューについて、そのジョブリストとともに要約情報が表示されます。
-F [ resource_name,... ]	-f の場合同様、キューばかりでなくすべてのジョブに関する情報も表示します。qstat はまた、オプション引数が省略された場合は、あらゆる資源、オプション引数が指定された場合は、その resource_name リストに含まれる資源の現在の可用性を示す、キューごとの詳細なリストを表示します。詳細は、下記の「出力形式の」の節の「完全形式」の説明を参照してください。
-g d	配列ジョブの詳細情報を表示します。デフォルトでは、配列ジョブはグループ分けされ、同じステータスのすべてのタスク (保留中のタスクのみ) が 1 行に表示されます。出力の配列ジョブのタスク ID 範囲フィールドは、対応するタスク群を示します (「出力形式」の節を参照)。現在 -g スイッチに使用できるオプション引数は d だけです。他のオプション引数は、今後の拡張用として予約されています。
-help	全オプションの一覧を表示します。
-j [job_list]	保留中のすべてのジョブ、または job_list に指定されたジョブのスケジューリングされない理由を表示します。

表 21 qstat のオプション (続き)

オプション	説明
-l <i>resource</i> [= <i>value</i> ],...	情報を要求されたジョブが必要とする資源またはキューが付与する資源を定義します。キューに対して一致検索が行われます。保留中のジョブは、上記のキューのいずれかで実行可能なジョブに限定されます。
-ne	-f と組み合わせられた場合に、空のキューの表示を抑止します。このことは、実際には実行中のジョブのないすべてのキューが表示されないことを意味します。
-pe <i>pe_name</i> ,...	コンマ区切りのオプション引数に列挙された並列環境の少なくとも 1 つに関連づけられているキューのステータス情報を表示します。選択されたキューのいずれかで実行されているか、保留中でそうしたキューにスケジューリングされる可能性があるジョブのいずれかのステータス情報が表示されます。
-q <i>queue</i> ,...	ジョブの情報を表示するキューを指定します。
-r	表示されたジョブの資源要求に関する拡張情報を表示します。詳細は、「出力形式」の節の「拡張形式」の説明を参照してください。
-s { <i>p r s z hu ho hs hj ha h</i> } [+]	指定された状態のジョブの情報だけ表示します。ただし、状態は任意に組み合わせることができます。-s <i>prs</i> は、-s なしの通常の <i>qstat</i> の出力に相当します。最近完了したジョブを表示するには、-s <i>z</i> を使用します。ユーザー / オペレータ / システムホールド状態のジョブを表示するには、-s <i>hu/ho/hs</i> オプションを使用します。-s <i>ha</i> オプションは、 <i>qsub -a</i> コマンドを使用して実行依頼されたジョブを表示します。 <i>qstat -s hj</i> は、ジョブ依存関係リストにエントリがないにかかわらず、実行対象になっていないすべてのジョブを表示します ( <i>qsub</i> (1) の -a および <i>hold_jid</i> オプションを参照)。

表 21 qstat のオプション (続き)

オプション	説明
-t	表示された並列ジョブの制御対象のタスクに関する拡張情報を表示します。詳細は、「出力形式」の節の「拡張形式」の説明を参照してください。並列ジョブのタスクと配列ジョブのタスクは異なります (上記の -g オプションと qsub(1) の -t オプションを参照)。
-U <i>user,...</i>	指定したユーザーがアクセス権を持つキューのステータス情報を表示します。選択されたキューのいずれかで実行されているか、保留中でそうしたキューにスケジューリングされる可能性があるジョブのいずれかのステータス情報が表示されます。
-u <i>user,...</i>	指定されたユーザーリストに含まれるユーザーに関連づけられているジョブおよびキューに関する情報だけを表示します。-f または -F オプションも指定されていて、指定されたユーザーがキューでジョブを実行している場合に、そのキューのステータス情報が表示されます。

## 出力形式

-alarm、-f、-F、-r、-t オプションの有無によって、3 通りの出力形式があります。Sun Grid Engine, Enterprise Edition システムの場合は、-ext オプションを使用して、各ジョブの追加情報を表示することができます。

### 簡略形式 (-f および -F なし)

ヘッダー行に続いて、1 つのジョブに 1 行の長さで以下の情報が表示されます。

- ジョブ ID
- qsub(1) の -p オプションまたは保留中のジョブリストの順番を決定する qalter(1) を使用して割り当てられているジョブの優先順位
- ジョブ名
- ジョブ所有者のユーザー名
- ジョブのステータス - d (削除中)、t (転送中)、r (実行中)、R (再開)、s (一時停止中)、S (一時停止中)、T (しきい値)、w (待機中)、h (ホールド中) のいずれか。

d (削除中) の状態は、qdel(1) を使用してジョブの削除が開始されていることを意味します。t (転送中) および r (実行中) は、ジョブが実行されようとしているか、すでに実行中であることを示すのに対し、s (一時停止中) と S (一時停止中)、T (しきい

値) はすでに実行中のジョブが一時停止されていることを示します。s の一時停止は、qmod(1) コマンドを使用してジョブを位置停止したときの状態であり、s の一時停止は、そのジョブを含むキューが一時停止され、そのためにジョブも一時停止していることを意味します。T のしきい値は、対応するキューがその一時停止しきい値の少なくとも1つを上回る状態になり (queue\_conf(5) を参照)、その結果としてジョブが一時呈されたことを意味します。R (再開) は、ジョブが再開されたことを示します。これは、ジョブの移動が原因で、あるいは qsub(1) コマンドの -r で説明している理由で発生します。

w (待機中) と h (ホールド) は、保留中のジョブに対してのみ現れます。h (ホールド中) 状態は、qhold(1) か qalter(1)、qsub(1) -h コマンドによってホールド状態が割り当てられているために、ジョブが現在実行対象になっていないか、qsub(1) または qalter(1) の -hold\_jid オプションを使用してジョブの依存関係が割り当てられているジョブの完了待ちになっていることを示します。

- ジョブの実行依頼または開始日時
- ジョブが割り当てられているキュー (実行中または一時停止中のジョブのみ)
- 実行中のジョブの役割 (MASTER または SLAVE で、並列ジョブの場合にのみ意味がある) 配列ジョブタスク ID。非配列ジョブの場合は空です。詳細は、qsub(1) の -t オプションと上記の -g オプションを参照してください。  
-t オプションが指定された場合は、各ジョブのステータス行に次の情報も含まれます。
- 並列タスク ID (並列タスクと配列ジョブのタスクは異なります)
- 並列タスクのステータス -r (実行中)、R (再開)、s (一時停止中)、S (一時停止中)、T (しきい値)、w (待機中)、h (ホールド中)、x (終了) のいずれか。
- CPU、メモリー、入出力の使用量 (Sun Grid Engine, Enterprise Edition のみ)
- 並列タスクの終了ステータス
- 並列タスクの失敗コードとメッセージ

## 完全形式 (-f または -F の場合)

ヘッダー行の後に、横方向の線で区切られて1つのキューの情報が1つの区画で提供されます。表示されるキューの情報としては以下があります。

- キュー名
- キューの種類 -B (バッチ)、I (対話形式)、C (チェックポイント)、P (並列)、T (転送) のいずれかか、その組み合わせ
- 使用ジョブスロット数と使用可能なジョブスロット数
- キューのホストの負荷平均
- キューのホストのアーキテクチャ

- キューの状態 - u (不明)、a (アラーム)、A (アラーム)、C (カレンダーに基づく一時停止)、s (一時停止)、S (従属)、d (使用不可)、D (使用不可)、E (エラー) のいずれかか、その組み合わせ。状態が u (不明) の場合は、対応する `sge_execd(8)` にアクセスできないことを意味します。

a (アラーム) の場合は、キューが、その構成 (`queue_conf(5)` を参照) の `load_thresholds` リストに定義されている負荷しきい値の少なくとも1つを上回っていて、新たにジョブをスケジューリングできない状態であることを意味します。

これに対し A (アラーム) は、キューがその一時停止しきい値の少なくとも1つを上回った状態であることを示します (`queue_conf(5)` を参照)。この結果、そのキューで実行中のジョブは、しきい値違反がなくなるまで連続して一時停止されません。

状態の s (一時停止) と d (使用不可) は、`qmod(1)` コマンドを使用してキューに割り当てたり、割り当て解除したりできます。キューを一時停止すると、そのキューで実行中のすべてのジョブが一時停止されます。

状態の D (使用不可) と C (カレンダーに基づく一時停止) は、それぞれキューが使用負荷状態であるか、Sun Grid Engine のカレンダー機能を使用して自動的に一時停止されていることを意味します (`calendar_conf(5)` を参照)。これに対して S (従属) の状態は、キューが、別のキューに従属しているために一時停止されていることを意味します (詳細は、`queue_conf(5)` を参照)。原因に関係なくキューが一時停止されると、そのキューで実行中のあらゆるジョブも一時停止されます。

キューに E (エラー) 状態が表示された場合は、そのキューのホスト上の `sge_execd(8)` が、ジョブを開始する際、同じホストにある `sge_shepherd(8)` 実行可能ファイルを検出できなかったことを意味します。この問題を解決する方法の手がかりを得るには、その `sge_execd(8)` のエラーログファイルを調べてください。問題を解決したら、`qmod(1)` コマンドの `-c` オプションを使用し手動でキューを使用可能にします。

`-F` オプションが使用された場合は、キューのステータス行に続いて、資源が使用可能かどうかの情報が表示されます。1つの資源 (`-F` のオプション引数で選択されている資源か、オプション引数が省略されている場合はすべての資源) について1行の長さで、次の情報が表示されます。

- 現在の資源可用性値が次のどちらの種類の値であるかを示す英字1字
  - g - グローバル (クラスタ全体)
  - h - ホスト全体
  - q - キュー関連の資源消費量
- 現在の資源可用性値の情報源を示す英字1字
  - l - 資源の負荷値報告
  - L - 管理者定義の負荷スケーリングの適用後の資源の負荷値
  - c - 消費可能資源機能から得られた可用性 (`complexes(5)` を参照)

- v - 負荷報告または消費可能資源更新で書き換えられていないデフォルトの複合構成値
- f - 非消費資源複合属性または固定資源制限から得られた固定可用性の定義
- 情報が表示されている資源名 (コロンの後)
- 現在の資源可用性値 (等号の後)

可用性値とその情報源として、可能なあらゆる組み合わせのうちの最小値がつねに表示されます。このため、たとえば「**qf:h\_vmem=4G**」は、現在 **4G** バイトまでの仮想メモリーを使用可能なキューが存在し、この値は固定値で (たとえばキュー構成の資源制限値)、キュー主体の値であること、すなわち、ホスト全体ではもっと多くの仮想メモリーが使用可能である可能性があるが、キューが許可する上限は **4G** バイトであることを示します。これに対し「**hl:h\_vmem=4G**」という行も、**4G** バイトまで仮想メモリーが使用可能であることを示しますが、この上限値はホストに関する負荷報告から得られたものであることを示します。つまり、キューはもっと多くの仮想メモリーを要求しているジョブを受け付けられるかもしれないものの、そのキューが存在しているホストで現在使用可能なのは **4G** バイトだけであることとなります。

-alarm オプションが使用された場合は、負荷または一時停止しきい値に違反している資源の情報が、

次の追加情報とともに、-F オプションのときと同じ形式で表示されます。

- キーワード **alarm** から始まる行
- 適切なしきい値の種類と値 - 資源値の後に付加されます。

キューのステータス行 (-f の場合) または資源の可用性情報 (-F の場合) の後には、そのキューで実行中のジョブごとに 1 行の情報が表示されます。このジョブステータス行には、以下の情報が含まれます。

- ジョブ ID
- ジョブ名
- ジョブの所有者名
- ジョブのステータス - t (転送中)、r (実行中)、R (再開)、s (一時停止中)、S (一時停止中)、T (しきい値) のいずれか (詳細は、「簡略形式」の節を参照)。
- ジョブの開始日時と役割 (MASTER または SLAVE で、並列ジョブの場合にのみ意味がある)
- ジョブの優先順位

-t オプションが指定された場合は、各ジョブのステータス行に次の情報も含まれません。

- タスク ID
- タスクのステータス - r (実行中)、R (再開)、s (一時停止中)、S (一時停止中)、T (しきい値)、w (待機中)、h (ホールド中)、x (終了) のいずれか (詳細は「簡略形式」を参照)。
- CPU、メモリー、入出力の使用量 (Sun Grid Engine, Enterprise Edition のみ)

- タスクの終了ステータス
- タスクの失敗コードとメッセージ

キューへの割り当て待ちになっているジョブがある場合は、一連のキューの区画に続いて、保留中のジョブリストが表示されることがあります。待機中のジョブごとに、実行中のジョブに似たステータス行が 1 行表示されます。両者の違いは、ジョブのステータスが w (待機中) か h (ホールド中) であること、開始時間の代わりに実行依頼日時が示されること、ジョブに何の機能も表示されないことです。

非常にまれではありますが、たとえば、ジョブまたはキュープールファイルに矛盾がある状態から `sge_qmaster(8)` が起動したり、あるいは `qconf(1)` のキューの後処理 (`-cq`) オプションが使用された場合、`qstat` が実行中または保留中のどちらのジョブであるのか判定できないことがあります。そうした場合は、ジョブステータスの矛盾 (たとえばジョブのステータスは実行中であるが、キューに割り当てられていないなど) が検出され、出力最後の区画でエラージョブとして報告されます。エラー発生ジョブ (ERROR JOBS) 区画は、`sge_qmaster(8)` を再起動すると消えます。そうしたジョブの原因または影響について不明な点がある場合は、サンにお問い合わせください。

## 拡張形式 (-r の場合)

`qstat` に `-r` オプションが指定された場合は、ジョブごとに次の情報 (ジョブの特性ごとに 1 行が使用される) が表示されます。

- `qsub(1)` の `-l` オプションで指定されたジョブのハードおよびソフト資源要求
- 特定のキューロット範囲などの並列環境要求 (`qsub(1)` の `-pe` オプションを参照)
- ジョブのチェックポイント環境要求 (`qsub(1)` の `-ckpt` オプションを参照)
- 付与された並列環境とキューロット数 (実行中のジョブの場合)

## Sun Grid Engine, Enterprise Edition の拡張出力 (-ext の場合)

ジョブごとに次の情報が追加表示されます。

- `project/` プロジェクト - `qsub(1)` の `-P` オプションに指定された、ジョブが割り当てられたプロジェクト
- `department/` 部署 - ユーザーが属する部署。 `qconf(1)` の `-sul` と `-su` オプションを使用して、現在の部署の定義を表示できます。
- `deadline/` 締め切り優先 - `qsub(1)` の `-dl` オプションで指定された、ジョブの締め切り優先開始時間
- `CPU` - ジョブの現在の累積 CPU 使用量
- `mem` - ジョブの現在の累積メモリー使用量



- `io` - ジョブの現在の累積入出力使用量
- `tckts` - ジョブに現在割り当てられている総チケット数
- `ovrts - galter(1)` の `-ot` オプションで割り当てられた一時優先チケット
- `otckt` - ジョブに現在割り当てられている総チケット数に一時優先が占める割合
- `dtckt` - ジョブに現在割り当てられている総チケット数に締め切り優先が占める割合
- `ftckt` - ジョブに現在割り当てられている総チケット数に業務優先が占める割合
- `stckt` - ジョブに現在割り当てられている総チケット数に基本割当が占める割合
- `share` - システム全体に対してジョブが現在受ける資格をもっている配分量

## 環境変数

表 22 は、`qstat` に関する環境変数をまとめています。

表 22 `qstat` 関係の環境変数

変数名	説明
<code>SGE_ROOT</code>	Sun Grid Engine の標準構成ファイルの場所を示します。
<code>SGE_CELL</code>	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、 <code>qstat</code> は以下を使用します (優先順)。 <ul style="list-style-type: none"> <li>• 環境変数 <code>SGE_CELL</code> に指定されているセル名 (変数が設定されている場合)</li> <li>• デフォルトのセル名 (例: <code>default</code>)</li> </ul>
<code>SGE_DEBUG_LEVEL</code>	設定されている場合は、デバッグ情報を <code>stderr</code> に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
<code>COMMD_PORT</code>	設定されている場合は、 <code>sge_commd(8)</code> が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。
<code>COMMD_HOST</code>	設定されている場合は、Sun Grid Engine が <code>qstat</code> クライアントとの通信に使用する特定の <code>sge_commd(8)</code> が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

Sun Grid Engine のマスターホストファイル -  
`<sge_root>/<cell>/common/act_qmaster`

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## qtcsch(1)

### 名前

`qtcsch - qrsh` を利用した透過的な遠隔実行機能を持つ `tcsh v6.09` です。

### 構文

`qtcsch [ tcsh options | -ABLR ]`

### 説明

`qtcsch` は、一般的な `csch(1)` を基にした `tcsh` の機能を拡張します。透過的な遠隔実行コマンドの入力を受け付けて、**Sun Grid Engine** による制御を可能にします。`qtcsch` は、`tcsh` シェルスクリプトの処理ばかりでなく、対話形式のコマンドインタプリタとしても使用することができます。

`qtcsch` を起動すると、`qtcsch` が遠隔実行するコマンドとそうでないコマンドを判別します。このため、`<sge_root>/<cell>/common/qttask` および `~/.qttask` のファイル処理します。それらのファイルの各行は、遠隔実行用のコマンドの定義です（ファイル形式の定義については、`qttask(5)` を参照）。ユーザーのホームディレクトリの `.qttask` ファイルには、そのユーザー個人の遠隔タスク指定が含まれるのに対し、`common` ディレクトリ内の `.qttask` ファイルは管理者によって管理され、クラスタ全体のデフォルトの動作を定義します。ユーザーの `qttask(5)` ファイルに適切な

エントリーがある場合、管理者提供の `qtask(5)` ファイルの内容は完全に無効になります。ただし、管理者の `qtask` ファイル内のコマンド名の先頭に感嘆符が付いている場合は、そのようになりません。

`qtcsh` はつねに `qrsh(1)` 使用して、指定されたタスクを遠隔実行しようとします。例外は以下のケースです。

- コマンド名単独ではなく、相対または絶対パス名を使用してそうしたコマンドが入力された場合。詳細は、`qtask(5)` を参照してください。
- `JOB_ID` 環境変数が設定されていて、このため `qtcsh` がすでに **Sun Grid Engine** ジョブ内で遠隔実行が行われているとみなし、タスクをローカルに実行した場合。この動作は不要な繰り返しを回避しますが、コマンド行オプション `-R` または組み込みコマンドの `qrshmode -R` を使用して無効にすることができます (下記の対応する説明を参照)。
- `qtcsh` が起動中に **Sun Grid Engine** との接続を確立できなかった場合。この動作によって、**Sun Grid Engine** のサービスが受けられない場合は、ブロックされることなく `qtcsh` をログインシェルとして使用することができます。

`qtcsh` は、3つの異なるモードで動作することができ、以下のことを行うかどうかを決定します。

- タスクの遠隔実行
- 即時またはバッチ実行の要求
- ステータスの詳細出力 (エラーの場合にのみ詳細出力するかどうか)

これらのモードは、`qtcsh` を起動する際に下記のコマンド行スイッチを使用するか、または 82 ページの「組み込みコマンド」の節で説明しているように、組み込みコマンドの `qrshmode` を使用して、動作中の `qtcsh` の中から制御することができます。

## オプション

表 23 にまとめているオプションは `qtcsch` 専用です。その他のオプションについては、`tcsh(1)` の説明を参照してください。

表 23 `qtcsch` のオプション

オプション	説明
-A	<code>qtcsch</code> を詳細モードで起動します。遠隔実行の場合は診断出力が行われます。
-B	遠隔タスク実行をバッチモードに切り替えます。Sun Grid Engine に実行依頼されたタスクをただちに開始できない場合は、キューに入れます。このため、キューに入れられたタスクが Sun Grid Engine によって開始可能になるまで、 <code>qtcsch</code> がブロックされることがあります。対話形式でのセッションではおそらくこの動作は望ましいものではありませんが、特定のタスク用の資源が一時的に使用できない場合のスクリプトの実行失敗が回避されるため、 <code>qtcsch</code> を使用してシェルスクリプトを実行する場合に非常に役立つことがあります。
-I	コマンドの遠隔実行というデフォルトの動作を無効にします。 <code>qtask(5)</code> ファイルのいずれかに含まれていても、すべてのコマンドがローカルに実行されます。
-R	<code>JOB_ID</code> 環境変数に設定されている場合でもコマンドを遠隔実行します。

## 組み込みコマンド

ここでは、標準の `tcsh(1)` では使用できない、追加のシェル組み込みコマンドだけを説明します。

### `qrshmode` [-ANBILR]

オプションなしで使用すると、`qtcsch` の現在の動作モードが表示されます。使用可能なオプションは以下のとおりです。

- -A - 詳細出力モードに切り替えます。
- -N - 非詳細出力モードに切り替えます。
- -B - バッチ実行モードに切り替えます。
- -I - 即時実行モードに切り替えます。
- -L - つねにコマンドをローカルに実行します。
- -R - 組み込まれているコマンドを遠隔実行します。

## 環境変数

表 24 は、qtcsch に関する環境変数をまとめています。

表 24 qtcsch 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、qtcsch は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>• 環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>• デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が qtcsch クライアントとの通信に使用する特定の sge_commd(8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## ファイル

- ユーザーの qtask ファイル - ~/.qtask
- クラスタの qtask ファイル - <sge\_root>/<cell>/common/qtask

## 関連項目

sge\_intro(1)、qrsh(1)、qtask(5)、tcsch(1)  
(<sge\_root>/3rd\_party/qtcsch にある)

## 著作権

qtcsch には、一部、Regents of the University of California が著作権を保有する tcsh のコードが含まれています。このため、次のことが適用されます。この製品には、カリフォルニア州立大学バークレー校とその寄稿者が開発したソフトウェアが含まれています。

他の権利および許諾条件文については、sge\_intro(1) と <sge\_root>/3rd\_party/qtcsch に提供されている情報を参照してください。

---

## submit (1)

### 名前

qsub - Sun Grid Engine にバッチジョブの実行依頼をします。

qsh - Sun Grid Engine に対話形式の X ウィンドウセッションの実行依頼をします。

qlogin - Sun Grid Engine に対話形式のログインセッションの実行依頼をします。

qrshn - Sun Grid Engine に対話形式の rsh セッションの実行依頼をします。

qalter - Sun Grid Engine の保留中のバッチジョブを変更します。

qresub - 既存の Sun Grid Engine ジョブのコピーの実行依頼をします。

### 構文

qsub [ options ] [ scriptfile | -- [ script\_args ] ]

qsh [ options ] [ -- xterm\_args ]

qlogin [ options ]

qrsh [ options ] [ command [ command\_args ] ]

qalter [ options ] job/task\_id\_list [ -- [ script\_args ] ]

qalter [ options ] -u user\_list | -uall [ -- [ script\_args ] ]

qresub [ options ] job\_id\_list

## 説明

`qsub` は、Sun Grid Engine のキューイングシステムにバッチジョブの実行依頼をします。Sun Grid Engine は、単一および複数ノードのジョブをサポートしています。`scriptfile` には、シェル (`sh(1)`、`csh(1)` など) を使用してジョブが実行するコマンドを含めます。ジョブスクリプトに渡す引数は、`script_args` で指定します。Sun Grid Engine のフラグは、`qsub` の引数として入力することも、`scriptfile` への埋め込みフラグとして入力することもできます。ただし、後者の場合、スクリプト行の最初の 2 文字は「`#$`」であるか、下記の `-c` オプションで定義した接頭辞文字列である必要があります。

`qsh` は、Sun Grid Engine に対話形式の X ウィンドウセッションの実行依頼をします。この場合は、実行マシンから `xterm(1)` が起動され、表示は `DISPLAY` 環境変数の示す X サーバーか、`qsh` の `-display` オプションで指定された X サーバーのいずれかに送られます。実行に使用できる資源がない場合、対話形式のジョブはスプールされません。即時の実行に適したマシンにディスパッチされるか、そのジョブの実行依頼をしたユーザーに、実行に適切な資源がないことが `qsh` から通知されます。

`xterm_args` は `xterm(1)` 実行可能ファイルに渡されます。

`qlogin` は、キューイングシステムに対話形式のジョブの実行依頼をする点で `qsh` に似ています。しかし、`qlogin` は X 表示で `xterm(1)` ウィンドウを開きません。`qlogin` はユーザー入出力に現在の端末を利用します。通常、`qlogin` は標準のクライアントおよびサーバー側コマンドを使用して遠隔ホストと `telnet(1)` 接続を確立します。そうしたコマンドは、`sge_conf(5)` のグローバルおよびローカル構成設定の `qlogin_daemon` (サーバー側) と `qlogin_command` (クライアント側) パラメータを使用して設定することができます。これらのパラメータが設定されていない場合は、それぞれ Sun Grid Engine の `telnetd` と `telnet` が使用され、設定する場合は、`/usr/sbin/in.telnetd` あるいは `/usr/bin/telnet` というような形式で設定する必要があります。クライアント側コマンドは、遠隔ホスト名と接続先のポート番号で自動的にパラメータ化されます (すなわち、`/usr/bin/telnet my_exec_host 2442` のようになる)。`qlogin` の起動動作は `qsh` と完全に同じですが、そのジョブは INTERACTIVE キューでのみ実行できます。`qlogin` ジョブは、`sge_execd(8)` が root アカウントの管理下で動作している場合にのみ使用できます。

`qrsh` は、キューイングシステムに対話形式のジョブの実行依頼をする点で `qlogin` に似ています。`qrsh` はユーザー入出力に現在の端末を使用し、通常、遠隔ホストとの間で `rsh(1)` 接続を確立します。`qrsh` にコマンドが指定されなかった場合は、`rlogin(1)` セッションが確立されます。使用するサーバー側コマンドは、`sge_conf(5)` のグローバルおよびローカル構成設定の `rsh_daemon` および `rlogin_daemon` パラメータを使用して設定することができます。これらのパラメータが設定されていない場合は、Sun Grid Engine の `rshd` または `rlogind` が使用されます。設定する場合は、`/usr/sbin/in.rshd` あるいは `/usr/sbin/in.rlogind` というような形式で設定する必要があります。クライアント側では、`sge_conf(5)` のグローバルおよびローカル構成設定で `rsh_command` および `rlogin_command` パラメータを設定することができます。これらのパラメータ

が設定されていない場合は、Sun Grid Engine 付属の `rsh(1)` と `rlogin(1)` がそれぞれ使用されます。クラスタ構成を使用して、オペレーティングシステムに付属している `ssh` や `rsh(1)`、`rlogin(1)` などの機能を組み込んでください。

`-now no` オプションが使用されていないのでない限り、`qrsh` ジョブは **INTERACTIV** キューでのみ実行することができます (下記を参照)。また、`qrsh` ジョブは、`sge_execd(8)` が `root` アカウントの管理下で動作している場合にのみ使用できます。

`qrsh` には、固有のコマンドシェルを提供する対話型のツールとの統合に便利な機能も用意されています。`QRSH_WRAPPER` 環境変数が設定されている場合、`qrsh` を起動すると、ユーザーのログインシェルあるいは `qrsh` のコマンド行に指定されたシェルではなく、その環境変数が示すコマンドインタプリタが実行されて、`qrsh` コマンドを実行します。

`qalter` は、保留中のジョブの属性の変更に使用することができます。ジョブの実行が開始された場合、そうした属性の変更はできなくなります。配列ジョブの場合は、一部のタスクが保留中で、別のタスクが実行中のことがあります (下記の `-t` オプションを参照)、`qalter` による変更は保留中のタスクにだけ適用されます。`qalter` は、スクリプトファイル組み込みフラグとして定義されている特性を含めて、ジョブの大部分の特性を変更することができます (下記の「オプション」の対応する説明を参照)。

`qresub` では、既存の保留中または実行中のジョブからジョブのコピーを作成することができます。作成されたジョブはコピー元のジョブと完全に同じ属性を持ちますが、新しいジョブ ID が割り当てられます。`qresub` がサポートするジョブのコピーに加えらるる唯一の変更は、`-h` オプションを使用してホールド状態が割り当てられることです。この状態を利用して、ジョブのコピーを作成した後、`qalter` を使用してその属性を変更することができます。

`qsub`、`qsh`、`qrsh`、`qlogin` の場合、管理者およびユーザーは、下記の任意のオプションを含むことが可能なデフォルト要求ファイルを定義することができます (`sge_request(5)` を参照)。`qsub` と `qlogin` が認識し、`qsh` が認識しないオプションがそうしたデフォルト要求ファイルに含まれていて、`qsh` が起動された場合、そのオプションは何のメッセージもなしに無視されます。このため、`qsub` と `qsh` の両方について共有デフォルト要求ファイルを持つことができます。

クラスタ全体のデフォルト要求ファイルは、`$sge_root/$sge_cell/common/sge_request` に置くことができます。ユーザー個人のデフォルト要求ファイルは、`$HOME/.sge_request` と `$cwd/.sge_request` の場所で処理されます。これらのデフォルト要求ファイルの優先順位は、作業ディレクトリのローカルのファイルの最も高く、次にホームディレクトリのファイル、クラスタ全体のファイルと続きます。オプション引数、スクリプト埋め込みフラグ、デフォルト要求ファイルのオプションは、次の順序で処理されます。

1. スクリプト行の左から右
2. デフォルト要求ファイルの左から右



3. スクリプトファイルの上から下 (qsub のみ)
4. デフォルト要求ファイルの上から下
5. コマンド行の左から右

言い替えれば、コマンド行は埋め込みフラグとデフォルト要求設定に優先する指定を行う目的に使用することができます。ただし、埋め込みフラグはデフォルトの設定に優先します。

---

**注** – デフォルト要求ファイルやスクリプト埋め込みフラグ、コマンド行で `-clear` オプションを使用することによって、いつでも以前の設定を廃棄することができます。ただし、`qalter` ではこの機能は使用できません。

---

表 25 で説明しているオプションは、ハードまたはソフトのどちらでも要求することができます。デフォルトでは、`-soft` オプション (下記を参照) が検出されない限り、すべての要求はハードとみなされます。ハード / ソフトのステータスは、対応する指定が再検出されない限り有効です。ジョブのハード要求が 1 つでも満たせない場合、そのジョブはスケジューリングされません。その時点で実行不可能なジョブはスプールされたままになります。

# オプション

表 25 は、さまざまな実行依頼コマンドのオプションをまとめています。

表 25 実行依頼コマンドのオプション

オプション	説明
<code>-@ optionfile</code>	<p><i>optionfile</i> に含まれているオプションを強制的に <code>qsub/qysh/qsh/qlogin</code> に使用させます。指定されたファイルには、正当な任意のオプションを含めることができます。コメント行は # 記号で始めます。</p>
<code>-a date_time</code>	<p><code>qsub/qysh/qsh/qlogin/qalter</code> で使用できます。ジョブを実行対象にする日時を設定または再設定します。</p> <p><i>Date_time</i> は <code>[[CC]]YY[MMDDhhmm].[ss]</code> の形式で指定します。</p> <p>CC 2桁で世紀を指定します。 YY 2桁で年を指定します。 MM 2桁で月を指定します。 DD 2桁で日を指定します。 hh 2桁で時を指定します。 mm 2桁で分を指定します。 ss 2桁で秒を指定します (デフォルトは 00) 省略可能な日付フィールドがどれか省略された場合は、そのときの日付の対応する値とみなされます。</p> <p>Sun Grid Engine プールのホストのクロックの同期がとられていない場合、このオプションを使用すると、予期しない結果が生じることがあります。また、このオプションが正しく機能するかどうかは、Sun Grid Engine のデーモンの <code>sge_qmaster(8)</code> と <code>sge_execd(8)</code> が起動されたときに、TZ 環境変数 (詳細は <code>date(1)</code> を参照) などに適切な時間帯が正しく設定されているかどうかにかかなり依存します。</p> <p><code>qalter</code> を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
<code>-ac variable[=value],...</code>	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。</p> <p>指定された名前と値のペア (複数指定可能) をジョブのコンテキストに追加します。<code>value</code> は省略可能です。その場合、Sun Grid Engine はジョブのコンテキスト変数のリストに、指定された引数を付加します。<code>-ac</code>、<code>-dc</code>、<code>-sc</code> オプションは複数指定できます。ここでは、順序が重要な意味を持ちます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>
<code>-A account_string</code>	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。</p> <p>ジョブの資源消費の課金先のアカウントを指定します。<code>account_string</code> には、任意の ASCII 英数字文字列を指定できますが、空白や区切り文字を含めることはできません。下線 (<code>_</code>) は区切り文字とはみなされません。このパラメータがない場合、Sun Grid Engine はジョブのアカウントレコードにデフォルトのアカウント文字列の <code>sge</code> を記録します。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-c <i>occasion_specifier</i>	<p>qsub と qalter でのみ使用できます。ジョブがチェックポイント生成対象であるかどうか、生成対象の場合は、生成のタイミングを定義または再定義します。このオプションで指定したチェックポイントの生成タイミングは、qsub の -ckpt スイッチで参照されるチェックポイント環境の <i>when</i> パラメータの定義に優先します (checkpoint (5) を参照)。<i>occasion_specifier</i> には、以下の値を指定できます。</p> <ul style="list-style-type: none"> <li>n - チェックポイントを生成しない。</li> <li>s - バッチサーバーの停止時にチェックポイントを生成</li> <li>m - 最小 CPU 間隔でチェックポイントを実行</li> <li>x - ジョブが一時停止されたときにチェックポイントを生成</li> </ul> <p>&lt;interval&gt;- 指定した時間間隔でチェックポイントを生成最小 CPU 間隔は、キュー構成で定義します (詳細は、queue_conf (5) を参照)。          &lt;interval&gt; は、hh:mm:ss の形式で指定する必要があります。&lt;interval&gt; が指定された場合は、その &lt;interval&gt; と キューの最小 CPU 間隔の大きい方が使用されます。これは、チェックポイントがあまり頻繁に生成されることによってマシンが過負荷にならないようにするためです。</p>
-ckpt <i>ckpt_name</i>	<p>qsub と qalter でのみ使用できます。ジョブのチェックポイント生成に使用するチェックポイント環境を選択します (checkpoint (5) を参照)。また、ジョブがチェックポイントジョブであると定義します。</p>
-clear	<p>qsub/qcrsh/qsh/qlogin で使用できます。このコマンドに指定された変更を適用する前に、ジョブのすべての要素を初期デフォルト状態にリセットします。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-cwd	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。現在の作業ディレクトリからジョブを実行します。対応する構成ファイルが存在する場合、このスイッチは <b>Sun Grid Engine</b> のパス別名機能を有効にします (sge_aliases(5) を参照)。このスイッチが指定された qalter がこの前の qsub または qalter と異なるディレクトリから実行された場合、現在の作業ディレクトリの以前の定義は書き換えられます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-C prefix_string	<p>qsub でのみ使用できます。prefix_string には、ジョブのスク립トファイルにおける qsub に対する指定であることを示す接頭辞を定義します。この接頭辞はジョブの属性ではありませんが、qsub の動作に影響します。-C オプションに対する指令接頭辞が null 文字列の場合、qsub はスク립トファイルを読み取りません。指令接頭辞は、ASCII 文字 2 文字で構成します。スク립ト行の先頭 2 バイトがそれら 2 文字の場合は、その後の文字列が <b>Sun Grid Engine</b> のコマンドであることを意味します (デフォルトの接頭辞は #)。最初の区切り文字を変更すると、予期できない副作用が生まれる可能性があることに注意してください。スク립トファイルの行の先頭バイト位置に # 以外の文字が含まれている場合、ジョブのシェルプロセスは行を拒否して、ジョブを途中で終了することがあります。スク립トファイルに含まれている -C オプションは、無視されます。</p>
-dc variable,...	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。ジョブのコンテキストから指定された変数 (複数指定可能) を削除します。-ac、dc、-sc オプションは複数指定できます。ここでは、順序が重要な意味を持ちます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>
-display display_specifier	<p>qsh でのみ使用できます。X サーバーにアクセスする際に display_specifier を使用する xterm(1) に指示します。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-dl <i>date_time</i>	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できません。このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。[[CC]YY]DDhhmm[.SS] の形式で締め切り優先開始時間を指定します (上記の -a オプションを参照)。締め切り優先開始時間とは、指定された締め切りまでに完了できるよう、締め切り優先ジョブが最高の優先順位に達する必要がある時間です。締め切り優先ジョブの優先順位は、締め切り優先開始時間以前から、Sun Grid Engine 管理者が設定した最大値に達するまで着実に高められていきます。このオプションは、締め切り優先ジョブの実行依頼が許可されているユーザーにのみ有効です。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-e <i>[hostname:]path,...</i>	<p>qsub と qalter でのみ使用できます。ジョブの標準エラー streams に使用するパスを設定または再設定します。path が絶対パス名の場合、ジョブのエラーパス属性は、ホスト名を含むその値に設定されます。path が相対パス名の場合は、現在の作業ディレクトリのパス (上記の -cwd スイッチも指定されている場合) かホームディレクトリのパスで展開されます。hostname が指定されていて、ジョブがそのホストで実行されている場合、標準エラー streams は対応する場所に書き込まれます。デフォルトでは、標準エラーのファイル名は job_name.ejob_id か job_name.ejob_id.task_id (配列ジョブのタスクの場合で、下記の -t オプションを参照) のいずれかの形式です。path がディレクトリの場合、ジョブの標準エラー streams は、そのディレクトリにデフォルトのファイル名で書き込まれます。パス名に疑似環境変数が含まれている場合、その値は実行時に展開されて、標準エラー streams のパス名の作成に使用されます。現在使用可能な疑似環境変数は以下のとおりです。</p> <p>\$HOME - 実行マシンのホームディレクトリ            \$USER - ジョブ所有者のユーザー ID            \$JOB_ID - 現在のジョブ ID            \$JOB_NAME - 現在のジョブ名 (-N オプションを参照)            \$HOSTNAME - 実行ホスト名            \$TASK_ID - 配列ジョブのタスクの添字番号</p> <p>csh(1) または ksh(1) のどちらでも、\$HOME の代わりにチルド記号 (~) を使用することができます。~ 記号はまた、ユーザー名とも組み合わせることができ、「~&lt;ユーザー名&gt;」は &lt;ユーザー&gt; のホームディレクトリに展開されます。当然、ジョブ所有者以外のユーザー ID を使用するには、対応するアクセス権が必要です。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-hard	<p>qsub/qrsh/qsh/qlogin/qalter で使用できません。コマンド行の以降のすべての資源要求がハード要求であり、ジョブをスケジューリングする前に完全に満たされる必要があることを意味します。Sun Grid Engine はコマンド行とスクリプトファイルからそのオプションとパラメータを読み取りながら、ジョブが要求する資源のリストを作成します。そうした資源要求はすべて、ジョブを開始するために不可欠であるとみなされます。読み取り中に下記の -soft オプションが検出されると、以降の資源要求は、実行に際しての「ソフト要求」、すなわち、「あればよいが、不可欠ではない」とみなされます。読み取りの後の方の段階で -hard フラグが検出された場合、以降のすべての資源要求は再び「不可欠」になります。要するに、-hard および -soft オプションは読み取り中に「トグル」の働きをします。</p>



表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-h   -h {u s o n U O S}...	<p>qsub/qcrsh/qsh/qlogin/qalter/qresub で使用できます。ジョブに以下のホールドを設定できます。</p> <p>u - ユーザーホールド  s - システムホールド  o - オペレータホールド  n - ホールドなし</p> <p>n 以外のホールドがジョブに割り当てられている場合、そのジョブは実行対象でなくなります。ホールドは <code>qalter</code> や <code>qrls(1)</code> で解除できます。<code>qalter</code> の場合は、<code>-h</code> スイッチの他のオプションを使用してホールドを解除することができます。</p> <p>U - ユーザーホールドの解除  S - システムホールドの解除  o - オペレータホールドの解除</p> <p><b>Sun Grid Engine</b> のマネージャはすべての種類のホールド、オペレータはユーザーとオペレータホールド、ユーザーはユーザーホールドのみ割り当てたり、割り当て解除したりできます。<code>qsub</code> の場合は、ユーザーホールドのみ設定できます。つまり、使用できるのは、<code>-h</code> スイッチの最初の形式のオプションだけです。これに対し、<code>qalter</code> では、上記の 2 番目の形式を使用する必要があります。もう 1 つホールドを割り当てる手段として、<code>qhold(1)</code> 機能があります。ジョブが配列ジョブの場合 (下記の <code>-t</code> オプションを参照)、<code>-t</code> で指定されたタスクはすべて <code>-h</code> 操作の影響を受けません。<code>qalter</code> を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p> <p><code>-help</code> 全オプションの一覧を表示します。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-hold_jid [job_id job_name],...	<p>qsub/qrsh/qsh/qlogin/qalter で使用できません。実行依頼されたジョブのジョブ依存関係リストを定義または再定義します。ジョブ名による参照は、参照先のジョブが参照元と同じユーザーによって所有されている場合にのみ受け付けられます。コンマ区切りの <i>job_id</i> か <i>job_name</i>、またはその両方のリストで参照されているすべてのジョブが正しく完了していない限り、実行依頼したジョブが実行対象になることはありません。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-inherit	<p>qrsh と qmake(1) でのみ使用できます。qrsh を使用して、すでにスケジューリングされている並列ジョブのタスクを開始することができます。-inherit オプションは、環境変数 <i>JOB_ID</i> からジョブ ID を読み取り、指定されているコマンドをそのジョブ内のタスクとして実行するよう qrsh に指示します。この場合、コマンドが実行されるホストのホスト名をコマンドの前に挿入する必要があり、構文は以下のようになることに注意してください。</p> <p>qrsh-inherit [ <i>other options</i> ]  <i>hostnamecommand</i> [ <i>command_args</i> ]-inherit</p> <p>と組み合わせると、他の大部分のコマンド行オプションは無視されます。解釈されるのは、-verbose、-v、-V オプションだけです。また、-cwd の代わりに、-v PWD を使用します。通常、タスクは対応するジョブと同じ環境 (現在の作業ディレクトリなど) で実行するようにします。このため、大部分のアプリケーションに -V オプションが適しています。コマンドポートをサービスとして構成していなくて、環境変数 <i>COMMD_PORT</i> を使用して構成しているシステムで、-inherit オプションを付けて qrsh または qmake を呼び出す場合は、必ずこの変数が設定されていることを確認してください。ジョブスクリプトの中から -inherit オプション付きで qrsh または qmake を呼び出す場合は、submit オプションまたは特殊なコメント -v <i>COMMD_PORT</i> を使用して <i>COMMD_PORT</i> をエクスポートします。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-j y n	<p>qsub と qalter でのみ使用できます。ジョブの標準エラーストリームを標準出力ストリームにマージするかどうかを指定します。-j y と -e オプションの両方が指定されている場合、Sun Grid Engine はエラーパス属性を設定しますが、無視します。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-l resource=value,...	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。指定された資源要求リストを満たす Sun Grid Engine キューでジョブを開始します。qalter の場合、以前の定義は指定された定義で置き換えられます。complex(5) は、使用可能な資源とそれに割り当てられている値の指定の取得方法を表します。-l スイッチは、1 つのコマンドに複数指定できます。同じコマンド行で複数の -l オプションをソフトまたはハードのどちらとしても要求できます。シリアルジョブの場合、複数の -l スイッチを使用して、対象のキューを細かく定義できます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-m b e a s n,...	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。ジョブの所有者または下記の -M オプションで定義したユーザーへのメールの送信環境を定義または再定義します。オプションの引数の意味は次のとおりです。</p> <ul style="list-style-type: none"> <li>b - ジョブの開始でメールを送信</li> <li>e - ジョブの終了でメールを送信</li> <li>a - ジョブの実行中または再スケジューリングでメールを送信</li> <li>s - ジョブの一時停止でメールを送信</li> <li>n - メールを生成しない。現在、ジョブが一時停止されたときメールは送信されません。デフォルト要求ファイルに qsh または qlogin が含まれている場合、ジョブの開始または終了時のメールは抑止されます。qalter を使用すると、ジョブの実行中でも b、e、a オプションを変更することができます。ただし、b オプション引数の変更が有効になるのは、ジョブの再開または移動後です。</li> </ul>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-M <i>user[@host],...</i>	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。ジョブを実行するサーバーがそのジョブに関するメールを送信するように設定されている場合に、その送信先のユーザーのリストを定義または再定義します。デフォルトは、実行依頼元のホストのジョブの所有者です。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>
-masterq <i>queue,...</i>	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。並列ジョブ、すなわち、-pe オプションとの組み合わせでのみ有効です。この並列ジョブのいわゆるマスターキューとして使用可能なキューのリストを定義または再定義します。マスターキューとは、並列ジョブが開始されるキューです。並列ジョブのタスク生成先の他のすべてのキューは、スレーブキューと呼ばれます。並列ジョブはマスターキューを1つだけ持ちます。このパラメータは資源要求のすべてのプロパティを持ち、上記の -l オプションから得られた要求とマージされます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-notify	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。このフラグが設定されると、Sun Grid Engine は実行中のジョブにシグナルそのものを送信する前に「警告」シグナルを送信します。SIGSTOP が保留中の場合、ジョブは SIGSTOP の数秒前に SIGUSR1 を受け取ります。SIGKILL が保留中の場合、SIGKILL の数秒前に SIGUSR2 を受け取ります。この遅延時間の長さは、個々のキュー構成の notify パラメータで制御します (queue_conf (5) を参照)。Linux オペレーティングシステムの現在の POSIX スレッド実装状態では、ユーザーシグナルの SIGUSR1 と SIGUSR2 が「誤用」されています。Linux 上でジョブからスレッドアプリケーションを実行する場合は、-notify オプションを使用しないことを推奨します。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-now y[es]   n[0]	<p>qsub/qrsh/qsh/qlogin で使用できます。</p> <p>-now y は、ジョブをただちに開始しようとし、開始できない場合は、実行しません。開始に成功した場合は 0、失敗した場合は 1 が返されます。1 は、ジョブをただちにスケジューリングできなかった場合にも返されます。</p> <p>qsh、qlogin、qrsh の場合、デフォルトは -now y です。-now n オプションの場合、ただちにられます。qsub の場合、デフォルトは -now n です。</p>
-N name	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。ジョブの名前です。ジョブ名は印刷可能文字の組み合わせで、英字で始めます。-N オプションが指定されていない場合、Sun Grid Engine は、ジョブスクリプト名からディレクトリパス名をすべて省略した名前をジョブに割り当てます。標準入力からスクリプトが読み取った場合、ジョブ名はデフォルトで STDIN になります。qsh または qlogin で -N オプションが指定されていない場合は、INTERACT という文字列がジョブに割り当てられます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>
-noshell	<p>qrsh コマンド行でのみ使用できます。-qrsh に与えられたコマンド行のユーザーのログインシェル内で開始しないで、ラッピングシェルなしでコマンド行を実行します。シェルの起動、シェルリソースファイルの供給などのオーバーヘッドが回避されるため、このオプションを使用して実行速度を上げることができます。このオプションを使用できるのは、シェル固有のコマンド行構文解析が必要ない場合だけです。コマンド行に環境変数の置換または (逆) 引用などの構文が含まれている場合は、シェルを起動する必要があります。このため、そうした場合は -noshell オプションを使わないか、コマンド行にシェルの呼び出しを含めてください。</p> <p>例:</p> <pre>qrsh echo '\$HOSTNAME'</pre> <p>-noshell オプションを使用して次のように呼び出すこともできます。</p> <pre>qrsh -noshell /bin/tcsh -f -c 'echo \$HOSTNAME'</pre>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-nostdin	qrsh でのみ使用できます。入力ストリーム STDIN を抑止します。qrsh は rsh(1) コマンドに -n オプションを渡します。このオプションは、qrsh を使用し、たとえば make(1) プロセスで複数のタスクを並列実行する場合に特に有用です。どのプロセスが入力を得るのかは、定義されません。
-o [hostname:]path,...	qsub と qalter でのみ使用できます。ジョブの標準出力ストリームに使用するパスです。path は、標準エラーストリーム用の -e オプションで説明しているように処理されます。デフォルトでは、標準出力のファイル名は job_name.ojob_id か job_name.ojob_id.task_id (配列ジョブのタスクの場合で、下記の -t オプションを参照) のいずれかの形式です。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。
-ot override_tickets	qalter でのみ使用できます。このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。指定されたジョブの一時優先チケット数を変更します。マネージャー / オペレータ特権が必要です。
-P project_name	qsub/qrsh/qsh/qlogin/qalter で使用できます。このオプションは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムでは使用できません。ジョブを割り当てるプロジェクトを指定します。管理者は、特定のプロジェクトにジョブの実行依頼をする権限を個々のユーザーに付与する必要があります (qconf(1) の -apri オプションを参照)。

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-p <i>priority</i>	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。他のジョブに対するジョブの相対的な優先順位を設定または再設定します。優先順位値は -1023 から 1024 の範囲の整数です。デフォルトの優先順位値は 0 です。Sun Grid Engine システムでは、ユーザーはジョブの優先順位を下げるのみ行うことができます。Sun Grid Engine マネージャーと管理者は、優先順位を上げることもできます。優先順位が高いほど、保留中のジョブが Sun Grid Engine スケジューラのディスパッチ対象になるのが早まります。ジョブの優先順位は、Sun Grid Engine は実行中のジョブには影響しません。Sun Grid Engine, Enterprise Edition では、ジョブの優先順位が基本割当ポリシーと業務優先ポリシーに影響します。ただし、締め切り優先ポリシーと一時優先ポリシーには影響しません (Sun Grid Engine, Enterprise Edition がサポートする資源管理ポリシーについての詳細は、share_tree(5) と sced_conf(5)、『Sun Grid Engine 5.3/Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』を参照)。基本割当ポリシーの場合、ユーザーは、-p で優先順位を割り当てることによって、現在自分が資格を受けているチケットを自分のジョブに分配することができます。ジョブの優先順位がすべて同じ場合、チケットは均等に分配されます。同じでない場合、ジョブは優先順位に比例してチケットを受け取ります。後者の場合、優先順位は基本割当ツリーの追加レベルのように扱われます。業務優先ポリシーの場合、優先順位は、業務優先ジョブカテゴリ内のジョブの重み付けに使用できます。ここでも、チケットは優先順位に比例して分配され、この優先順位は、業務優先ジョブカテゴリ下の仮想の配分レベルとして扱われます。基本割当ツリーおよび業務優先ポリシーのどちらも使用されている場合、ジョブの優先順位は両方のポリシーに作用し、それぞれで得られるチケットが、各ジョブのチケット合計に加算されます。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-pe <i>parallel_environment</i> <i>n</i> [- <i>m</i> ][/ <i>n</i> - <i>m</i> ,...	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。インスタンス化する並列プログラミング環境 (PE) です。並列環境名の後の範囲記述子は、実行する並列プロセス数を示します。Sun Grid Engine は、適切な資源が使用可能になると割り当てます。並列環境の定義と現在有効な並列環境のリストの取得方法については、sge_pe (5) のマニュアルページを参照してください。並列環境名にワイルドカード文字 (*) を使用することができます。たとえば pvm* という要求は、名前が文字列 pvm から始まるすべての並列環境に一致します。範囲は、<i>n</i>-<i>m</i> (<i>n</i> および <i>m</i> ともに、ゼロ以外の正の整数) の形式の範囲式のリストです。<i>m</i> は <i>m</i>-<i>m</i> の省略形で、-<i>m</i> は <i>1</i>-<i>m</i> の省略形、<i>n</i>- は <i>n</i>- 無限の省略形になります。この範囲指定は次のように処理されます。要求されたキューの最大数をまずチェックします。指定された属性リストを満たすのに十分な数のキューがある場合は、すべてを割り当てます。次にキュー数を少なくしてチェックします。このようにして、キュー数を小さくしてチェックしています。-1 オプションも指定されている場合、並列ジョブの実行資格があるキュー群は制限されます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-q <i>queue</i> , ...	<p>qsub/qrsh/qsh/qlogin/qalter で使用できます。このジョブの実行に使用可能なキューのリストを定義または再定義します。このパラメータは資源要求のすべてのプロパティを持ち、上記の -1 オプションから得られた要求とマージされます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>
-r <i>y</i>   <i>n</i>	<p>qsub と qalter でのみ使用できます。ジョブの再実行を許可するかどうかを指定します。-r の値が <i>y</i> で、ジョブを実行していたノードがクラッシュしたなどの理由でジョブの実行が中止された場合は、ジョブが再実行されます。-r が <i>n</i> の場合は、そうした条件に関係なくジョブは再実行されません。qsh または qlogin で実行依頼した対話形式のジョブは再実行できません。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>



表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-sc <i>variable[=value],...</i>	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。指定された <i>name/valu</i> のペア (複数指定可能) をジョブのコンテキストとして設定します。<i>value</i> は省略可能です。Sun Grid Engine は、定義されていたジョブのコンテキストを、引数に指定されたコンテキストに置き換えます。-ac、-dc、-sc オプションは複数指定できません。ここでは、順序が重要な意味を持ちます。コンテキストは、動的にジョブに meta 情報を関連づけまたは関連づけ解除する手段です。その環境では、コンテキスト変数はジョブの実行コンテキストに渡されません。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。</p>
-soft	<p>qsub/qcrsh/qsh/qlogin/qalter で使用できます。コマンド行の以降のすべての資源要求がソフト要求、すなわち、使用可能になった時点で満たせばよい要求であることを意味します。Sun Grid Engine はコマンド行とスクリプトファイルからそのオプションとパラメータを読み取りながら、ジョブが要求する資源のリストを作成します。そうした資源要求はすべて、ジョブを開始するために不可欠であるとみなされます。読み取り中に下記の -soft オプションが検出されると、以降の資源要求は、実行に際しての「ソフト要求」、すなわち、「あればよいが、不可欠ではない」とみなされます。読み取りの後の方の段階で -hard フラグが検出された場合、以降のすべての資源要求は再び「不可欠」になります。要するに、-hard および -soft オプションは読み取り中に「トグル」の働きをします。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-s [ <i>host:</i> ] <i>pathname</i> ,...	<p>qsub/qrsh/qlogin/qalter で使用できません。ジョブの解釈を行うシェルを指定します。<i>host</i> 指定のない <i>pathname</i> のみも有効であり、特定のホストのパス名だけ指定することもできます。シェルのパスにホストが指定されていて、そのホストが実行ホストの場合、そのパスは解釈を行うシェルの定義になります。ホスト指定のないシェルのパスは、実行ホストに一致するホストがリストにない場合に使用されます。また、上記の -e オプションで説明している疑似環境変数を使用して、<i>pathname</i> を作成することもできます。qsh の場合は、対話形式のジョブのために起動された xterm(1) で、指定されたシェルのパスを使用して、対応するコマンドインタプリタが実行されます。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-t n[-m[:s]]	<p>qsub と qalter でのみ使用できます。いわゆる配列ジョブ、すなわち、添え字番号とで区別され、Sun Grid Engine によって一続きのジョブのように扱われる同一のタスクからなる配列の実行依頼をします。-t のオプション引数には、配列ジョブのタスク数とそれらのタスクに関連づけられている添え字番号を指定します。添え字番号は、SGE_TASK_ID 環境変数を使用してジョブのタスクにエクスポートされます。タスク ID 範囲としては、単一の番号、n-m の形式の簡単な範囲、ステップ付きの範囲を指定することができます。たとえば、タスク ID 範囲として 2-10:2 を指定すると、タスク ID が添字 2、4、6、8、10 の合計で 5 つのタスク (それぞれ、この 5 つの添字番号の 1 つを含む SGE_TASK_ID 環境変数を持つ) を指定したことになります。配列ジョブのタスクはすべて、-t オプションを除き、qsub または qalter コマンド行に指定された資源要求と属性定義を継承します。タスクは個々にスケジューリングされますが、十分な資源がある場合は、独立したジョブのように並行スケジューリングされます。ただし、配列ジョブあるいは部分配列は、qmod(1) または qdel(1) ようなコマンドでは、全体を 1 つとしてアクセスできます。詳細は、対応するマニュアルページを参照してください。一般に配列ジョブは、変化する入力データセットをタスクの添え字番号に相互に関連づけ、それらデータセットに同じ種類の処理を行う用途に使用されます。配列ジョブのタスク数に制限はありません。デフォルトでは、配列ジョブのタスクの STDOUT と STDERR は次の場所にある異なるファイルに書き出されます。</p> <p>&lt;ジョブ名&gt;.[ 'e' 'o']&lt;ジョブ ID&gt;.'&lt;タスク ID&gt;</p> <p>このデフォルトの場所は、上記の -e および -o と疑似環境変数の \$HOME、\$USER、\$JOB_ID、\$JOB_NAME、\$HOSTNAME、\$SGE_TASK_ID を組み合わせることによって変更できます。出力のリダイレクト機能を使用して、すべてのタスクの出力を同じファイルに送ることができますが、そのようにした場合の結果は定義されていません。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-u <i>username,...</i>   -uall	qalter でのみ使用できます。username リストに指定されたユーザーが実行依頼したジョブにだけ変更を加えます。マネージャーの場合は、qalter -uall コマンドを使用してすべてのユーザーのすべてのジョブを変更できます。-u または -uall スイッチを使用した場合、job/task_id_list は指定できません。
-v <i>variable[=value],...</i>	qsub/qrush/qsh/qlogin/qresub/qalter で使用できます。ジョブの実行コンテキストにエクスポートする環境変数を定義または再定義します。-v オプションが指定された場合、Sun Grid Engine はそのスイッチ引数に定義されている環境変数とその値 (値が指定されている場合) をジョブの実行コンテキストに追加します。qalter を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。
-verbose	qrsh と qmake(1) でのみ使用できます。qsh や qlogin と異なり、qrsh は、標準の rsh(1) あるいは rlogin(1) システムコールに準拠したセッションを確立する際にいかなる情報メッセージも出力しません。しかし、-verbose オプションが指定されている場合は、qsh や qlogin コマンドのような詳細な動作をして、rsh(1) または rlogin(1) セッションの確立過程でその情報を出力します。
-verify	qsub/qrush/qsh/qlogin/qalter/qresub で使用できます。ジョブの実行依頼をしないで、現在のコマンド行や関係するあらゆる外部影響などの、ジョブに関する情報だけ表示します。
-V	qresub/qsub/qrush/qsh/qlogin/qalter で使用できます。qsub ユーティリティ内でアクティブなすべての環境変数をジョブのコンテキストにエクスポートするよう指示します。

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
-w e w n v	<p>qsub/qrsh/qsh/qlogin/qalter/qresub で使用できます。qsub、qlogin、qsh の場合は実行依頼するジョブ、qalter の場合はキュー内の指定されたジョブに適用する検査レベルを指定します。表示される情報は、他のジョブのない空のシステムを仮定して、ジョブのスケジューリングが可能かどうかを示します。設定済みの最大しきい値を超える要求、あるいは使用不可能な資源属性を要求する資源要求があると、ジョブがこの検査を通らないことがあります。e、w、n、v はそれぞれ次の検査モードを指定します。</p> <p>e - エラー: 不正な要求のジョブは拒否されます。qrsh、qsh、qlogin のデフォルトです。</p> <p>w - 警告のみ: 不正な要求に対する警告だけ表示します。</p> <p>n - なし: 検査を無効にします。qalter と qsub のデフォルトです。</p> <p>v - 検査: ジョブの実行依頼を行わないで、広範囲にわたる検査レポートを表示します。パフォーマンスが低下するため、デフォルトでは検査は無効になっています。キューの資源可用性の面からジョブ要求が無効になった理由が、qstat (1) - F オプションで説明している形式で表示されます (qstat (1) のマニュアルページの「出力形式」の節の「完全形式」の説明を参照)。</p>
<i>job/task_id_list</i>	<p>次の形式で指定します。</p> <p><i>job_id</i>[.<i>task_range</i>][<i>job_id</i>[.<i>task_range</i>],...] この指定がある場合、<i>task_range</i> は、qrel の処理対象を、ジョブ ID の接尾辞として指定された配列ジョブのタスク範囲に限定します (配列ジョブについての詳細は、qsub (1) の -t オプションを参照)。タスク範囲は、<i>n</i>[-<i>m</i>:<i>s</i>] の形式で指定します。範囲として、単一の番号、<i>n-m</i> の形式の簡単な範囲、ステップ付きの範囲を指定することができます。<i>job/task_id_list</i> 代わりにキーワードの <i>all</i> を使用して、現在のユーザーのすべてのジョブを変更することができます。</p>
scriptfile	<p>qsub でのみ使用できます。ジョブのスクリプトファイルです。この指定がないか、オペランドが単一文字の文字列 (-) の場合、qsub は標準出力からスクリプトを読み取ります。</p>

表 25 実行依頼コマンドのオプション (続き)

オプション	説明
<code>script_args</code>	<code>qsub</code> と <code>qalter</code> でのみ使用できます。ジョブに渡す引数です。標準入力からスクリプトを入力する場合は使用できません。 <code>qalter</code> を使用すると、ジョブの実行中でもこのオプションを変更できます。ただし、パラメータの変更が有効になるのは、ジョブの再開または移動後です。
<code>xterm_args</code>	<code>qsh</code> でのみ使用できます。構成に定義されている <code>xterm(1)</code> 実行可能ファイルに渡す引数です。詳細は、 <code>sge_conf(5)</code> を参照してください。

## 環境変数

表 26 は、`submit` に関する環境変数をまとめています。

表 26 `submit` 関係の環境変数

変数名	説明
<code>SGE_ROOT</code>	Sun Grid Engine の標準構成ファイルの場所を示します。
<code>SGE_CELL</code>	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、 <code>qtcsh</code> は以下を使用します (優先順)。 <ul style="list-style-type: none"> <li>環境変数 <code>SGE_CELL</code> に指定されているセル名 (変数が設定されている場合)</li> <li>デフォルトのセル名 (例: <code>default</code>)</li> </ul>
<code>SGE_DEBUG_LEVEL</code>	設定されている場合は、デバッグ情報を <code>stderr</code> に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
<code>COMMD_PORT</code>	設定されている場合は、 <code>sge_commd(8)</code> が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。
<code>COMMD_HOST</code>	設定されている場合は、Sun Grid Engine が <code>qsub</code> や <code>qsh</code> 、 <code>qlogin</code> 、 <code>qalter</code> クライアントとの通信に使用する特定の <code>sge_commd(8)</code> が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

上記の `-v` または `-V` オプションを使用して、ジョブにエクスポートするように指定された環境変数に加えて、`qsub` や `qsh`、`qlogin` は、指示された値とともに、表 27 に示す変数を変数リストに追加します。

表 27 その他の `qsub`、`qsh`、`qlogin` 関係の環境変数

変数名	説明
<code>SGE_O_HOME</code>	実行依頼クライアントのホームディレクトリ
<code>SGE_O_HOST</code>	実行依頼クライアントが動作しているホスト名
<code>SGE_O_LOGName</code>	実行依頼ホストの <code>LOGNAME</code>
<code>SGE_O_MAIL</code>	実行依頼クライアントの <code>MAIL</code> ( <code>mail</code> ディレクトリ)。
<code>SGE_O_PATH</code>	実行依頼クライアントの実行ファイル検索パス
<code>SGE_O_SHELL</code>	実行依頼ホストの <code>SHELL</code>
<code>SGE_O_TZ</code>	実行依頼クライアントの時間帯
<code>SGE_O_WORKDIR</code>	実行依頼クライアントの現在の作業ディレクトリの絶対パス

Sun Grid Engine ソフトウェアはまた、ジョブの環境に表 28 に示すような変数も組み込みます。

表 28 `submit` コマンドによってジョブ環境に組み込まれる変数

変数名	説明
<code>ARC</code>	ジョブが実行されているノードの Sun Grid Engine アーキテクチャ名。コンパイルで <code>sge_execd(8)</code> バイナリに組み込まれます。
<code>SGE_CKPT_ENV</code>	チェックポイントジョブが実行されるチェックポイント環境 ( <code>-ckpt</code> オプションで選択された環境)。チェックポイントジョブの場合にのみ設定されます。
<code>SGE_CKPT_DIR</code>	チェックポイントジョブの場合にのみ設定されます。 <code>ckpt_dir</code> パスが含まれます (チェックポイントインタフェースの <code>checkpoint(5)</code> を参照)。
<code>SGE_STDERR_PATH</code>	ジョブの標準エラーestreamの書き出し先のファイルのパス名。一般には、 <i>prolog</i> 、 <i>epilog</i> 、並列環境の開始 / 停止、チェックポイントスクリプトからのエラーメッセージを含む出力の強化に使用されます。
<code>SGE_STDOUT_PATH</code>	ジョブの標準出力estreamの書き出し先のファイルのパス名。一般には、 <i>prolog</i> 、 <i>epilog</i> 、並列環境の開始 / 停止、チェックポイントスクリプトからのエラーメッセージを含む出力の強化に使用されます。

表 28 submit コマンドによってジョブ環境に組み込まれる変数 (続き)

変数名	説明
SGE_JOB_SPOOL_DIR	ジョブの実行中に <code>sgc_shepherd</code> (8) がジョブ関連のデータの格納に使用するディレクトリ。このディレクトリは <code>root</code> または <code>Sun Grid Engine</code> 管理アカウントによって所有され、一般に、通常のユーザーが読み取り / 書き込みアクセスで開くことはできません。
SGE_TASK_ID	現在の配列ジョブのタスクの添え字番号 (上記の <code>-t</code> オプションを参照)。各配列ジョブで一意的な番号で、たとえば、さまざまな入力データレコードの参照に利用することができます。非配列ジョブの場合、この環境変数は <code>undefined</code> に設定されます。
ENVIRONMENT	この変数には、 <code>Sun Grid Engine</code> の制御下で実行中のジョブであることを示す <code>BATCH</code> が設定されます。
HOME	<code>passwd</code> (5) ファイルから読み取られたユーザーのホームディレクトリパス
HOSTNAME	ジョブが実行されているノードのホスト名
JOB_ID	ジョブが実行依頼されたときに <code>sgc_qmaster</code> (8) によって割り当てられた一意の識別子。ジョブ ID は 1 ~ 99999 の範囲の 10 進整数です。
JOB_NAME	ジョブ名。対話形式のジョブの場合の <code>INTERACT</code> か、 <code>qsub</code> スクリプトファイル名とピリオド、ジョブ ID の数字からなる名前がどれかです。このデフォルトは、 <code>-N</code> オプションで書き換えることができます。
LOGNAME	<code>passwd</code> (5) ファイルから読み取られたユーザーのログイン名
NHOSTS	並列ジョブが使用するホスト数
NQUEUES	ジョブに割り当てられたキュー数 (シリアルジョブの場合はつねに 1)
NSLOTS	並列ジョブが使用するキューホスト数
PATH	デフォルトのシェル検索パス: <code>/usr/local/bin:/usr/ucb:/bin:/usr/bin</code>
PE	ジョブが実行される並列環境 (並列ジョブのみ)
PE_HOSTFILE	<code>Sun Grid Engine</code> が並列ジョブに割り当てる仮想並列マシンの定義を含むファイルのパスこのファイルの形式についての詳細は、 <code>sgc_pe</code> (5) の <code>\$pe_hostfile</code> パラメータの説明を参照してください。この環境変数は、並列ジョブに対してのみ使用できます。
QUEUE	ジョブが実行されているキューの名前
REQUEST	バッチジョブにのみ使用できます。上記の <code>-N</code> オプションで指定されたか、ジョブスクリプトファイル名として与えられたジョブの要求名



表 28 submit コマンドによってジョブ環境に組み込まれる変数 (続き)

変数名	説明
RESTARTED	システムのクラッシュ後または移動 (チェックポイントジョブの場合) 後にジョブが再開された場合に 1 に設定されます。それ以外の場合は 0 です。
SHELL	passwd (5) ファイルから読み取られたユーザーのログインシェル。注: これは、必ずしもジョブが使用しているシェルではありません。
TMPDIR	ジョブの一時作業ディレクトリへの絶対パス
TMP	TMPDIR 同じです。NQS との互換性を維持するために提供されています。
TZ	sge_execd (8) からインポートされた時間帯変数 (設定されている場合)
USER	passwd (5) ファイルから読み取られたユーザーのログイン名

## 制限事項

Sun Grid Engine の下では、バッチジョブ用の制御端末はありません。このため、制御端末からテストあるいは操作しようとしても失敗します。`.login` あるいは `.cshrc` ファイルにそうした操作が含まれていると、たいていジョブの実行は打ち切られます。

`.login` ファイルのコマンドの前に、バッチジョブに無関係の次のテストを挿入してみてください。

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

シェルの起動ファイル内のこのコードの前にシェルの検索パスを設定することを忘れないでください。

## 終了ステータス

終了ステータスとして返される値には、以下があります。

- 0 - 処理に生成しました。

- 25 - *max\_u\_jobs* または *max\_jobs* 制限が設定されているため、新しいジョブを登録できませんでした。 *sgc\_conf* (5) で追加情報を得られることがあります。
- >0 - エラーが発生しました。

## 使用例

以下は、Sun Grid Engine スクリプトファイルの最も簡単な例です。

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

以下は、もう少し複雑な Sun Grid Engine スクリプトの例です。

```
#!/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
```

## ファイル

- ジョブ #JID の STDOUT - `$REQUEST.oJID[.taskid]`
- ジョブ #JID の STDERR - `$REQUEST.eJID[.taskid]`
- 並列環境ジョブの STDOUT - `$REQUEST.poJID[.taskid]`
- 並列環境ジョブの STDERR - `$REQUEST.peJID[.taskid]`
- 並列環境ジョブのホストファイル - `$REQUEST.hostsJID[.taskid]`
- cwd パスの別名 - `$cwd/.sge_aliases`
- cwd デフォルト要求 - `$cwd/.sge_request`
- ユーザーパスの別名 - `$HOME/.sge_aliases`
- ユーザーのデフォルト要求 - `$HOME/.sge_request`
- クラスタパスの別名 - `<sge_root>/<cell>/common/.sge_aliases`
- クラスタのデフォルト要求 - `<sge_root>/<cell>/common/.sge_request`
- Sun Grid Engine のマスターホストファイル - `<sge_root>/<cell>/common/act_qmaster`

## 関連項目

`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)`

## 著作権

正しく組み込まれている場合、`qrsh` および `qlogin` には、一部、**Regents of the University of California** が著作権を保有する `rsh`、`rshd`、`telnet`、`telnetd` のコードが含まれています。このため、`qrsh` および `qlogin` に関しては、次のことが適用されます。この製品には、カリフォルニア州立大学バークレー校とその寄稿者が開発したソフトウェアが含まれています。

他の権利および許諾条件文については、`sge_intro(1)` と `<sge_root>/3rd_party/qrsh` および `<sge_root>/3rd_party/qlogin` に提供されている情報を参照してください。

---

# access\_list (5)

## 名前

access\_list - Sun Grid Engine のアクセスリストファイルの形式

## 説明

Sun Grid Engine 製品ではアクセスリストを使用して、キュー (queue\_conf (5) を参照) や 並列環境 (seg\_pe (5) を参照) に対するユーザーのアクセス権を定義します。構成済みのアクセスリストのリストは、qconf (1) の -sul オプションを使用して表示することができます。このリストの個々のアクセスリストの内容は、-su スイッチを使用して表示することができます。この出力は、access\_list 形式の説明に従っています。qconf (1) の -au および -du オプションを使用してそれぞれ、アクセスリストの新規作成、変更を行うことができます。

## 形式

1 行にユーザーまたは UNIX ユーザーグループを 1 つ入力します。シンボリック名のみ使用できます。グループ名は、名前の先頭に @ 記号を付けることによってユーザー名と区別します。

## 関連項目

sg\_intro (1)、qconf (1)、sg\_pe (5)、queue\_conf (5)

## 著作権

著作権および許諾条件全文については、sg\_intro (1) を参照してください。

---

# accounting (5)

## 名前

accounting - Sun Grid Engine のアカウントリングファイルの形式

## 説明

Sun Grid Engine の accounting ファイルには、ジョブが 1 つ完了するたびにアカウントリングレコードが 1 つ書き込まれます。qacct(1) はこの accounting ファイルを処理して、アカウントリング統計を集計します。

## 形式

accounting ファイルでは、1 つのジョブが 1 行で表されます。空の行または文字数が 1 文字以下の行は無視されます。アカウントリングレコードのエントリは、コロン(:) 記号で区切られます。以下では、エントリをその出現順に示しています。

- qname - ジョブが実行されたキュー名
- hostname - 実行ホスト名
- group - ジョブの実行に使用された、ジョブ所有者のグループ ID
- owner - ジョブの所有者
- job\_name - ジョブ名
- job\_number - ジョブ ID または ジョブ番号
- account - qsub(1) または qalter(1) の -A オプションで指定されたアカウント文字列
- priority - キュー構成の優先順位パラメータに従ってジョブに割り当てられていた優先順位値(queue\_conf(5) を参照)。
- submission\_time - 秒単位の実行依頼時間 (エポック形式)
- start\_time - 秒単位の開始時間 (エポック形式)。
- end\_time - 秒単位の終了時間 (エポック形式)
- failed - ジョブの所有者が実行ホストマシンで正当なアカウントを持っていなかったなど、実行ホストでジョブを実行できなかった場合の理由を示します。Sun Grid Engine がジョブの実行を複数回試みると、アカウントリングファイルに同じジョブ ID に対応するエントリが複数生成されることがあります。

- `exit_status` - ジョブスクリプトに終了ステータス (また特定のエラー状態が発生した場合は、**Sun Grid Engine** 固有のステータス)。
- `ru_wallclock` - 上記の `end_time` と `start_time` の差  
このカテゴリ内のアカウントングエントリの残りの部分は、`getrusage(2)` に表されている UNIX 標準の `rusage` 構造の内容に準じています。以下のエントリが提供されます。
  - `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` - ジョブが割り当てられていたプロジェクト。プロジェクトは、**Sun Grid Engine, Enterprise Edition** システムでのみ使用できます。
- `department` - ジョブが割り当てられていた部署。部署は、**Sun Grid Engine, Enterprise Edition** システムでのみ使用できます。
- `granted_pe` - ジョブに選択されていた並列環境
- `slots` - スケジューラがジョブにディスパッチしたスロット数
- `task_number` - 配列ジョブタスクの添え字番号
- `cpu` - 秒単位の CPU 時間使用量
- `mem` - G バイト秒単位の統合メモリー使用量
- `io` - 入出力処理で転送されたデータ量
- `category` - ジョブのカテゴリを示す文字列
- `iow` - 秒単位の入出力待ち時間
- `pe_taskid` - この ID がある場合は、並列ジョブを構成するタスクで、`qrsh -inherit` インタフェースを使用して **Sun Grid Engine, Enterprise Edition** に渡されたことを示します。
- `maxvmem` - バイト単位の最大 `vmem` サイズ

## 関連項目

sg\_e\_intro(1)、qacct(1)、qalter(1)、qsub(1)、getrusage(2)、queue\_conf(5)

## 著作権

著作権および許諾条件全文については、sg\_e\_intro(1) を参照してください。

---

# calendar\_conf(5)

## 名前

calendar\_conf - Sun Grid Engine のカレンダー構成ファイルの形式

## 説明

calendar\_conf は、Sun Grid Engine のカレンダー構成の形式を表します。カレンダーの定義を使用して、時刻、曜日、日付に基づいて Sun Grid Engine キューの「当番」時間と「非番」時間が指定されます。さまざまなカレンダーを作成することができ、特定のクラスのジョブに合ったカレンダー定義をキューに関連づけることができます。

calendar\_conf エントリは、qconf(1) の -Acal や -acal、-Mcal、-mcal、-scal、-scall オプション、あるいはグラフィカルユーザーインターフェースの qmon(1) のカレンダー構成ダイアログボックスを使用して追加、変更、表示することができます。以下に、カレンダー構成の形式を示します。

## 形式

- calendar\_name - キューに関連づける際またはカレンダー定義を管理する際に使用するカレンダー名
- year - 日付に基づくキューステータスの定義。一般に、このフィールドは、カレンダーを関連づけるキューが特定の状態に切り替わる日付 (その日の時刻も任意で指定可能) を示します。year フィールドの構文は以下の通りです。

```
year :=
```



```
{year_day_range_list[=daytime_range_list][=state]
|[year_day_range_list=]daytime_range_list[=state]
|[year_day_range_list=][daytime_range_list=]state} ...
```

各パラメータの意味は次のとおりです。

- 必ず *year\_day\_range\_list*、*daytime\_range\_list*、*state* の少なくとも 1 つが存在する必要があります。
- *year\_day\_range\_list* が省略された場合は、その年の毎日とみなされます。
- *daytime\_range\_list* が省略された場合は、終日とみなされます。
- *state* が省略された場合は、キューは非番 (すなわち、使用不可) とみなされます。
- *year\_day\_range\_list* の省略によって暗黙または明示的に参照されていない日、キューは使用可能とみなされます。

意味構文要素は以下のように定義します。

```
year_day_range_list := {yearday-yearday|yearday},...
daytime_range_list := hour[:minute][:second]-hour[:minute][:second],...
state := {on|off|suspended}
year_day := month_day.month.year
month_day := {1|2|...|31}
month := {jan|feb|...|dec|1|2|...|12}
year := {1970|1971|...|2037}
```

- *week* - 曜日に基づくキューステータスの定義。一般に、このフィールドは、カレンダーを関連づけるキューが特定の状態に切り替わる曜日 (その日の時刻も任意で指定可能) を示します。

*week* フィールドの構文は以下の通りです。

```
week :=
```

```
{week_day_range_list[=daytime_range_list][=state]
|[week_day_range_list=]daytime_range_list[=state]
|[week_day_range_list=][daytime_range_list=]state} ...
```

各パラメータの意味は次のとおりです。

- 必ず *week\_day\_range\_list*、*daytime\_range\_list*、*state* の少なくとも 1 つが存在する必要があります。
- *week\_day\_range\_list* が省略された場合は、その週の毎日とみなされます。
- *daytime\_range\_list* および *state* の構文と意味構文は、上記の *year* フィールドで定義されているのと同じです。
- *week\_day\_range\_list* の省略によって暗黙または明示的に参照されていない日、キューは使用可能とみなされます。

*week\_day\_range\_list* は以下のように定義します。  
*week\_day\_range\_list* := {weekday-weekday|weekday},...  
*week\_day* := {mon|tue|wed|thu|fri|sat|sun}

## 意味構文

*year* と *week* フィールド (両者は空白で区切る) への連続したエントリは、以下の規則に従って結合します。

- *off* - *Off* と *on* または *suspended* が重なっている場合は、*on* または *suspended* が優先します。

たとえば、*week 12-18 tue=13-17=on* というエントリは、このカレンダーを参照するキューは、使用可能になる木曜の 13.00 ~ 17.00 を除いてその週の間ずっと使用不可であることを意味します。

## 使用例

ここで紹介する例は、`$sge_root/util/resources/calendars` ディレクトリに含まれています。

- 夜間と週末、祝祭日カレンダー - 「**night**」キューを祝祭日に明示的に使用可能にします。、平日、キューは 6.00 ~ 20.00 の間使用不可になります。土曜と日曜は、暗黙で使用可能な時間とみされなされます。

```
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
week mon-fri=6-20
```

- 日別カレンダー - 平日、キューは 20.00 ~ 6.00 までの夜間の間閉じられます。すなわち、キューは、月曜の 0.00 ~ 6.00、金曜の 20.00 ~ 24.00 の間も閉じられます。土曜と日曜は、使用不可になります。

```
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
week mon-fri=20-6 sat-sun
```

- 一時停止の入った夜間、週末、祝祭日カレンダー - 基本的に最初の例のシナリオと同じですが、キューは「非番」になるのではなく、一時停止されます。

```
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
week mon-fri=6-20=suspended
```

- 一時停止の入った日別カレンダー - 基本的に 2 つ目の例のシナリオと同じですが、キューは「非番」になるのではなく、一時停止されます。

```
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
```

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# checkpoint (5)

## 名前

`checkpoint` - Sun Grid Engine のチェックポイント環境構成ファイルの形式

## 説明

チェックポイント機能は、実行中のプログラムまたはジョブの完全な状態を保存し、システムクラッシュなどで基のプログラムまたはジョブの実行が中止された場合に、その保存した状態を復元して、そのいわゆるチェックポイント から実行を再開する機能です。

Sun Grid Engine では、さまざまなレベルのチェックポイント機能をサポートしています ( `sge_ckpt` (1) を参照)。ここで説明するチェックポイント環境は、Sun Grid Engine クラスタ全体またはその一部で使用されているさまざまなタイプのチェックポイント機能を構成する手段を提供します。チェックポイント環境では、チェックポイント生成の開始に際して実行する必要がある操作や、別のホストへのチェックポイントの移動、チェックポイントが生成されたアプリケーションの再起動、チェックポイントの生成方法を設定可能なキューのリストを定義することができます。

さまざまなオペレーティングシステムをサポートしようとする、チェックポイント構成ファイルの構成でオペレーティングシステムに依存する要素を Sun Grid Engine が取り込まなければならなくなることがよくあります。また、サポートしているオペレーティングシステムのバージョンの更新で、実装の詳細が変わることもよくあります。詳細は、`<sge_root>/doc/checkpointing.asc` ファイルを参照してください。

コマンド行からチェックポイント環境を操作するには、`qconf` (1) コマンドの `-ackpt`、`-dckpt`、`-mckpt`、`-sckpt` を使用するか、対応する `qmon` (1) ダイアログボックス を使用します。`qmon` では、X-Windows を使用した対話形式で構成を行うことができます。

## 形式

checkpoint ファイルの形式は以下のとおりです。

- `ckpt_name` - チェックポイント環境名。 `qsub` (1) の `-ckpt` スイッチまたは上記の `qconf` (1) のオプションで使用されます。
- `interface` - 使用するチェックポイントの種類。現在有効なのは次の種類です。
  - `hibernator` - **Hibernator** というカーネルレベルのチェックポイントインタフェースに接続します。
  - `cpr` - **SGI** というカーネルレベルのチェックポイントインタフェースを使用します。
  - `cray-ckpt` - **Cray** というカーネルレベルのチェックポイントインタフェースとみなされます。
  - `transparent` - このチェックポイントインタフェースを参照して実行依頼されたジョブは、パブリックドメインパッケージの **Condor** が提供するチェックポイントライブラリを使用するとみなされます。

- **userdefined** - このチェックポイントインタフェースを参照して実行依頼されたジョブは、専用のチェックポイント方法を使用するとみなされます。
- **application-level** - 下記の **restart\_command** を除いて、カーネルレベルのチェックポイントインタフェース (**cpr**、**cray-ckpt** など) などのチェックポイントオブジェクトに構成されているインタフェースコマンドをすべてを使用します。この場合、**restart\_command** は設定されていても使用されず、ジョブスクリプトは再開ではなく起動されます。
- **queue\_list** - この並列環境に属する並列ジョブがアクセス権をもつキューのコンマ区切りのリスト
- **ckpt\_command** - チェックポイントを開始する際に **Sun Grid Engine** が実行するコマンド行タイプのコマンド文字列
- **migr\_command** - 別のホストにチェックポイントジョブを移動する際に **Sun Grid Engine** が実行するコマンド行タイプのコマンド文字列
- **restart\_command** - 以前にチェックポイントが生成されたアプリケーションを再起動する際に **Sun Grid Engine** が実行するコマンド行タイプのコマンド文字列
- **clean\_command** - チェックポイントが生成されたアプリケーションの終了後の後処理の際に **Sun Grid Engine** が実行するコマンド行タイプのコマンド文字列
- **ckpt\_dir** - かなりの大きさになる可能性があるチェックポイントを格納するファイルシステムの場所
- **queue\_list** - ジョブの実行依頼時にチェックポイント環境が指定された場合にジョブの実行が可能なキュー名の、コンマまたは空白区切りのリスト
- **ckpt\_signal** - チェックポイントの生成を開始する際に **Sun Grid Engine** がジョブに送信する UNIX シグナル。このフィールドの値としては、**kill(1)** コマンドの **-1** オプションで生成されるリストのシンボリック名、またはチェックポイントに使用するシステムで有効なシグナルを示す整数番号のどちらでも指定できます。
- **when** - チェックポイントを生成するタイミングを指定します。このパラメータの有効な値は、文字 **s**、**m**、**x**、**r** かまたはその組み合わせで構成します。組み合わせる場合、区切り文字はいりません。**qsub(1)** コマンドの **-c** オプションに対するのと同じ文字を使用することができ、その場合は、使用されるチェックポイント環境の定義を書き換えます。文字フィールドの意味は以下のとおりです。
  - **s** - 対応する **sgc\_execd(8)** が停止した場合に、チェックポイントが生成されて、ジョブの実行が打ち切られ、可能な場合は移動されます。
  - **m** - ジョブが実行されるキューに定義されている **min\_cpu\_interval** 間隔で定期的にチェックポイントが生成されます (**queue\_conf(5)** を参照)。
  - **x** - ジョブが手動または自動で一時停止されると、チェックポイントが生成されて、ジョブの実行が打ち切られ、可能な場合は移動されます。
  - **r** - ジョブが実行されているホストが不明な状態になってい、グローバル / ローカルクラスタ構成に定義されている **reschedule\_unknown** 間隔を超えそうな場合に、ジョブがスケジューリングし直されます (**sgc\_conf(5)** を参照)。

## 制限事項

---

注 – Sun Grid Engine のディストリビューションにデフォルトで付属しているあらゆるチェックポイント生成、移動、再開プロシージャの機能、さらには、デフォルトのすべてのチェックポイント環境の *ckpt\_command*, *migr\_command*, *restart\_command* パラメータのそれらプロシージャの起動方法を変更しないでください。変更した場合に機能するかどうかの責任は、すべてチェックポイント環境を構成する管理者にあります。Sun Grid Engine ソフトウェアは単にそれらプロシージャを起動して、その終了ステータスを評価するだけです。プロシージャが正しくその仕事を行わないか、正しい方法で起動されなかった場合は、チェックポイント機能が予期しない動作をすることがあります。Sun Grid Engine システムは、そのことを検出できません。

---

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## sge\_request(5)

### 名前

`sge_request` - Sun Grid Engine のデフォルト要求定義ファイルの形式

### 説明

`sge_request` は、デフォルトの要求プロファイルを定義するファイルの形式を表します。デフォルト要求ファイルが存在する場合は、ジョブの実行依頼で、ジョブスクリプトに埋め込まれた実行依頼オプションと `qsub(1)` または `qsh(1)` コマンド行が

検討される前にそのファイルが読み取られて、処理されます。このため、コマンド行およびスクリプト埋め込みオプションによって、デフォルト要求ファイルの設定が書き換えられることがあります (詳細は、`qsub(1)` または `qsh(1)` を参照)。

デフォルト要求定義ファイルには、クラスタ全体のグローバルファイルとユーザー個人のファイル、作業ディレクトリのローカルファイルがあります。作業ディレクトリのローカルデフォルト要求ファイルの優先順位が最も高く、その後、ユーザー個人、クラスタ全体のグローバルファイルの順になります。

---

**注** – デフォルト要求ファイルやスクリプト埋め込みフラグ、`qsub(1)` または `qsh(1)` コマンド行オプションの以前の設定は、`qsub(1)` または `qsh(1)` の `-clear` オプションを使用することによって、いつでも以前の設定を廃棄することができます。

---

デフォルト要求定義ファイルの形式は以下のとおりです。

- デフォルト要求ファイルには、任意の数の行を含むことができます。空白行と先頭文字位置に # 記号がある行は無視されます。
- 無視する行以外の各行には、任意の `qsub(1)` オプションを含めることができます (『Sun Grid Engine 5.3/Sun Grid Engine, Enterprise Edition 5.3 リファレンスマニュアル』を参照)。1 行に複数のオプションを指定することができます。バッチスクリプトファイルとバッチスクリプトに対する引数オプションは、`qsub(1)` オプションとみなされません。このため、デフォルト要求ファイルでは使用できません。

## 使用例

以下は、デフォルト要求定義ファイルの簡単な例です。

```
# 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
```

このようなデフォルト要求定義ファイルを作成して、次のようにジョブの実行依頼をすると、

```
qsub test.sh
```

以下のコマンドで実行依頼したのと完全に同じ結果になります。

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

## ファイル

- グローバルデフォルトファイル - `<sgc_root>/<cell>/common/sgc_requestst`
- ユーザー個人のデフォルトファイル - `$HOME/.sgc_request`
- `cwd` ディレクトリのデフォルトファイル - `$cwd/.sgc_request`

## 関連項目

`sgc_intro(1)`、`qsh(1)`、`qsub(1)`、『Sun Grid Engine 5.3 管理およびユーザーマニユアル』、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニユアル』

## 著作権

著作権および許諾条件全文については、`sgc_intro(1)` を参照してください。

---

# sgc\_aliases(5)

## 名前

`sgc_aliases` - Sun Grid Engine のパス別名ファイルの形式



## 説明

Sun Grid Engine のパス別名設定機能を使用して、管理者およびユーザーは、分散環境の複雑で不統一なファイルシステム構造 (ホストによってユーザーのホームディレクトリが異なるパスにマウントされているなど) を表わし、Sun Grid Engine がバッチジョブ実行用の適切な作業ディレクトリを確実に見つけられるようにすることができます。

パス別名設定ファイルはシステム全体のグローバルファイルとユーザー個人のローカルファイルがあり、どちらの形式も、`sgc_aliases` で定義します。

- 空白行と先頭文字位置に # 記号がある行は無視されます。
- 空白行と # で始まる行以外の各行には、任意の数の空白文字またはタブで区切った 4 つの文字列が含まれる必要があります。
- 最初の文字列がソースパス、2 つ目が実行依頼ホスト、3 つ目が実行ホスト、4 つ目がソース置換パスを表します。
- 実行依頼ホストおよび実行ホストのエントリは、任意のホストを意味する \* 記号だけ指定できます。

`qsub(1)` の `-cwd` フラグが指定されていると、パス別名設定機能が有効になり、ファイルは以下のように処理されます (この処理は `-cwd` が指定されている場合だけで、それ以外の場合は、ジョブの実行用として実行ホスト上のユーザーのホームディレクトリが選択されます)。

- クラスタ全体のグローバルのパス別名設定ファイルが存在する場合は、`qsub` が物理的な現在の作業ディレクトリのパスを検索した後で、そのファイルが読み取られます。ユーザーのパス別名設定ファイルは、グローバルファイルの最後に付加されているかのように、後で読み取られます。
- ファイルの先頭から省略する行以外の行が 1 行ずつ読み取られ、必要に応じて、それらの行に指定された置換内容が保存されます。
- 置換内容が保存されるのは、実行依頼ホストのエントリが `qsub(1)` コマンドの実行されるホストに一致し、ソースパスが、すでに保存されている現在の作業ディレクトリまたはソース置換パスの先頭部分の構成要素になっている場合だけです。
- 両方のファイルの読み取りを終えるとただちに、保存されているパス別名設定情報が実行依頼されたジョブとともに渡されます。
- 実行ホストで、別名設定情報が評価されます。パス別名の実行ホストエントリが実行ホストに一致する場合は、現在の作業ディレクトリの先頭部分が、ソース置換パスに置き換えられます。

---

**注** – この場合は現在の作業ディレクトリ文字列が変更されること、また以降のパス別名が適用する新しい作業ディレクトリパスに一致する必要があることに注意してください。

---

## 使用例

以下は、`automount(8)` が使用された場合にみられる、パスの不統一に関する問題を解決するパス別名設定ファイルの簡単な例です。

```
# Path Aliasing File
# src-pathsub-hostexec-hostreplacement
/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
```

## ファイル

- グローバル別名ファイル - `<sge_root>/<cell>/common/sge_aliases`
- ユーザーのローカル別名ファイル - `$HOME/.sge_aliases`

## 関連項目

`sge_intro(1)`、`qsub(1)`、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## `sge_conf(5)`

## 名前

`sge_conf` - Sun Grid Engine の構成ファイル

## 説明

`sge_conf` は、Sun Grid Engine のグローバルおよびローカルの構成を定義し、そうした構成は、`qconf(1)` の `-sconf` および `-mconf` オプションを使用して表示・変更することができます。`sge_conf` を変更できるのは、`root` またはクラスタ管理者だけです。

初めて起動されると、`sge_qmaster(8)` は Sun Grid Engine の内部ディレクトリ階層内の既知の場所に有効な Sun Grid Engine 構成が存在するかどうかを調べ、存在する場合は、その構成情報を読み取って次の処理に進みます。存在しない場合は、デフォルト値を含む汎用の構成を同じ場所に書き込みます。起動後、Sun Grid Engine の実行デーモンの `sge_execd(8)` は `sge_qmaster(8)` からその構成を読み出します。

`sge_qmaster(8)` と `sge_execd(8)` 両方の実際の構成は、いわゆるグローバル構成と、マスターまたは実行デーモンが存在するホストに対するローカル構成を組み合わせたものです。ローカル構成が存在する場合、そのエントリによってグローバル構成の対応するエントリが書き換えられます。

---

**注** - ローカル構成が構成エントリのすべてを含む必要はありませんが、グローバルエントリを変更する必要があるエントリだけは必ず含める必要があります。

---

## 形式

ここでは、Sun Grid Engine クラスタのグローバルおよびローカル構成を構成する個々のパラメータを簡単に説明します。

- `qmaster_spool_dir` - マスタープールディレクトリが存在する場所。このディレクトリにアクセスする必要があるのは、`sge_qmaster(8)` と `sge_shadowd(8)` だけです。ただし、`root` には、読み取り・書き込みアクセス権が必要です。クラスタの規模およびジョブ数によっては、マスタープールディレクトリ、具体的にはジョブディレクトリとメッセージログファイルがかなり大きなサイズになることがあります。このため、必ず十分なディスク領域を確保し、たとえば `cron(8)` 使用して定期的にログファイルを整理してください。

インストール時に設定されるパラメータのため、実行中のシステムで `qmaster_spool_dir` を変更することはできません。

マスタープールディレクトリのデフォルトの場所は以下のとおりです。  
`<sge_root>/<cell>/spool/qmaster`

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- `execd_spool_dir` - 実行デーモンのプールディレクトリのパス。ここでもまた `root` に読み取り・書き込みアクセス権が必要です。グローバル構成のこのパラメータのエントリは、実行ホストのローカル構成で書き換えられることがあります。す

なわち、各 `sge_execd(8)` はそれぞれに異なるパスの専用のスプールディレクトリを持つことができ、その場合は、対応する実行ホストの `root` アカウントに読み取り・書き込みアクセス権を付与する必要があります。

`execd_spool_dir` の下には、実行ホストのホスト名に応じた名前のディレクトリが作成され、そのディレクトリにディスクにスプールするすべての情報が書き込まれます。このため、すべての実行ホストの `execd_spool_dirs` が物理的に同じディレクトリパスを参照するように設定することができます (ただし、上記の `root` アクセス制限は満たされる必要がある)。

インストール時に設定されるパラメータのため、実行中のシステムで `execd_spool_dir` を変更することはできません。

実行デーモンのスプールディレクトリのデフォルトの場所は  
`<sge_root>/<cell>/spool`。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- **`binary_path`** - Sun Grid Engine のバイナリが存在するディレクトリのパス。Sun Grid Engine のコンポーネントが他のコンポーネントを検出、起動する際に使用されます。

バイナリの検索は、ここで指定されたパスと、現在のオペレーティングシステムのアーキテクチャと同じディレクトリ名を持つ下位のディレクトリに対して行われます。このため、対応するバイナリが `aix43` や `cray`、`glinux`、`hp10`、`irix6`、`osf4`、`solaris` などの名前のサブディレクトリに存在する場合は、どのアーキテクチャでも、`/usr/Sun Grid Engine/bin` が使用できます。

`sge_execd(8)` はそれぞれに専用のバイナリパスを持つことができます。バイナリパスを変更すると、`sge_execd(8)` に対してただちに有効になります。

バイナリパスのデフォルトの場所は以下のとおりです。

`<sge_root>/bin`

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- **`mailer`** - 使用システムの電子メール送信システムへの絶対パス名。次の構文を受け付ける必要があります。

`mailer -s <メールの件名> <受信者>`

`sge_execd(8)` はそれぞれに専用のメールエージェントを使用することができます。`mailer` を変更すると、ただちに有効になります。

`mailer` のデフォルトは、Sun Grid Engine のマスターのインストールが実行されたホストのオペレーティングシステムによって異なります。一般的な値は `/bin/mail` または `/usr/bin/Mail` です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *xterm* - X Window システムの端末エミュレータ *xterm*(1) への絶対パス名。  
*sge\_execd*(8) はそれぞれに専用のメールエージェントを使用することができます。*xterm* を変更すると、ただちに有効になります。  
デフォルトは `/usr/bin/X11/xterm` です。  
この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。
- *load\_sensor* - 実行可能なシェルスクリプトのパス、または *sge\_execd*(8) が実行するプログラムの、コンマ区切りのリスト。このリストを使用して、構成変更が可能なサイト別の負荷情報 (特定のディスクパーティションの空き領域など) を読み出されます。  
*sge\_execd*(8) はそれぞれに専用の *load\_sensor* プログラムまたはスクリプトを使用できます。*load\_sensor* を変更すると、2つの負荷レポート間隔において有効になります (*load\_repot\_time* を参照)。負荷センサー実行可能ファイルのファイル変更時間が変更されると、自動的に負荷センサーが再起動されます。  
この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。  
*sge\_execd*(8) は、*load\_sensor* で設定された負荷センサーの他に、実行ホストの **Sun Grid Engine** バイナリディレクトリパスに *qloadsensor* という名前の実行可能ファイルが存在しないか調べます。そうしたファイルが見つかった場合は、*load\_sensor* に定義されている構成変更が可能な負荷センサーのように扱われます。この機能は、デフォルトの負荷センサーが事前にインストールされていることを前提にしています。
- *prolog* - 後で実行する **Sun Grid Engine** ジョブに対するのと同じ環境設定で、そのジョブの実行前に実行するシェルスクリプトのパス。省略可能な接頭辞の「*user@*」は、このプロシージャを実行するユーザーを示します。このプロシージャの目的は、**Sun Grid Engine** 管理者が、ジョブと同じコンテキスト情報を必要とする一時ファイルシステムの準備などの一般的なサイト固有の作業を自動化できるようにすることにあります。*sge\_execd*(8) はそれぞれに専用の *prolog* スクリプトを使用することができます。このため、実行ホストのローカル構成は、キュー構成で書き換えられることがあります (*queue\_conf*(5) を参照)。*prolog* を変更すると、ただちに有効になります。

---

**注** - *prolog* はジョブスクリプトそのものとして実行されます。このため、下記の *shell\_start\_mode* と *login\_shells* パラメータで説明しているすべての暗黙事項が適用されます。

---

*prolog* のデフォルト値は特殊な値の `NONE` で、プロログスクリプトの実行を禁止します。

*prolog* プロシージャが解釈する必要のある他のすべての文字列に加えて、実行時に展開される以下の特殊な変数を使用して、コマンド行を構成することができます。

- *\$host* - *prolog* または *epilog* プロシージャが実行されるホスト名

- *\$job\_owner* - ジョブ所有者のユーザー名
- *\$job\_id* - Sun Grid Engine システムの一意的ジョブ識別番号
- *\$job\_name* - ジョブの名前
- *\$processors* - マスターキュー (*prolog* および *epilog* プロシージャが起動されるキュー) のキュー構成に含まれている *processors* 文字列 (*queue\_conf*(5) を参照)
- *\$queue* - マスターキュー (*prolog* および *epilog* プロシージャが実行されるキュー)。この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。
- *epilog* - 完了した Sun Grid Engine ジョブに対するのと同じ環境設定で、そのジョブの実行後に実行するシェルスクリプトのパス。省略可能な接頭辞の「*user@*」は、このプロシージャを実行するユーザーを示します。このプロシージャの目的は、Sun Grid Engine 管理者が、ジョブと同じコンテキスト情報を必要とする一時ファイルシステムの後処理などの一般的なサイト固有の作業を自動化できるようにすることにあります。*sge\_execd*(8) はそれぞれに専用の *epilog* スクリプトを使用することができます。このため、実行ホストのローカル構成は、キュー構成で書き換えられることがあります (*queue\_conf*(5) を参照)。*epilog* を変更すると、ただちに有効になります。

---

**注** - *epilog* はジョブスクリプトそのものとして実行されます。このため、下記の *shell\_start\_mode* と *login\_shells* パラメータについて説明しているすべての暗黙事項が適用されます。

---

*epilog* のデフォルト値は特殊な値の *NONE* で、エピローグスクリプトの実行を禁止します。*prolog* と同じ環境変数を使用して、コマンド行を構成することができます。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *shell\_start\_mode* - 実行ホストで実際にジョブスクリプトを起動するのに使用する機構を指定します。有効な値は以下のとおりです。
  - *unix\_behavior* - UNIX 上で単にジョブのシェルスクリプト名を入力することによって対話形式でジョブを起動した場合、オペレーティングシステムの実行可能ファイルローダーは、そのスクリプトの先頭行にある「*#!/bin/csh*」などのコメントに提供されている情報を使用して、スクリプトを解釈するために起動するコマンドインタプリタを検出します。*shell\_start\_mode* として *unix\_behavior* が指定されている場合、Sun Grid Engine はジョブを起動する際にこの機構を使用します。
  - *posix\_compliant* - POSIX は、「*#!/bin/csh*」などの先頭スクリプト行のコメントを意味のあるものとみなしません。このため、バッチキューイングシステムに関する POSIX 規格 (P1003.2d) では、準拠しているキューイングシステムにそうした行を無視して、代わりにユーザー指定またはデフォルトに設定されているコマンドインタプリタを使用するよう求めています。このため、*shell\_start\_mode* が *posix\_compliant* に設定されている場合、Sun Grid

Engine は `qsub(1)` コマンドの `-S` オプション、または使用するキューの `shell` パラメータに指定されたコマンドインタプリタのいずれかを使用します (詳細は、`queue_conf(5)` を参照)。

- `script_from_stdin-shell_start_mode` を `posix_compliant` または `unix_behavior` のいずれかに設定する場合は、`sgexecd(8)` に対して使用する `umask` を設定して、ユーザーの誰もが対応する実行デーモンのスプールディレクトリ内の `active_jobs` ディレクトリに読み取りアクセスできるようにします。`prolog` および `epilog` スクリプトを設定している場合は、ジョブを実行するユーザーの誰もがそれらのスクリプトを読み取れるようにする必要もあります。

サイトのセキュリティポリシーでこのことが認められない場合は、`shell_start_mode` を `script_from_stdin` に設定することを推奨します。このように設定すると、Sun Grid Engine は、ジョブ所有者のユーザーアカウントに切り替える前にジョブスクリプトばかりでなく、`root` でエピログやプロログスクリプトを強制的に開くことによって、それらのスクリプトを `STDIN` に読み込めるようにします。開かれたスクリプトは、`qsub(1)` コマンドの `-S` オプション、または使用するキューの `shell` パラメータに指定されたコマンドインタプリタの `STDIN` ストリームに取り込まれます (詳細は、`queue_conf(5)` を参照)。

また、`shell_start_mode` を `script_from_stdin` に設定するということは、`poix_compliant` の動作にすることも意味します。

---

**注** – ジョブスクリプトの内部で `rsh(1)` などのコマンドを起動すると、コマンドインタプリタへの `STDIN` ストリームへのスクリプトの取り込みで問題が発生することがあります。これは、そうしたコマンドもコマンドインタプリタの `STDIN` ストリームを処理するためです。通常、この種の問題は、`/dev/null` からコマンドの `STDIN` チャンネルがくるようにリダイレクトすることによって解決することができます (例: `rsh host date < /dev/null`)。

---

---

**注** – 実行シェルには、ジョブに関係するコマンド行オプションがすべて渡されません。有効なシェルオプションと認識されない場合でも、シェルは単にそれらのオプションをジョブに転送します。

---

`shell_start_mode` に対する変更は、ただちに有効になります。デフォルトは `posix_compliant` です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- `login_shells` -Sun Grid Engine では、ジョブスクリプトの起動に Bourne シェル (`sh(1)` を参照) あるいは C シェル (`csh(1)` を参照) などの UNIX コマンドインタプリタを使用することができます。こうした UNIX コマンドインタプリタは、ログインシェルとして起動することも、あるいはコマンド実行用にだけ起動することもできます。ログインシェルとして起動した場合は、起動時に `.login` または

.profile などのシステムおよびユーザーのデフォルトリソースファイルがすべて処理され、ユーザーがログインしたばかりであるかのようにジョブの環境が構成されます。コマンドの実行用にだけ起動した場合は、.cshrc などのシェルに固有のリソースファイルだけが処理され、Sun Grid Engine によって最小のデフォルトの環境が構成されます (qsub (1) を参照)。login\_shells パラメータには、ログインシェルとして起動するコマンドインタプリタの実行可能ファイル名の、コマンド区切りのリストを指定します。

login\_shells に対する変更はただちに有効になります。デフォルトは sh、csh、tcsh、ksh です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **min\_uid** - クラスタを使用可能なユーザー ID の下限を設定します。ユーザー ID (getpwnam(3) で返される) が min\_uid より小さいユーザーは、クラスタでのジョブの実行が許可されません。

min\_uid に対する変更はただちに有効になります。デフォルトは 0 です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **min\_gid** - クラスタを使用可能なグループ ID の下限を設定します。デフォルトのグループ ID (getpwnam(3) で返される) が min\_gid より小さいユーザーは、クラスタでのジョブの実行が許可されません。

min\_gid に対する変更はただちに有効になります。デフォルトは 0 です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **user\_lists** - access\_list(5) で説明している、いわゆるユーザーアクセスリストの、コマンド区切りのリストです。このリストに登録されているアクセスリストの少なくとも 1 つに登録されているユーザーは誰でもクラスタにアクセスできます。user\_lists パラメータが NONE (デフォルト) に設定されている場合は、下記の xuser\_lists パラメータで明示的に除外されていない限り、ユーザーの誰もがクラスタにアクセスできます。ユーザーが xuser\_lists に登録されているアクセスリストと user\_lists に登録されているアクセスリストの両方に登録されている場合、そのユーザーのクラスタへのアクセスは拒否されます。

user\_lists に対する変更はただちに有効になります。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **xuser\_lists** - access\_list(5) で説明している、いわゆるユーザーアクセスリストの、コマンド区切りのリストです。このリストに登録されているアクセスリストの少なくとも 1 つに登録されているユーザーのクラスタへのアクセスは拒否されます。xuser\_lists パラメータが NONE (デフォルト) に設定されている場合、ユーザーの誰もがクラスタにアクセスできます。ユーザーが xuser\_lists に登録されているアクセスリストと上記の user\_lists に登録されているアクセスリストの両方に登録されている場合、そのユーザーのクラスタへのアクセスは拒否されます。



*user\_lists* に対する変更はただちに有効になります。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *administrator\_mail* - 内部生成される問題レポートの送信先のクラスタ管理者の電子メールアドレスの、コンマ区切りのリストです。メールアドレスの書式は、使用電子メールシステムとその構成によって異なります。詳細は、システムのマニュアルを参照してください。

*administrator\_mail* を変更するとただちに有効になります。デフォルトでは、メールリストは空です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *projects* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

*projects* は、Sun Grid Engine, Enterprise Edition システムへのアクセス権を付与するすべてのプロジェクトのリストです。プロジェクトのどれにも属さないユーザーは、Sun Grid Engine, Enterprise Edition システムを使用できません。*projects* リストと下記の *xprojects* に登録されているプロジェクトに属している場合も、システムを使用することはできません。

*projects* を変更するとただちに有効になります。デフォルトは `none` です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *xprojects* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

*xprojects* は、Sun Grid Engine, Enterprise Edition システムへのアクセスを拒否するすべてのプロジェクトのリストです。プロジェクトのいずれかに属するユーザーは、Sun Grid Engine, Enterprise Edition システムを使用できません。上記の *projects* リストと *xprojects* に登録されているプロジェクトに属している場合も、システムを使用することはできません。

*xprojects* を変更するとただちに有効になります。デフォルトは `none` です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *load\_report\_time* - 実行デーモンは定期的にシステム負荷を `sge_qmaster(8)` に報告します。*load\_report\_time* は、負荷報告の間隔を定義します。

`sge_execd(8)` はそれぞれに専用の負荷報告間隔を使用することができます。*load\_report\_time* を変更すると、ただちに有効になります。

---

**注** - *load\_report\_time* を変更するときは注意してください。特に実行ホストの数が多く、あまり頻繁に負荷報告を行うと、*sgc\_qmaster*(8) の動作の妨げになることがあります。また、一般にシステム負荷は滑らかに増減するため、負荷報告回数を多くしてもほとんどメリットはありません。

---

デフォルトは 40 秒です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *reschedule\_unknown* - *unknown* 状態にホスト上のジョブをスケジューリングし直すか、つまり、他のホストに送信するかどうかを決定します。*sgc\_qmaster*(8) がホストの *sgc\_execd*(8) との通信を確立できない場合、ホストは *unknown* と登録されます (*max\_unheard* を参照)。考えられる理由としては、ホストそのものの停止や両者間のネットワーク接続の停止ばかりでなく、*sgc\_execd*(8) がホストで動作していない可能性も考えられます。

どの場合も、Sun Grid Engine はそうしたで実行されているジョブを別のシステムにスケジューリングし直すことができます。*reschedule\_unknown* は、ホストが不明な状態になった後、ジョブの再スケジューリングを開始するまでの待ち時間を制御します。時間を指定する形式は *hh:mm:ss* です。特殊な値 *00:00:00* が指定された場合、このホストからジョブが再スケジューリングされることはありません。

再スケジューリングは、*rerun* フラグが有効なジョブに対してだけ開始されます (*qsub*(1) の *-r y* オプションと *queue\_conf*(5) の *rerun* オプションを参照)。並列ジョブは、そのマスタータスクが実行されているホストが *unknown* 状態の場合にのみ再スケジューリングされます。チェックポイントジョブは、対応するチェックポイント環境の *when* オプションの適切なフラグが設定されている場合にのみ再スケジューリングされます (*checkpoint*(5) を参照)。対話形式のジョブは再スケジューリングされません (*qsh*(1)、*qrsh*(1)、*qtcsh*(1) を参照 *j*)。

*reschedule\_unknown* のデフォルトは *00:00:00* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *stat\_log\_time* - Sun Grid Engine はクラスタに構成されているキューのステータスのスナップショットを定期的にディスクに記録します。*stat\_log\_time* は、このスナップショットを記録する間隔 (秒単位) で定義します。

*stat\_log\_time* を変更するとただちに有効になります。デフォルトは 2 時間 30 分です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *max\_unheard* - *max\_unheard* 秒の間 *sgc\_qmaster*(8) がホストの実行デーモンにアクセスできないか、実行デーモンからアクセスがない場合、そのホスト上のすべてのキューは *unknown* ステータスに設定されます。*sgc\_qmaster*(8) が負荷

レポートを得るには、少なくとも実行デーモンからアクセスされる必要があります。このため、*max\_unheard* には、*load\_report\_time* よりも大きな値を設定します。

*max\_unheard* を変更するとただちに有効になります。デフォルトは 2 時間 30 秒です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *loglevel - sge\_qmaster(8)* や *sge\_execd(8)* などの Sun Grid Engine のコンポーネントは、マスターおよび実行デーモンのスプールディレクトリに情報メッセージや警告・エラーメッセージを書き込みます (上記の *qmaster\_spool\_dir* および *execd\_spool\_dir* の説明を参照)。このパラメータは、そうしたメッセージをどのくらいの詳細さで生成させるかを指定します。指定できるメッセージレベルは以下のとおりです。

- *log\_err* - 検出したすべてのエラーイベントを記録します。
- *log\_warning* - 検出されたすべてのエラーイベントとすべての誤作動の兆候を記録します。
- *log\_info* - 検出されたエラーイベントと誤作動の兆候、各種の情報メッセージをすべて記録します。

*loglevel* を変更すると、ただちに有効になります。

デフォルトは *log\_info* です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *max\_aj\_instances -1* つの配列ジョブで同時に実行されるようスケジューリングするタスクの最大数を定義します。配列ジョブのタスクの場合は、スケジューラから開始順位を受け取ると、マスターデーモン内にインスタンスが作成され、タスクが終了すると、インスタンスが破壊されます。つまり、このパラメータは、主としてマスターおよびスケジューラデーモン内の配列ジョブのメモリー消費量を制御することを可能にします。このパラメータが特に有用なのは、非常に大きなクラスタの非常に大きな配列ジョブに対してです。デフォルトは 2000 個です。値 0 はこの制限を無効にして、クラスタに適切な資源がある限り、スケジューラが配列ジョブのタスクを開始することを可能にします。

*max\_aj\_instances* を変更すると、ただちに有効になります。このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *max\_aj\_tasks -1* つの配列ジョブに含まれる最大タスク数を定義します。*sge\_qmaster(8)* は、*max\_aj\_tasks* 数を超えるタスクを要求する配列ジョブの実行依頼をすべて拒否します。デフォルトは 75000 個です。値 0 はこの制限を無効にします。

*max\_aj\_tasks* を変更すると、ただちに有効になります。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **max\_u\_jobs -Sun Grid Engine** ユーザー 1 人がシステムで一度に持つことができるアクティブなジョブ (未完了ジョブ) 数を制御します。0 より大きな値を指定すると、この制限が課せられます。デフォルト値の 0 は「無制限」を意味します。ジョブの実行依頼が **max\_u\_jobs** 制限を超える場合、その実行依頼コマンドは終了ステータス 25 と対応するエラーメッセージで終了します。

**max\_u\_jobs** を変更すると、ただちに有効になります。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **max\_jobs -Sun Grid Engine** システムで一度に実行できるアクティブなジョブ数 (未完了ジョブ数) を制御します。0 より大きな値を指定すると、この制限が課せられます。デフォルト値の 0 は「無制限」を意味します。ジョブの実行依頼が **max\_jobs** 制限を超える場合、その実行依頼コマンドは終了ステータス 25 と対応するエラーメッセージで終了します。

**max\_jobs** を変更すると、ただちに有効になります。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **enforce\_projects** - このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ使用できます。**Sun Grid Engine** では使用できません。

true に設定した場合、ユーザーはジョブの実行依頼で必ずプロジェクトを要求するよう求められます。詳細は、**qsub (1)** の **-P** オプションを参照してください。

**enforce\_project** を変更すると、ただちに有効になります。デフォルトは **false** です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **enforce\_user** - このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ使用できます。**Sun Grid Engine** では使用できません。

true に設定した場合は、**Sun Grid Engine, Enterprise Edition** の **user (5)** が存在して、ジョブの実行依頼を許可する必要があります。対応するユーザーが存在しない場合、ジョブは拒否されます。

**enforce\_user** を変更すると、ただちに有効になります。デフォルトは **false** です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **set\_token\_cmd** - このパラメータは、使用している **Sun Grid Engine** システムが AFS をサポートするライセンスを供与されている場合のみ存在します。

*Set\_token\_cmd* は、Sun Grid Engine ジョブ用に AFS トークンを設定・延長するコマンドを示します。Sun Grid Engine の標準の AFS ディストリビューションでは、AFS は 2 つのコマンド行パラメータを受け付けるスクリプトになっています。このスクリプトは STDIN からトークンを読み取り、トークンの有効期間を延長して、トークンを設定します。

```
<set_token_cmd> <user> <token_extend_after_seconds>
```

このコマンドは、シェルスクリプトして次のプログラムを呼び出します。

- SetToken
- forge

これらのプログラムは、ディストリビュータからソースコードで提供されます。スクリプトの内容は以下のようになっています。

```
#!/bin/sh
# set_token_cmd
forge -u $1 -t $2 | SetToken
```

*forge* が秘密の AFS サーバー鍵を読み取る必要があるため、サイトでは、*set\_token\_cmd* の部分を、AFS サーバーで自己書き込みのデーモンに接続するコマンドに置き換えることを推奨します。トークンは AFS サーバーで *forge* し、*SetToken* が実行されるローカマシンに復帰させる必要があります。

*set\_token\_cmd* を変更すると、ただちに有効になります。デフォルトは *none* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *pag\_cmd* - このパラメータは、使用している Sun Grid Engine システムが AFS をサポートするライセンスを供与されている場合のみ存在します。

このパラメータを使用して、*pagsh* へのパスを指定します。*sgc\_shepherd(8)* およびジョブは、*pagsh* 内で実行されます。詳細は、AFS の管理者にお尋ねください。

*pag\_cmd* を変更すると、ただちに有効になります。デフォルトは *none* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *token\_extend\_time* - このパラメータは、使用している Sun Grid Engine システムが AFS をサポートするライセンスを供与されている場合のみ存在します。

AFS トークンを定期的に期間延長する期間を設定します。Sun Grid Engine は、ジョブが完了して、対応するトークンが必要なくなるまでは、トークンが期限切れになる 30 分前にトークン延長を呼び出します。

`token_extend_time` を変更すると、ただちに有効になります。デフォルトは 24:0:0 すなわち 24 時間です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- `gid_range` - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

`gid_range` は、 $n$ - $m$  ( $n$  および  $m$  ともに、ゼロ以外の正の整数の番号) の形式の範囲式のリストです。 $m$  は  $m$ - $m$  の省略形になります。`sge_execd(8)` では、これらの番号を使用して、同じジョブに属するプロセスを識別します。

`sge_execd(8)` は、この目的のためにそれぞれ独立した 1 組のグループ ID を使用できます。グループ ID 範囲内の番号はすべて、`sge_execd(8)` で実行されるシステムで使用されていない補助グループ ID である必要があります。

`gid_range` を変更すると、ただちに有効になります。デフォルトはありません。管理者は、Sun Grid Engine, Enterprise Edition のインストール中に `gid_range` 用の値を割り当てておく必要があります。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- `qmaster_params` - Sun Grid Engine の `qmaster` には、追加パラメータを渡すことができます。認識される値は以下のとおりです。
  - `ENABLE_FORCED_QDEL` - このパラメータを設定すると、非管理ユーザーが `qdel(1)` の `-f` オプションを使用して自身のジョブを強制的に削除できるようになります。設定されていない場合は、Sun Grid Engine のマネージャーまたはオペレータのみジョブの強制削除を行うことができます。

---

**注** - ユーザーが Sun Grid Engine の管理者であるかどうかによって、ジョブの強制削除の動作は異なります。管理ユーザーの場合、ジョブは Sun Grid Engine の内部データベースからただちに削除されます。通常の場合、通常の `qdel(1)` と同等の処理がまず実行され、通常のリソース不足が失敗した場合にのみ、強制的な削除が行われます。

---

`qmaster_params` を変更すると、ただちに有効になります。デフォルトは `none` です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- `FORBID_RESCHEDULE` - このパラメータを設定すると、ユーザーの制御下にあるジョブスクリプトによってジョブの再キューイングを開始できなくなります。設定されていない場合、値 99 を返したジョブはスケジューリングし直されます。このパラメータは、たとえば現在のマシンに十分な資源がない場合に、別のマシンでジョブを再開させる目的に利用することができます。
- `DISABLE_AUTO_RESCHEDULING` - `true` または 1 に設定すると、`reschedule_unknown` パラメータが考慮されなくなります。

`qmaster_params` を変更すると、ただちに有効になります。デフォルトは `none` です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- `schedd_params` - このパラメータの目的は、**Sun Grid Engine** のスケジューラに追加のパラメータを渡せるようにすることにあります。現時点で認識される値は以下のとおりです。

- `FLUSH_SUBMIT_SEC`、`FLUSH_FINISH_SEC` - これらのパラメータは、システムのスケジューリング動作をチューニングする目的に使用します。デフォルトでは、スケジューラは、スケジューラ構成 `schedconf` (5) に定義されているスケジュール間隔 (`schedule_interval` パラメータ) で実行されます。

`FLUSH_SUBMIT_SEC`/`FLUSH_FINISH_SEC` パラメータは、スケジューラ実行とジョブの実行依頼 / 完了の時間のずれを定義します。

最速のスケジューラ応答は、両方の値を 0 に設定することによって有効にすることができます。デフォルトのスケジューラ動作は、両方のパラメータを削除するか、両方を -1 に設定すると有効になります。

- `CLASSIC_SGEEEE_SCHEDULING` - `true` または 1 に設定すると、保留中ジョブに対して **Sun Grid Engine, Enterprise Edition** の元のスケジューリングアルゴリズムが使用されます。この元のアルゴリズムでは、アクティブなすべてのジョブと保留中のすべてのジョブの間で各スケジューリングポリシーのチケットを分配することによって、チケットを再計算します。そして、それらのジョブに割り当てられたチケットに基づいて、保留中のジョブリストのジョブを順序づけます。デフォルト値は `false` です。このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ有効です。

- `POLICY_HIERARCHY` - このパラメータは、ポリシーの依存関係を設定します。ポリシーの依存関係では、前のポリシーがその後のポリシーに影響します。代表的なのは、基本割当ポリシーに一時優先ポリシーを優先させる設定です。この場合、同じユーザーまたはプロジェクト間で基本割当チケットを分配する方法が一時優先ポリシーによって左右されます。ポリシー階層の定義に関係なく、すべてのポリシーが特定のジョブに割り当てられたチケット数に関係していることに注意してください。ポリシーごとに求められたチケットは、`POLICY_HIERARCHY` の設定によって変わることがあります。

`POLICY_HIERARCHY` パラメータには、4 つあるポリシーの先頭文字 (基本割当の `S`、業務優先の `F`、締め切り優先の `D`、一時優先の `O`) を 4 文字まで組み合わせることができます。このため、`OFSD` という値は、一時優先ポリシーが業務優先ポリシーに優先し、基本割当ポリシーは業務優先ポリシーの影響を受けて、締め切り優先ポリシーに優先することを意味します。4 文字より少ない場合は、ポリシーの一部が他のポリシーの影響を受けなくなることを意味します。このため、`FS` という値は、基本割当ポリシーだけが業務優先ポリシーによって左右され、他のポリシーは影響を受けないことを意味します。

特殊な値の `NONE` は、ポリシー階層を無効にします。

- `SHARE_OVERRIDE_TICKETS` - `true` または `1` に設定すると、一時優先オブジェクトのインスタンスの一時優先チケットが、そのオブジェクトに関連づけられているすべてのオブジェクト間で均等に分配されます。`false` または `0` に設定すると、各ジョブがそのオブジェクトに割り当てられている一時優先チケット全体を受け取ります。デフォルト値は `true` です。このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ有効です。
- `SHARE_DEADLINE_TICKETS` - `true` または `1` に設定すると、締め切り優先チケット全体がすべての締め切り優先ジョブに分配されます。`false` または `0` に設定すると、締め切りに達したとき、各ジョブが締め切り優先チケット全体を受け取ります。デフォルト値は `true` です。このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ有効です。
- `MAX_FUNCTIONAL_JOBS_TO_SCHEDULE` - 業務優先ポリシーでスケジューリングする保留中のジョブの最大数を定義します。デフォルト値は `200` です。このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ有効です。
- `MAX_PENDING_TASKS_PER_JOB` - 保留中の `1` つの配列ジョブに対してスケジューリングするタスクの最大数を定義します。このパラメータの目的は、スケジューリングのオーバーヘッドを軽減することにあります。デフォルト値は `50` です。このパラメータは、**Sun Grid Engine, Enterprise Edition** システムでのみ有効です。
- `PROFILE` - 設定すると、スケジューラによって、各スケジューリングを要約したプロファイル情報がログに記録されます。

`schedd_params` を変更すると、ただちに有効になります。デフォルトは `none` です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- `execd_params` - このパラメータの目的は、**Sun Grid Engine** の実行デーモンに追加のパラメータを渡せるようにすることにあります。認識される値は以下のとおりです。
  - `ACCT_RESERVED_USAGE` - `true` に設定すると、CPU、mem、io のアカウントingエントリに、測定された使用量ではなく、予約使用量が使用されます。
  - `KEEP_ACTIVE` - このパラメータは、デバッグの目的にのみ使用します。`true` に設定すると、`sge_shepherd(8)` が管理するジョブのプールディレクトリを実行デーモンが削除しなくなります。
  - `NO_REPRIORITIZATION` - `true` または `1` に設定すると、ジョブの動的な優先順位付けが行われなくなります。設定されていない場合は、`sge_execd(8)` 内の `PTF` モジュールによって動的な最優先順位付けが行われるため、`sched_conf(5)` の優先順位パラメータがずっと有効になります (**Sun Grid Engine, Enterprise Edition** システムの場合)。
  - `PTF_MIN_PRIORITY`、`PTF_MAX_PRIORITY` - これらのパラメータは、**Sun Grid Engine, Enterprise Edition** でのみ使用できます。



それぞれ、Sun Grid Engine, Enterprise Edition がジョブに割り当てる最高および最低の優先順位を指定します。nice(2) システムコールを使用した優先順位の設定が可能なシステムの場合、一般にこれらの値は、-20 (最高) ~ 19 (最低) の範囲の負または正の値です。これ以外のシステムの場合は、この範囲とは異なることがあります。

これらのパラメータを削除するか、-999 に設定すると、デフォルトの優先順位 (システムによって異なる) が設定されます。

ホストで事前に定義されているデフォルト値については、実行デーモンの messages ファイルを参照してください。これらの値は、実行デーモンの起動中に記録されます。

- NOTIFY\_KILL - このパラメータを使用すると、SIGKILL シグナルの通知シグナルを変更できます (qsub(1) の -notify オプションを参照)。有効な値はシグナル名 (kill(1) の -l オプションを使用) か 特殊名 none のいずれかです。none に設定した場合、通知シグナルは送信されません。たとえば TERM などのシグナル名を設定すると、通知シグナルとしてそのシグナルが送信されます。
- NOTIFY\_SUSP - このパラメータを使用すると、SIGSTOP シグナルの通知シグナルを変更できます (qsub(1) の -notify オプションを参照)。有効な値はシグナル名 (kill(1) の -l オプションを使用) か 特殊名 none のいずれかです。none に設定した場合、通知シグナルは送信されません。たとえば TSTP などのシグナル名を設定すると、通知シグナルとしてそのシグナルが送信されます。
- SET\_SGE\_ENV、SET\_COD\_ENV、SET\_GRD\_ENV - qsub(1) では、個々の Sun Grid Engine, Enterprise Edition ジョブにエクスポートされる環境変数を一覧表示することができます。そうした環境変数名の一部は、接頭辞 SGE\_ で始まります。以前の Sun Grid Engine, Enterprise Edition 製品 (Codine、GRD) では、そうした変数に接頭辞 COD\_ および GRD\_ が使用されていました。このため、これらの環境に依存するジョブスクリプトはすべて更新する必要があります。SET\_SGE\_ENV と SET\_COD\_ENV、SET\_GRD\_ENV パラメータを利用すると、Codine/GRD から Sun Grid Engine, Enterprise Edition への移行をスムーズに行うことができます。

これらのパラメータはそれぞれ、qsub(1) で取り上げられていて、対応する接頭辞で始まる一群の環境変数がジョブ環境にエクスポートされるようにします。これらのどのパラメータも、true (1) か false (0) に設定することができます。デフォルトは SET\_SGE\_ENV=1、SET\_COD\_ENV=0、SET\_GRD\_ENV=0

3つのパラメータがすべて false か 0 に設定されている場合、対応する環境変数が何も設定されないため、ジョブの実行は失敗する可能性があります。このため、その場合は、接頭辞 SGE\_ で始まる環境変数が設定されます。

- SHARETREE\_RESERVED\_USAGE - このパラメータを true に設定すると、Sun Grid Engine, Enterprise Edition の基本割当ツリーの資源消費量が、計測された使用量ではなく、予約使用量とみなされます。

- **USE\_QSUB\_GID** - このパラメータを **true** に設定すると、ジョブを実行依頼したときにアクティブであった主グループ ID がジョブ実行時の主グループ ID になるように設定されます。設定されていない場合は、実行ホストの **passwd(5)** に定義されているジョブ所有者の主グループ ID が使用されます。

この機能は、**qsub(1)**、**qrsh(1)**、**qmake(1)**、**qtcsh(1)** で実行依頼されたジョブにのみ使用できます。また、**Sun Grid Engine** に付属している **rsh** および **rshd** コンポーネントが使用された場合は (すなわち、**rsh\_daemon** および **rsh\_command** パラメータがデフォルトから変更されている可能性がある)、**qrsh(1)** のジョブにしか機能しません (このため、実際には、**qtcsh(1)** と **qmake(1)** にも機能することになる)。

**execd\_params** を変更すると、ただちに有効になります。デフォルトは **none** です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- **admin\_user** - **Sun Grid Engine** があらゆる内部ファイル処理 (ステータスのスプールやメッセージの記録など) に使用する管理ユーザーアカウント。このパラメータは、たとえば、グローバルな **root** 読み取り・書き込みアクセス権のない共有ファイルシステムであるため、**root** アカウントに対応するファイルアクセス権がない場合に利用できます。

インストール時に設定されるパラメータのため、実行中のシステムで **admin\_user** を変更することはできません。停止中のクラスタで変更できますが、**Sun Grid Engine** のスプール領域へのアクセスが妨害されると、予測不能な動作になります。

このパラメータにはデフォルト値はなく、マスターインストールで定義します。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **finished\_jobs** - **Sun Grid Engine** は最近完了したジョブを一定数保存して、事後のステータス情報を提供します。**finished\_jobs** は、そのために保存する完了ジョブの個数を定義します。最大数に達した場合は、完了ジョブリストに完了した新しいジョブが追加されるたびに、古い完了ジョブが廃棄されます。

**finished\_jobs** を変更すると、ただちに有効になります。デフォルトは **0** です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- **qlogin\_daemon** - **qlogin(1)** 要求のサーバー側で起動する実行可能ファイルを指定します。通常は、サーバー側のシステムの **telnet** デーモンの絶対パス名です。値が指定されなかった場合は、特別に用意された **Sun Grid Engine** コンポーネントが使用されます。

**qlogin\_daemon** を変更すると、ただちに有効になります。デフォルトは **none** です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *qlogin\_command* - *qlogin*(1) 要求のクライアント側で起動するコマンドを指定します。通常は、クライアント側のシステムの *telnet* クライアントプログラムの絶対パス名です。値が指定されなかった場合は、特別に用意された **Sun Grid Engine** コンポーネントが使用されます。このコマンドは、ターゲットホストとポート番号をパラメータとして使用して自動的に起動されます。

*qlogin\_command* を変更すると、ただちに有効になります。デフォルトは *none* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *rlogin\_daemon* - *qrsh* (1) 要求に遠隔実行するコマンド引数の指定がない場合に、そのサーバー側で起動する実行可能ファイルを指定します。通常は、サーバー側のシステムの *rlogin* デーモンの絶対パス名です。値が指定されなかった場合は、特別に用意された **Sun Grid Engine** コンポーネントが使用されます。

*rlogin\_daemon* を変更すると、ただちに有効になります。デフォルトは *none* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *rlogin\_command* - *qrsh* (1) 要求に遠隔実行するコマンド引数が指定されていない場合に、そのクライアント側で起動するコマンドを指定します。通常は、クライアント側のシステムの *rlogin* クライアントプログラムの絶対パス名です。値が指定されなかった場合は、特別に用意された **Sun Grid Engine** コンポーネントが使用されます。このコマンドは、*telnet* (1) に似てターゲットホストとポート番号をパラメータとして使用して自動的に起動されます。**Sun Grid Engine** の *rlogin* クライアントは、ポート番号を引数として受け付けて使用するよう機能拡張されています。使用できるのは、同様にこの構文を認識する *ssh* などのクライアントだけです。

*rlogin\_command* を変更すると、ただちに有効になります。デフォルトは *none* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *rsh\_daemon* - *qrsh* (1) 要求に遠隔実行するコマンド引数が指定されている場合に、そのサーバー側で起動する実行可能ファイルを指定します。通常は、サーバー側のシステムの *rsh* デーモンの絶対パス名です。値が指定されなかった場合は、特別に用意された **Sun Grid Engine** コンポーネントが使用されます。

*rsh\_daemon* を変更すると、ただちに有効になります。デフォルトは *none* です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *rsh\_command* - *qrsh* (1) 要求に遠隔実行するコマンド引数が指定されている場合に、そのクライアント側で起動するコマンドを指定します。通常は、クライアント側のシステムの *rsh* クライアントプログラムの絶対パス名です。値が指定されなかった場合は、特別に用意された **Sun Grid Engine** コンポーネントが使用さ

れます。このコマンドは、telnet(1) に似たターゲットホストとポート番号、それに遠隔実行する引数付きのコマンドをパラメータとして使用して自動的に起動されます。Sun Grid Engine の rsh クライアントは、ポート番号を引数として受け付けて使用するよう機能拡張されています。使用できるのは、同様にこの構文を認識する ssh などのクライアントだけです。

*rsh\_command* を変更すると、ただちに有効になります。デフォルトは none です。

この値のグローバル構成エントリは、実行ホストのローカル構成によって書き換えられることがあります。

- *ignore\_fqdn* - このパラメータを設定すると、システムがホスト名の絶対パス指定されたドメイン名の部分を無視します。Sun Grid Engine クラスタに属するすべてのホストが 1 つの DNS ドメインのメンバーの場合に設定してください。true または 1 に設定すると、この機能が有効になります。有効にすると、クラスタでホスト名の解決方法が異なるために発生する、負荷レポートの問題が解決されることがあります。

インストール時に設定されるパラメータのため、実行中のシステムで *ignore\_fqdn* を変更することはできません。デフォルト値は true です。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

- *default\_domain* - Sun Grid Engine クラスタを構成しているホストが複数の DNS ドメインにまたがっている場合にのみ設定する必要があるパラメータです。ホスト名の解決で、DNS ドメインの 1 つのホストにパス指定ありとなし両方のホスト名が生成される場合に使用することができます。*default\_domain* の値がパス指定なしのホスト名に付加されて、完全指定のホスト名が作成されます。*ignore\_fqdn* が true に設定されている場合、このパラメータは機能しません。

インストール時に設定されるパラメータのため、実行中のシステムで *default\_domain* を変更することはできません。デフォルトは none で、この機能は使用されません。

このパラメータはグローバル構成にだけ存在します。実行ホストのローカル構成で書き換えることはできません。

## 関連項目

sg\_e\_intro(1)、csh(1)、qconf(1)、qsub(1)、rsh(1)、sh(1)、getpwnam(3)、queue\_conf(5)、sched\_conf(5)、sg\_e\_execd(8)、sg\_e\_qmaster(8)、sg\_e\_shepherd(8)、cron(8)、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## `sge_h_aliases(5)`

### 名前

`sge_h_aliases` - Sun Grid Engine のホスト別名ファイルの形式

### 説明

あらゆる Sun Grid Engine コンポーネントは、`sge_commd(8)` が提供するホスト名解決サービスを利用して、一意のホスト名でホストを識別します。`sge_commd(8)` そのももは、DNS や NIS、`/etc/hosts` などの UNIX 標準のディレクトリを参照することによって、ホスト名を解決します。まれにですが、こうした標準のサービスがクリーンに作成できず、さまざまなホストで動作している Sun Grid Engine 通信デーモンが、1 つまたはすべてのホストに対して、あらゆるホストで使用可能な一意のホスト名を自動的に決定できないことがあります。そうした場合は、Sun Grid Engine のホスト別名ファイルを使用して、矛盾のない専用のホスト名解決用データベースを通信デーモンに提供することができます。

ホスト別名ファイルのデフォルトの場所は以下のとおりです。

```
<sge_root>/<cell>/common/host_aliases
```

これとは別のホスト別名ファイルを使用するには、`-a` コマンド行オプションを使用して、`sge_commd(8)` に明示的にそのファイルを指示する必要があります。

### 形式

ホストごとに、空白かコンマ、セミコロン 1 つで区切ったホスト名の別名リストを 1 行で指定する必要があります。最初の別名は、`sge_commd(8)` のホスト名別名設定サービスを使用して、あらゆる Sun Grid Engine コンポーネントが使用する一意のホスト名として定義します。

## 関連項目

`sge_intro(1)`、`sge_commd(8)`

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## `sge_pe(5)`

### 名前

`sge_pe` - Sun Grid Engine の並列環境構成ファイルの形式

### 説明

並列環境は、並列化された共有メモリーまたは分散メモリーアプリケーションの実行を可能にする並列プログラミングおよび実行時間強です。通常、並列アプリケーションを実行するには、そのための並列環境を構成する必要があります。一般的な並列環境としては、たとえば、共有メモリー並列オペレーティングシステムや分散メモリー環境の PVM (Parallel Virtual Machine) あるいは MPI (Message Passing Interface) があります。

`sge_pe` では、任意の並列環境に対するインタフェースを定義することができます。`qconf(1)` の `-ap` または `-mp` オプションを使用して並列環境を定義または変更すると、`qsub(1)` の `-pe` スイッチを使用して、ジョブが割り当てる並列プロセス数の範囲要求とともに、その環境を要求することができます。`-l` オプションを使用して、さらに詳細にジョブの要求を指定することもできます。

### 形式

`sge_pe` ファイルの形式は以下のとおりです。

- `pe_name` - 並列環境名。`qsub(1)` の `-pe` スイッチで使用します。

- *queue\_list* - この並列環境に属する並列ジョブがアクセス権をもつキューのコンマ区切りのリスト
- *slots* - 並列環境で並行実行可能な並列プロセス数
- *user\_lists* - ユーザーアクセスリスト名のコンマ区切りのリスト (*access\_list*(5) を参照)。このリストに登録されているアクセスリストの少なくとも 1 つに登録されているユーザーは誰でも並列環境にアクセスできます。*user\_lists* パラメータが NONE (デフォルト) に設定されている場合は、下記の *xuser\_lists* パラメータで明示的に除外されていない限り、ユーザーの誰もがクラスタにアクセスできます。ユーザーが *xuser\_lists* に登録されているアクセスリストと *user\_lists* に登録されているアクセスリストの両方に登録されている場合、そのユーザーの並列環境へのアクセスは拒否されます。
- *xuser\_lists* - *access\_list*(5) で説明している、いわゆるユーザーアクセスリストの、コンマ区切りのリスト。このリストに登録されているアクセスリストの少なくとも 1 つに登録されているユーザーは誰でも並列環境にアクセスできます。*xuser\_lists* パラメータが NONE (デフォルト) に設定されている場合、ユーザーの誰もがクラスタにアクセスできます。ユーザーが *xuser\_lists* に登録されているアクセスリストと *user\_lists* に登録されているアクセスリストの両方に登録されている場合、そのユーザーの並列環境へのアクセスは拒否されます。
- *start\_proc\_args* - 並列環境用の起動プロシージャの起動コマンド行。起動プロシージャは、ジョブスクリプトを実行する前に *sge\_shepherd*(8) によって起動されます。このプロシージャの目的は、要求に従って並列環境が構成されるようにすることにあります。省略可能な接頭辞の「*user@*」は、このプロシージャを実行するユーザーを示します。起動プロシージャの標準出力は、ジョブの作業ディレクトリ (*qsub*(1) を参照) 内のファイル *REQName.pojID* (*REQName* は *qstat*(1) で表示されるジョブ名、*JID* はジョブの識別番号) にリダイレクトされます。同様に、標準エラー出力は *REQName.pejID* にリダイレクトされます。

起動および停止プロシージャが解釈する必要のある他のすべての文字列に加えて、実行時に展開される以下の特殊な変数を使用して、コマンド行を構成することができます。

- *\$pe\_hostfile* - 起動プロシージャが構成する並列環境のレイアウトに関する詳細な説明の入ったファイルのパス名。ファイルの各行は、並列プロセスを実行するホストに対する参照です。各行の先頭エントリがホスト名、2 つ目がホストで実行する並列プロセス数、3 つ目が使用するプロセッサ範囲 (マルチプロセッサマシンの場合) を表します。
- *\$host* - 起動または停止プロシージャが実行されるホスト名
- *\$job\_owner* - ジョブ所有者のユーザー名
- *\$job\_id* - Sun Grid Engine システムの一意のジョブ識別番号
- *\$job\_name* - ジョブの名前
- *\$pe* - 使用する並列環境名。
- *\$pe\_slots* - ジョブに割り当てるスロット数
- *\$processors* - マスターキュー (起動および停止プロシージャが起動されるキュー) のキュー構成に含まれている *processors* 文字列 (*queue\_conf*(5) を参照)

- `$queue` - マスターキュー (起動および停止プロシージャが起動されるキュー)。
- `stop_proc_args` - 並列環境用の停止プロシージャの起動コマンド行。停止プロシージャは、ジョブスクリプトの完了後に `sge_shepherd(8)` によって起動されます。このプロシージャの目的は、並列環境を停止して、すべての参加システムからその環境が削除されるようにすることにあります。省略可能な接頭辞の「`user@`」は、このプロシージャを実行するユーザーを示します。停止プロシージャの標準出力は、ジョブの作業ディレクトリ (`qsub(1)` を参照) 内のファイル `REQName.pojID` (`REQName` は `qstat(1)` で表示されるジョブ名、`JID` はジョブの識別番号) にリダイレクトされます。同様に、標準エラー出力は `REQName.pejID` にリダイレクトされます。

`start_proc_args` のときと同じ特殊変数を使用して、コマンド行を構成することができます。

- `signal_proc_args` - 並列環境用のシグナル伝達プロシージャの起動コマンド行。シグナル伝達プロシージャは、`qmod(1)` か `qdel(1)` を使用してシグナルが並列ジョブに送信された後必ず、あるいは移動要求があった場合に、`sge_shepherd(8)` によって起動されます。このプロシージャの目的は、並列環境のすべてコンポーネントおよび関係するアプリケーションにシグナルを送信されるようにすることにあります。シグナル伝達プロシージャの標準出力は、ジョブの作業ディレクトリ (`qsub(1)` を参照) 内のファイル `REQName.pojID` (`REQName` は `qstat(1)` で表示されるジョブ名、`JID` はジョブの識別番号) にリダイレクトされます。同様に、標準エラー出力は `REQName.pejID` にリダイレクトされます。

`start_proc_args` のときと同じ特殊変数を使用して、コマンド行を構成することができます。

- `allocation_rule` - 割り当て規則は `sge_schedd(8)` によって解釈され、使用可能なマシンに並列プロセスを分散させる方法をスケジューラが決定するのに役立ちます。たとえば共有メモリアプリケーション専用の並列環境の構成では、使用できる適切なマシンの台数に関係なく、すべての並列プロセスを単一マシンに割り当てる必要があります。これに対し、分散メモリー方式に従った並列環境の構成では、マシン間で均等にプロセスが分散されるようにした方がよいこととなります。

原稿バージョンのスケジューラは、次の割り当て規則のみ理解します。

- **<整数>** - ホスト 1 つあたりの固定プロセス数を示す整数。1 の場合は、すべてのプロセスがさまざまなホストに分散されるようにする必要があります。特殊デノミネータの `$pe_slots` が使用された場合は、割り当てるジョブに対して、その範囲のどの値が最終的に選択されたかに関係なく、`qsub(1)` の `-pe` スイッチで指定された全範囲のプロセスを単一のホストに割り当てる必要があります。
- `$fill_up` - 最適なホスト / キューから始めて、使用可能なすべてのスロットが割り当てられます。並列タスク用のスロットのジョブの要求がある限り、ホストおよびキューが順に「埋められて」いきます。



- `$round_robin` - 並列ジョブが要求するすべてのタスクがディスパッチされるまで、適切なホストからスロットが1つずつ割り当てられます。検出された適切なホスト数を超えるタスクが要求された場合は、最初のホストから割り当てが繰り返されます。この割り当て方法では、適切な順に適切なホストが巡回されます。
- `control_slaves` - このパラメータに設定可能な値は TRUE か FALSE (デフォルト) です。Sun Grid Engine に `sge_execd(8)` および `sge_sshepherd(8)` を使用して並列アプリケーションのスレーブタスクを生成させて、そのアプリケーションのすべてのプロセスを完全に制御するかどうかを指定します。このようにすると、資源制限や適切なアカウンティングなどの機能が使用できるようになりますが、スレーブタスクを制御するには、Sun Grid Engine の機能と緊密に連携する高度な並列環境インタフェースが必要になります。そうした並列環境インタフェースは、サンから入手することができます。

上記以外の場合は、`control_slaves` パラメータを `false` に設定してください。

- `job_is_first_task` - このパラメータは、上記の `control_slaves` が TRUE に設定されている場合、すなわち、Sun Grid Engine が `sge_execd(8)` と `sge_shepherd(8)` を使用してスレーブタスクを作成する場合にのみチェックされます。この場合は、並列環境と Sun Grid Engine を緊密に結合する、高度な並列環境インタフェースが必要になります。そのような並列環境インタフェースに付属する資料では、`job_is_first_task` を設定することを推奨しています。

`job_is_first_task` パラメータは、TRUE または FALSE に設定することができます。TRUE の場合は、Sun Grid Engine ジョブスクリプトにすでに並列アプリケーションのタスクの1つが含まれていることを意味します。FALSE の場合は、ジョブスクリプトとその子プロセスが並列アプリケーションの構成要素ではないことを意味します。

## 制限事項

---

注 - 起動、停止、シグナル伝達プロシージャが機能するかどうかの責任は、すべて並列環境を構成する管理者にあります。Sun Grid Engine ソフトウェアは単にそれらプロシージャを起動して、その終了ステータスを評価するだけです。プロシージャが正しくその仕事を行わなかったとしても、あるいは並列環境または並列アプリケーションが正しく動作しなかったとしても、Sun Grid Engines はそのことを検出できません。

---

## 関連項目

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

# 著作権

著作権および許諾条件全文については、`sgc_intro(1)` を参照してください。

---

## complex(5)

### 名前

`complex` - Sun Grid Engine の複合構成ファイルの形式

### 説明

`complex` は、Sun Grid Engine の複合構成の形式を表します。複合の定義は、`qsub(1)` の `-l` オプションを使用して Sun Grid Engine ジョブ用にユーザーが要求可能な資源属性と Sun Grid Engine システムにおけるそれらパラメータの解釈方法に関するすべての情報を提供します。

複合構成ファイルには、直接アクセスしないでください。複合を追加または変更する場合は、`qconf(1)` オプションの `-Ac`、`-ac`、`-Mc`、`-mc` オプションを使用します。`-Ac` および `-Mc` オプションは引数として複合構成ファイルをとるのに対し、`-ac` および `-mc` オプションは、エディタで複合構成用のテンプレートまたは既存の複合の構成を開きます。

Sun Grid Engine の複合オブジェクトは、次の 4 種類の複合を統合します。

- キュー複合 - 特殊名の *queue* で参照します。

デフォルトでは、キュー複合には、`queue_conf(5)` に定義されているキュー構成のパラメータ群が含まれます。基本的にユーザーがジョブ用に要求可能なキュー構成パラメータは以下のとおりです。

- *qname*
- *hostname*
- *notify*
- *calendar*
- *min\_cpu\_interval*
- *tmpdir*
- *seq\_no*
- *s\_rt*
- *h\_rt*
- *s\_cpu*

- *h\_cpu*
- *s\_data*
- *h\_data*
- *s\_stack*
- *h\_stack*
- *s\_core*
- *h\_core*
- *s\_rss*
- *h\_rss*

キュー複合を拡張して、追加の属性をすべてのキューで使用可能にすることができます。キュー複合は、そこに含まれる属性の、データ型などの特性を定義します。キュー複合属性の値は、上記の標準のパラメータの場合はキューごとのキュー構成、デフォルトのキュー構成にパラメータを追加した場合はキュー構成の *complex\_values* エントリ (詳細は *queue\_conf* (5) を参照) で定義します。後者の場合に、*complex\_values* エントリに値の設定がない場合は、下記の *value* フィールドに定義されている値が設定されます。

- **ホスト複合** - ホスト複合は特殊名の *host* で参照され、ホスト単位で管理することを意図したすべての属性の特性定義が含まれます。ホスト関連の属性の標準セットは2つのカテゴリから構成されますが、上記のキュー複合と同様に拡張することができます。最初のカテゴリは、特にホスト単位の管理に適したいくつかのキュー構成の属性で構成されます。それらの属性は以下のとおりです。
  - *slots*
  - *s\_vmem*
  - *h\_vmem*
  - *s\_fsize*
  - *h\_fsize*

(詳細は *queue\_conf* (5) を参照)

---

**注** - キュー構成でこれらの属性があることと、ホスト複合でこれらの属性を定義することとの間に矛盾はありません。そうすることによって、ホストレベルとキューレベルの両方で対応する資源を管理することができます。たとえば、ホストに関して使用可能な仮想メモリ全体 (*h\_vmem*) を管理できるとともに、その一部をそのホストのキューに関連づけることができます。

---

標準のホスト複合の2つ目の属性カテゴリはデフォルトの負荷値です。各 *sge\_execd* (8) は定期的に負荷を *sge\_qmaster* (8) に報告します。報告される負荷値は、CPU 負荷平均 (*uptime* (1) を参照) などの、Sun Grid Engine 標準の負荷値か、Sun Grid Engine 管理者が定義した負荷値のいずれかです (詳細は、クラスタ構成 *sge\_conf* (5) の *load\_sensor*、『Sun Grid Engine Installation and Administration Guide』、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』を参照)。標準の負荷値の特性定義がデフォルトのホスト複合に含まれるのに対して、管理者定義の負荷値はホスト複合の拡張を必要とします。標準の負荷値セットについての詳細は、*<sge\_root>/doc/load\_parameters.asc* ファイルを参照してください。

一般にホスト複合は、標準以外の負荷パラメータを追加する目的ばかりでなく、ホストに割り当てられるソフトウェアライセンス数やホストのローカルファイルシステムの使用可能なディスク容量などのホスト関連の資源を管理する目的でも拡張されます。

グローバル複合属性の具体的な値は、キュー構成から得られる値の場合は関係するキュー構成、負荷レポート、あるいは対応するホスト構成 (`host_conf(5)` を参照) の `complex_values` エントリの明示的な定義のいずれかで決定されます。上記のどれも該当しない場合は (値は負荷パラメータとされているのだが、`sge_execd(8)` からその負荷値の報告がないなど)、下記の `value` フィールドが使用されます。

- **グローバル複合** - 特殊名の *global* で参照します。

グローバル複合構成のエントリは、特定のソフトウェアの使用可能な「包括」ライセンス数や、ネットワーク全体で使用可能なファイルシステムの空きディスク容量などのクラスタ全体の資源属性を表します。負荷レポートに GLOBAL 識別子を含めることによって、グローバル資源属性を負荷レポートに関連づけ (詳細は、『Sun Grid Engine Installation and Administration Guide』か『Sun Grid Engine 5.3 管理およびユーザーマニュアル』の対応する節を参照)、クラスタ内の任意のホストからグローバル負荷値を報告させることもできます。デフォルトでは、Sun Grid Engine が報告するグローバル負荷値はないため、デフォルトのグローバル複合構成はありません。

ローバル複合属性の具体的な値は、グローバル負荷レポートか、**global** ホスト構成の `complex_values` パラメータの明示的な定義のいずれかで決定されます (`host_conf(5)` を参照)。たとえば、負荷値がまだ報告されていないなどの理由で、このどちらにも存在しない場合は、下記の `value` フィールドが使用されます。

- **ユーザー定義の複合** - Sun Grid Engine 管理者は、ユーザー定義の複合を構成することによって、Sun Grid Engine が管理する属性セットを拡張したり、特定のキューかホスト、またはその両方に対するそれら属性の影響を制限したりすることができます。ユーザー複合は、単に属性と、Sun Grid Engine におけるそれら属性の取り扱い方法に関する定義の、名前付きの集合です。 `complex_list` キューとホスト構成パラメータを使用して、ユーザー定義の複合をキューかホスト、またはその両方に関連づけることができます (`queue_conf(5)` および `host_conf(5)` を参照)。デフォルトの複合属性のほかに、関連づけられた複合に定義されているすべての属性が、キューおよびホストから利用できるようになります。

ユーザー定義の複合の属性の具体的な値は、キューまたはホスト構成の `complex_values` パラメータで設定する必要があります。設定されていない場合は、下記の `value` フィールドが使用されます。

## 形式

*complex* 構成の主要形式は表形式のリスト形式です。# 文字で始まる行はコメント行になります。コメント行以外の各行には、複合の 1 つの要素を定義します。1 つの要素定義行は、出現順に次の 6 つの列エントリで構成されます。

- **name** - `qsub(1)` の `-l` オプションでジョブ用にこの属性を要求するときに使用する複合要素名。属性名は複合全体で 1 回しか使用できません。すなわち、複合属性の定義はあらゆる複合にまたがって一意です。
- **shortcut** - `qsub(1)` の `-l` オプションでジョブ用にこの属性を要求するときに使用可能な、属性名のショートカット。あいまいな複合属性の参照が行われるのを避けるため、属性のショートカットは複合全体で 1 回しか使用できません。
- **type** - 値の比較または負荷複合エントリの負荷スケーリングで Sun Grid Engine が内部的に値をどのように取り扱うかは、この設定によって決まります。
  - INT - 整数のみ使用できます。
  - DOUBLE - 倍精度の浮動小数点数 (10 進の科学的記数法形式) を指定できます。
  - TIME - 時間指定子を使用できます。書式については、`queue_conf(5)` を参照してください。
  - MEMORY - メモリーサイズ指定子を使用できます。書式については、`queue_conf(5)` を参照してください。
  - BOOL - TRUE と FALSE 文字列を使用できます。負荷式で使用された場合、TRUE と FALSE はそれぞれ 1 と 0 にマッピングされます (`sched_conf(5)` を参照)。
  - STRING - あらゆる文字列を使用できます。比較には `strcmp(3)` を使用します。
  - CSTRING - 比較で大文字と小文字が区別されることを除けば STRING に似ています。
  - HOST - CSTRING に似ていますが、文字列が有効なホスト名である必要があります。
- **value** - 属性の定義済みの値を指定します。この値は、キュー、ホスト、Sun Grid Engine クラスタに対して属性の具体的な値を決定する際に、その値が上書きされない場合にのみ意味を持ちます。値フィールドは、以下の値によって上書きされることがあります。
  - 参照先のキューのキュー構成値
  - ホスト固有およびクラスタ関連の負荷値
  - キューまたはホスト構成の `complex_values` パラメータに明示的に指定された値 (詳細は `queue_conf(5)` および `host_conf(5)` を参照)
 上記のどれも該当しない場合に、属性に値が設定されます。
- **relop** - 関係演算子です。関係演算子は、このパラメータに関してユーザーが要求した値を、検討対象のキューに設定されている、対応する値と比較するときに使用します。比較の結果が偽の場合、このキューでジョブを実行することはできません。使用できる演算子は、`==` と `<`、`>`、`<=`、`>=` です。文字列型の属性に有効な演算子は `==` だけです。

- *requestable* - このフィールドが *y* または *yes* に設定されていると、*qsub*(1) 資源要求でこのエントリを使用することができます。 *n* または *no* の場合は、ユーザーがこのエントリを使用してキューまたはキュークラスを要求することはできません。 *force* または *f* の場合、ジョブは必ずこの属性を要求する必要があります。要求がない場合、ジョブは拒否されます。
- *consumable* - *consumable* パラメータは、*yes* (または *y*) または *no* (または *n*) のいずれかに設定できます。 *yes* に設定できるのは、*INT*、*MEMORY*、*TIME* の数値型の属性に対してだけです (上記の *type* を参照)。 *yes* に設定すると、*Sun Grid Engine* の内部ブックキーピング機能で、対応する資源の消費量を管理できるようになります。この場合、*Sun Grid Engine* は、実行中のすべてのジョブについて消費可能資源の消費量を把握し、使用可能な消費可能資源が十分にあることが、その内部ブックキーピングによって明らかである場合にのみジョブがディスパッチされるようにします。消費可能資源は、使用可能なメモリー、ファイルシステムの空き容量、ネットワーク帯域幅、包括的ソフトウェアライセンスなどの限られた資源を管理する効率的な手段です。

消費可能資源はデフォルトまたはユーザー定義の負荷パラメータと結合することができます (*sge\_conf*(5) および *host\_conf*(5) を参照)。すなわち、消費可能属性に関する負荷値を報告させたり、その逆に負荷属性に消費可能フラグを設定したりすることができます。その場合、*Sun Grid Engine* の消費可能資源の管理機能は、負荷 (資源の可用性の測定によって得られる) と内部ブックキーピングの両方を考慮して、どちらも指定されている制限を超えないようにします。

消費可能資源の管理機能を使用するには、資源の基本的な可用性を定義する必要があります。この定義は、クラスタ全体のグローバルレベル、ホストレベル、キューレベルで行うことができ、これらのカテゴリは、列挙したのとは逆の順序で別のカテゴリに優先することができます。すなわち、ホストはクラスタ資源を制限することができ、キューはホストおよびクラスタ資源を制限することができます。資源の可用性の定義は、*host\_conf*(5) および *queue\_conf*(5) の *complex\_values* エントリを使用して行います。 *global* ホストの *complex\_values* の定義では、クラスタ全体のグローバル消費可能資源の設定をします。 *complex\_values* リスト内の消費可能資源の複合属性ごとに、その資源の最大使用可能量を表す値を割り当てます。内部ブックキーピングは、この全体量から、ジョブの資源要求から推定される、実行中のすべてのジョブの資源消費量を差し引きます。

---

**注** - 要求可能パラメータの「強制」値を使用して、ジョブに強制的に資源要求させ、推定消費量を指定することができます (上記の説明を参照)。

---

---

**注** - 管理者は、ジョブが消費可能資源属性を明示的に要求していない場合にデフォルトで使用される資源消費量を事前に定義しておくことができます (下記の *default* パラメータを参照)。これは、前述したように、属性の要求が行われていない場合のみ意味があります。

---

消費可能資源機能の使用例については、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』または『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』を参照してください。

- **default** - デフォルト値が意味があるのは、消費可能な複合属性だけです (上記の *consumable* パラメータを参照)。Sun Grid Engine は、消費可能属性を管理するホストまたはキューにディスパッチされるジョブが、このパラメータに指定された資源量を暗黙で消費するとみなします。qsub(1) の -1 オプションを使用して属性が明示的に要求された場合は、その要求がデフォルト値に優先します。

## 関連項目

sge\_intro(1)、qconf(1)、qdel(1)、qmod(1)、qsub(1)、access\_list(5)、sge\_qmaster(8)、sge\_schedd(8)、sge\_shepherd(8)、『Sun Grid Engine Installation and Administration Guide』、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、sge\_intro(1) を参照してください。

---

# host\_conf(5)

## 名前

host\_conf - Sun Grid Engine の実行ホスト構成ファイルの形式

## 説明

host\_conf は、実行ホスト構成用のテンプレートファイルの形式を表します。qconf(1) の -ae および -me オプションを使用して、クラスタに実行ホストを追加したり、実行ホストの構成を変更したりできます。特定のホストから初めて sge\_execd(8) が sge\_qmaster(8) に登録されると、自動的にデフォルトの実行ホストエントリが追加されます。qconf(1) の -sel スイッチを使用して、Sun Grid Engine システムに構成されている実行ホストのリストを表示することができます。-se オプションは、特定のホストの実行ホスト構成を表示します。

特殊なホスト名の `global` は使用すると、クラスタ全体のグローバル構成を定義することができます。

## 形式

`host_conf` ファイルの形式は以下のとおりです。

- `hostname` - 実行ホスト名
- `load_scaling` - クラスタ全体のグローバル「`host`」複合に定義されていて、ホスト上の `sge_execd(8)` によって報告される負荷値のすべてまたは一部に適用するスケール値の、コンマ区切りのリスト (`complex(5)` を参照)。負荷スケール係数の目的は、実行ホスト間のハードウェアあるいはオペレーティングシステム固有の違いを吸収することにあります。たとえば、マルチプロセッサマシンと単一プロセッサマシンの負荷平均値（「`host`」複合の `load_avg` で設定: 関連項目: `uptime(1)`）を比較できるようにするには、単一 CPU ホストから報告される負荷に、マルチプロセッサ（同じ CPU ハードウェアと仮定）の負荷を基準にした重みを付ける必要があります。負荷スケール係数は、重み付けした負荷値を構成する際に、報告された負荷量に対する乗数となる整数です。つまり、上記の例に従うと、比較できるようにするには、単一プロセッサマシンの負荷値を単一プロセッサマシンのプロセッサ数で乗算する必要があります。

負荷係数を指定する構文は以下のようになります。最初に負荷値名（「`host`」複合で定義）を指定し、等号で区切って、負荷スケール値を指定します。

`load_scaling` 値文字列の途中で空白を使用することはできません。

また、`load_scaling` パラメータは、「`global`」ホストの定義では意味はありません。

- `complex_list` - ホストに関連づける管理者定義の複合の、コンマ区切りのリスト（詳細は、`complex(5)` を参照）。下記の `complex_values` リストで使用できるのは、このリストに列挙された複合に含まれる複合属性と「`global`」および「`host`」複合（この 2 つは暗黙ですべてのホストに関連づけられる）の複合属性だけです。ただし、「`global`」ホストの場合、「`host`」複合は関連づけられず、「`global`」ホストの `complex_values` リストにおいて、デフォルトで使用できるのは、「`global`」複合属性だけです。

このパラメータのデフォルト値は `NONE` で、管理者定義の複合はホストに関連づけられません。

- `complex_values` - このホストで管理されている資源属性に対する割当量を定義します。複合属性の後に等号（`=`）を付け、その後に複合属性の型に従って値を指定します (`complex(5)` を参照)。割当量の指定はコンマで区切ります。属性は、上記の `complex_list` で定義されているもののみ使用できます。

消費可能資源の場合、割当量はホスト上のすべてのジョブの資源消費量に関係し、非消費可能資源の場合は、ジョブスロット単位で解釈されます（消費可能資源についての詳細は、`complex(5)` を参照）。一般に消費可能資源属性は空きメモ



リーや空きディスク領域、使用可能な包括的ソフトウェアライセンスの管理に使用されるのに対し、非消費可能資源属性は、通常、設置されているハードウェアの種類などの特性を定義します。

消費可能資源属性の場合、使用可能な資源量は、*complex\_values* リストの割当量から、ホストで実行中のすべてのジョブの現在の資源消費量を減算することによって求められます。ジョブは、その資源要求が、この方法で求められた資源の可用性を超えない場合にのみホストにディスパッチできます。特定の資源の負荷が監視され、報告された負荷値がその割当量よりも厳密な場合、*complex\_values* リストの割当量の定義は、報告された現在の負荷値で自動的に置き換えられます。こうして、資源の過剰予約が回避されます。

---

**注** – スケーリング (上記の *load\_scaling* を参照) か負荷調整 (*sched\_conf* (5) を参照)、またはその両方が行われるため、割当量の指定に置き換わる負荷値の方が厳密であることがあります。*qstat* (1) の *-F* オプションあるいは、*qmon* (1) のキュー制御ダイアログボックス (*Shift* キーを押しながら、キューのアイコンをクリックすることによって表示) の負荷表示を利用して、消費可能資源の実際の可用性と現在考慮されている値の情報源の詳細な情報を得ることができます。

---

---

**注** – 可用性の計算に使用される実行中のジョブの資源消費量およびディスパッチ待ちのジョブの資源要求量は、ジョブの実行依頼時の明示的なユーザー要求 (*qsub* (1) の *-1* オプションを参照) あるいは、管理者が定義した属性の「*default*」値から得ることができます (*complex* (5) を参照)。*qstat* (1) の *-r* オプションを使用して、システム上のすべてのジョブの実際の資源要求に関する詳細を確認することができます。

---

非消費可能資源の場合、Sun Grid Engine は、単に複合属性定義の関係演算子を考慮しながら、ジョブの属性要求と *complex\_values* の対応する値指定とを比較するだけです。この比較結果が「真」の場合、その属性に関してホストはジョブの実行に適していることとなります。並列ジョブの場合は、並列タスクが占有するすべてのジョブスロットが同じ資源属性値を提供するようになっています。

---

**注** – 消費可能資源として定義できるのは、数値型の複合属性だけです。非数値型の属性はつねにジョブスロット単位で扱われます。

---

このパラメータのデフォルト値は *NONE* で、管理者定義の資源割当量はホストに関連づけられません。

- *load\_values* - このエントリを設定することはできません。*qconf* (1) *-se* コマンドを使用すると表示されます。すべての負荷値は、ホスト上の *sge\_execd* (8) から報告された値がコンマ区切りのリストの形式で表示されます。負荷値はすべて名前から始まり、等号、報告された値と続きます。

- *processors* - このエントリーを設定することはできません。qconf(1) -se コマンドを使用すると表示されます。ホスト上の sge\_execd(8) によって検出されたプロセッサ数です。

- *usage\_scaling* - このエントリーは、Sun Grid Engine, Enterprise Edition システムにのみ存在します。Sun Grid Engine では使用できません。

形式は上記の *load\_scaling* と同じです。ただし、スケーリング対象として有効な属性は、CPU 時間消費量を示す CPU、ジョブの実行時間に集計されたメモリー消費量を示す mem、入出力デバイスを使用して転送されたデータ量を示す io だけです。デフォルトの NONE は「スケーリングなし」を意味します。すなわち、スケーリング係数はすべて 1 です。

- *resource\_capability\_factor* - このエントリーは、Sun Grid Engine, Enterprise Edition システムにのみ存在します。Sun Grid Engine では使用できません。

資源能力係数は、実行ホストにジョブを割り当てる際に Sun Grid Engine, Enterprise Edition によって使用されます。資源能力係数は、実行ホスト間の資源 (CPU、メモリー、入出力など) の比較方法を Sun Grid Engine, Enterprise Edition に指示します。これは、大量の資源 (すなわち、大量のチケット) でを要求するジョブが使用可能な資源が大量にある実行ホストにディスパッチされるようにするのに役立ちます。また、ディスパッチにあたっては、その実行ホストの負荷状況も考慮され、ホストが十分に強力で、負荷が軽い状態になるようにします。

たとえば、各実行ホストに対して、CPU 数や CPU の速度、取り付けられている主メモリーの大きさに基づく資源能力係数を設定することを検討していると仮定します。

$\#\_of\_CPUs * (MHz/200) + GB\_of\_memory$

ここでは、32 個の 200MHz CPU と 10G バイトのメモリーを搭載した実行ホストに資源能力係数 42 を与えられるのに対して、24 個の 200 MHz CPU と 40G バイトのメモリーを搭載して実行ホストには 64 の資源能力係数が与えられます。すなわち、この例では、メモリーが重要な影響力を持ちます。

資源能力係数の設定で検討すべきその他の要素としては、以下があります。

- ジョブのタイプ - CPU 集約またはメモリー集約のジョブ
- CPU ベンチマーク - CPU ベンダーによる比較
- MFLOPS - 複雑な計算の能力
- 入出力能力 - ディスク / ネットワーク速度
- 使用可能なディスク容量 - 実行ホストでの容量

資源能力係数は、倍精度の浮動小数点値として保存されます。使用する値の範囲は重要ではありません。Sun Grid Engine, Enterprise Edition は、さまざまなホストの値と値の関係を考慮するだけです。

## 関連項目

`sge_intro(1)`、`qconf(1)`、`uptime(1)`、`complex(5)`、`sge_execd(8)`、`sge_qmater(8)`

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# hostgroup (5)

## 名前

`hostgroup` - ホストグループエントリファイルの形式

## 説明

`hostgroup` エントリは、ホスト名をグループにマージするために使用します。1つのグループを1つの `hostgroup` エントリファイルで定義します。グループは、名前の先頭に @ 記号を付けることによって参照します。現在の実装状態では、`usermapping(5)` 構成で `hostgroups` を使用することができます。グループ定義ファイル内から、他のグループを参照することもできます。そのようにして参照されるグループをサブグループと呼びます。

`hostgroup` エントリファイルの各行には、そのグループに属するホスト名またはグループを指定します。

既存の `hostgroup` エントリは、`qconf(1)` の `-shgrp1` オプションを使用して一覧表示することができます。特定の `hostgroup` エントリの内容は、`-shgrp` スイッチを使用して表示することができます。この出力は、`hostgroup` 形式の説明に従っています。`hostgroup` エントリは、`qconf(1)` の `-ahgrp` や `-mhgrp`、`-dhgrp` オプションを使用して作成したり、変更したりできます。

## 形式

hostgroup エントリには、少なくとも 2 つのパラメータが含まれます。

- `group_name` キーワード -hostgroup 名は、`group_name` キーワードで定義します。テキスト行の `group_name` キーワード以降は、hostgroup 名値とみなされません。
- `hostname - group_name` で指定されたグループのメンバーになっているホスト名。`hostname` の先頭文字が @ 記号の場合は、名前が hostgroup (5) の参照に使用され、このグループのサブグループとみなされます。

## 使用例

以下は、一般的な hostgroup エントリの例です。

```
group_name bigMachines
@calculate
speedhost
```

このエントリは、bigMachines という新しい hostgroup を定義します。この hostgroup は、ホスト speedhost と hostgroup の calculate のすべてのメンバーで構成されます。

## 関連項目

qconf (1)、usermapping (5)

## 著作権

著作権および許諾条件全文については、sge\_intro (1) を参照してください。

---

# project (5)

## 名前

project - Sun Grid Engine, Enterprise Edition のプロジェクトエントリファイルの形式

## 説明

プロジェクトオブジェクトは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine には、プロジェクトオブジェクトはありません。

Sun Grid Engine, Enterprise Edition では、プロジェクトにジョブの実行依頼をすることができます。プロジェクトは、業務優先または一時優先ポリシーを使用して特定のレベルの重要性で割り当てることができます。指定された重要性レベルは、そのプロジェクトで実行されるジョブによって継承されます。

既存のプロジェクトは、qconf (1) の -sprj1 オプションを使用して表示することができます。このリストの個々のプロジェクトの定義内容は、-sprj スイッチを使用して表示することができます。この出力は、project 形式の説明に従っています。qconf (1) の -aprj、-mprj、-dprj オプションを使用してそれぞれ、プロジェクトの新規作成、変更を行うことができます。

## 形式

プロジェクトの定義には、以下のパラメータが含まれます。

- name - プロジェクト名
- oticket - プロジェクトに割り当てられている一時優先チケット数
- fshare - プロジェクトの現在の業務優先配分量
- facl - プロジェクトへのジョブの実行依頼が許可されているユーザーを参照するユーザーアクセスリストのリスト (access\_list (5) を参照)
- fxacl - プロジェクトへのジョブの実行依頼が禁止されているユーザーを参照するユーザーアクセスリストのリスト (access\_list (5) を参照)

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# qtask (5)

## 名前

`qtask - qtask` ファイルのファイル形式

## 説明

`qtask` ファイルには、`qtcsch(1)` における遠隔実行で、**Sun Grid Engine** に実行依頼するコマンドを定義します。このファイルには、オプションで `qrsh(1)` コマンド行パラメータを含めることができます。指定したパラメータは、コマンドの実行依頼をする際に `qtcsch` が使用する `qrsh(1)` コマンドに渡されます。

クラスタ全体のデフォルト値を定義する グローバルな `qtask` ファイルとユーザー固有の `qtask` ファイルがあり、ユーザー固有の `qtask` ファイルはグローバルな定義に優先し、グローバルな定義を拡張します。グローバルファイルは `<sge_root>/<cell>/common/qtask` に置かれ、ユーザー固有のファイルは `~/.qtask` にあります。管理者は、グローバルファイルでコマンド定義の前に感嘆符を使用することによって、ユーザーによるコマンドの優先指定を拒否することができます。

## 形式

`qtask` ファイルの主要形式は表形式のリスト形式です。# 文字で始まる行はコメント行になります。コメント行以外各行はそれぞれ、遠隔実行するコマンドを 1 つ定義します。

定義はコマンド名で始まり、このコマンドは、`qtcsh(1)` コマンド行で入力するのと完全に同じである必要があります。パス名は使用できません。コマンドそのものは `qtask` ファイルに定義されているものと同じであっても、`qtcsh(1)` コマンド行の絶対または相対パス名はつねにローカル実行されることになります。

コマンド名の後には、任意の個数の `qrsh(1)` オプション引数を続けることができ、それらの引数は、`qtcsh(1)` によって `qrsh(1)` に渡されます。グローバル `qtask` ファイル内でコマンド名の前に感嘆符を使用すると、ユーザー提供の `qtask` ファイルが優先されなくなります。

## 使用例

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

上記の `qtask` ファイルは、**Sun Grid Engine** を使用し、対話形式で `netscape`、`grep`、`verilog` アプリケーションを遠隔実行するよう指定しています。`Netscape` は、`DISPLAY` 変数が `myhost:0` に設定された `Solaris64` アーキテクチャでのみ実行するよう要求しています。`grep` は `filesurfer` というホストでのみ実行されます。`verilog` は、遠隔実行する際に `verilog` ライセンスを使用できるようにすることを要求しています。

## 関連項目

`sge_intro(1)`、`qtcsh(1)`、`qrsh(1)`

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# queue\_conf (5)

## 名前

queue\_conf - Sun Grid Engine のキュー構成ファイルの形式

## 説明

queue\_conf は、キュー構成用のテンプレートファイルの形式を表します。qconf(1) の -aq および -mq オプションを使用して、クラスタにキューを追加したり、キューの構成を変更したりできます。

queue\_conf パラメータは、値として文字列、10 進整数、ブール値、時間、メモリー指定ばかりでなく、コンマ区切りのリストも受け付けます。時間は、正の 10 進か 16 進、または 8 進の整数で指定するか、コロンで区切った 3 つの整数で指定します。整数の場合、値は秒数とみなされます。3 つの整数の場合は、最初の数字が時、2 つ目が分、3 つ目が秒を表します。数字がゼロの場合、その数字は省略できますが、区切りのコロンは残す必要があります。たとえば 1:0:1 は 1::0 と同じことで、1 時間 1 秒を意味します。

メモリー指定では、正の 10 進か 16 進、8 進の整数の後に乗数を表す文字を付けます。有効な乗数文字は k、K、m M で、k は値の 1000 倍、K は値の 1024 倍、m は 1000\*1000 倍、M は 1024\*1024 倍を意味します。乗数がない場合は、単にバイト数とみなされます。

## 形式

以下は、queue\_conf のパラメータをまとめています。

- *qname* - ノード上のキュー名 (文字列型で、テンプレートのデフォルト値は *template*)
- *hostname* - ノードの完全指定のホスト名 (文字列型で、テンプレートのデフォルト値は *host.dom.dom.dom*)
- *seq\_no* - *queue\_sort\_method* の設定によっては (*sched\_conf* (5) を参照)、ホストの負荷状況との関連で、このパラメータが、ジョブのディスパッチ先として適切なキューをスケジューリングするときの、このキューの順番を示すことがあります。



qstat(1) は、queue\_sort\_method の設定に関係なく seq\_no の値で定義される順序でキュー情報を報告します。このパラメータに一本調子で増加する連続番号を設定してください。

- **load\_thresholds** - 負荷しきい値のリスト。しきい値をすでに超えている場合、このノードのキューにジョブがそれ以上スケジューリングされることはなく、qmon(1) はこのノードが過負荷状態であることを示します。「host」および「global」構成で定義されている任意の負荷値を使用することができます (詳細は、complex(5) を参照)。

このリストはコンマ区切りのリストで、各要素は負荷値名と等号、過負荷状態を開始するしきい値で構成します (例: load.avg=175,users\_logged\_in=5)。

---

**注** - 対応する実行ホストの定義で指定されている場合は、負荷値および消費可能資源は、ホストによってスケジューリングが異なることがあります (詳細は、host\_conf(5) を参照) 負荷しきい値は、スケジューリングされた負荷および消費可能資源値をと比較照合されます。

---

- **suspend\_thresholds** - 一時停止しきい値のリスト。構文は、指定されたしきい値を超えると、キュー内の複数のジョブのいずれかの一時停止が開始されることを除けば、上記の **load\_thresholds** パラメータと同じです。一時停止されるジョブ数についての詳細は、下記の **nsuspend** パラメータを参照してください。
- **nsuspend** - **suspend\_thresholds** リストの少なくとも 1 つの一時停止しきい値が超えられた場合に一定の時間が経過するたびに一時停止するか、あるいは **suspend\_threshold** 違反がなくなった場合に一定の時間が経過するたびに使用可能にするジョブ数。すべての **suspend\_thresholds** 値を下回った状態にならない限り、一定の時間が経過するたびに **nsuspend** 個のジョブが一時停止され、キュー内のすべてのジョブが一時停止されることになります。すべての **suspend\_threshold** 値を下回る状態になると、同様の方法でジョブは使用可能にされます。ジョブの一時停止が繰り返される時間間隔は、下記の **suspend\_interval** で定義します。
- **suspend\_interval** - キューが存在するホストの現在の負荷が **suspend\_thresholds** 値の 1 つを超えた場合に **nsuspend** 個のジョブの一時停止を繰り返す時間間隔。この時間間隔は、ジョブを使用可能にする場合にも使用されます。
- **priority** - このキューでジョブを実行するときの **nice**(2) 値を指定します。数値型で、デフォルトはゼロ (明示的に **nice** 値を設定しない) です。負の値 (-20 まで) を設定すると、スケジューリングの優先順位が高くなります。正の値 (+20 まで) を設定すると、スケジューリングの優先順位が低くなります。Sun Grid Engine, Enterprise Edition では、優先順位を動的に調整することによって資源利用資格の目標を実現するため、**priority** 値は何の働きもしません。この動的な優先順位調整は、グローバルまたは実行デーモンのロールクラス構成で **execd\_param** NO\_REPRIORITIZATION を true に設定することによって無効にすることができます。その場合は、Sun Grid Engine, Enterprise Edition システムでも **priority** の設定が意味を持ちます。
- **min\_cpu\_interval** - 透過的なチェックポイントジョブに対する自動的なチェックポイント生成間隔。qsub(1) でユーザーが要求した時間とキュー構成で定義されている時間の大きい方が、チェックポイント間隔として使用されます。チェックポ

イントファイルはかなりの大きさなることがあり、ファイルシステムに書き込むとコストがかさむため、十分に大きな間隔を選択することを推奨します。  
*min\_cpu\_interval* は時間型で、デフォルトは 5 分です (通常、この時間はテストするのに適しています)。

- *processors* - マルチプロセッサ実行ホストの場合に、キューで実行されるジョブが関連づける事が可能なプロセッサのリスト。このパラメータの値の型は、*qsub(1)* の *-pe* オプションのような範囲 (例: 1-4,8,10) で、使用するプロセッサグループのプロセッサ番号を表します。当然、こうした値の解釈は、オペレーティングシステムの仕様によって異なるため、キューホストで動作する *sge\_execd(8)* 内部で行われます。このため、このパラメータの構文解析機能は実行デーモンが提供する必要があり、パラメータは *sge\_qmaster(8)* を介して文字列として渡されます。

現在、サポートされているのは、IRIX 6.2 を実行する SGI マルチプロセッサマシンと Digital UNIX マルチプロセッサマシンだけです。Digital UNIX の場合は、1 つのプロセッサセットで一度にジョブを 1 つだけ実行することができます。すなわち、この場合、下記の *slots* を 1 に設定します。

- *qtype* - キューの種類。現在は、バッチ、対話形式、並列、チェックポイントのいずれかかその任意の組み合わせが可能で、コマンドで区切って指定します。文字列型で、デフォルトは *batch interactive parallel* です。
- *rerun* - キューホストのシステムクラッシュまたは Sun Grid Engine システム全体 (ジョブの *sge\_shepherd(8)* およびそのプロセス階層を含む) の *kill(1)* を使用した手動停止が原因でジョブの実行が中止された場合の、ジョブのデフォルトの動作を定義します。ジョブの再開が可能な場合、*sge\_execd(8)* が再起動され、ジョブの実行がこのような理由で中止されたことを検出すると、ただちにジョブを再開することができます。ただし、たとえば、データベースが更新されている場合 (データベース / ファイルの同じレコードが、読み取られた後、書き込みが行われている場合) は、ジョブの実行中止によってデータベースの整合性が失われている可能性があるため、ジョブを再開することはできません。ジョブの所有者は、*qsub(1)* の *-r* オプションを使用することによってキュー内のジョブのデフォルトの動作を無効にすることができます。

このパラメータはブール型で、TRUE または FALSE のいずれかを指定できます。デフォルトは FALSE で、ジョブが自動的に再開されることはありません。

- *slots* - キューで並行実行可能なジョブの最大数。数値型です。
- *tmpdir* - 一時ディレクトリファイルシステムのベースへの絶対パスを指定します。ジョブを起動すると、*sge\_execd(8)* は、ジョブの実行中のスクラッチファイルを保持する目的でこのファイルシステムに一意の名前を持つディレクトリを作成します。ジョブが終了すると、作成されたディレクトリとその内容は自動的に削除されます。環境変数の *TMPDIR* と *TMP* には、各ジョブのスクラッチディレクトリのパスを設定します (文字列型で、デフォルトは */tmp*)。
- *shell* - *sge\_conf(5)* で *shell\_start\_mode* パラメータが *posix\_compliant* または *script\_from\_stdin* に設定されている場合に、*shell* パラメータはキューで実行されるジョブスクリプトの処理に使用するコマンドインタプリタ (*sh(1)* あるいは *csh(1)* など) の実行可能ファイルのパスを示します。この *shell* の定義は、*qsub(1)* の *-S* オプションを使用して無効にすることができます。

パラメータの型は文字列型で、デフォルトは `/bin/csh` です。

- *shell\_start\_mode* - 実行ホストで実際にジョブスクリプトを起動するのに使用する機構を指定します。認識される値は以下のとおりです。
  - *unix\_behavior* - UNIX 上で単にジョブのシェルスクリプト名を入力することによって対話形式でスクリプトを起動した場合、オペレーティングシステムの実行可能ファイルローダーは、そのスクリプトの先頭行にある「`#!/bin/csh`」などのコメントに提供されている情報を使用して、スクリプトを解釈するために起動するコマンドインタプリタを検出します。*shell\_start\_mode* として *unix\_behavior* が指定されている場合、Sun Grid Engine はジョブを起動する際にこの機構を使用します。
  - *posix\_compliant* - POSIX は、「`#!/bin/csh`」などの先頭スクリプト行のコメントを意味のあるものとみなしません。このため、バッチキューイングシステムに関する POSIX 規格 (P1003.2d) では、準拠しているキューイングシステムにそうした行を無視して、代わりにユーザー指定またはデフォルトに設定されているコマンドインタプリタを使用するよう求めています。このため、*shell\_start\_mode* が *posix\_compliant* に設定されている場合、Sun Grid Engine は `qusb(1)` コマンドの `-S` オプション、または使用するキューの *shell* パラメータに指定されたコマンドインタプリタのいずれかを使用します (上記を参照)。
  - *script\_from\_stdin* - *shell\_start\_mode* を *posix\_compliant* または *unix\_behavior* のいずれかに設定する場合は、`sge_execd(8)` に対して使用する `umask` を設定して、ユーザーの誰もが対応する実行デーモンのスプールディレクトリ内の `active_jobs` ディレクトリに読み取りアクセスできるようにします。*prolog* および *epilog* スクリプトを設定している場合は、ジョブを実行するユーザーの誰もがそれらのスクリプトを読み取れるようにする必要があります。

サイトのセキュリティポリシーでこのことが認められない場合は、*shell\_start\_mode* を *script\_from\_stdin* に設定することを推奨します。このように設定すると、Sun Grid Engine は、ジョブ所有者のユーザーアカウントに切り替える前にジョブスクリプトばかりでなく、`root` でエピログやプロログスクリプトを強制的に開くことによって (`sge_execd(8)` が `root` で起動されている場合)、それらのスクリプトを STDIN に読み込めるようにします。開かれたスクリプトは、`qsub(1)` コマンドの `-S` オプション、または使用するキューの *shell* パラメータに指定されたコマンドインタプリタの STDIN ストリームに取り込まれます (上記を参照)。

また、*shell\_start\_mode* を *script\_from\_stdin* に設定するということは、*poix\_compliant* の動作にすることも意味します。

---

**注** - ジョブスクリプトの内部で `rsh(1)` などのコマンドを起動すると、コマンドインタプリタへの STDIN ストリームへのスクリプトの取り込みで問題が発生することがあります。これは、そうしたコマンドもコマンドインタプリタの STDIN ストリームを処理するためです。通常、この種の問題は、`/dev/null` からコマンドの STDIN チャネルがくるようにリダイレクトすることによって解決することができます (例: `rsh host date < /dev/null`)。

---

---

**注** - 実行シェルには、ジョブに関係するコマンド行オプションがすべて渡されます。有効なシェルオプションと認識されない場合でも、シェルは単にそれらのオプションをジョブに転送します。

---

デフォルトは *posix\_compliant* です。

- **prolog** - 後で実行する Sun Grid Engine ジョブに対するのと同じ環境設定で、そのジョブの実行前に実行するシェルスクリプトのパス。省略可能な接頭辞の「*user@*」は、このプロシーダを実行するユーザーを示します。このプロシーダの目的は、Sun Grid Engine 管理者が、ジョブと同じコンテキスト情報を必要とする一時ファイルシステムの準備などの一般的なサイト固有の作業を自動化できるようにすることにあります。グローバルまたは実行ホスト固有の *prolog* の定義は、このキュー構成エントリによって書き換えられます (*sge\_conf*(5) を参照)。

---

**注** - *prolog* はジョブスクリプトそのものとして実行されます。このため、下記の *shell\_start\_mode* と *login\_shells* パラメータで説明しているすべての暗黙事項が適用されます。

---

*prolog* のデフォルト値は特殊な値の *NONE* で、プロローグスクリプトの実行を禁止します。コマンド行の構成では、クラスタ構成の *prolog* 定義で使用できるのと同じ特殊変数を使用できます (*sge\_conf*(5) を参照)。

- **epilog** - 完了した Sun Grid Engine ジョブに対するのと同じ環境設定で、そのジョブの実行後に実行するシェルスクリプトのパス。省略可能な接頭辞の「*user@*」は、このプロシーダを実行するユーザーを示します。このプロシーダの目的は、Sun Grid Engine 管理者が、ジョブと同じコンテキスト情報を必要とする一時ファイルシステムの後処理などの一般的なサイト固有の作業を自動化できるようにすることにあります。グローバルまたは実行ホスト固有の *epilog* の定義は、このキュー構成エントリによって書き換えられます (*sge\_conf*(5) を参照)。

---

**注** - *epilog* はジョブスクリプトそのものとして実行されます。このため、下記の *shell\_start\_mode* と *login\_shells* パラメータで説明しているすべての暗黙事項が適用されます。

---

*epilog* のデフォルト値は特殊な値の *NONE* で、エピローグスクリプトの実行を禁止します。コマンド行の構成では、クラスタ構成の *prolog* 定義で使用できるのと同じ特殊変数を使用できます (*sge\_conf*(5) を参照)。

- **starter\_method** - バッチジョブの起動用のジョブ起動機能として使用する、すなわち、ジョブを起動する際に、設定されているシェルの代わりに実行する実行可能ファイルのパスを指定します。ジョブの引数は、ジョブ起動機能の引数として扱われます。また、次の環境変数を使用して、ジョブの起動用に設定されているか、要求されたシェル環境に関する情報をジョブ起動機能に渡すことができます。

- `SGE_STARTER_SHELL_PATH` - ジョブの起動用として要求するシェル名
- `SGE_STARTER_SHELL_START_MODE` - 設定されている `shell_start_mode`.
- `SGE_STARTER_USE_LOGIN_SHELL` - シェルをログインシェルとして使用する場合に `true` に設定されます (`sge_conf(5)` の `login_shells` を参照)。

`qsh`、`qlogin`、`qrsh` が `rlogin` として機能する場合、`starter_method` は起動されません。

- `suspend_method`、`resume_method`、`terminate_method` - これらのパラメータを使用して、ジョブの一時停止、停止解除、ジョブの終了で **Sun Grid Engine** がデフォルトで使用する方法を書き換えることができます。デフォルトでは、これらの処理を行う際、ジョブにはそれぞれ `SIGSTOP`、`SIGCONT`、`SIGKILL` が送信されます。しかし、このデフォルトの動作はアプリケーションによっては適切でないことがあります。

実行可能ファイルのパスが指定されていない場合、**Sun Grid Engine** はデフォルトのシグナルの代わりに送信するシグナルとして、これらのパラメータに指定されたエントリを使用します。シグナルは、正数、あるいは `kill -1` で出力される `SIG` から始まるシグナル名 (例: `SIGTERM`) で指定する必要があります。

実行可能ファイルのパス (「/」で始まる絶対パスである必要がある) が指定されている場合、**Sun Grid Engine** は、その引数とともに指定されたコマンドを実行して、適切な処理を行います。プロシージャが解釈する必要のある他のすべての文字列に加えて、実行時に展開される以下の特殊な変数を使用して、コマンド行を構成することができます。

- `$host` - プロシージャを実行するホスト名
- `$job_owner` - ジョブ所有者のユーザー名
- `$job_id` - **Sun Grid Engine** システムの一意のジョブ識別番号
- `$job_name` - ジョブの名前
- `$queue` - キューの名前
- `$job_pid` - ジョブの PID
- `notify` - `qsub(1)` の `-notify` オプションを使用してジョブが実行依頼された場合に、`SIGUSR1/SIGUSR2` 通知シグナルを送信してから一時停止 / 終了シグナルを送信するまでの待ち時間。
- `owner_list` - このキューを一時停止する権限を持つユーザーのログイン名の、コマ区切りのリスト (デフォルトでは、**Sun Grid Engine** のオペレータとマネージャはキューを一時停止できます)。通例では、このフィールドは、対話セッションと **Sun Grid Engine** ジョブの間で計算資源が共有される対話型のワークステーション上のキューの場合に設定し、ワークステーションの所有者が優先的なアクセス権を持つようにします (文字列型で、デフォルトは `NONE`)。
- `user_lists` - `access_list(5)` で説明している、いわゆるユーザーアクセスリストの、コマ区切りのリスト。このリストに登録されているアクセスリストの少なくとも 1 つに登録されているユーザーは誰でもキューにアクセスできます。  
`user_lists` パラメータが `NONE` (デフォルト) に設定されている場合は、下記の `xuser_lists` パラメータで明示的に除外されていない限り、ユーザーの誰もが

キューにアクセスできます。ユーザーが *xuser\_lists* に登録されているアクセスリストと *user\_lists* に登録されているアクセスリストの両方に登録されている場合、そのユーザーのキューへのアクセスは拒否されます。

- *xuser\_lists* - *access\_list* (5) で説明している、いわゆるユーザーアクセスリストの、コンマ区切りのリスト。このリストに登録されているアクセスリストの少なくとも 1 つに登録されているユーザーは、このキューへのアクセスが拒否されます。*xuser\_lists* パラメータが NONE (デフォルト) に設定されている場合は、ユーザーの誰もがこのキュー環境にアクセスできます。ユーザーが *xuser\_lists* に登録されているアクセスリストと *user\_lists* に登録されているアクセスリストの両方に登録されている場合、そのユーザーのキューへのアクセスは拒否されます。
  - *projects* - キューへのアクセス権を持つプロジェクトの、コンマ区切りのリスト。このリストに登録されているプロジェクトの、このキューへのアクセスは拒否されます。デフォルトの NONE に設定されている場合は、下記の *xprojects* パラメータで明示的に除外されていない限り、あらゆるプロジェクトがこのキューにアクセスできます。プロジェクトが *projects* と *xprojects* パラメータの両方に指定されている場合、そのプロジェクトのキューへのアクセスは拒否されます。このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。
  - *xprojects* - キューへのアクセスを拒否するプロジェクトの、コンマ区切りのリスト。デフォルトの NONE に設定されている場合は、上記の *projects* パラメータに基づいてアクセスが拒否されるものを除いて、プロジェクトのアクセスが拒否されることはありません。プロジェクトが *projects* と *xprojects* パラメータの両方に指定されている場合、そのプロジェクトのキューへのアクセスは拒否されます。このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。
  - *subordinate\_list* - この Sun Grid Engine キューと同じホスト上にあつて、このキューで実行中のジョブが特定の個数に達したときに一時停止するキューのリスト。リストの指定形式は、上記の *load\_threshods* パラメータと同じで、たとえば *low\_pri\_q=5,small\_q* というように指定します。数字は、従属キューの一時停止を開始するまでに埋められる必要があるキューのジョブスロット数を示します。値が設定されていない場合は、キューのすべてのスロットが埋まった場合に、一時停止が開始されます。
- 複数キューのホストになっているノードでは、特定のクラスのジョブにより良いサービスを提供するようにすることを推奨します。たとえば、並列処理専用のキューは、通常の生産キューより優先順位を高くする、といった具合です。デフォルトは NONE です。
- *complex\_list* - キューに関連づける管理者定義の複合の、コンマ区切りのリスト (詳細は、*complex* (5) を参照)。下記の *complex\_values* リストで使用できるのは、このリストに列挙された複合に含まれる複合属性と「*global*」、「*host*」、「*queue*」複合 (この 3 つは暗黙ですべてのキューに関連づけられる) の複合属性だけです。

このパラメータのデフォルト値は NONE で、管理者定義の複合はキューに関連づけられません。

- *complex\_values* - このキューで管理されている資源属性に対する割当量を定義します。このパラメータで使用可能な複合属性は、上記の *complex\_list* で定義します。構文は、上記の *load\_thresholds* と同じです。消費可能資源の場合、割当量はキュー内のすべてのジョブの資源消費量に関係し、非消費可能資源の場合は、ジョブスロット単位で解釈されます (消費可能資源についての詳細は、*complex(5)* を参照)。一般に消費可能資源属性は空きメモリーや空きディスク領域、使用可能な包括的ソフトウェアライセンスの管理に使用されるのに対し、非消費可能資源属性は、通常、設置されているハードウェアの種類などの特性を定義します。

消費可能資源属性の場合、使用可能な資源量は、*complex\_values* リストの割当量から、キューで実行中のすべてのジョブの現在の資源消費量を減算することによって求められます。ジョブは、その資源要求が、この方法で求められた資源の可用性を超えない場合のみキューにディスパッチできます。特定の資源の負荷が監視され、報告された負荷値がその割当量よりも厳密な場合、*complex\_values* リストの割当量の定義は、報告された現在の負荷値で自動的に置き換えられます。こうして、資源の過剰予約が回避されます。

---

**注** - スケーリング (*host\_conf(5)* を参照) か負荷調整 (*sched\_conf(5)* を参照)、またはその両方が行われるため、割当量の指定に置き換わる負荷値の方が厳密であることがあります。*qstat(1)* の *-F* オプションあるいは、*qmon(1)* のキュー制御ダイアログボックス (*Shift* キーを押しながら、キューのアイコンをクリックすることによって表示) の負荷表示を利用して、消費可能資源の実際の可用性と現在考慮されている値の情報源の詳細な情報を得ることができます。

---

---

**注** - 可用性の計算に使用される実行中のジョブの資源消費量およびディスパッチ待ちのジョブの資源要求量は、ジョブの実行依頼時の明示的なユーザー要求 (*qsub(1)* の *-1* オプションを参照) あるいは、管理者が定義した属性の「*default*」値から得ることができます (*complex(5)* を参照)。*qstat(1)* の *-r* オプションを使用して、システム上のすべてのジョブの実際の資源要求に関する詳細を確認することができます。

---

非消費可能資源の場合、*Sun Grid Engine* は、単に複合属性定義の関係演算子を考慮しながら、ジョブの属性要求と *complex\_values* の対応する値指定とを比較するだけです。この比較結果が「真」の場合、その属性に関してキューはジョブの実行に適していることとなります。並列ジョブの場合は、並列タスクが占有するすべてのジョブスロットが同じ資源属性値を提供するようになっています。

---

**注** - 消費可能資源として定義できるのは、数値型の複合属性だけです。非数値型の属性はつねにジョブスロット単位で扱われます。

---

このパラメータのデフォルト値は *NONE* で、管理者定義の資源割当量はホストに関連づけられません。

- *calendar* - このキューに使用するカレンダーを指定します。デフォルトは NONE です。カレンダーは、時刻や週、年に基づいてキューの可用性を定義します。Sun Grid Engine のカレンダーについての詳細は、`calendar_conf(5)` を参照してください。

---

注 - ジョブでは、`qsub(1)` の `-l c=<cal_name>` を使用して特定のカレンダーモデルを持つキューを要求することができます。

---

- *initial\_state* - 初めてシステムに追加したとき、あるいはキューが存在するホストで `sge_execd(8)` が起動したときのキューの初期状態を定義します。指定可能な値は以下のとおりです。
  - *Default* - システムに追加されたときに使用可能になり、`sge_execd(8)` が起動したときは前の状態にリセットされます (この動作は、*initial\_state* をサポートしていなかった以前のリリースの Sun Grid Engine の動作に相当します)。
  - *Enabled* - どちらの場合もキューを使用可能にします。これは、手動で明示的な `qmod -e` コマンドの動作に相当します (`qmod(1)` を参照)。
  - *Disabled* - どちらの場合もキューを使用不可にします。これは、手動で明示的な `qmod -d` コマンドの動作に相当します (`qmod(1)` を参照)。
- *fshare* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

キュー、すなわち、ジョブクラスの業務優先配分量を指定します。業務優先ポリシーが有効な場合、このキューで実行されるジョブは、その相対的な重要性に基づいて求められる業務優先チケットを得ることができます。
- *oticket* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

キュー、すなわち、ジョブクラスの一時優先チケット数を指定します。Sun Grid Engine, Enterprise Edition は、このキューで実行されるすべてのジョブに、指定された数の一時優先チケットを分配します。

## 資源利用の制限

Sun Grid Engine には、`s_rt` と `h_rt` という 2 つの資源制限パラメータが実装されています。これらのパラメータは、ジョブが開始されてから経過した「実時間」(経過時間) または時計時間を定義します。キューで実行中のジョブが `h_rt` を超えた場合は、SIGKILL シグナルによってただちに実行が中止されます (`kill(1)` を参照)。`s_rt` を超えた場合は、最初に SIGUSR1 による「警告」を受け (ジョブが捕捉可能)、最終的に上記のキュー構成パラメータの *notify* に定義されている通知時間の経過後に実行が中止されます。

`s_cpu` と `h_cpu` は、ジョブを制限するための資源制限パラメータとして実装されています。これらのパラメータは、ジョブを構成するすべてのプロセスが消費する CPU 時間に制限を課します。キューで実行中のジョブが `h_cpu` を超えた場合は、



SIGKILL シグナルによってただちに実行が中止されます (kill(1) を参照)。s\_cpu を超えた場合は、ジョブが捕捉可能な SIGXCPU シグナルがジョブに送信されます。ジョブが「警告」を受けて、終了させられる前に自身で終了できるようにする場合は、h\_cpu よりも小さい値を s\_cpu に設定します。並列プロセスの場合は、スロットレベルで制限が適用され、このことは、ジョブが使用しているスロット数が制限値に掛けられることを意味します。

s\_vmem と h\_vmem は、ジョブを制限するための資源制限パラメータとして実装されています。これらのパラメータは、ジョブを構成するすべてのプロセスが消費する仮想メモリー量に制限を課します。キューで実行中のジョブが h\_vmem を超えた場合は、SIGKILL シグナルによってただちに実行が中止されます (kill(1) を参照)。s\_vmem を超えた場合は、ジョブが捕捉可能な SIGXCPU シグナルがジョブに送信されます。ジョブが「警告」を受けて、終了させられる前に自身で終了できるようにする場合は、h\_vmem よりも小さい値を s\_vmem に設定します。並列プロセスの場合は、スロットレベルで制限が適用され、このことは、ジョブが使用しているスロット数が制限値に掛けられることを意味します。

キュー構成用テンプレートの残りのパラメータには、setrlimit(2) システムコールによって実装されるソフトおよびハード資源制限をジョブレベルで指定します。詳細は、システムのマニュアルページを参照してください。デフォルトでは、制限フィールドはすべて無限 (setrlimit(2) のマニュアルページに説明があるように RLIM\_INFINITY) に設定されています CPU 時間制限の s\_cpu と h\_cpu の値の型は時間です。その他の制限の値の型はメモリーです。

---

**注** - 必ずしもすべてのシステムが setrlimit(2) をサポートしているわけではありません。

---

---

**注** - s\_vmem と h\_vmem (仮想メモリー) は、RLIMIT\_VMEM をサポートしているシステムでのみ使用できます (オペレーティングシステムの setrlimit(2) を参照)。

---

SGI/Cray の UNICOS オペレーティングシステムは、setrlimit(2) システムコールをサポートしていません。このオペレーティングシステムは、専用の資源制限設定システムコールを採用しています。以下の説明は、UNICOS システムにのみ該当します。

- s\_cpu - プロセス 1 つあたりの CPU 時間制限 (秒単位)
- s\_core - プロセス 1 つあたりの最大コアファイルサイズ (バイト単位)
- s\_data - プロセス 1 つあたりの最大メモリー制限 (バイト単位)
- s\_vmem - s\_data と同じ (両方が設定されている場合は、小さい方が使用される)。
- h\_cpu - ジョブ 1 つあたりの CPU 時間制限 (秒単位)
- h\_data - ジョブ 1 つあたりの最大メモリー制限 (バイト単位)
- h\_vmem - h\_data と同じ (両方が設定されている場合は、小さい方が使用される)。
- h\_fsize - ジョブが作成可能な総ディスクブロック数

## 関連項目

sgc\_intro(1)、csh(1)、qconf(1)、qmon(1)、qrestart(1)、qstat(1)、qsub(1)、sh(1)、nice(2)、setrlimit(2)、access\_list(5)、calendar\_conf(5)、sgc\_conf(5)、complex(5)、host\_conf(5)、sched\_conf(5)、sgc\_execd(8)、sgc\_qmaster(8)、sgc\_shepherd(8)

## 著作権

著作権および許諾条件全文については、sgc\_intro(1) を参照してください。

---

# sgc\_conf(5)

## 名前

sgc\_conf - Sun Grid Engine のデフォルトスケジューラ構成ファイル

## 説明

sgc\_conf は、sgc\_schedd(8) が提供する Sun Grid Engine システムのデフォルトスケジューラ用構成ファイルの形式を定義します。構成を変更するには、グラフィカルユーザーインターフェースの qmon(1) か qconf(1) コマンドの -msconf オプションを使用します。デフォルトの構成は、Sun Grid Engine ディストリビューションパッケージに付属しています。

## 形式

sgc\_conf に次のパラメータが存在する場合、Sun Grid Engine のスケジューラはそれらのパラメータを認識します。

- *algorithm* - 別のスケジューリングアルゴリズムを選択できます。ただし、現在指定できる設定は default だけです。

- *load\_formula* - ホストごとに *sge\_execd*(8) から報告される負荷パラメータの一部またはすべてと、ホストごとに管理されている消費可能資源 (*complex*(5) を参照) の一部またはすべてから 1 つの重み付き負荷値を得る際に使用する簡単な代数式です。この負荷式の構文は、重み付きの負荷値を集計する構文になります。

$load\_val1[*w1][[+|-]load\_val2[*w2][[+|-]...]]$

---

**注** – 負荷式で空白は使用できません。

---

負荷値と消費可能資源 (*load\_val1*, ...) は、複合に定義されている名前を使用して指定します (*complex*(5) を参照)。

---

**注** – 管理者定義の負荷値 (詳細は *sge\_conf*(5) の *load\_sensor* パラメータを参照) やすべてのホストに使用可能な消費可能資源 (*complex*(5) を参照) ばかりでなく、*Sun Grid Engine* のデフォルトの負荷パラメータも使用することができます。

---

重み付け係数 (*w1*, ...) は、正の整数です。ホストごとに式が評価された後、結果がホストに割り当てられます。こうして、重みが付けられた負荷に従ってホストがソートされ、ソートされたホストリストを使用してキューがソートされます。

デフォルトの負荷式は *load\_avg* です。

- *job\_load\_adjustments* - システムで実行中の *Sun Grid Engine* ジョブによって課される負荷は時間とともに変化し、しばしば、オペレーティングシステムが適切な報告するのにそれなりの時間が必要になることがあります (たとえば CPU 負荷のため)。このため、ジョブの開始直後の場合、報告される負荷は、ジョブによってホストにすでに課されている負荷を十分に表していないことがあります。報告される負荷は時間とともに実際の負荷に近づいていきますが、低すぎる間は、そのホストが過剰な実行依頼を受ける可能性があります。管理者は、*Sun Grid Engine* スケジューラがこの問題の補正に使用する *job\_load\_adjustments* を指定することができます。

*job\_load\_adjustments* は、任意の負荷パラメータか消費可能資源の後に等号を挿入し、関連づける負荷補正値を指定した、コンマ区切りのリストの形式で指定します。*sge\_schedd*(8) によってジョブがホストにディスパッチされるたびに、ホストの負荷パラメータおよび消費可能資源値セットが、*job\_load\_adjustments* リストに指定された値分増加します。これらの補正値は、ジョブが開始されてから *load\_adjustment\_decay\_time* が経過して、補正がゼロになるまで直線的に減少します。*job\_load\_adjustments* に特殊なデノミネータの *NONE* が設定されている場合、負荷の補正は行われません。

補正された負荷および消費可能資源値は、ホストの結合および重み付き負荷と上記の *load\_formula* との比較や、キュー構成に定義されている負荷しきい値リストとの負荷および消費可能資源値の比較照合に使用されます (*queue\_conf*(5) を参照)。*load\_formula* が単に CPU 負荷平均パラメータの *load\_avg* で、ジョブが非常に計算集約型の場合は、*job\_load\_adjustments* リストに *load\_avg=100* を設定する

ことを推奨します。これは、ホストにディスパッチされる新しいジョブのすべてが、100%のCPU時間を要求し、マシンの負荷がただちに100増加することを意味します。

- `load_adjustment_decay_time` - 上記の「`job_load_adjustments`」リストの負荷補正は、ジョブが開始されてから (ジョブの開始時点では、対応する負荷または消化可能資源パラメータが補正值分フルに増加する)、「`load_adjstment_decay_time`」の時間が経過して、補正がゼロになるまで直線的に減少します。  
「`load_adjustment_decay_time`」の適切な値は、使用する負荷または消費可能資源パラメータと、使用しているオペレーティングシステムによってかなり異なります。このため、現場で試行錯誤しながら決定することを推奨します。デフォルトの `load_avg` 負荷パラメータの場合は、`load_adjustment_decay_time` を7分にすると、それなりの結果を得られることが証明されています。

- `maxujobs` - 1人のユーザーが一度に Sun Grid Engine クラスタで実行可能なジョブの最大数。デフォルトのゼロに設定すると、ユーザーは任意の数のジョブを実行できます。

- `schedule_interval` - `sge_schedd(8)` が当初 `sge_qmaster` に登録する時間。`schedule_interval` は、`sge_qmaster(8)` がスケジューリングイベントの更新を `sge_schedd(8)` に送信する時間間隔を設定する目的で使用されます。スケジューリングイベントとは、`sge_qmaster(8)` 内で発生するステータスの変更で、スケジューラの決定を促したり、スケジューラの決定に影響を及ぼすことがあります (たとえば、ジョブが完了して、割り当てられていた資源が再び使用可能になるなど)。

Sun Grid Engine のデフォルトのスケジューラでは、スケジューライベントレポートが到着すると、スケジューラが実行されます。それ以外の場合、スケジューラはイベントレポートの待ち状態になります。

`Schedule_interval` は時間値です (`queue_conf(5)` の時間値の構文の定義を参照)。

- `user_sort` - 通常、Sun Grid Engine システムは、先にきたものに先にサービスを提供する方式でユーザージョブをスケジューリングします。ユーザーが非常に短い時間の間に大量のジョブの実行依頼をした場合は、後で実行依頼をするすべてのユーザーが、そのユーザーのジョブの大半が完了するまで待たされることになるため、かなり不公平な状況になる可能性があります。このため、Sun Grid Engine では、この方式をいわゆる均等配分ソート方式に変更できるようにしています。この方式では、あるユーザーのジョブが実行されると、そのユーザーの他のジョブはただちに保留中のジョブリストの最後に回されます。こうすることによって、他のすべてのユーザーの最初のジョブが、キューを見つける同等の機会を得るようになります。

---

注 - 均等配分ソート方式は、同じジョブの優先順位カテゴリ内でのみ適用されます (詳細は、`qsub(1)` と `qalter(1)` コマンドの `mp` オプションを参照)。

---

`user_sort` のデフォルトは `FALSE` です。

- *queue\_sort\_method* - キューリストをソートする際にいくつかの基準を考慮する順番を定義します。有効な設定は *seq\_no* と *load* の 2 つです。ただし、どちらの場合も、Sun Grid Engine ソフトウェアは、第 1 の基準として特定のジョブに関してキューが満たすソフト要求数が最大になるようにします (*qsub(1)* の *-s* オプションを参照)。

*queue\_sort\_method* が *seq\_no* に設定されている場合、Sun Grid Engine は現在のキュー構成に設定されている *seq\_no* パラメータを第 1 の基準として使用して、キューリストをソートします (*queue\_conf(5)* を参照)。上記の *load\_formula* が意味を持つのは、2 つのキューの連続番号が同じ場合だけです。

*queue\_sort\_method* が *load* に設定されている場合、*load\_formula* に一致する負荷が、ジョブのソフト要求を最大にした後の第 1 の条件になります。この設定は、2 つのホストの負荷が同じ場合にのみ連続番号が使用されます。

連続番号によるソートは、たとえば、コストの低い資源順というように、キューを埋める一定の順番を定義する場合に特に役立ちます。

このパラメータのデフォルトは *load* です。

- *sgeee\_schedule\_interval* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

このパラメータには、Sun Grid Engine, Enterprise Edition の動的スケジューラが行うジョブの優先順位の調整間隔を指定します。*queue\_conf(5)* で指定するのと同じ時間型の値を指定します。

- *halftime* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

基本割当ポリシーを適用する際、Sun Grid Engine, Enterprise Edition は、基本割当ツリーに定義されている資源配分をフレックスな時間枠で実現できるよう、時間の経過とともに使用量を減少させます。*halftime* は、累積利用量を元の値の半分に減少させる時間間隔を定義します。*queue\_conf(5)* で指定するのと同じ時間型の値を指定します。

- *usage\_weight\_list* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

Sun Grid Engine, Enterprise Edition は、CPU 時間、メモリー、入出力の資源消費を考慮することによって、ジョブがシステムに課す使用量を決定します。個々の値に重みを掛け、合計することによって、これら 3 つの入力パラメータから 1 つの使用量が求められます。*usage\_weight\_list* には、その重みを定義します。リストの形式は以下のとおりです。

```
cpu=wcpu, mem=wmem, io=wio
```

*wcpu*、*wmem*、*wio* は設定可能な重みです。重みは実数で指定し、3 つの重みの合計が 1 になるようにします。

- *compensation\_factor* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

このパラメータは、基本割当ツリーに定義されている資源配分を下回るか上回る過去の資源利用を Sun Grid Engine, Enterprise Edition が補正する速さを定義します。推奨値は 2 から 10 の範囲で、10 が高速な補正を意味します。

- *weight\_user* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

業務優先ポリシーにおけるユーザー配分の相対的な重要性を定義します。値の型は実数です。

- *weight\_project* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

業務優先ポリシーにおけるプロジェクト配分の相対的な重要性を定義します。値の型は実数です。

- *weight\_jobclass* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

業務優先ポリシーにおけるジョブクラス (すなわち、キュー) 配分の相対的な重要性を定義します。値の型は実数です。

- *weight\_department* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

業務優先ポリシーにおける部署配分の相対的な重要性を定義します。値の型は実数です。

- *weight\_job* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

業務優先ポリシーにおけるジョブ配分の相対的な重要性を定義します。値の型は実数です。

- *weight\_tickets\_functional* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

Sun Grid Engine, Enterprise Edition が分配可能な業務優先チケットの最大数です。業務優先ポリシーの相対的な重要性を決定します。

- *weight\_tickets\_share* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

Sun Grid Engine, Enterprise Edition が分配可能な基本割当チケットの最大数です。基本割当ポリシーの相対的な重要性を決定します。

- *weight\_deadline* - このパラメータは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine では使用できません。

Sun Grid Engine, Enterprise Edition が分配可能な締め切り優先チケットの最大数です。締め切り優先ポリシーの相対的な重要性を決定します。

- *schedd\_job\_info* - デフォルトのスケジューラは、前回のスケジューラ実行中にジョブをスケジューリングできなかった理由を記録することができます。このパラメータは、この監視を有効または無効にします。値 true は監視を有効にし、false は無効にします。

特定のジョブについてだけ監視を有効にすることもできます。この監視を行うには、`job_list` に続けて、ジョブ ID からなるコンマ区切りのリストを指定します。収集された情報は、`qstat -j` コマンドを使用して見ることができます。

## ファイル

- `sgc_schedd` 構成 - `<sgc_root>/<cell>/common/sched_configuration`

## 関連項目

`sgc_intro(1)`、`qalter(1)`、`qconf(1)`、`qstat(1)`、`qsub(1)`、`complex(5)`、`queue_conf(5)`、`sgc_execd(8)`、`sgc_qmaster(8)`、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、`sgc_intro(1)` を参照してください。

---

# share\_tree(5)

## 名前

`share_tree` - Sun Grid Engine, Enterprise Edition の基本割当ツリーファイルの形式

## 説明

基本割当ツリーオブジェクトは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine には、基本割当ツリーオブジェクトはありません。

基本割当ツリーは、ユーザー / プロジェクトおよびそれらの任意のグループからなる階層の長期の資源利用資格を定義します。

現在の基本割当ツリーは、`qconf(1)` の `-sstree` オプションを使用して表示することができます。この出力は、`share_tree` の説明に従っています。`qconf(1)` の `-astree` および `-dstree` オプションを使用してそれぞれ、基本割当ツリーの新規作成、変更を行うことができます。個々の基本割当ツリーノードは、`qconf(1)` の `-astnode`、`-dstnode`、`-mstnode`、`-sstnode` オプションを使用してそれぞれ作成、変更、削除、表示することができます。

## 形式

基本割当ツリーファイルの形式は以下のとおりです。

- 新規ノードは、ノードの属性 `id` と等号、識別番号 (数値) から始まります。別の `id` キーワードが検出されるまで、同じノードの属性が続きます。
- 属性 `childnodes` には、このノードの子ノードの、コンマ区切りのリストが含まれます。
- パラメータ `name` は、ノードの任意の名前を表します。ただし、ノードが基本割当ツリーのリーフノードの場合は、対応するユーザー (`user(5)` を参照) かプロジェクト (`project(5)` を参照) を表します。ツリーのルートノードの名前は、規則で「`Root`」になります。
- パラメータ `shares` は、同じ親ノードを持つノード間のノードの配分を定義します。
- 基本割当ツリーでは、「`default`」という名前のユーザーリーフノードをプロジェクトノードの子孫として定義することができます。`default` ノードは、プロジェクトに属するものの、そのプロジェクトにユーザーノードが定義されていないユーザーに対する配分を定義します。`default` ユーザーノードは、単一のノードで、すべてのユーザーにプロジェクト資源を均等に配分する便利な手段です。`default` ノードは単独で指定することも、プロジェクトの下の同じレベルの他の `user(5)` ノードとともに指定することもできます。ユーザーノードとして明示的に指定されているか、あるいは `default` ユーザーノードに対応づけられているかに関係なく、配分を受けるには、すべてのユーザーに対して対応する `user(5)` オブジェクトが定義されている必要があります。`default` という名前の `user(5)` オブジェクトを作成することはできません。

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。



---

# user (5)

## 名前

user - Sun Grid Engine, Enterprise Edition のユーザーエン트리ファイルの形式

## 説明

ユーザーオブジェクトは、Sun Grid Engine, Enterprise Edition システムでのみ使用できます。Sun Grid Engine システムには、ユーザーオブジェクトはありません。

Sun Grid Engine, Enterprise Edition では、ユーザーエントリを使用してユーザー別のチケットおよび資源利用情報を記録します。Sun Grid Engine, Enterprise Edition をユーザー基本割当ポリシーに基づいて運用する場合は、システムに参加するすべてのユーザーに対するユーザーエントリを作成する必要があります。

構成済みのユーザーエントリのリストは、qconf(1) の `-suserl` オプションを使用して表示することができます。このリストの個々のユーザーエントリの定義内容は、`-suser` スイッチを使用して表示することができます。この出力は、*user* 形式の説明に従っています。qconf(1) の `-auser`、`-muser`、`-duser` オプションを使用してそれぞれ、ユーザーエントリの新規作成、変更、削除を行うことができます。

## 形式

ユーザーエントリは、4つのパラメータで構成されます。

- *name* - ユーザー名
- *oticket* - ユーザーに割り当てられている一時優先チケット数
- *fshare* - ユーザーの現在の業務優先配分量
- *default\_project* - ユーザーのデフォルトプロジェクト

## 関連項目

sge\_intro(1)、qconf(1)

## 著作権

著作権および許諾条件全文については、`sgc_intro(1)` を参照してください。

---

## usermapping(5)

### 名前

`usermapping` - ユーザーマッピングエントリファイルの形式

### 説明

`usermapping` エントリでは、クラスタユーザーの別名を定義します。スケジューリングシステムが認識しているユーザーの名前はクラスタユーザー名といいます。クラスタ名が実行ホスト上のユーザーアカウント名と一致しない場合は、`usermapping` 機能でこの問題を解決することができます。

`usermapping` エントリファイルでは、1 行に、ユーザー名 1 つと、そのユーザーがアカウントを持っているホストを指定します。

既存の `usermapping` エントリは、`qconf(1)` の `-sumapl` オプションを使用して一覧表示することができます。このリストの個々の `usermapping` エントリの内容は、`-sumap` スイッチを使用して表示することができます。この出力は、`usermapping` 形式の説明に従っています。`usermapping` エントリは、`qconf(1)` の `-aumap` や `-mumap`、`-dumap` オプションを使用して作成、変更、削除することができます。

### 形式

`usermapping` エントリには、少なくとも 3 つのパラメータが含まれます。

- `cluster_user` キーワード - クラスタユーザー名を定義します。テキスト行の `cluster_user` キーワード以降は、クラスタユーザー値とみなされます。
- `name` - 実行ホスト上のユーザー名
- `hostname` - `name` で指定したユーザーがアカウントを持っているホスト名。ホスト名の先頭文字が `@` 記号の場合、この名前を使用して `hostgroup(5)` が参照されます。

## 使用例

以下に、クラスタユーザーマッピングの一般的な `usermapping` エントリを示します。

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

このエントリは、クラスタシステムに定義されているユーザー `office` をホスト `wonderland` と `realworld` 上のユーザー `alice` に対応づけます。@ 記号を使用して、ホストグループが参照されています。このため、ユーザー `office` は、ホストグループ `office2` に指定されているすべてのホスト上の `alice` にも対応づけられます。このあたりの詳細は、`hostgroup(5)` を参照してください。

## 関連項目

`qconf(1)`、`hostgroup(5)`

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

## `sge_commd(8)`

### 名前

`sge_commd` - Sun Grid Engine の通信エージェント

### 形式

```
sge_commd [ -S ] [ -a aliasfile ] [ -dhr ] [ -help ]
[ -ll loglevel ] [ -ml fname ] [ -nd ] [ -p port ]
[ -s service ]
```

## 説明

Sun Grid Engine クラスタ内のあらゆるネットワーク通信は、通信デーモンの `sge_commd` 経由で行われます。 `qsub(1)` や `qstat(1)` のようなクライアントプログラムばかりでなく、 `sge_qmaster(8)` や `sge_execd(8)` などの Sun Grid Engine のデーモンも `sge_commd` が提供するサービスを使用して、他の Sun Grid Engine コンポーネントとメッセージのやりとりをします。

`sge_commd` は、任意の数の並行同期または非同期通信を処理します。通常、 `sge_qmaster(8)`、 `sge_execd(8)`、 `sge_schedd(8)` が起動されたホストごとに 1 つ `sge_commd` が自動的に起動されます。ただし、1 つの `sge_commd` を使用して複数のホストを接続したり、Sun Grid Engine の実行依頼または管理ホストから起動された Sun Grid Engine クライアントプログラム用の通信エージェントとして、そのホストで `sge_commd` を使用したりすることもできます。

# オプション

表 29 は、`sgc_commd` のオプションをまとめています。

表 29 `sgc_commd` のオプション

オプション	説明
<code>-S</code>	<code>sgc_commd</code> と他の Sun Grid Engine コンポーネント、または <code>sgc_commd</code> 間の通信に強制的にセキュリティ保護されたポートを使用します。このためには、 <code>root</code> アクセス権ですべての Sun Grid Engine デーモンを起動し、クライアントプログラムを <code>set-uid root</code> 設定する必要があります。そうすることによって、 <code>root</code> 以外のアカウントについて、許可されていない通信が確実に禁止されます。
<code>-a aliasfile</code>	クラスタ内の全ホストの一意の Sun Grid Engine ホスト名を解決するために <code>sgc_commd</code> が使用する Sun Grid Engine ホストの別名を含むファイル。 <code>&lt;sgc_root&gt;/&lt;cell&gt;/common/host_aliases</code> にあるデフォルトの <code>host_aliases</code> ファイルに優先して使用されます。 <code>sgc_commd</code> のホスト名解決サービスは、他のすべての Sun Grid Engine コンポーネントによっても使用されます。ファイル形式とその要素の意味については、 <code>sgc_h_aliases(5)</code> に説明があります。
<code>-dhr</code>	一部 OS アーキテクチャでは、 <code>gethostent(3)</code> 、 <code>gethostbyname(3)</code> 、 <code>gethostbyaddr(3)</code> などのホスト名解決 C ライブラリ関数によってある種のキャッシュが行われます。このため、DNS (Domain Name Service) や NIS などのサービスによって提供されるネットワーク全体のホスト名データベースは、数分の遅延で更新されます。この影響を受けるのは、 <code>sgc_commd</code> などの繰り返しホスト名を解決するアプリケーションだけです。プログラムの起動時に、最新の情報がアクセスするため、 <code>telnet(1)</code> や <code>nslookup(1)</code> などのコマンドが影響を受けることはありません。ただし、 <code>sgc_commd</code> の場合、どんなときでもホスト名が解決されるようにするのは意味のあることではなく (いずれにしても、返される情報が古い可能性があるため)、ネットワークが過負荷であったり、NIS あるいは DNS サーバーが非常にビジーであったりした場合、解決はかなりコストのかかる操作になる可能性があります。このため、 <code>sgc_commd</code> は、時々にはホスト名を解決しません。それでも、たとえばネットワークの負荷のためにホスト名の解決で問題が発生する場合は、 <code>-dhr</code> スイッチを使用して無効にすることができます。ホスト名解決を無効にした場合、管理者は、ホスト名データベースが大幅に変更されたらすぐに、 <code>sgc_commd</code> を再起動することを忘れないでください。
<code>-help</code>	全オプションの一覧を表示します。
<code>-ll loglevel</code>	エラー監視のログレベルを設定します。エラートレース情報は、 <code>/tmp/commd/err.&lt;pid&gt;</code> ファイルに書き込まれます。この <code>/tmp/commd</code> は必ず存在する必要があるため、存在しない場合、トレース出力は廃棄されます。現在のところ、有効なログレベルは 255 だけです。

表 29 sge\_commd のオプション (続き)

オプション	説明
-ml <i>fname</i>	ファイルへのメッセージの記録を有効にします。
-nd	デーモン化しません。-nd を指定して起動された場合、sge_commd は制御端末との接続を維持し、自身が起動された端末にトレース情報を直接出力します。このトレース情報は、上記の -ll オプションを使用したときに得られる情報と同じです。
-p <i>port_number</i>	他の commd との通信に指定された TCP ポートを使用します。
-s <i>service_name</i>	他の commd との通信に指定されたサービスと TCP ポートを使用します。

## 環境変数

表 30、sge\_commd に関する環境変数をまとめています。

表 30 sge\_commd 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
COMMD_PORT	設定されている場合は、sge_commd が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	(sge_commd の動作には影響しませんが、sge_commd にアクセスする他の Sun Grid Engine コンポーネントの動作に影響します。) 設定されている場合は、任意の Sun Grid Engine クライアントプログラムまたはデーモンとの通信に使用する特定の sge_commd (8) が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 制限事項

通常、sge\_commd は、sge\_qmaster(8) および sge\_execd(8) を起動すると起動され、root 権限で動作します。通常のユーザーが起動する場合、セキュリティモードで root 権限が必要になるため、-s スイッチは使用できません (上記を参照)。

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# sge\_execd(8)

## 名前

`sge_execd` - Sun Grid Engine のジョブ実行エージェント

## 形式

```
sge_execd [ -help ] [ -lj log_file ] [ -nostart-commd ]
```

## 説明

`sge_execd` は、自身が動作するマシンにローカルの Sun Grid Engine キューを制御し、それらのキューで実行されるよう `sge_qmaster(8)` から送信されたジョブを実行、制御します。

`sge_execd` が起動されたマシンに `sge_commd(8)` が存在しない場合は、`sge_commd(8)` も自動的に起動されません。

## オプション

表 31 は `sgc_execd` のオプションをまとめています。

表 31 `sgc_execd` のオプション

オプション	説明
<code>-help</code>	全オプションの一覧を表示します。
<code>-lj log_file</code>	ジョブのログ記録を有効にします。ジョブを受け取ってから <code>sgc_qmaster(8)</code> にそのジョブを返すまでの <code>sgc_execd</code> のあらゆる動作が <code>log_file</code> に記録されます。この機能は、 <code>sgc_qmaster(8)</code> デーモンにもあります。
<code>-nostart-commd</code>	<code>sgc_execd</code> とともに <code>sgc_commd(8)</code> を自動的に起動しないで、 <code>COMMD_HOST</code> 環境変数を評価して、多入出力する <code>sgc_commd(8)</code> を検出します。

## 負荷センサー

グローバルまたは実行ホストに固有のクラスタ構成で、`sgc_execd` 用の負荷センサーが設定されている場合は、`sgc_execd` によってその実行可能ファイルが定期的起動され、実行ホストまたはクラスタ全体に関する 1 つ以上の負荷情報 (実行ホストに現在ログイン中のユーザー、あるいはネットワーク全体のスクラッチファイルシステムの空きディスク容量など) が提供されます。この負荷センサーはスクリプトでも、バイナリ形式の実行可能ファイルでもかまいません。どちらの場合も、その標準入出力ストリームの処理および制御の流れは次の規則に従っている必要があります。

負荷センサーは、特定の地点で STDIN からの入力を待つ無限ループとして作成する必要があります。STDIN から文字列 `quit` を読み取ったら、負荷センサーを終了します。STDIN から行の終わりを読み取ったら、ただちに負荷データの読み出しサイクルを開始します。このサイクルでは、負荷センサーは目的の負荷値の計算に必要なあらゆる処理を行い、サイクルの終わりで結果を `stdout` に書き込みます。負荷センサーの形式は次のとおりです。

- 負荷値レポートは、`start` という文字列だけを含まる行で開始します。
- 負荷値はそれぞれ改行で区切ります。
- 1 つの負荷値レポートは、空白なしのコロン (`:`) で区切られた 3 つの部分で構成します。
- 負荷値情報の最初の部分は、その負荷の情報元のホスト名か特殊名 `global` です。



- 2つ目の部分は、ホストまたはグローバル構成リストに定義されている負荷値のシンボリック名です (詳細は、complex (5) を参照)。ホストまたはグローバル複合リストにエントリのない負荷値が報告された場合、その負荷値は使用されません。
- 3つ目の部分は負荷測定値です。
- 負荷値レポートは、end という文字列の行で終了します。

## 環境変数

表 32 は、sge\_execd に関する環境変数をまとめています。

表 32 sge\_execd 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、sge_execd は以下を使用します (優先順)。 <ul style="list-style-type: none"> <li>• 環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li> <li>• デフォルトのセル名 (例: default)</li> </ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd (8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。
COMMD_HOST	設定されている場合は、Sun Grid Engine が sge_execd クライアントとの通信に使用する特定の sge_commd (8) が存在するホストを示します。sge_execd コマンド行に -nostart-commd オプションが指定された場合にのみ評価されます。デフォルトでは、ローカルホストが使用されます。

## 制限事項

通常、sge\_execd は Sun Grid Engine プール内の各マシンの root から起動されます。通常、ユーザーが起動する場合は、そのユーザーが読み取り・書き込みアクセス権を持つスプールディレクトリを使用する必要があります。この場合、そのユーザーによって実行依頼されたジョブだけが正しく処理されます。

## ファイル

- Sun Grid Engine のグローバル構成 - `<sg_e_root>/<cell>/common/configuration`
- Sun Grid Engine のホスト固有の構成 -  
`<sg_e_root>/<cell>/common/local_conf/<host>`
- 実行ホストのデフォルトのスパールディレクトリ -  
`<sg_e_root>/<cell>/spool/<host>`
- Sun Grid Engine のマスターホストファイル -  
`<sg_e_root>/<cell>/common/act_qmaster`

## 関連項目

`sg_e_intro(1)`、`sg_e_conf(5)`、`complex(5)`、`sg_e_commd(8)`、`sg_e_qmaster(8)`

## 著作権

著作権および許諾条件全文については、`sg_e_intro(1)` を参照してください。

---

# sg\_e\_qmaster(8)

## 名前

`sg_e_qmaster` - Sun Grid Engine のマスターコントロールデーモン

## 形式

```
sg_e_qmaster [ -help ] [ -lj log_file ] [ -nohist ]  
[ -nostart-commd ] [ -s ]
```

```
sg_e_qmaster -show-license
```

## 説明

`sgc_qmaster` は、クラスタ内の Sun Grid Engine 全体の動作を制御します。ジョブのスケジューリングでは、`sgc_qmaster` は `sgc_schedd(8)` と協力します。`sgc_qmaster` が起動されたマシンに `sgc_commd(8)` が存在しない場合は、通常、`sgc_commd(8)` も自動的に起動されます。

## オプション

表 33 は、`sgc_qmaster` のオプションをまとめています。

表 33 `sgc_qmaster` のオプション

オプション	説明
<code>-help</code>	全オプションの一覧を表示します。
<code>-lj log_file</code>	ジョブのログ記録を有効にします。実行依頼からジョブの終了までの <code>sgc_qmaster</code> のあらゆる動作が <code>log_file</code> に記録されます。この機能は、 <code>sgc_execd(8)</code> デーモンにもあります。
<code>-nohist</code>	通常動作では、 <code>sgc_qmaster</code> はキュー、複合、ホスト構成の変更履歴を履歴データベースにダンプします。このデータベースは主として <code>qacct(1)</code> コマンドで使用し、 <code>qacct(1)</code> コマンド行で <code>qsub(1)</code> に似た <code>-l resource</code> 要求を行うことを可能にします。このスイッチは、履歴データベースへの書き込みを抑制します。
<code>-nostart-commnd</code>	<code>sgc_qmaster</code> とともに自動的に <code>sgc_commd(8)</code> を起動しません。
<code>-s</code>	<code>sgc_qmaster</code> のサイレントモードを有効にします。通常、 <code>sgc_qmaster</code> は起動時にライセンス情報を表示して、Return が押されるのを待ちます。 <code>-s</code> スイッチを使用すると、 <code>sgc_qmaster</code> がそうした情報を表示しないで起動します。

## 環境変数

表 34 は、sge\_qmaster に関する環境変数をまとめています。

表 34 sge\_qmaster 関係の環境変数

変数名	説明
SGE_ROOT	Sun Grid Engine の標準構成ファイルの場所を示します。
SGE_CELL	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、sge_qmaster は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 SGE_CELL に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: default)</li></ul>
SGE_DEBUG_LEVEL	設定されている場合は、デバッグ情報を stderr に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
COMMD_PORT	設定されている場合は、sge_commd(8) が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに services マップエントリを使用して、このポートを定義します。

## 制限事項

通常、sge\_qmaster は、クラスタのマスターまたはシャドワマスターマシンの root から起動されます (シャドワマスターホストの構成についての詳細は、『Sun Grid Engine Installation and Administration Guide』を参照)。通常、ユーザーが起動する場合は、そのユーザーが読み取り・書き込みアクセス権を持つマスタープールディレクトリを使用する必要があります。この場合、そのユーザーによって実行依頼されたジョブだけが正しく処理されます。

## ファイル

- Sun Grid Engine のグローバル構成 - `<sge_root>/<cell>/common/configuration`
- Sun Grid Engine のホスト固有の構成 - `<sge_root>/<cell>/common/local_conf/<host>`
- 履歴データベース - `<sge_root>/<cell>/common/history`
- sge\_qmaster 引数ファイル - `<sge_root>/<cell>/common/qmaster_args`
- デフォルトのマスタープールディレクトリ - `<sge_root>/<cell>/spool`

## 関連項目

`sgc_intro(1)`、`sgc_conf(5)`、`sgc_commd(8)`、`sgc_execd(8)`、`sgc_schedd(8)`、`sgc_shadowd(8)`、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』、『Sun Grid Engine, Enterprise Edition 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、`sgc_intro(1)` を参照してください。

---

# sgc\_schedd(8)

## 名前

`sgc_schedd` - Sun Grid Engine のジョブスケジューリングエージェント

## 形式

```
sgc_schedd [ -help ]
```

## 説明

`sgc_schedd` は Sun Grid Engine クラスタ内でスケジューリングの決定をします。この決定に必要な情報は、イベントインタフェースを使用し、`sgc_qmaster(8)` から読み出されます。スケジューリングアルゴリズムの適用後、`sgc_schedd` は Sun Grid Engine GDI を介してその決定を `sgc_qmaster` に返します。`sgc_schedd` の実行を開始するには、`sgc_qmaster(8)` が定期的な間隔でクラスタステータスの変更をサンプリングし、`sgc_schedd` に通知する必要があります。

`sgc_schedd` が起動されたマシンに `sgc_commd(8)` が存在しない場合は、`sgc_commd(8)` も自動的に起動されます。

qconf(1) コマンドの `-tsm` オプションを使用するとよって、`sge_schedd` が強制的に次のスケジューリングのトレースメッセージをファイル `<sge_root>/<cell>/common/schedd_runlog` に出力します。メッセージには、そのスケジューリングでジョブおよびキューが選択されなかった理由が示されます。

## オプション

表 35 は `sge_schedd` のオプションをまとめています。

表 35 `sge_schedd` のオプション

オプション	説明
<code>-help</code>	スケジューラのバージョン番号を表示します。

## 環境変数

表 36 は、`sge_schedd` に関する環境変数をまとめています。

表 36 `sge_schedd` 関係の環境変数

変数名	説明
<code>SGE_ROOT</code>	<code>Sun Grid Engine</code> の標準構成ファイルの場所を示します。
<code>SGE_CELL</code>	設定されている場合は、デフォルトの <code>Sun Grid Engine</code> セルを示します。 <code>Sun Grid Engine</code> セルにアドレッシングする際、 <code>sge_schedd</code> は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 <code>SGE_CELL</code> に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: <code>default</code>)</li></ul>
<code>SGE_DEBUG_LEVEL</code>	設定されている場合は、デバッグ情報を <code>stderr</code> に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
<code>COMMD_PORT</code>	設定されている場合は、 <code>sge_commd(8)</code> が通信要求を待機することになっている <code>TCP</code> ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。

## ファイル

- `sge_schedd` のスプールディレクトリ -  
`<sge_root>/<cell>/spool/qmaster/schedd`
- `sge_schedd` のトレース情報 - `<sge_root>/<cell>/common/sched_runlog`
- `sge_schedd` 構成 - `<sge_root>/<cell>/common/sched_configuration`
- スケジューラ構成ファイルについての詳細は、`sge_conf(5)` を参照してください。

## 関連項目

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

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# sge\_shadowd(8)

## 名前

`sge_shadowd` - Sun Grid Engine のシャドウマスターデーモン

## 形式

`sge_shadowd`

## 説明

`sge_shadowd` は、Sun Grid Engine クラスタ内のいわゆるシャドウマスターホスト上で実行可能な「軽量」プロセスです。`sge_shadowd` は Sun Grid Engine の現在のマスターデーモンの `sge_qmaster(8)` のエラーを検出し、自身が動作しているホス

トで新しい `sge_qmaster(8)` を起動します。クラスタ内で複数のシャドウデーモンがアクティブな場合、それらデーモンはプロトコルを実行して、そのうちの 1 つだけが新しいマスターデーモンを起動するようにします。

シャドウマスターホストとして使用するのに適したホストは、`<sge_root>/<cell>/common` ディレクトリばかりでなく、マスターデーモンズプールディレクトリ (デフォルトは `<sge_root>/<cell>/spool/qmaster`) への共有 root 読み取り・書き込みアクセス権を持っている必要があります。シャドウマスターホストは、`<sge_root>/<cell>/common/shadow_masters` ファイルに登録されている必要があります。

## 制限事項

`sge_shadowd` は root からのみ起動できます。

## 環境変数

表 37 は、`sge_shadowd` に関する環境変数をまとめています。

表 37 `sge_shadowd` 関係の環境変数

変数名	説明
<code>SGE_ROOT</code>	Sun Grid Engine の標準構成ファイルの場所を示します。
<code>SGE_CELL</code>	設定されている場合は、デフォルトの Sun Grid Engine セルを示します。Sun Grid Engine セルにアドレッシングする際、 <code>sge_shadowd</code> は以下を使用します (優先順)。 <ul style="list-style-type: none"><li>環境変数 <code>SGE_CELL</code> に指定されているセル名 (変数が設定されている場合)</li><li>デフォルトのセル名 (例: <code>default</code>)</li></ul>
<code>SGE_DEBUG_LEVEL</code>	設定されている場合は、デバッグ情報を <code>stderr</code> に書き込みます。また、生成するデバッグ情報の詳細度を指定することができます。
<code>COMMD_PORT</code>	設定されている場合は、 <code>sge_commd(8)</code> が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。
<code>COMMD_HOST</code>	設定されている場合は、Sun Grid Engine が <code>sge_shadowd</code> クライアントとの通信に使用する特定の <code>sge_commd(8)</code> が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。



## ファイル

- デフォルトの構成ディレクトリ - `<sge_root>/<cell>/common`
- シャドウマスターホスト名ファイル - `<sge_root>/<cell>/common/shadow_masters`
- マスターデーモンのデフォルトのプールディレクトリ - `<sge_root>/<cell>/spool/qmaster`

## 関連項目

`sge_intro(1)`、`sge_conf(5)`、`sge_commd(8)`、`sge_qmaster(8)`、『Sun Grid Engine Installation and Administration Guide』、『Sun Grid Engine 5.3 管理およびユーザーマニュアル』

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。

---

# sge\_shepherd(8)

## 名前

`sge_shepherd` - Sun Grid Engine の単一ジョブ制御エージェント

## 形式

`sge_shepherd`

## 説明

`sgc_shepherd` は、単一の Sun Grid Engine ジョブに親プロセス機能を提供します。ジョブの完了後、UNIX システムで資源利用情報呼び出すには、こうした親機能が必要です (`getrusage(2)` を参照)。また、`sgc_shepherd` は、一時停止、使用可能、終了、Sun Grid Engine チェックポイントシグナル (`sgc_ckpt(1)` を参照) などのシグナルをジョブに転送します。

`sgc_shepherd` は起動するジョブに関する情報を `sgc_execd(8)` から受け取り、実際には、ジョブの実行中に最高 5 つの子プロセスを起動します。1 つ目が *prolog* スクリプト (クラスタ構成の *prolog* パラメータでこの機能が有効になっている場合、`sgc_conf(5)` を参照)、次に並列環境起動スクリプト (ジョブが並列ジョブの場合、`sgc_pe(5)` を参照)、その後、ジョブそのものと、並列環境の停止プロシージャ (並列ジョブの場合)、そして最後に *epilog* スクリプト (クラスタ構成の *epilog* パラメータで要求されている場合) です。*prolog* および *epilog* スクリプトと並列環境の起動・停止プロシージャは、Sun Grid Engine の管理者が提供し、それらの目的は、実際のユーザージョブの実行前後に行うサイト固有の処理を行うことにあります。

ジョブが完了し、*epilog* スクリプトが処理されると、`sgc_shepherd` はジョブに関する資源利用統計を読み取って、`sgc_execd(8)` のスプールディレクトリのジョブ固有のサブディレクトリに書き込み、終了します (`sgc_execd(8)` は、この書き込まれた情報を使用して、資源利用報告を行う)。

## 制限事項

`sgc_shepherd` が手動で起動しないで、`sgc_execd(8)` に起動させてください。

## ファイル

ジョブ固有のディレクトリ - `<execd_spool>/job_dir/<job_id>`

## 関連項目

`sgc_intro(1)`、`sgc_conf(5)`、`sgc_execd(8)`

## 著作権

著作権および許諾条件全文については、`sgc_intro(1)` を参照してください。

---

# sgecommdcntl(8)

## 名前

sgecommdcntl - Sun Grid Engine の通信エージェント制御コマンド

## 形式

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

## 説明

sgecommdcntl を使用して、sge\_commd(8) の動作を制御したり、実行中の sge\_commd(8) から情報を受け取ったりすることができます。

# オプション

表 38 は、`sgecommdcnt1` のオプションをまとめています。

表 38 `sgecommdcnt1` のオプション

オプション	説明
<code>-d</code>	実行中の <code>sge_commd(8)</code> プロセスの内部構成を <code>/tmp/commd/commd.dump</code> にダンプします。ディレクトリ <code>/tmp/commd</code> が存在していて、 <code>sge_commd(8)</code> がそのディレクトリに対する書き込みアクセス権を持っている必要があります。この条件が満たされない場合、要求は無視されます。このオプションの主な目的はデバッグにあります。ダンプ対象の <code>sge_commd(8)</code> コマンドの機能が影響を受けることはありません。
<code>-k</code>	対象の <code>sge_commd(8)</code> を終了します。終了要求時点で保留中の通信はただちに廃棄されます。ただし、 <code>sge_commd(8)</code> の停止によって、中止プロセスに接続されていたプロセスが矛盾した状態に置かれたままになることはありません。
<code>-t level</code>	<code>sgecommdcnt1</code> は <code>sge_commd(8)</code> との接続を確立し、 <code>level</code> で指定されたトレースレベルに応じた連続するトレース出力を表示します。この出力は、 <code>-l1</code> オプションを指定して <code>sge_commd(8)</code> が起動された場合に表示されるトレース出力のサブセットです。現在サポートされているトレースレベルは 255 だけです。
<code>-gid commprocname</code>	<code>commprocname</code> の通信プロセス識別番号を読み取ります。 <code>sge_commd(8)</code> に登録されていて他の Sun Grid Engine プロセスと通信することが可能な Sun Grid Engine コンポーネントは、名前と識別番号からなる一意の識別情報を使い、 <code>sge_commd(8)</code> によって登録されます。識別名は、Sun Grid Engine コンポーネントの名前と同じです (たとえば <code>sge_qmaster(8)</code> の場合は <code>sge_qmaster</code> )。識別番号は、この <code>-gid</code> オプションを使用して読み出すことができます。
<code>-help</code>	全オプションの一覧を表示します。

表 38 sgecommdcntl のオプション (続き)

オプション	説明
-p <i>commdport</i>	sge_commd (8) にアドレッシングする際に使用するポート番号
-S	sge_commd と他の Sun Grid Engine コンポーネント、または sge_commd 間の通信に強制的にセキュリティ保護されたポートを使用します。このためには、root アクセス権ですべての Sun Grid Engine デーモンを起動し、クライアントプログラムを set-uid root 設定する必要があります。そうすることによって、root 以外のアカウントについて、許可されていない通信が確実に禁止されます。
-unreg <i>commprocname id</i>	sge_commd (8) に ID <i>id</i> で登録されている Sun Grid Engine コンポーネントの <i>commprocname</i> を登録解除します ( <i>commprocname</i> と <i>id</i> については、上記の -gid を参照)。sge_commd (8) からの Sun Grid Engine コンポーネントの登録解除が必要になる可能性があるのは、sge_qmaster (8) や sge_execd (8)、sge_schedd (8) などの Sun Grid Engine デーモンの実行が制御されていない方法で打ち切られ (たとえば kill (1) を使用した SIGKILL シグナルの送信)、sge_commd (8) が、「error enrolling to commd: COMMPROC ALREADY REGISTERED」というメッセージを表示して、そのコンポーネントの再起動を拒否した場合です。sge_commd (8) の登録機能の目的は、冗長な Sun Grid Engine デーモンがホストで実行されるのを回避することにあります。Sun Grid Engine コンポーネントの実行が打ち切られたにもかかわらず、sge_commd (8) から登録解除できない場合は、数分のタイムアウト時間が経過するか、通信プロセスを手動で登録解除しない限り、その登録が有効な状態に置かれます。

## 環境変数

表 39 は、`sgecommdcnt1` に関する環境変数をまとめています。

表 39 `sgecommdcnt1` 関係の環境変数

Name of Variable	説明
<code>COMMD_PORT</code>	設定されている場合は、 <code>sgecommdcnt1</code> が通信要求を待機することになっている TCP ポートを示します。大部分のインストールサイトでは、代わりに <code>services</code> マップエントリを使用して、このポートを定義します。
<code>COMMD_HOST</code>	設定されている場合は、Sun Grid Engine が <code>sgecommdcnt1</code> との通信に使用する特定の <code>sge_commd(8)</code> が存在するホストを示します。デフォルトでは、ローカルホストが使用されます。

## 関連項目

`sge_intro(1)`、`sge_commd(8)`、`sge_execd(8)`、`sge_qmaster(8)`、`sge_schedd(8)`

## 著作権

著作権および許諾条件全文については、`sge_intro(1)` を参照してください。