

1. 概要

Tencube/JM ジョブマネージャは、DualCore、QuadCore、Xeon 等の SMP の WindowsPC を対象にしたジョブ管理システムです。Tencube/WM から起動されて、Gaussian(G03W)や GAMESS(WinGAMESS,PC-GAMESS)、MOPAC を効率的に実行することができます。

2. 使用方法

(1) 起動

WM からジョブを投入 (G03W や GAMESS を実行) しようとする、下図のような JM のウィンドウが立ち上がり、1 番目の実行待ちキューに入り、過去のジョブは下のキューにシフトします。実行状況は、まず wait (実行待ち) となり、Priority と実行 core 数等に従ってジョブが順序よく run 状態に移って実行されます。wait の時の開始時刻は実行キューに入った時刻ですが、run 状態に変わった時に開始時刻が変わります。MOPAC では、設定画面で JM を使う方式と使わない方式の選択ができます。JM を使わない方式では、MOPAC 計算後に自動的に構造が読み込まれますが、JM を使った場合は自動読み込みされませんので、Import 操作が必要になります。

JM は WM からジョブを投入した時に起動されて、WM を停止しても JM は停止しません。詳細は、(7) JM の起動と停止を参照ください。

(2) MaxCores の設定

MaxCores は実行するジョブが使用できる総コア数で、これを超えないようにジョブが実行されます。MaxCores には、実行 PC のコア数以下を指定するとよく、例えば 4 コアの PC では、3 か 4 を指定します。MaxCores =3 にすると cpu 使用率に余力があるので、同じ PC で他の仕事をしていてもストレスなく使えます。MaxCores =4 では PC の性能を最大限に利用できることになります。

The screenshot shows the Tencube/JM Job Manager V0.92 interface. At the top, there are settings for MaxCores (set to 2), RunJobs (1), RunCores (1), and AutoShutdown (unchecked). The Jobname is 20090209_130901. The Status is 'run', Priority is 5, and Start Time is 2009/02/09 13:11. The Job Name is E:\home#ccs#winmos#tencjm2.gjf.bat. Below this, a list of jobs is shown with columns for Status, Priority, Start Time, and Job Name. A job with ID 4 is highlighted in blue and is in 'run' status. Callout boxes point to the 'Priority' column, the 'run' status, the 'Job Name' column, the 'Start Time' column, and the 'End Time' column.

Status	Priority	Job ID	Job Name	Start Time	End Time
wait	5	1	E:\home#ccs#winmos#tencjm3.inp.bat	2009/02/09 13:11	
wait	5	2	E:\home#ccs#winmos#tencjm2.inp.bat	2009/02/09 13:10	
end	5	1	E:\home#ccs#winmos#tencjm1.inp.bat	2009/02/09 13:11	
run	5	4	E:\home#ccs#winmos#tencjm2.gjf.bat	2009/02/09 13:11	
end	5	2	E:\home#ccs#winmos#tencjm3.inp.bat	2009/02/09 13:07	2009/02/09 13:07
end	5	2	E:\home#ccs#winmos#tencjm1.inp.bat	2009/02/09 13:07	2009/02/09 13:07
end	5	1	E:\home#ccs#winmos#tencjm2.inp.bat	2009/02/09 13:07	2009/02/09 13:07
end	5	1	E:\home#ccs#winmos#tencjm1.inp.bat	2009/02/09 13:07	2009/02/09 13:07
end	5	1	E:\home#ccs#winmos#tencjm2.gjf.bat	2009/02/09 13:07	2009/02/09 13:07
end	5	1	E:\home#ccs#winmos#ethylene.gjf.bat	2009/02/08 10:22	2009/02/08 10:23

(3) ジョブの実行順序

実行順序は、wait ジョブの番号の古いジョブから基本的に優先されますが、Priority を変えると実行順序を変えることができます。Priority は 0 ~ 9 まであり、小さい方が高い優先度です。下図の例で、run 中のジョブは 1 個で、1 core を使用中で 1 core 分の実行は可能であるが、次の実行順序にある 2 番目のジョブは 2 core を要求しているため、2 core 空くまで wait 状態が続きます。ここで、1 番目のジョブの Priority を 4 以上に上げると、1 番目は 1 core ジョブなので、2 番目を待たずに 1 番目が run になります。

(4) ジョブの強制終了

ジョブが正常終了する場合は、ジョブマネージャが自動的に認識して end 状態になりますが、強制終了したい時には、DOS 窓の[x] (閉じるボタン) を押すことで終了できます。run 状態のジョブの行をクリックして Status を end に変更することでも終了できる場合がありますが、MOPAC2009 では、DOS 窓の[x]の押下でしか終了できません。

ジョブ名は実行キューに入った時刻を元にしたユニークな名前がつけられています。run 状態のジョブの行をクリックすると実行中の DOS 窓が前面に出るので、実効中の画面を確認することができます。

(5) 開始時刻と強制実行

開始時刻を未来に設定することで、実行を遅らすことができます。これによって、深夜からジョブを開始する設定も可能です。一旦 run になってしまったジョブは (4) の強制終了した後、未来の開始時刻にして、Status を wait に変えることで未来時刻の wait ジョブが設定できます。

開始時刻で wait のジョブがあった時は、それより若い番号の wait ジョブはすべて実行待ちになります。これは、空き core 待ちの wait ジョブがある時と同じです。wait ジョブの実行順序を変えるのは、Priority を変更することで可能です。

一旦実行してみて、エラーがすぐに出ないことを確認しておくのは良い方法です。このような目的で、wait のジョブの Status を run に変更することで、キューの順序や Priority 等の全ての条件を無視して強制実行することができます。end ジョブも run にすることができます。

(6) 並列計算

実行 core 数 は、G03W の場合は%nproc=の値、GAMESS の場合は NCPUS の値が設定されます。G03W は並列計算版が必要で、最大 4core までの制限があります。

複数の並列計算を同時に実行することも可能です。

(7) JM の起動と停止

JM は、WM からジョブを投入した時に起動されますが、別プログラムなので WM を停止しても JM は停止しません。JM を停止しないで、WM からジョブを連続して投入することができます。JM は二重起動が防止されていますので、複数の JM が立ち上がることはありません。JM のメニューから JM 自身を停止すると、それ以降は wait ジョブは実行されません。誤って JM を停止した場合は、WM のメニューのその他、[Job Manager]で起動すると、その時点から wait ジョブの実行や end ジョブの検出が行われます。複数の WM からジョブを投入しても、一つの実行キューを共用するので問題ありません。

(8) 実行キューについて

実行キューの最大は 50 個です。50 個を超えると、古いものから削除されます。キューから削除されても、実行中の計算が停止することはありません。キューを削除したり、順序を変える方法はありません。削除の代わりに、Status を wait から end に変えれば、run になることはありません。実行順序を変えるには、Priority や開始時刻を変更します。

(9) 省電力機能について

JM の起動中は、時間設定によってスタンバイ (スリープ) や休止状態に入ることを、JM が防止しています。キーボード操作でスタンバイ等にはなりますが、テレビ録画ソフトのように、スタンバイ等の状態から復帰する機能はありませんので、ご注意ください。

AutoShutdown にチェックしておく、すべてのジョブが end 状態になると、自動的にシャットダウンと電源オフを行います。