

## Linux の資源管理機構 Control Groups の性能調査

富樫 荘太<sup>†</sup> 片山 吉章<sup>††</sup> 出口 昌弘<sup>††</sup> 毛利 公一<sup>†</sup><sup>†</sup>立命館大学情報理工学部 <sup>††</sup>三菱電機株式会社情報技術総合研究所

## 1 はじめに

Linux には Cgroups(Control Groups)[1] と呼ばれる資源管理機構がある。Cgroups は、プロセスを単位とした資源管理を可能とする。具体的には、任意のプロセスを階層的にグループ化して、CPU や I/O、メモリ、ネットワークなどの計算機資源毎に割当て量を指定することができる。本論文では、Cgroups がどの程度の時間的精度で資源割当てを可能としているのか、特に CPU リソースコントローラの CPU 資源の割当て間隔の精度について調査したのでその結果を報告する。

## 2 Cgroups

Linux 2.6.24 以降、カーネルの機能として Cgroups が提供されている。Cgroups は、図 1 に示すように、Cgroups 仮想ファイルシステムを通じて利用可能である。Cgroups は、図 1 のコントローラ A、コントローラ B、コントローラ C のように、CPU、メモリ、I/O、ネットワークといった資源毎にリソースコントローラを割り当てて資源管理を行う。図 1 では、コントローラ A、コントローラ B が階層 1 に、コントローラ C が階層 2 にマウントされている。階層は、1 つ以上のプロセスグループから構成される。グループは、Cgroups 仮想ファイルシステムによって、ディレクトリとして表され、マウントされたリソースコントローラのパラメータファイルと、所属プロセスが記述された Tasks ファイルから構成される。図 1 の階層 2 のように、作成直後の階層は、ルートグループのみからなる。資源管理方法をグループ毎に分ける場合、図 1 の階層 1 のように、サブグループを生成しておのおの管理方法を分ける。サブグループは、特に指定しない限りは親のグループの管理方法を継承する。

CPU 資源を扱うリソースコントローラとして、cpu と cpuset[2] がある。cpu リソースコントローラは、パラメータ shares によって値を設定することで、その値の比率に従ってグループ毎の CPU 時間を割り当てる。cpuset リソースコントローラは、パラメータ cpus や mems によって、個々の物理 CPU とメモリノードを割り当てる。

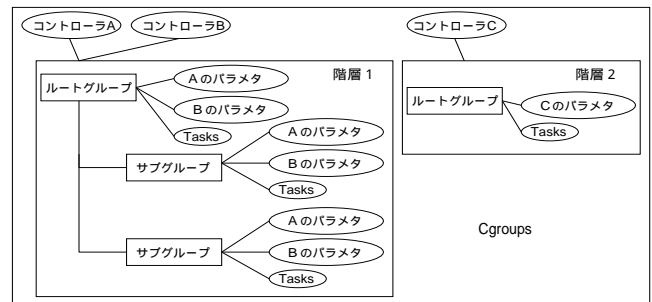


図 1 Cgroups の構成

## 3 cpu リソースコントローラの評価

Cgroups の cpu リソースコントローラの評価を行った。具体的には、CPU 資源の割当て間隔がどの程度の精度であるかを評価する。cpu リソースコントローラでは、複数のグループが存在する場合、パラメータ shares の値の比率に従って CPU 利用率を保証する。本稿では、この保証に cpu リソースコントローラが CPU 時間をどの程度の時間解像度で割り当てるのかを明らかにするため、測定周期を変更しながら、CPU 利用率を測定する。また、負荷をかけ、CPU 資源管理のアイソレーションがどの程度保証されているかを検討する。

## 3.1 評価環境と評価用設定

評価は、表 1 のような環境で行った。評価では、図 2 に示すように cpu リソースコントローラと、CPU 数の割当てを設定する cpuset リソースコントローラの 2 つを使用する。なお、Linux の制約により、ルートグループの CPU 数は変更不可能であり、システムプロセスはルートグループに所属しなければならない。よって、ルートグループのサブグループとして、計測対象のプロセス(グループ A)、計測処理をするプロセス(グループ B)、負荷を生成するプロセス(グループ C)の 3 つのグループに分割した。グループ A は、CPU0 を割り当て、サブグループ A1~A4 を持つ。サブグループ A1~A4 は、それぞれビジーグループを 1 個実行する。グループ B は、CPU3 を割り当て、CPU 利用率を得るために top コマンドを実行する。top は、周期 10, 15, ..., 100msec の各測定周期で 1000 回の計測を行う。また、グループ C は、CPU1~2 を割り当て、負荷なし、CPU 負荷、I/O 負荷、メモリ負荷の 4 通りの負荷を生成するプログラムを動作させた。なお、どのグループも、shares をデフォルトの 1024 に設定し、割り当てられた CPU を均等に使用する。

Performance Evaluation of Linux Resource Controller “Control Groups”

Souta Togashi<sup>†</sup>, Yoshiaki Katayama<sup>††</sup>, Masahiro Deguchi<sup>††</sup> and Koichi Mouri<sup>†</sup><sup>†</sup>College of Information Science and Engineering, Ritsumeikan University<sup>††</sup>Information Technology R&D Center, Mitsubishi Electric Corporation

表 1 評価環境

項目	内容
CPU	Intel Core2 Quad Q6600 2.4GHz
Memory	2.0GB
OS	Fedora Core 14
Kernel	Linux Kernel 2.6.35.14-96.fc14.i686

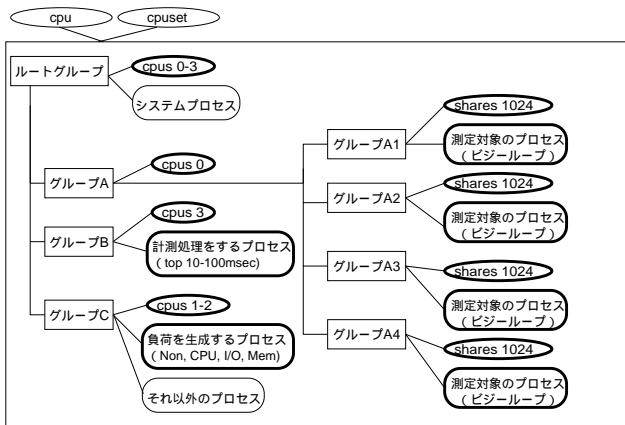


図 2 Cgroups の評価用設定

### 3.2 結果と考察

前項の評価環境にて、ビジーループ 1 個分の CPU 利用率を計測した。

図 3 は、測定周期 10, 30, 50, 70msec における CPU 利用率の頻度分布である。ビジーループが 4 個均等に CPU を用いたときの CPU 利用率は 25% になると予想できる。ここで、測定周期が 10, 30msec の場合は、CPU 利用率が 0% となる計測結果が得られた。また、0% として計測される頻度は、測定周期が長くなるほど低くなり、測定周期を約 30msec より大きくすると 0 になった。よって、Cgroups による CPU の割当て間隔の精度は、30msec 程度であることがわかる。すなわち、Cgroups が有効に機能するためには最小でも 30msec 間隔の周期が必要になる。

図 4 は、各測定周期における負荷毎の標準偏差である。10 ~ 30msec で値が減少し、以降は、30, 70, 100msec のときに極小値をとりつつ、減少していく。また、図 3 で、測定周期 30, 70msec の CPU 利用率の分布は、ほぼ期待される 25% のみであるのに対して、測定周期 10, 50msec の場合は、25% を挟む形で分布が分かれている。したがって、Cgroups は、約 30 ~ 40msec の整数倍の周期であれば、常に約 25% の CPU 利用率を保証できることが分かる。

また、図 4 にて、負荷による影響は極小値付近でみられ、特に I/O 負荷、メモリ負荷が大きくなった。これは、グループ A とグループ C で、CPU 以外の資源は分離さ

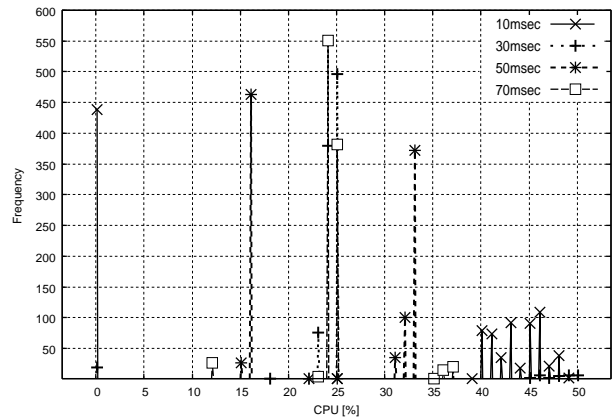


図 3 CPU 利用率の頻度分布 (測定周期 10, 30, 50, 70msec)

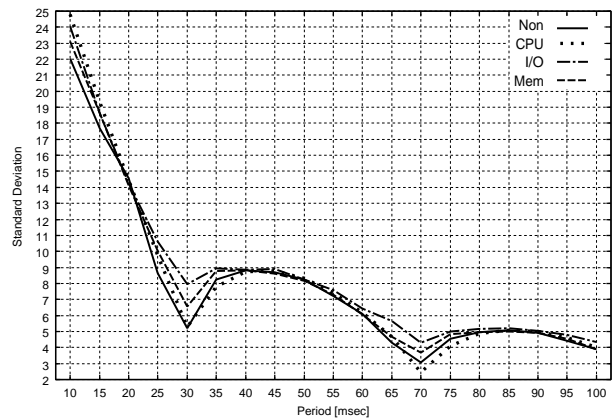


図 4 各測定周期における負荷ごとの標準偏差

せていないため、グループ C での負荷がグループ A の処理に影響するためである。

### 4 おわりに

本稿では、Linux におけるプロセス群の資源管理機能である Cgroups について、その cpu リソースコントローラの CPU 割当て間隔の精度の評価を行った。今後は、cpu リソースコントローラの実タイムプロセスの資源管理について性能評価を行う。

### 参考文献

- [1] Paul Menage: "Linux Kernel Documentation/cgroups/cgroups.txt," <http://www.kernel.org/doc/Documentation/cgroups/cgroups.txt> (2011).
- [2] Simon Derr: "Linux Kernel Documentation/cgroups/cpusets.txt," <http://www.kernel.org/doc/Documentation/cgroups/cpusets.txt> (2011).