

第3章

プロセス

初期のコンピュータシステムでは、一度に一つのプログラムしか実行できなかった。このプログラムがシステムの制御を完全に握り、すべてのシステム資源にアクセスできた。これに対し、現在のコンピュータシステムは複数のプログラムをメモリにロードし、並列に実行することができる。このようにするためには、より確かな制御と種々のプログラムのさらなる区分化が必要であった。結果としてプロセス (process) の概念が出てきた。プロセスとは実行中のプログラムである。プロセスは現代のタイムシェアリングシステムにおける作業の単位である。

オペレーティングシステムが複雑になればなるほど、ユーザのために期待されることが多くなる。その主たる関心事はユーザプログラムの実行であるが、カーネル自身の外に出たほうがよい、様々なシステムタスクの面倒も見必要がある。したがって、システムはプロセスの集合からなる。システムコードを実行するオペレーティングシステムプロセスと、ユーザコードを実行するユーザプロセスがある。これらのすべてのプロセスは、潜在的には、単一あるいは複数の CPU をプロセス間で多重化することによって並列に実行できる。プロセス間で CPU を切り替えることによって、オペレーティングシステムはコンピュータをより生産的にすることができる。

3.1 プロセスの概念

オペレーティングシステムの議論で生じる一つの問題は、すべての CPU 活動を何と呼ぶかということである。バッチシステムはジョブ (job) を実行するが、タイムシェアリングシステムは「ユーザプログラム」 (user program) あるいは「タスク」 (task) を持つ。マイクロソフト社の Windows のような単一ユーザシステムですら、ユーザは、ワープロ、ウェブブラウザ、電子メールなど、いくつかのプログラムを一度に実行できる。たとえ一度に一つのプログラムしか実行できないとしても、オペレーティングシステムは、メモリ管理など、その内部にプログラム化された活動を行わなければならない。多くの点で、これらすべての活動は類似しているので、これらをすべてプロセス (process) と呼ぶ。

本書では、「ジョブ」という用語と「プロセス」という用語をほとんど同じように使う。個人的には「プロセス」という用語の方が好きだが、多くのオペレーティングシステム理論や用語は、オペレーティングシステムの主要な活動が「ジョブ処理」 (job processing) であったころに開発されている。単にプロセスがジョブに取って代わったということだけで、「ジョブスケジューリング」 (job scheduling) といった「ジョブ」という言葉を含む一般に受け入れられている用語を使わないとすると誤解を招きかねない。

3.1.1 プロセス

ざっくりばらんに言うと、前述したとおり、プロセスとは実行中のプログラムである。プロセスは単なるプログラムコードではない。プログラムコードは、時にはテキストセクション (text

section) として知られる。プロセスは、プログラムカウンタ (program counter) の値およびプロセッサのレジスタの内容によって表現されている、現在の活動も含む。さらに、プロセスは、一般に、関数引数、戻り先アドレス、局所変数といった一時的なデータを含むプロセススタック (process stack)、および大域変数を含む データセクション (data section) も含む。プロセスはまた、プロセスの実行時に動的に割り当てられたメモリである ヒープ (heap) も含む。メモリ内のプロセスの構造は図 3.1 のようになる。

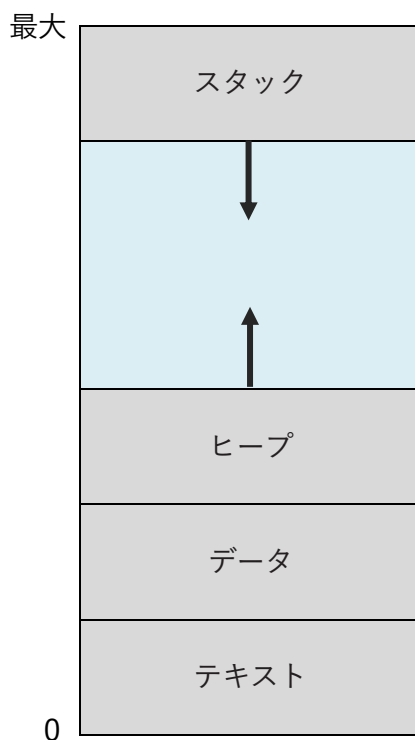


図 3.1 メモリ内のプロセス

プログラムそれだけではプロセスではない。プログラムはディスク上に格納された命令のリストを含むファイル (実行可能ファイル (executable file) ということが多い) のような「受動的な」実体であるのに対し、プロセスは実行すべき次の命令を指定するプログラムカウンタや、関連付けられた資源の集合を持つ「能動的な」実体である。実行可能ファイルがメモリにロードされるとプロセスになる。実行可能ファイルをロードするための一般的な技法としては、実行可能ファイルを表しているアイコンをダブルクリックする方法と、コマンド行に実行可能ファイル名を (“prog.exe” とか “a.out” といったように) 入れる方法とがある。

二つのプロセスが同じプログラムに関連付けられることも起こりうるが、それらは二つの個別の実行列と考える。たとえば、何人ものユーザがメールプログラムの相異なるコピーを実行しているかもしれないし、一人のユーザがウェブブラウザプログラムのいくつものコピーを使っているかもしれない。これらの各々はそれぞれ個別のプロセスであり、テキストセクションは同じであっても、データセクション、ヒープ、スタックは異なっている。走るにつれて子プロセスをたくさん生むプロセスもある。