**Định nghĩa:**Quá trình (process) là trạng thái tức thời của một chương trình đang chạy trên máy tính. Nó bao gồm bộ nhớ cần thiết để chạy chương trình (không gian địa chỉ của quá trình) và khả năng kiểm soát hiện trạng của bộ xử lý trong quá trình thực thi chương trình (tiến trình điều khiển của quá trình). Luồng (thread) tương tự như quá trình nhưng chỉ bao gồm tiến trình điều khiển. Nhiều luồng sử dụng không gian địa chỉ của một quá trình.  
Quá trình và luồng có chung một mục đích: buộc máy tính phải làm nhiều việc hơn tại một thời điểm. Để làm điều đó, bộ xử lý (hay các bộ xử lý) phải chuyển đổi một cách trơn tru giữa các tác vụ, điều này đòi hỏi chương trình ứng dụng phải được thiết kế để chia sẻ tài nguyên máy tính.  
Đó là lý do tại sao lập trình viên cần chia những gì chương trình phải làm thành quá trình và luồng.  
Mỗi chương trình chạy trên một máy tính cần ít nhất là một quá trình. Quá trình đó bao gồm không gian địa chỉ (phần bộ nhớ máy tính mà ở đó chương trình chạy) và tiến trình điều khiển (cách thức để biết được phần nào của chương trình đang được bộ xử lý thực thi tại bất kỳ thời điểm nào). Nói cách khác, quá trình là một vùng làm việc và cách thức quản lý những gì chương trình đang thực hiện. Khi một số chương trình chạy cùng một thời điểm, mỗi chương trình sẽ có không gian địa chỉ và tiến trình điều khiển của riêng nó (xem sơ đồ).  
Để phục vụ nhiều người dùng, một quá trình có thể cần phải phân nhánh, hay tạo bản sao của chính nó để tạo ra một quá trình con. Cũng giống như quá trình mẹ, quá trình con cũng có không gian địa chỉ và tiến trình điều khiển riêng. Tuy nhiên, thường thì khi quá trình mẹ chấm dứt, mọi quá trình con mà nó khởi động cũng sẽ tự động chấm dứt.  
Hệ điều hành đa nhiệm như Unix hay Windows thực hiện việc chuyển đổi qua lại giữa các quá trình, lần lượt phân thời gian sử dụng CPU cho từng quá trình. Nếu máy tính có nhiều CPU, mỗi quá trình có thể được gán riêng cho một trong các CPU.  
Điều này thích hợp với các chương trình đơn giản. Các ứng dụng phức tạp hiện nay như xử lý văn bản hay bảng tính có thể xem như là nhiều chương trình khác nhau với yêu cầu chuyển đổi qua lại và giao tiếp giữa các quá trình liên tục. Đây là một vấn đề vì phải mất thời gian để chuyển đổi giữa các quá trình.   
CPU hiện đại có bộ quản lý bộ nhớ (memory management unit - MMU) để ngăn bất kỳ quá trình nào vi phạm không gian bộ nhớ của quá trình khác. Chuyển từ một quá trình này sang quá trình khác - được gọi là chuyển ngữ cảnh - có nghĩa là lập trình lại MMU để chỉ đến không gian địa chỉ khác cùng với việc lưu và phục hồi thông tin của một quá trình. Hệ điều hành chịu trách nhiệm quản lý chi tiết của việc chuyển ngữ cảnh nhưng nó cũng tiêu tốn thời gian của CPU. Do mỗi quá trình đều được cách ly với những quá trình khác, giao tiếp giữa các quá trình đòi hỏi phải có những chức năng đặc biệt. Tương tự việc chuyển ngữ cảnh, truyền thông giữa các quá trình cũng chiếm thời gian của bộ xử lý.  
Tất cả thời gian trên sẽ cộng dồn lên khi nhiều chương trình chạy cùng lúc hay khi có nhiều người dùng mà mỗi người đều yêu cầu chạy nhiều quá trình cũng lúc. Càng nhiều quá trình chạy thì càng tốn nhiều thời gian của CPU và hệ điều hành để thực hiện công việc chuyển ngữ cảnh.  
Nếu số quá trình đủ nhiều, máy chủ có thể phải dành toàn bộ thời gian để thực hiện việc chuyển đổi giữa các quá trình mà không thể thực sự xử lý được công việc nào.

**Phân Luồng**

**Để tránh tình trạng trên**, lập trình viên có thể dùng luồng (thread). Luồng cũng giống một quá trình con, ngoại trừ đặc điểm là mọi luồng kết hợp với một quá trình nào đó chia sẻ cùng không gian địa chỉ.

|  |
| --- |
| http://www.pcworld.com.vn/pcworld/info/misc/2003/12/01A_CN_Can-ban-cap-toc2.gif |

Ví dụ, nếu có nhiều người dùng cùng chương trình, lập trình viên có thể viết ứng dụng sao cho ứng với mỗi người dùng sẽ có một luồng mới được tạo ra.   
Mỗi luồng có tiến trình kiểm soát riêng nhưng nó lại chia sẻ cùng không gian địa chỉ và hầu hết dữ liệu với tất cả luồng khác chạy trong cùng quá trình. Đối với từng người dùng có thể nói rằng chương trình dường như chỉ chạy cho một mình họ.  
**Ưu điểm là gì?** Thời gian chuyển đổi giữa các luồng ít hơn hẳn so với giữa các quá trình vì không cần phải chuyển đổi không gian địa chỉ. Ngoài ra, vì chúng chia sẻ không gian địa chỉ nên các luồng trong một quá trình có thể giao tiếp với nhau dễ dàng hơn nhiều.  
Trên máy tính có nhiều bộ xử lý, chương trình dạng một quá trình đơn chỉ chạy trên một CPU, còn chương trình dạng luồng có thể chia các luồng cho tất cả các bộ xử lý. Vì thế, nếu bạn chuyển chương trình dạng luồng sang máy chủ nhiều bộ xử lý thì nó sẽ chạy nhanh hơn.  
**Khuyết điểm?** Chương trình dạng luồng khó viết và kiểm lỗi hơn. Không phải mọi thư viện lập trình đều được thiết kế để dùng với luồng và không phải mọi ứng dụng cũ đều có thể làm việc tốt với ứng dụng dạng luồng. Một vài công cụ lập trình cũng làm cho việc thiết kế và thử nghiệm mã luồng khó khăn hơn.  
Lỗi liên quan đến luồng cũng khó phát hiện hơn. Các luồng trong một quá trình có thể bị chồng chéo dữ liệu với nhau. Hệ điều hành có thể hạn chế số luồng thực thi chẳng hạn đọc và ghi dữ liệu cùng lúc. Việc định thời cho các luồng khác nhau để tránh xung đột là rất khó khăn.  
Tuy nhiên, khi các ứng dụng dùng chung phức tạp và máy chủ nhiều bộ xử lý ngày càng phổ biến thì luồng sẽ ngày càng được dùng nhiều để thực hiện đa xử lý.