Tổng quan về hệ thống phân tán

Giới thiệu

Mạng máy tính ở khắc mọi nơi.Internet là một,cũng như nhiều mạng mà nó cấu thành. Mạng điện thoại di động, mạng công ty, mạng nhà máy,mạng trong khuân viên trường,mạng trong xe – tất cả những mạng này điều có sự riêng biệt và kết hợp,điều có điểm chung thiết yếu trở thành những đối tượng nghiên cứu thích hợp trong hệ thống phân tán. Trong cuốn sách này, chúng tôi mong muốn giải thích các đặc điểm của máy tính nối mạng ảnh hưởng đến các nhà thiết kế và triển khai hệ thống, đồng thời trình bày các khái niệm kĩ thuật chính được phát triển để trợ giúp cho các nhiệm vụ thiết kế và triển khai hệ thống dựa trên chúng.

Chúng tôi định nghĩa hệ thống phân tán là một hệ thống trong đó các thành phần phần cứng hoặc phần mềm đặt tại các máy tính nối mạng giao tiếp và điều phối các hành động của chúng bằng cách chuyển các thông điệp. định nghĩa đơn giản này bao gồm phạm vi hệ thống trong đó các máy tính nối mạng có thể triển khai một cách hữu ích.

Các máy tính được kết nối bởi một mạng có thể được phân tách về mặt không gian bởi bất kỳ khoảng cách nào. Họ có thể ở các quốc gia riêng biệt,trong cùng một toà nhà hoặc cùng một phòng. Định nghĩa của chúng tôi về hệ thống phân tán có những hậu quả đáng kể sau:

Đồng thời: trong một mạng máy tính,thực hiện chương trình đồng thời là một tiêu chuẩn. tôi có thể thực hiện công việc của tôi trên máy tính của tôi trong khi bạn thực hiện công việc của bạn trên máy tính của bạn,chia sẻ tài nguyên như trang wed hoặc tệp khi cần thiết. năng lực của hệ thống để xử lý tài nguyên dùng chung có thể được tăng lên bằng cách thêm nhiều tài nguyên hơn (ví dụ:máy tính) vào mạng. chúng tôi sẽ mô tả những cách mà năng lực bổ sung này có thể được triển khai hữu ích ở nhiều điểm trong cuốn sách này. Việc phối hợp thực hiện đồng thời các chương trình chia sẻ tài nguyên cũng là một chủ đề quan trọng và thường xuyên.

Không đồng bộ ở mọi nơi: khi các chương trình cần hợp tác, họ phối hợp hành động của mình bằng cách trao đổi thông điệp. sự phối hợp chặt chẽ thường phụ thuộc vào ý tưởng chung về thời gian mà các hành động của chương trình xảy ra. Nhưng hoá ra có những giới hạn đối với độ chính xác mà các máy tính trong mạng có thể đồng bộ hoá thời gian của chúng – không có khái niệm toàn cầu về thời gian chính xác. Đây là hệ quả trực tiếp của thực tế là giao tiếp bằng cách gửi thông điệp qua mạng. ví dụ những vấn đề thời gian này và giải pháp cho chúng được mô tả trong chương 14.

Lỗi độc lập: tất cả các hệ thống máy tính đều có thể bị lỗi,và trách nhiệm của các nhà thiết kế hệ thống là phải lập kế hoạch cho các hậu quả của các lỗi có thể xảy ra. Hệ thống phân tán có thể bị lỗi theo những cách mới. lỗi trong mạng dẫn đến tách các máy tính được kết nối với nó, nhưng điều đó không có nghĩa là ngừng chạy. trên thực tế, các chương trình trên chúng có thể không phát hiện được mạng bị lỗi hoặc trở nên chậm bất thường. tương tự như vậy, sự cố của một máy tính hoặc sự cố bất ngờ của một chương trình ở đâu đó trong hệ thống (treo) không được biết ngay lập tức đối với các thành phần khác mà đang giao tiếp với nó. Mỗi thành phần của hệ thống có thể bị lỗi một cách độc lập, khiến các thành phần khác vẫn chạy. hậu quả của đặc tính này của hệ thống phân tán sẽ là một chủ đề lặp đi lặp lại xuyên suốt cuốn sách.

Động lực chính để xây dựng và sử dụng các hệ thống phân tán bắt nguồn từ mong muốn chi sẻ tài nguyên. Thuật ngữ “tài nguyên” là một thuật ngữ khá là trừu tượng, nhưng nó mô tả tốt phạm vi những thứ có thể được chia sẻ hữu ích trong một hệ thống máy tính nối mạng. nó mở rộng từ các thành phần phần cứng như đĩa và máy in đến các thực thể do phần mềm đĩnh nghĩa như tệp,cơ sở dữ liệu và đối tượng dữ liệu thuộc mỗi loại. nó bao gồm khung hình video xuất hiện từ máy quay video kỹ thuật số và kết nối âm thanh mà một cuộc gọi điện thoại di động thể hiện.

1 khái quát về hệ phân tán

Hệ thống phân tán hay còn được biết đến với tên khác là điện toán phân tán – là một hệ thống bao gồm nhiều thành phần hoạt động đọc lập, trao đổi và làm việc với nhau (qua mạng) để giải quyết một vấn đề cụ thể.

Các thành phần trong hệ thống phân tán có thể là một máy tính (máy vật lý hoặc máy ảo), container, hay bất kỳ thiết bị kết nối mạng nào, có bộ nhớ local và có thể giao tiếp với nhau thông qua mạng máy tính.

Nếu coi tất cả các thành phần trong hệ thống phân tán là các máy tính, thì có thể hiểu nhanh

Hệ thống phân tán = các máy tính + Mạng máy tính

+lợi ích của việc triển khai hệ thống phân tán

-Có khả năng mở rộng theo chiều ngang: Vì mỗi thành phần trong hệ thống phân tán là một máy tính (hoặc các thành phần khác như đã chỉ ra ở trên) nên có thể dễ dàng thêm bớt khi cần, chi phí cũng rẻ.

-Khả năng chịu lỗi cao: Vì mỗi thành phần đều hoạt động riêng biệt, nên việc có một thành phần ngừng hoạt động cũng không ảnh hưởng tới toàn hệ thống. Nếu xảy ra lỗi, sẽ dễ dàng thay thế hoặc khôi phục.

-Hiệu năng: Một trong những lợi ích quan trọng nhất của hệ thống phân tán đó là có hiệu năng cao. Do các công việc sẽ được chia đều cho các máy nên chúng có thể xử lý song song.

Tuy nhiên không phải trong trường hợp nào cũng có thể triển khai hệ thống phân tán

-đặc tính của hệ thống phân tán

+chia sẻ tài nguyên: Tiết kiệm chi phí đầu tư => số lượng các thiết bị ngoại vi đầu tư cho máy tính giảm => giảm suất đầu tư trên từng người sử dụng

Việc cho phép NSD kết nối các tài nguyên ở xa và các máy khác nhau làm tăng khả năng sẵn sàng của hệ thống

+Trong suốt: Tính trong suốt là khả năng cung cấp một khung cảnh lôgic của hệ thống cho người dùng, độc lập với hạ tầng vật lý. Hệ thống luôn là duy nhất đối với người dùng song nó sẽ che giấu được tính phân tán của hệ phân tán phía dưới.

+tính mở : dễ dàng mở rộng phần cứng (thiết bị ngoại vi ,bộ nhớ ,các giao diện truyền thông..) và phần mềm(các mô hình HĐH,các giao thức truyền thông,…) -> tính mở mang ý nghĩa bao hàm dễ dàng cấu hình cả phần cứng và phần mềm.

+tính co giãn: Là 1 khả năng của hệ thống có thể đáp ứng được các thay đổi của hạ tầng của môi trường xung quanh

+các loại hệ thống phân tán

Máy khách-máy khách—Máy khách liên hệ với máy chủ để tìm dữ liệu, sau đó định dạng nó và hiển thị nó cho người dùng cuối. Người dùng cuối cũng có thể thực hiện thay đổi từ phía máy khách và cam kết nó trở lại máy chủ để làm cho nó vĩnh viễn.

Ba tầng —Thông tin về máy khách được lưu trữ ở tầng giữa thay vì trên máy khách để đơn giản hóa việc triển khai ứng dụng. Mô hình kiến trúc này là phổ biến nhất cho các ứng dụng web.

n-tier—Thường được sử dụng khi một ứng dụng hoặc máy chủ cần chuyển tiếp các yêu cầu đến các dịch vụ doanh nghiệp bổ sung trên mạng.

Peer-to-peer —Không có máy bổ sung nào được sử dụng để cung cấp dịch vụ hoặc quản lý tài nguyên. Trách nhiệm được phân phối thống nhất giữa các máy trong hệ thống, được gọi là đồng nghiệp, có thể phục vụ như máy khách hoặc máy chủ.

1. 2 các ví dụ về hệ thống phân tán

+tìm kiếm bằng wed:

-là một ngành tăng trưởng chính trong thập kỉ qua .nhiệm vụ của công cụ tìm kiếm wed là lập chỉ mục toàn bộ nội dung của world wide wed

-có thể coi đây là một mô hình công cụ trong đó tác vụ thu thập dữ liệu wed,lập chỉ mục và xử lý truy vấn được phân phối giữa nhiều máy tính và mạng

- Ban đầu, hầu hết các công cụ tìm kiếm được hỗ trợ bởi một siêu máy tính duy nhất . Tuy nhiên, trong những năm gần đây, hầu hết đã chuyển sang mô hình phân tán. Ví dụ, tìm kiếm của Google dựa vào hàng ngàn máy tính thu thập dữ liệu Web từ nhiều vị trí trên toàn thế giới.

- Trong hệ thống tìm kiếm phân tán của Google, mỗi máy tính tham gia lập chỉ mục thu thập dữ liệu và đánh giá một phần web, lấy URL và theo dõi mọi liên kết có sẵn từ nó (trừ những liên kết được đánh dấu để loại trừ). Máy tính thu thập kết quả thu thập dữ liệu từ các URL và gửi thông tin đó trở lại máy chủ tập trung ở định dạng nén. Máy chủ tập trung sau đó điều phối thông tin đó trong cơ sở dữ liệu , cùng với thông tin từ các máy tính khác liên quan đến lập chỉ mục.

- Khi người dùng nhập truy vấn vào trường tìm kiếm, phần mềm máy chủ tên miền (DNS) của Google chuyển tiếp truy vấn đến cụm máy tính hợp lý nhất, dựa trên các yếu tố như sự gần gũi của nó với người dùng hoặc mức độ bận rộn của nó. Tại cụm người nhận, phần mềm máy chủ Web phân phối truy vấn đến hàng trăm hoặc hàng ngàn máy tính để tìm kiếm đồng thời. Hàng trăm máy tính quét chỉ mục cơ sở dữ liệu để tìm tất cả các hồ sơ liên quan. Máy chủ chỉ mục biên dịch kết quả, máy chủ tài liệu tập hợp các tiêu đề và tóm tắt và trình tạo trang tạo các trang kết quả tìm kiếm.

Google,công ty dẫn đầu thị trường về công nghệ tìm kiếm trên wed,đã nỗ lực đáng kể trong việc thiết kế cơ sở hạ tầng hệ thống phân tán tinh vi để hỗ trợ tìm kiếm(và thực tế là các ứng dụng và dịch vụ khác của google như Google Earth).điều này đại diện cho một trong những cài đặt hệ thống phân tán lớn và phức tạp nhất trong lịch sử máy tính do đó yêu cầu kiểm tra chặt chẽ.điểm nổi bật của cơ sở hạ tầng này gồm:

* Một sơ sở hạ tầng vật lý cơ bản bao gồm một số lượng rất lớn các máy tính nối mạng đặt tại các trung tâm dữ liệu trên toàn thế giới;
* Một hệ thống tệp phân tán được thiết kế để hỗ trợ các tệp rất lớn và được tối ưu hoá nhiều cho phong cách sử dụng theo yêu cầu tìm kiếm và ứng dụng khác của Google(đặc biệt là đọc từ tệp với tốc độ cao và ổn định);
* Một hệ thống lưu trữ phân tán có cấu trúc liên kết cung cấp khả năng truy cập nhanh vào các tập dữ liệu rất lớn;
* Một dịch vụ khoá cung cấp các chức năng hệ thống phân tán như khoá phân tán và thoả thuận;
* Một mô hình lập trình hỗ trợ quản lý các phép tính song song và phân tán rất lớn trên cơ sở hạ tầng vật lý cơ bản

+ trò chơi trực tuyến hỗ trợ nhiều người chơi cùng một lúc(MMOG)

-Trò chơi trực tuyến nhiều người chơi mang đến trải nghiệm nhập vai, theo đó số lượng lớn người dùng tương tác qua Internet với một thế giới ảo liên tục. Ví dụ hàng đầu về các trò chơi như vậy bao gồm Sony’s EverQuest II và EVE Online của công ty Phần Lan CCP Games. Những thế giới như vậy đã tăng lên đáng kể về độ tinh vi và giờ đây bao gồm các đấu trường chơi phức tạp (ví dụ như EVE, Trực tuyến bao gồm một vũ trụ với hơn 5.000 hệ sao) và các hệ thống kinh tế và xã hội đa dạng. Số lượng người chơi cũng đang tăng lên, với các hệ thống có thể hỗ trợ hơn 50.000 người chơi trực tuyến đồng thời (và tổng số người chơi có lẽ gấp mười lần con số này).

-Kỹ thuật của MMOG là một thách thức lớn đối với các công nghệ hệ thống phân tán, đặc biệt là do yêu cầu thời gian phản hồi nhanh để duy trì trải nghiệm của người dùng trong trò chơi. Những thách thức khác bao gồm việc truyền bá các sự kiện trong thời gian thực cho nhiều người chơi và duy trì quan điểm nhất quán về thế giới được chia sẻ. Do đó, điều này cung cấp một ví dụ tuyệt vời về những thách thức mà các nhà thiết kế hệ thống phân tán hiện đại phải đối mặt.

+giao dịch tài chính

Ngành tài chính từ lâu đã đi đầu trong công nghệ hệ thống phân tán với nhu cầu của nó, đặc biệt, để truy cập thời gian thực vào nhiều nguồn thông tin (ví dụ, giá cổ phiếu hiện tại và xu hướng, phát triển kinh tế và chính trị). Ngành này sử dụng các ứng dụng giám sát và giao dịch tự động

-Lưu ý rằng trọng tâm trong các hệ thống như vậy là liên lạc và xử lý các mục quan tâm, được gọi là các sự kiện trong hệ thống phân tán, với nhu cầu cũng là cung cấp các sự kiện một cách đáng tin cậy và kịp thời cho một số lượng rất lớn khách hàng có mối quan tâm đã nêu. trong các mục thông tin đó. Ví dụ về các sự kiện như vậy bao gồm giảm giá cổ phiếu, công bố số liệu thất nghiệp mới nhất, v.v. Điều này đòi hỏi một phong cách kiến ​​trúc cơ bản rất khác với các phong cách được đề cập ở trên (ví dụ: máy khách-máy chủ) và các hệ thống như vậy thường sử dụng những gì được gọi là hệ thống dựa trên sự kiện phân tán.

-ta có thể lấy ví dụ như Blockchain, Blockchain như một cuốn số cái kế toán công cộng. Trong đó, mọi thông tin được lưu trữ và truyền tải một cách minh bạch, toàn vẹn, không thể nào thay đổi hay gian lận được. Đây là một công nghệ mới, giúp cải thiện được rất nhiều những mặt hạn chế của cách lưu trữ và trao đổi thông tin truyền thống. Bởi lý do này, mà blockchain ngày càng được ứng dụng rộng rãi trong nhiều lĩnh vực: kinh tế tài chính, giáo dục, nông nghiệp, công nghiệp, lĩnh vực giải trí, y tế hay giáo dục….

-Sâu về kỹ thuật hơn, Blockchain là một database phân tán (phi tập trung) mà trong đó các dữ liệu được lưu trữ dưới dạng các blocks. Body của một block mang theo các transactions trên dữ liệu (như state machine). Block được kết nối với nhau theo dạng linked list (danh sách liên kết) dưới dạng mã hóa SHA256.

* 1. các xu hướng của hệ thống phân tán

các hệ thống phân tán đang trải qua một thời kỳ thay đổi đáng kể và điều này có thể được tìm ra một số xu hướng có ảnh hưởng:

-sự xuất hiện của công nghệ mạng lan toả

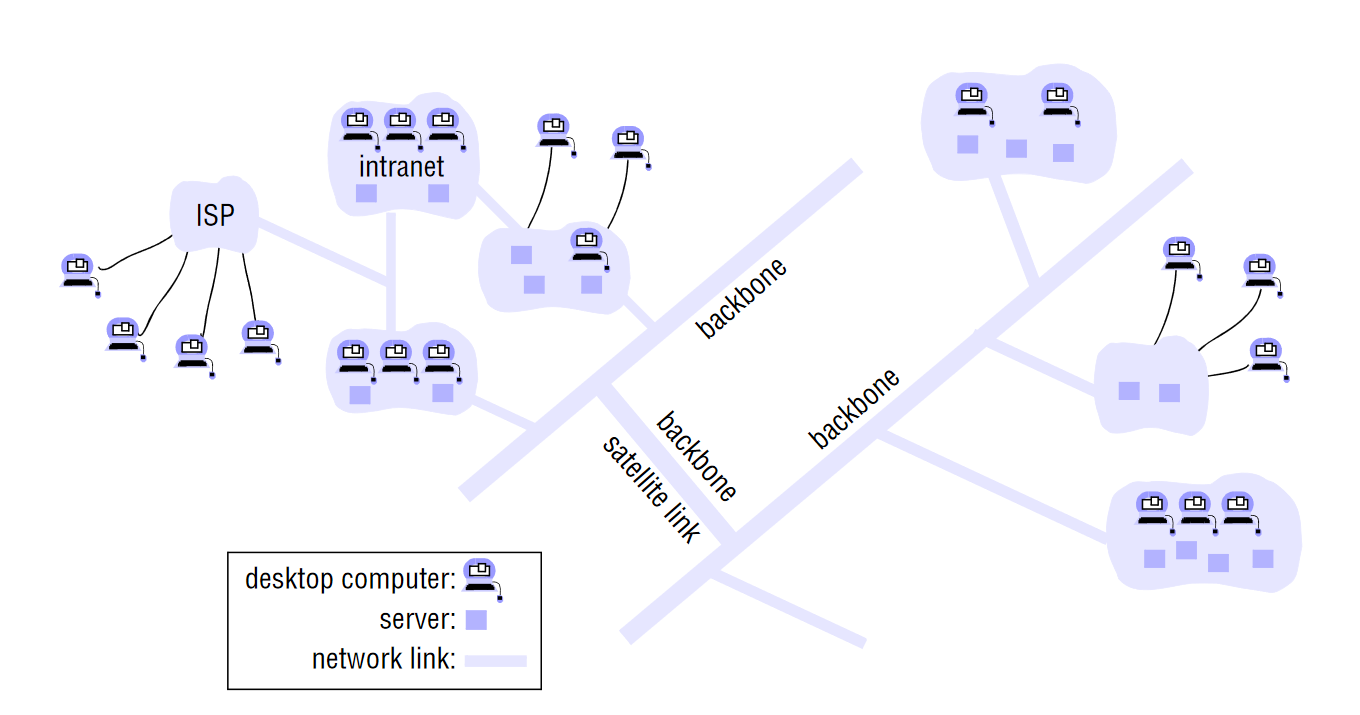
-sự xuất hiện của máy tính phổ biến cùng với mong muốn hỗ trợ người dùng tính di động trong hệ thống phân tán

-nhu cầu ngày càng tăng đối với các dịch vụ đa phương tiện

-xem hệ thống phân tán như là một tiện ích

1.3.1 mạng lan toả và Internet hiện đại

-là một tập hợp các mạng máy tính kết nối với nhau của nhiều loại khác nhau,với số lượng ngày càng tăng với một loạt công nghệ truyền thông không dây như Wifi,Wimax,Blutoolh và các mạng điện thoại di động thế hệ thứ 3



Hình trên minh hoạ một phần của internet, các chương trình chạy trên các máy tính được kết nối với nó tương tác bằng cách truyền thông điệp,thiết kế và xây dựng kết nối internet các cơ chế(giao thức internet) là một thành tựu kĩ thuật quan trọng, cho phép trương trình đang chạy ở bất kỳ đâu để giải quyết các thông báo tới các chương trình ở bất kỳ nơi nào khác và trừu tượng hơn vô số công nghệ được đề cập ở trên. Internet cũng là một hệ thống phân tán rất lớn. nó cho phép người dùng, dù họ ở đâu cũng có thể sử dụng các dịch vụ như World Wide Wed, email, và truyền tệp( thật vậy , đôi khi wed còn bị đánh đồng không chính xác với internet)

-tập hợp các dịch vụ thường theo kiểu mở - đóng nó có thể được mở rộng bằng cách bổ sung các máy chủ và các loại dịch vụ mới.. hình trên cho thấy một tập hợp các mạng nội bộ - mạng con được vận hành bởi các công ty và tổ chức khác nhau thường được bảo vệ bởi tường lửa. vai trò của tường lửa là bảo vệ mạng nội bộ bằng cách ngăn các thông điệp trái phép rời khỏi hoặc xâm nhập

-tường lửa thực hiện bằng cách lọc các bài luận đến và đi, việc lọc có thể được thực hiện bởi nguồn hoặc đích hoặc chỉ cho phép nhưng thông điệp liên quan đến email và quyền truy cập wed vào và ra khỏi mạng nội bộ và nó bảo vệ,nhà cung cấp dịch vụ internet(ISP) là các công ty cung cấp liên kết băng thông và dịch vụ mạng,một khi người dùng cá nhân và tổ chức nhỏ kết nối với nhau, cho phép họ truy cập các dịch vụ ở bất kỳ đâu trên internet cũng như cung cấp các dịch vụ cục bộ như email và lưu trữu wed,các mạng nội bộ liên kết với nhau thành các backbones là một đường mạng với tốc độ cao,cáp quang…..

-tuy nhiên có một vài tổ chức sẽ không muốn kết nối mạng nội bộ của họ với internet ví dụ như cảnh sát,các cơ quan bảo mật và thực thi pháp luật

-việc triển khai internet và các dịch vụ mà nó hỗ trợ kéo theo sự phát triển của các giải pháp thực tế cho nhiều hệ thống phân tán

1.3.2 mạng di động phổ biến

Những tiến bộ công nghệ trong việc thu nhỏ thiết bị và mạng không dây ngày càng tăng dẫn đến việc tích hợp các thiết bị máy tính nhỏ và di dộng và các hệ thống phân tán các thiết bị này bao gồm:

-máy tính cá nhân

-Thiết bị cầm tay, bao gồm điện thoại di động, điện thoại thông minh, thiết bị hỗ trợ GPS, máy nhắn tin, trợ lý kỹ thuật số cá nhân (PDA), máy quay video và máy ảnh kỹ thuật số.

-Thiết bị đeo được, chẳng hạn như đồng hồ thông minh có chức năng tương tự như PDA.

-Các thiết bị nhúng trong các thiết bị như máy giặt, hệ thống hi-fi, ô tô và tủ lạnh.

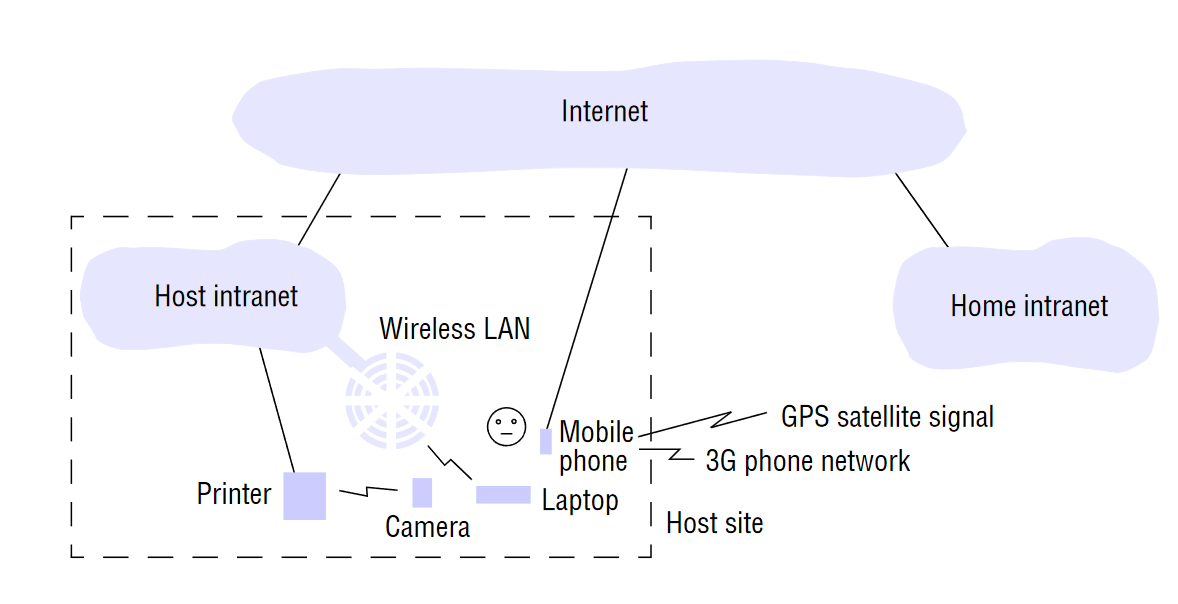
-tính di động của nhiều thiết bị này cùng với khả năng kết nối thuận tiện với các mạng ở những nơi khác nhau ,làm cho mạng di động trở nên khả thi. Điện toán di động là việc thực hiện các tác vụ tính toán trong khi người dùng đang di chuyển hoặc đến thăm các địa điểm khác với môi trường thông thường của họ, trong điện toán di động,người dùng ở xa mạng nội bộ ‘tại nhà’ của họ(nơi làm việc hoặc nơi ở) vẫn được cung cấp quyền truy cập vào các tài nguyên thông qua thiết bị họ mang bên mình,Họ có thể tiếp tục truy cập Internet; họ có thể tiếp tục truy cập tài nguyên trong mạng nội bộ gia đình của họ; và ngày càng có nhiều điều khoản cho phép người dùng sử dụng các tài nguyên như máy in hoặc thậm chí các điểm bán hàng gần đó một cách thuận tiện khi họ di chuyển.

-Tính di động gây ra một số thách thức đối với các hệ thống phân tán, bao gồm nhu cầu đối phó với khả năng kết nối thay đổi và thực sự là ngắt kết nối, và nhu cầu duy trì hoạt động khi đối mặt với tính di động của thiết bị.

- điện toán phổ biến là việc khai thác nhiều thiết bị tính toán nhỏ,rẻ tiền hiện diện trong môi trường vật lý của người sử dụng, bao gồm gia đình,văn phòng hay thậm chí là môi trường tự nhiên, thuật ngữ “phổ biến” nhằm gợi ý rằng các thiết bị nhỏ cuối cùng cũng trở trên phổ biến trong các vật dụng hằng ngày đến mức chúng hiếm khi được chú ý đến. Có nghĩa là, hành vi tính toán của chúng sẽ được gắn chặt với chức năng vật lý của chúng một cách minh bạch và mật thiết.

-sự hiện diện của máy tính ở khắp mọi nơi chỉ hữu dụng khi chúng có thể giao tiếp với nhau ví dụ người dùng có thể sử dụng điện thoại hoặc thiết bị đa năng để điều khiển máy giặt tương tự máy giặt có thể thông báo cho người dùng thông qua biểu tượng thông minh hoặc thông báo khi giặt xong.

-điện toán phổ biến và điện toán di động có thể dan xen lẫn nhau về nguyên tắc máy tính có thể hữu dụng ở nhiều nơi nhưng chúng có khác biết điện toán phổ biến mang lại lợi ích cho người dùng khi họ ở trong một môi trường duy nhất nhà, bệnh viện ương tự, điện toán di động có những lợi thế ngay cả khi nó chỉ liên quan đến các máy tính và thiết bị thông thường, rời rạc như máy tính xách tay và máy in.



ở hình trên một người dùng đang truy cập vào một tổ chức lưu trữ ,cho thấy rằng mạng nội bộ gia đình và mạng nội bộ máy chủ cả 2 được kết nối với phần còn lại của internet.

-với cơ sở hạ tầng hệ thống phù hợp người dùng có thể thực hiện các tác vụ đơn giản bằng các tác vụ mà họ mạng theo Trong khi di chuyển đến trang web lưu trữ, người dùng có thể lấy giá cổ phiếu mới nhất từ ​​máy chủ web bằng điện thoại di động và cũng có thể sử dụng phần mềm tìm đường và GPS tích hợp để chỉ đường đến vị trí trang web. Trong cuộc họp với người dẫn chương trình của họ, người dùng có thể cho họ xem một bức ảnh gần đây bằng cách gửi trực tiếp từ máy ảnh kỹ thuật số đến máy in hoặc máy chiếu được kích hoạt thích hợp (cục bộ) trong phòng họp (được phát hiện bằng cách sử dụng dịch vụ định vị). Điều này chỉ yêu cầu liên kết không dây giữa máy ảnh và máy in hoặc máy chiếu. Và về nguyên tắc, họ có thể gửi tài liệu từ máy tính xách tay của mình đến cùng một máy in, sử dụng các liên kết mạng LAN không dây và Ethernet có dây đến máy in.

1.3.3 hệ thống truyền phát đa phương tiện

-một xu hướng quan trọng khác là yêu cầu hỗ trợ các dịch vụ đa phương tiện trong hệ thống phân tán, một hệ thống đa phương tiện phân tán có thể thực hiện các chức năng tương tự như phát âm thanh và video, nghĩa là nó có thể lưu trữ và định vị các tệp âm thanh hoặc video sau đó truyền chúng qua mạng (có thể trong thời gian thực) để hỗ trợ trình bày các loại phương tiện cho người dùng và tùy chọn cũng để chia sẻ các loại phương tiện trên một nhóm người dùng.

-đặc điểm của các loại phương tiện truyền thông là bao gồm trong một khoảng ,tính toàn vẹn phụ thuộc về cơ bản phụ thuộc vào việc duy trì mối quan hệ giữa thời gian thực giữa các phần tử Ví dụ: trong bản trình bày video, cần phải duy trì thông lượng nhất định về số khung hình trên giây và đối với các luồng thời gian thực, độ trễ hoặc độ trễ tối đa nhất định để phân phối khung hình

-lợi ích của điện toán đa phương tiện là cung cấp các dịch vụ và ứng dụng mới (đa phương tiện) trên máy tính để bàn bao gồm quyền truy cập vào các chương trình truyền hình trực tiếp hoặc được ghi trước, truy cập vào các thư viện phim cung cấp các dịch vụ video theo yêu cầu , quyền truy cập vào các thư viện âm nhạc, cung cấp các thiết bị hội nghị âm thanh và video và các tính năng điện thoại tích hợp bao gồm điện thoại IP hoặc các công nghệ liên quan như Skype,

-Webcasting là một ứng dụng của công nghệ đa phương tiện phân tán. Webcasting là khả năng phát các phương tiện liên tục, thường là âm thanh hoặc video, qua Internet. Hiện nay, các sự kiện thể thao hoặc âm nhạc lớn thường được phát sóng theo cách này, thường thu hút một lượng lớn người xem (ví dụ: buổi hòa nhạc Live8 vào năm 2005 đã thu hút khoảng 170.000 người dùng đồng thời vào lúc đỉnh điểm)

- Các ứng dụng đa phương tiện phân tán như webcasting đặt ra nhu cầu đáng kể đối với cơ sở hạ tầng phân tán cơ bản về mặt:

• cung cấp hỗ trợ cho một loạt các định dạng mã hóa và mã hóa (có thể mở rộng), chẳng hạn như loạt tiêu chuẩn MPEG (bao gồm cả tiêu chuẩn MP3 phổ biến được biết đến như vậy) như MPEG-1, Audio Layer 3) và HDTV;

• cung cấp một loạt các cơ chế để đảm bảo rằng chất lượng dịch vụ mong muốn có thể được đáp ứng;

• cung cấp các chiến lược quản lý tài nguyên liên quan, bao gồm các chính sách lập lịch trình thích hợp để hỗ trợ chất lượng dịch vụ mong muốn;

• đưa ra các chiến lược thích ứng để đối phó với tình huống không thể tránh khỏi trong các hệ thống mở mà chất lượng dịch vụ không thể đáp ứng hoặc duy trì.

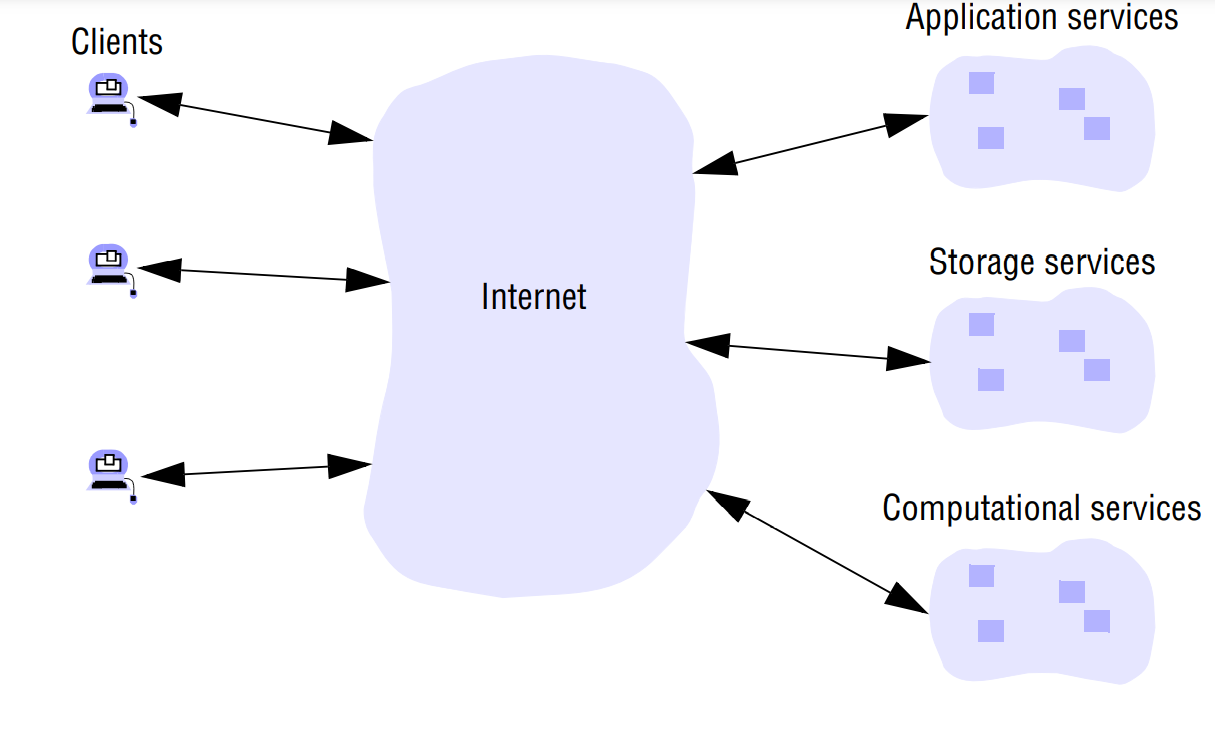
1.3.4 điện toán phân tán như một tiện ích

-Với sự phát triển ngày càng tăng của cơ sở hạ tầng hệ thống phân tán, một số công ty đang thúc đẩy việc coi tài nguyên phân tán như một loại hàng hóa hoặc tiện ích, tạo nên sự tương đồng giữa tài nguyên phân tán và các tiện ích khác như nước hoặc điện. Với mô hình này, các nguồn lực được cung cấp bởi các nhà cung cấp dịch vụ phù hợp và được thuê một cách hiệu quả chứ không phải thuộc sở hữu của người dùng cuối. Mô hình này áp dụng cho cả tài nguyên vật lý và các dịch vụ logic hơn:

+Các tài nguyên vật lý như lưu trữ và xử lý có thể được cung cấp cho các máy tính nối mạng, loại bỏ nhu cầu tự sở hữu các tài nguyên đó. Ở một đầu của phổ, người dùng có thể chọn một phương tiện lưu trữ từ xa cho các yêu cầu lưu trữ tệp (ví dụ: đối với dữ liệu đa phương tiện như ảnh, nhạc hoặc video) và / hoặc để sao lưu. Tương tự, cách tiếp cận này sẽ cho phép người dùng thuê một hoặc nhiều nút tính toán, để đáp ứng nhu cầu tính toán cơ bản của họ hoặc thực sự để thực hiện tính toán phân tán. Ở đầu kia của phổ, người dùng có thể truy cập các trung tâm dữ liệu phức tạp (các cơ sở được nối mạng cung cấp quyền truy cập vào kho lưu trữ thường có khối lượng dữ liệu lớn cho người dùng hoặc tổ chức) hoặc thực sự là cơ sở hạ tầng tính toán bằng cách sử dụng các loại dịch vụ hiện được cung cấp bởi các công ty như Amazon và Google . Ảo hóa hệ điều hành là một công nghệ cho phép quan trọng đối với phương pháp này, ngụ ý rằng người dùng có thể thực sự được cung cấp dịch vụ bởi một nút ảo chứ không phải là một nút vật lý. Điều này mang lại sự linh hoạt hơn cho nhà cung cấp dịch vụ về mặt quản lý tài nguyên

+Các dịch vụ phần mềm cũng có thể được cung cấp trên Internet toàn cầu bằng cách sử dụng phương pháp này. Thật vậy, nhiều công ty hiện cung cấp một loạt các dịch vụ để cho thuê hiệu quả, bao gồm các dịch vụ như email và lịch phân tán. Google, chẳng hạn, gói một loạt các dịch vụ kinh doanh dưới biểu ngữ Google Apps [www.google.com I].

-thuật ngữ điện toán đám mây cũng được sử dụng để nắm bắt tầm nhìn về điện toán như một tiện ích . cloud được định ngĩa là một tập hợp các dịch vụ ứng dụng lưu trữ và điện toán dựa trên Internet đủ để hỗ trợ hầu hết các nhu cầu của người dùng, do đó cho phép họ phân phối phần lớn hoặc toàn bộ phần mềm ứng dụng và lưu trữ dữ liệu cục bộ. Thuật ngữ này cũng thúc đẩy quan điểm về mọi thứ như một dịch vụ, từ cơ sở hạ tầng vật lý hoặc ảo thông qua phần mềm, thường được trả tiền trên cơ sở mỗi lần sử dụng hơn là mua. Lưu ý rằng điện toán đám mây giảm bớt các yêu cầu đối với thiết bị của người dùng, cho phép các thiết bị di động hoặc máy tính để bàn rất đơn giản truy cập vào nhiều loại tài nguyên và dịch vụ tiềm năng.



-Các đám mây thường được thực hiện trên các máy tính cụm để cung cấp quy mô và hiệu suất cần thiết theo yêu cầu của các dịch vụ đó. Máy tính cụm là một tập hợp các máy tính được kết nối với nhau hợp tác chặt chẽ để cung cấp một khả năng tính toán hiệu suất cao, được tích hợp duy nhất. Xây dựng dựa trên các dự án như Dự án NOW (Mạng lưới các máy trạm) tại Berkeley [Anderson et al. 1995, now.cs.berkeley.edu] và Beowulf tại NASA [www.beowulf.org], xu hướng sử dụng phần cứng hàng hóa cho cả máy tính và mạng kết nối.

-Mục tiêu chung của máy tính cụm là cung cấp một loạt các dịch vụ đám mây, bao gồm khả năng tính toán hiệu suất cao, lưu trữ hàng loạt (ví dụ thông qua các trung tâm dữ liệu) và các dịch vụ ứng dụng phong phú hơn như tìm kiếm trên web (Google chẳng hạn dựa vào một cụm lớn kiến trúc máy tính để triển khai công cụ tìm kiếm và các dịch vụ khác của nó,

-Điện toán cũng có thể được xem như một hình thức điện toán đám mây. Các thuật ngữ phần lớn đồng nghĩa và đôi khi không được xác định, nhưng điện toán lưới thường có thể được coi là tiền thân của mô hình chung hơn của điện toán đám mây với sự thiên vị đối với sự hỗ trợ cho các ứng dụng khoa học.

1.4 Tập trung vào chia sẻ tài nguyên

Người dùng đã quá quen với những lợi ích của việc chia sẻ tài nguyên nên họ có thể dễ dàng bỏ qua ý nghĩa của chúng. Chúng tôi thường xuyên chia sẻ tài nguyên phần cứng như máy in, dữ liệu tài nguyên như tệp và tài nguyên có chức năng cụ thể hơn như tìm kiếm động cơ.

Nhìn từ quan điểm cung cấp phần cứng, chúng tôi chia sẻ thiết bị như như máy in và đĩa để giảm chi phí. Nhưng ý nghĩa lớn hơn nhiều đối với người dùng là sự chia sẻ của các tài nguyên cấp cao hơn đóng một phần trong các ứng dụng của họ và hàng ngày của họ công việc và hoạt động xã hội. Ví dụ: người dùng quan tâm đến việc chia sẻ dữ liệu trong dạng cơ sở dữ liệu dùng chung hoặc một tập hợp các trang web - không phải đĩa và bộ xử lý trên đó chúng được thực hiện. Tương tự, người dùng nghĩ về các tài nguyên được chia sẻ, chẳng hạn như công cụ tìm kiếm hoặc công cụ chuyển đổi tiền tệ, không tính đến máy chủ hoặc các máy chủ cung cấp những. Trên thực tế, các mô hình chia sẻ tài nguyên rất khác nhau về phạm vi và cách thức người dùng chặt chẽ làm việc cùng nhau. Ở một khía cạnh nào đó, công cụ tìm kiếm trên Web cung cấp cho người dùng trên toàn thế giới, những người dùng không cần tiếp xúc với một trực tiếp khác. Ở một khía cạnh khác, trong hoạt động hợp tác có sự hỗ trợ của máy tính (CSCW), một nhóm người dùng hợp tác chia sẻ trực tiếp các tài nguyên như tài liệu trong một nhóm nhỏ, khép kín. Mô hình chia sẻ và phân bố địa lý của những người dùng cụ thể xác định những cơ chế nào hệ thống phải cung cấp để điều phối hành động của người dùng. Chúng tôi sử dụng thuật ngữ dịch vụ cho một phần riêng biệt của hệ thống máy tính quản lý tập hợp các tài nguyên liên quan và trình bày chức năng của chúng cho người dùng và ứng dụng. Ví dụ: chúng tôi truy cập các tệp được chia sẻ thông qua một dịch vụ tệp; chúng tôi gửi tài liệu đến máy in thông qua một dịch vụ in ấn; chúng tôi mua hàng hóa thông qua một dịch vụ thanh toán điện tử. Quyền truy cập duy nhất mà chúng tôi có đối với dịch vụ là thông qua tập hợp các hoạt động mà nó xuất ra. Đối với ví dụ, một dịch vụ tệp cung cấp các thao tác đọc, ghi và xóa trên tệp. Thực tế là các dịch vụ hạn chế quyền truy cập tài nguyên vào một nhóm hoạt động được xác định rõ ràng là trong phần thực hành kỹ thuật phần mềm tiêu chuẩn. Nhưng nó cũng phản ánh vật lý tổ chức hệ thống phân tán. Tài nguyên trong một hệ thống phân tán là vật lý được đóng gói trong máy tính và chỉ có thể được truy cập từ các máy tính khác bằng phương tiện truyền thông. Để chia sẻ hiệu quả, mỗi tài nguyên phải được quản lý bởi một chương trình cung cấp giao diện truyền thông cho phép truy cập tài nguyên và được cập nhật một cách đáng tin cậy và nhất quán. Thuật ngữ máy chủ có lẽ đã quen thuộc với hầu hết bạn đọc. Nó đề cập đến một chương trình đang chạy (một quy trình) trên một máy tính nối mạng chấp nhận các yêu cầu từ các chương trình đang chạy trên các máy tính khác để thực hiện một dịch vụ và đáp ứng một cách thích hợp. Yêu cầu các quy trình được gọi là máy khách và cách tiếp cận tổng thể được gọi là máy khách-máy chủ tin học. Trong cách tiếp cận này, các yêu cầu được gửi dưới dạng tin nhắn từ máy khách đến máy chủ và trả lời được gửi trong tin nhắn từ máy chủ đến máy khách. Khi khách hàng gửi một yêu cầu một hoạt động được thực hiện, chúng tôi nói rằng khách hàng yêu cầu một hoạt động trên máy chủ. Một tương tác hoàn chỉnh giữa máy khách và máy chủ, từ điểm khi máy khách gửi yêu cầu đến khi nó nhận được phản hồi của máy chủ, được gọi là lời kêu gọi từ xa. Quá trình tương tự có thể vừa là máy khách vừa là máy chủ, vì máy chủ đôi khi gọi các hoạt động trên các máy chủ khác. Các thuật ngữ "máy khách" và "máy chủ" chỉ áp dụng cho các vai trò chơi trong một yêu cầu duy nhất. Máy khách đang hoạt động (đưa ra yêu cầu) và máy chủ bị động (chỉ thức dậy khi nhận được yêu cầu); máy chủ chạy liên tục, trong khi máy khách chỉ tồn tại miễn là các ứng dụng mà chúng tạo thành một phần. Lưu ý rằng mặc dù theo mặc định, các thuật ngữ 'máy khách' và 'máy chủ' đề cập đến các quy trình thay vì so với các máy tính mà chúng thực thi, theo cách nói hàng ngày, các thuật ngữ đó cũng đề cập đến cho chính các máy tính. Một sự khác biệt khác, mà chúng ta sẽ thảo luận trong Chương 5,

là trong một hệ thống phân tán được viết bằng ngôn ngữ hướng đối tượng, các tài nguyên có thể được đóng gói dưới dạng các đối tượng và được truy cập bởi các đối tượng khách, trong trường hợp đó chúng ta nói về một ứng dụng khách đối tượng gọi một phương thức trên một đối tượng máy chủ. Nhiều, nhưng chắc chắn không phải tất cả, các hệ thống phân tán có thể được xây dựng hoàn toàn trong hình thức tương tác giữa máy khách và máy chủ. World Wide Web, email và mạng máy in đều phù hợp với mô hình này. Chúng tôi thảo luận về các lựa chọn thay thế cho hệ thống máy khách-máy chủ trong Chương 2. Một trình duyệt web đang thực thi là một ví dụ về một ứng dụng khách. Trình duyệt web giao tiếp với một máy chủ web, để yêu cầu các trang web từ nó. Chúng tôi coi Web và kiến trúc máy khách-máy chủ liên quan của nó chi tiết hơn trong Phần 1.6

1.5 Thách thức

Các ví dụ trong Phần 1.2 nhằm minh họa phạm vi của các hệ thống phân tán và đề xuất các vấn đề nảy sinh trong thiết kế của họ. Trong nhiều người trong số họ, đáng kể những thách thức đã gặp phải và vượt qua. Như phạm vi và quy mô của phân phối hệ thống và ứng dụng được mở rộng như nhau và những thách thức khác có thể đã gặp. Trong phần này, chúng tôi mô tả những thách thức chính.

1.5.1 Tính không đồng nhất

-Internet cho phép người dùng truy cập các dịch vụ và chạy các ứng dụng không đồng nhất bộ sưu tập máy tính và mạng. Tính không đồng nhất (nghĩa là đa dạng và khác biệt) áp dụng cho tất cả những điều sau:

• mạng lưới;

• phần cứng máy tính;

• các hệ điều hành;

• ngôn ngữ lập trình;

• triển khai bởi các nhà phát triển khác nhau.

Mặc dù Internet bao gồm nhiều loại mạng khác nhau (được minh họa trong Hình 1.3), sự khác biệt của chúng được che đậy bởi thực tế là tất cả các máy tính gắn liền với chúng sử dụng các giao thức Internet để giao tiếp với nhau. Ví dụ, một máy tính được gắn vào Ethernet có triển khai các giao thức Internet qua Ethernet, trong khi một máy tính trên một loại mạng khác sẽ cần triển khai của các giao thức Internet cho mạng đó. Chương 3 giải thích cách thức các giao thức Internet được triển khai trên nhiều mạng khác nhau.

Các kiểu dữ liệu như số nguyên có thể được biểu diễn theo nhiều cách khác nhau trên các loại khác nhau của phần cứng - ví dụ, có hai lựa chọn thay thế cho thứ tự byte của các số nguyên. Những khác biệt về cách đại diện này phải được xử lý nếu các thông điệp được trao đổi giữa các chương trình chạy trên các phần cứng khác nhau.

Mặc dù hệ điều hành của tất cả các máy tính trên Internet cần phải bao gồm triển khai các giao thức Internet, chúng không nhất thiết phải cung cấp cùng một giao diện lập trình ứng dụng cho các giao thức này. Ví dụ, các cuộc gọi cho trao đổi tin nhắn trong UNIX khác với cuộc gọi trong Windows.

-Các ngôn ngữ lập trình khác nhau sử dụng các biểu diễn khác nhau cho các ký tự và cấu trúc dữ liệu như mảng và bản ghi. Những khác biệt này phải được giải quyết nếu các chương trình được viết bằng các ngôn ngữ khác nhau để có thể giao tiếp với nhau. Các chương trình được viết bởi các nhà phát triển khác nhau không thể giao tiếp với nhau trừ khi họ sử dụng các tiêu chuẩn chung, ví dụ, cho giao tiếp mạng và biểu diễn các mục dữ liệu nguyên thủy và cấu trúc dữ liệu trong thông điệp. Đối với điều này để xảy ra, các tiêu chuẩn cần được đồng ý và thông qua - cũng như các giao thức Internet.

- Phần mềm trung gian • Thuật ngữ phần mềm trung gian áp dụng cho một lớp phần mềm cung cấp lập trình trừu tượng cũng như che giấu sự không đồng nhất của phần bên dưới mạng, phần cứng, hệ điều hành và ngôn ngữ lập trình. Phổ biến Nhà môi giới yêu cầu đối tượng (CORBA), được mô tả trong Chương 4, 5 và 8, là một thí dụ. Một số phần mềm trung gian, chẳng hạn như Java Remote Method Invocation (RMI) (xem Chương 5), chỉ hỗ trợ một ngôn ngữ lập trình duy nhất. Hầu hết phần mềm trung gian là được triển khai qua các giao thức Internet, chính các giao thức này che giấu sự khác biệt của mạng cơ bản, nhưng tất cả phần mềm trung gian đều đề cập đến sự khác biệt trong hệ điều hành và phần cứng - cách thực hiện điều này là chủ đề chính của Chương 4.

-Ngoài việc giải quyết các vấn đề về tính không đồng nhất, phần mềm trung gian cung cấp một mô hình tính toán thống nhất để các lập trình viên của máy chủ sử dụng và được phân phối các ứng dụng. Các mô hình có thể bao gồm gọi đối tượng từ xa, sự kiện từ xa thông báo, truy cập SQL từ xa và xử lý giao dịch phân tán. Ví dụ, CORBA cung cấp lệnh gọi đối tượng từ xa, cho phép một đối tượng trong chương trình chạy trên một máy tính để gọi một phương thức của một đối tượng trong một chương trình đang chạy trên một máy tính khác. Việc triển khai nó che giấu thực tế là các thông điệp được chuyển qua một mạng để gửi yêu cầu gọi và trả lời của nó.

Tính không đồng nhất và mã di động • Thuật ngữ mã di động được sử dụng để chỉ mã chương trình có thể được chuyển từ máy tính này sang máy tính khác và chạy tại đích - Java applet là một ví dụ. Mã phù hợp để chạy trên một máy tính không nhất thiết phải thích hợp để chạy trên một chương trình khác vì các chương trình thực thi thường cụ thể cả hai vào tập lệnh và hệ điều hành máy chủ.

-Phương pháp tiếp cận máy ảo cung cấp một cách làm cho mã có thể thực thi trên một nhiều loại máy tính chủ: trình biên dịch cho một ngôn ngữ cụ thể tạo mã cho một máy ảo thay vì mã đặt hàng phần cứng cụ thể. Ví dụ, Java trình biên dịch tạo ra mã cho một máy ảo Java, máy này thực thi nó bằng cách diễn giải. Máy ảo Java cần được triển khai một lần cho mỗi loại máy tính để cho phép các chương trình Java chạy. Ngày nay, dạng mã di động được sử dụng phổ biến nhất là Javascript bao gồm chương trình trong một số trang web được tải vào trình duyệt của khách hàng. Phần mở rộng này của Web công nghệ sẽ được thảo luận thêm trong Phần 1.6.

1.5.2 Tính cởi mở

Tính mở của hệ thống máy tính là đặc điểm quyết định liệu hệ thống có thể được mở rộng và thực hiện lại theo nhiều cách khác nhau. Tính mở của phân phối hệ thống được xác định chủ yếu bởi mức độ mà các dịch vụ chia sẻ tài nguyên mới có thể được thêm vào và có sẵn để sử dụng bởi nhiều chương trình khách hàng.

Không thể đạt được tính cởi mở trừ khi đặc điểm kỹ thuật và tài liệu của giao diện phần mềm chính của các thành phần của hệ thống được cung cấp cho phần mềm các nhà phát triển. Trong một từ, các giao diện chính được xuất bản. Quá trình này tương tự như tiêu chuẩn hóa các giao diện, nhưng nó thường bỏ qua các thủ tục tiêu chuẩn hóa chính thức, thường cồng kềnh và di chuyển chậm.

Tuy nhiên, việc xuất bản các giao diện chỉ là điểm khởi đầu cho việc thêm và mở rộng các dịch vụ trong một hệ thống phân tán. Thách thức đối với các nhà thiết kế là giải quyết sự phức tạp của hệ thống phân tán bao gồm nhiều thành phần được thiết kế bởi người khác.

Các nhà thiết kế giao thức Internet đã giới thiệu một loạt các tài liệu được gọi là 'Yêu cầu nhận xét' hoặc RFC, mỗi yêu cầu trong số đó được biết đến bằng một con số. Các đặc điểm kỹ thuật của các giao thức truyền thông Internet đã được xuất bản trong loạt bài này trong đầu những năm 1980, tiếp theo là các thông số kỹ thuật cho các ứng dụng chạy trên chúng, chẳng hạn như truyền tệp, email và mạng viễn thông vào giữa những năm 1980. Thực hành này đã tiếp tục và hình thành cơ sở của tài liệu kỹ thuật của Internet. Loạt bài này bao gồm các cuộc thảo luận cũng như các thông số kỹ thuật của các giao thức. Có thể lấy các bản sao từ [www.ietf.org]. Do đó, việc xuất bản các giao thức truyền thông Internet ban đầu đã cho phép nhiều hệ thống và ứng dụng Internet bao gồm cả Web sẽ được xây dựng. RFC không phương tiện xuất bản duy nhất. Ví dụ: World Wide Web Consortium (W3C) phát triển và xuất bản các tiêu chuẩn liên quan đến hoạt động của Web [www.w3.org].

Các hệ thống được thiết kế để hỗ trợ chia sẻ tài nguyên theo cách này được gọi là mở hệ thống phân tán để nhấn mạnh thực tế là chúng có thể mở rộng. Chúng có thể được gia hạn ở cấp độ phần cứng bằng cách bổ sung máy tính vào mạng và ở phần mềm bằng cách giới thiệu các dịch vụ mới và thực hiện lại các dịch vụ cũ, cho phép các chương trình ứng dụng để chia sẻ tài nguyên. Một lợi ích khác thường được trích dẫn để mở hệ thống là sự độc lập của họ với các nhà cung cấp riêng lẻ.

Tóm lại:

• Hệ thống mở được đặc trưng bởi thực tế là các giao diện chính của chúng được xuất bản.

• Hệ thống phân tán mở dựa trên việc cung cấp một giao tiếp thống nhất cơ chế và các giao diện được xuất bản để truy cập vào các tài nguyên được chia sẻ.

• Hệ thống phân tán mở có thể được xây dựng từ phần cứng không đồng nhất và phần mềm, có thể từ các nhà cung cấp khác nhau. Nhưng sự phù hợp của mỗi thành phần của tiêu chuẩn đã công bố phải được kiểm tra và xác nhận cẩn thận nếu hệ thống hoạt động chính xác

1.5.3 Bảo mật

Nhiều tài nguyên thông tin được cung cấp và duy trì trong hệ thống có giá trị nội tại cao đối với người dùng của họ. Do đó, an ninh của họ là tầm quan trọng đáng kể. Bảo mật cho tài nguyên thông tin có ba thành phần: tính bảo mật (bảo vệ chống tiết lộ cho các cá nhân trái phép), tính toàn vẹn (bảo vệ chống lại sự thay đổi hoặc tham nhũng) và tính khả dụng (bảo vệ chống lại can thiệp vào các phương tiện để truy cập các tài nguyên).

Phần 1.1 đã chỉ ra rằng mặc dù Internet cho phép một chương trình trong một máy tính để giao tiếp với một chương trình trong một máy tính khác bất kể

vị trí, rủi ro bảo mật liên quan đến việc cho phép truy cập miễn phí vào tất cả các tài nguyên trong một mạng nội bộ. Mặc dù tường lửa có thể được sử dụng để tạo rào cản xung quanh mạng nội bộ, hạn chế lưu lượng có thể ra vào, điều này không giải quyết được việc đảm bảo việc sử dụng tài nguyên thích hợp của người dùng trong mạng nội bộ hoặc với việc sử dụng thích hợp tài nguyên trên Internet, không được bảo vệ bởi tường lửa.

Trong hệ thống phân tán, máy khách gửi yêu cầu truy cập dữ liệu do máy chủ quản lý, liên quan đến việc gửi thông tin trong tin nhắn qua mạng. Ví dụ:

1. Một bác sĩ có thể yêu cầu quyền truy cập vào dữ liệu bệnh nhân của bệnh viện hoặc gửi bổ sung cho dữ liệu đó.

2. Trong thương mại điện tử và ngân hàng, người dùng gửi số thẻ tín dụng của họ qua Internet.

Trong cả hai ví dụ, thách thức là gửi thông tin nhạy cảm trong một tin nhắn qua mạng một cách an toàn. Nhưng bảo mật không chỉ là vấn đề che giấu nội dung trong số các thông báo - nó cũng liên quan đến việc biết chắc chắn danh tính của người dùng hoặc tác nhân khác trên tin nhắn đã được gửi thay mặt cho ai. Trong ví dụ đầu tiên, máy chủ cần biết rằng người dùng thực sự là một bác sĩ và trong ví dụ thứ hai, người dùng cần chắc chắn về danh tính của cửa hàng hoặc ngân hàng mà họ đang giao dịch. Thách thức thứ hai ở đây là xác định một người dùng từ xa hoặc đại lý khác một cách chính xác. Cả hai thách thức này đều có thể được đáp ứng bằng cách sử dụng kỹ thuật mã hóa được phát triển cho mục đích này. Chúng được sử dụng rộng rãi trên Internet và được thảo luận trong Chương 11.

Tuy nhiên, hai thách thức an ninh sau vẫn chưa được đáp ứng đầy đủ:

Tấn công từ chối dịch vụ: Một vấn đề bảo mật khác là người dùng có thể muốn làm gián đoạn dịch vụ vì một số lý do. Điều này có thể đạt được bằng cách bắn phá dịch vụ với một số lượng lớn các yêu cầu vô nghĩa đến mức người dùng nghiêm túc không thể sử dụng nó. Đây được gọi là tấn công từ chối dịch vụ. Đã có một số từ chối dịch vụ tấn công vào các dịch vụ web nổi tiếng. Hiện tại các cuộc tấn công như vậy được chống lại bởi cố gắng bắt và trừng phạt thủ phạm sau sự kiện, nhưng đó không phải là một giải pháp chung cho vấn đề. Các biện pháp đối phó dựa trên những cải tiến trong quản lý các mạng đang được phát triển và những điều này sẽ được đề cập trong Chương 3.

Bảo mật của mã di động: Mã di động cần được xử lý cẩn thận. Xem xét người nhận được một chương trình thực thi dưới dạng phần đính kèm thư điện tử: những tác động có thể xảy ra khi chạy chương trình là không thể đoán trước; ví dụ, nó có vẻ như để hiển thị một bức tranh thú vị nhưng trong thực tế, nó có thể truy cập các tài nguyên địa phương hoặc có thể là một phần của cuộc tấn công từ chối dịch vụ. Một số biện pháp để bảo mật mã di động là được nêu trong Chương 11.

1.5.4 Khả năng mở rộng

Hệ thống phân tán hoạt động hiệu quả và hiệu quả ở nhiều quy mô khác nhau, từ một mạng nội bộ nhỏ đến Internet. Một hệ thống được mô tả là có thể mở rộng nếu nó vẫn hiệu quả khi có sự gia tăng đáng kể về số lượng tài nguyên và số lượng của người dùng. Số lượng máy tính và máy chủ trên Internet đã tăng lên đáng kinh ngạc. Hình 1.6 cho thấy số lượng máy tính và máy chủ web ngày càng tăng trong lịch sử 12 năm của Web tính đến năm 2005 [zakon.org]. Thật thú vị khi lưu ý tăng trưởng đáng kể trong cả máy tính và máy chủ web trong giai đoạn này, nhưng cũng

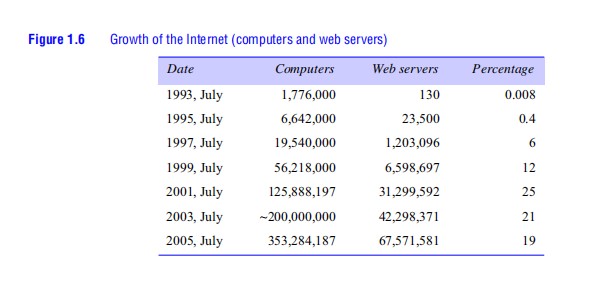
tỷ lệ phần trăm tương đối đang giảm dần - một xu hướng được giải thích bởi sự tăng trưởng của máy tính cá nhân di động. Một máy chủ web cũng có thể ngày càng được lưu trữ trên nhiều máy tính.

Việc thiết kế các hệ thống phân tán có thể mở rộng đưa ra những thách thức sau:

Kiểm soát chi phí tài nguyên vật chất: Khi nhu cầu về tài nguyên ngày càng tăng, nó nên có thể mở rộng hệ thống, với chi phí hợp lý, để đáp ứng nó. Ví dụ, tần suất mà các tệp được truy cập trong mạng nội bộ có thể sẽ tăng lên khi số lượng người dùng và máy tính tăng lên. Phải có thể thêm máy chủ để tránh tắc nghẽn hiệu suất sẽ phát sinh nếu một máy chủ tệp phải xử lý tất cả các yêu cầu truy cập tệp. Nói chung, để một hệ thống có n người dùng có thể mở rộng, số lượng tài nguyên vật chất cần thiết để hỗ trợ chúng tối đa phải là O (n) - tức là là, tỷ lệ với n. Ví dụ: nếu một máy chủ tệp duy nhất có thể hỗ trợ 20 người dùng, thì hai máy chủ như vậy sẽ có thể hỗ trợ 40 người dùng. Mặc dù điều đó nghe có vẻ hiển nhiên nó không nhất thiết phải dễ dàng đạt được trong thực tế, như chúng tôi đã trình bày trong Chương 12.

Kiểm soát việc mất hiệu suất: Xem xét việc quản lý một tập hợp dữ liệu có kích thước tỷ lệ thuận với số lượng người dùng hoặc tài nguyên trong hệ thống - ví dụ: bảng có sự tương ứng giữa tên miền của máy tính và Địa chỉ Internet do Hệ thống tên miền nắm giữ, được sử dụng chủ yếu để tìm kiếm lên tên DNS chẳng hạn như www.amazon.com. Các thuật toán sử dụng cấu trúc phân cấp mở rộng quy mô tốt hơn so với những loại sử dụng cấu trúc tuyến tính. Nhưng ngay cả với cấu trúc phân cấp sự gia tăng về kích thước sẽ dẫn đến một số tổn thất về hiệu suất: thời gian cần thiết để truy cập Dữ liệu có cấu trúc phân cấp là O (log n), trong đó n là kích thước của tập dữ liệu. Cho một hệ thống có thể mở rộng, sự mất hiệu suất tối đa sẽ không tồi tệ hơn thế này.

Ngăn chặn việc cạn kiệt tài nguyên phần mềm: Một ví dụ về việc thiếu khả năng mở rộng là được hiển thị bằng các số được sử dụng làm địa chỉ Internet (IP) (địa chỉ máy tính trong Internet). Vào cuối những năm 1970, người ta đã quyết định sử dụng 32 bit cho mục đích này, nhưng theo ý muốn được giải thích trong Chương 3, nguồn cung cấp các địa chỉ Internet khả dụng đang cạn kiệt. Vì lý do này, một phiên bản mới của giao thức với địa chỉ Internet 128 bit đang được được thông qua, và điều này sẽ yêu cầu sửa đổi nhiều thành phần phần mềm. Để có sự công bằng



đối với các nhà thiết kế đầu tiên của Internet, không có giải pháp chính xác cho vấn đề này. Nó là khó dự đoán nhu cầu sẽ được đặt trên một hệ thống nhiều năm tới. Hơn thế nữa, bù đắp quá mức cho sự tăng trưởng trong tương lai có thể tồi tệ hơn việc thích ứng với sự thay đổi khi chúng tôi buộc phải làm như vậy - các địa chỉ Internet lớn hơn sẽ chiếm thêm dung lượng trong thư và trong bộ nhớ máy tính.

Tránh tắc nghẽn hiệu suất: Nói chung, các thuật toán nên được phân cấp để tránh bị tắc nghẽn hiệu suất. Chúng tôi minh họa điểm này bằng cách tham khảo tiền thân của Hệ thống tên miền, trong đó bảng tên được giữ trong tệp chính duy nhất có thể được tải xuống bất kỳ máy tính nào cần nó. Đó là tốt khi chỉ có vài trăm máy tính trên Internet, nhưng nó sẽ sớm thôi trở thành một hiệu suất nghiêm trọng và tắc nghẽn hành chính. Tên miền Hệ thống đã gỡ bỏ nút thắt cổ chai này bằng cách phân vùng bảng tên giữa các máy chủ được đặt trên toàn mạng Internet và được quản lý cục bộ - xem Chương 3 và 13.

Một số tài nguyên được chia sẻ được truy cập rất thường xuyên; ví dụ, nhiều người dùng có thể truy cập vào cùng một trang web, làm giảm hiệu suất. Chúng ta sẽ thấy trong Chương 2 rằng bộ nhớ đệm và sao chép có thể được sử dụng để cải thiện hiệu suất của tài nguyên được sử dụng rất nhiều.

Tốt nhất, hệ thống và phần mềm ứng dụng không cần thay đổi khi thang đo của hệ thống tăng lên, nhưng điều này khó đạt được. Vấn đề về quy mô là một yếu tố chi phối chủ đề trong sự phát triển của các hệ thống phân tán. Các kỹ thuật đã được thành công được thảo luận nhiều trong cuốn sách này. Chúng bao gồm việc sử dụng dữ liệu sao chép (Chương 18), kỹ thuật liên quan của bộ nhớ đệm (Chương 2 và 12) và triển khai nhiều máy chủ để xử lý các tác vụ thường được thực hiện, cho phép một số các nhiệm vụ tương tự được thực hiện đồng thời.

1.5.5 Xử lý lỗi Hệ thống máy tính đôi khi bị lỗi.

Khi lỗi xảy ra trong phần cứng hoặc phần mềm, chương trình có thể tạo ra kết quả không chính xác hoặc có thể dừng lại trước khi chúng hoàn thành dự định tính toán. Chúng ta sẽ thảo luận và phân loại một loạt các loại lỗi có thể xảy ra xảy ra trong các quá trình và mạng bao gồm một hệ thống phân tán trong Chương 2.

Các lỗi trong hệ thống phân tán là một phần - nghĩa là, một số thành phần bị lỗi trong khi những người khác tiếp tục hoạt động. Do đó việc xử lý các hỏng hóc đặc biệt khó khăn. Các kỹ thuật sau để đối phó với thất bại được thảo luận trong suốt cuốn sách:

Phát hiện hư hỏng: Có thể phát hiện một số hỏng hóc. Ví dụ, tổng kiểm tra có thể là được sử dụng để phát hiện dữ liệu bị hỏng trong một tin nhắn hoặc một tệp. Chương 2 giải thích rằng nó là khó hoặc thậm chí không thể phát hiện một số hỏng hóc khác, chẳng hạn như điều khiển từ xa bị rơi máy chủ trên Internet. Thách thức là quản lý khi có những thất bại không thể bị phát hiện nhưng có thể bị nghi ngờ.

Các lỗi tạo mặt nạ: Một số lỗi đã được phát hiện có thể được ẩn hoặc làm ít hơn dữ dội. Hai ví dụ về việc che giấu thất bại:

1. Tin nhắn có thể được truyền lại khi chúng không đến được.

2. Dữ liệu tệp có thể được ghi vào một cặp đĩa để nếu một đĩa bị hỏng, đĩa kia có thể vẫn đúng.

Chỉ đánh rơi một tin nhắn bị hỏng là một ví dụ của việc làm cho lỗi trở nên ít nghiêm trọng hơn - nó có thể được truyền lại. Người đọc có thể sẽ nhận ra rằng các kỹ thuật được mô tả để che giấu các lỗi không đảm bảo hoạt động trong các trường hợp xấu nhất; cho ví dụ, dữ liệu trên đĩa thứ hai cũng có thể bị hỏng hoặc thông báo có thể không vượt qua trong một thời gian hợp lý tuy nhiên nó thường được truyền lại.

Chịu đựng lỗi: Hầu hết các dịch vụ trên Internet đều có lỗi - nó sẽ không thực tế để họ cố gắng phát hiện và che giấu tất cả các lỗi có thể xảy ra trong một mạng lớn với nhiều thành phần như vậy. Khách hàng của họ có thể là được thiết kế để chịu đựng các lỗi, thường liên quan đến việc người dùng chấp nhận chúng như tốt. Ví dụ: khi một trình duyệt web không thể kết nối với một máy chủ web, nó sẽ không làm cho người dùng chờ đợi mãi mãi trong khi nó tiếp tục cố gắng - nó thông báo cho người dùng về sự cố, để họ tự do để thử lại sau. Các dịch vụ chịu được lỗi được thảo luận trong đoạn trên dư thừa dưới đây.

Khôi phục sau lỗi: Phục hồi liên quan đến việc thiết kế phần mềm để trạng thái dữ liệu vĩnh viễn có thể được phục hồi hoặc 'khôi phục lại' sau khi máy chủ gặp sự cố. Trong nói chung, các tính toán được thực hiện bởi một số chương trình sẽ không hoàn chỉnh khi xảy ra lỗi và dữ liệu vĩnh viễn mà họ cập nhật (tệp và tài liệu khác được lưu trữ trong kho lưu trữ vĩnh viễn) có thể không ở trạng thái nhất quán. Phục hồi được mô tả trong Chương 17.

Dự phòng: Các dịch vụ có thể được thực hiện để chịu đựng các lỗi do sử dụng dự phòng các thành phần. Hãy xem xét các ví dụ sau:

1. Luôn phải có ít nhất hai tuyến đường khác nhau giữa hai bộ định tuyến bất kỳ trong Internet.

2. Trong Hệ thống tên miền, mọi bảng tên được sao chép trong ít nhất hai các máy chủ khác nhau.

3. Một cơ sở dữ liệu có thể được sao chép trong một số máy chủ để đảm bảo rằng dữ liệu vẫn còn có thể truy cập sau sự cố của bất kỳ máy chủ đơn lẻ nào; máy chủ có thể được thiết kế để phát hiện lỗi ở bạn bè của họ; khi một lỗi được phát hiện trong một máy chủ, các máy khách sẽ được chuyển hướng đến các máy chủ còn lại.

Việc thiết kế các kỹ thuật hiệu quả để giữ cho các bản sao của dữ liệu thay đổi nhanh chóng được cập nhật mà không làm giảm hiệu suất quá mức là một thách thức. Phương pháp tiếp cận là được thảo luận trong Chương 18.

Hệ thống phân tán cung cấp mức độ sẵn sàng cao khi đối mặt với các lỗi phần cứng. Tính khả dụng của một hệ thống là thước đo tỷ lệ thời gian mà nó có sẵn cho sử dụng. Khi một trong các thành phần trong hệ thống phân tán bị lỗi, chỉ có công việc sử dụng thành phần không thành công bị ảnh hưởng. Người dùng có thể chuyển sang máy tính khác nếu máy tính mà họ đang sử dụng không thành công; một quá trình máy chủ có thể được bắt đầu trên một máy tính khác.

1.5.6 Đồng tiền

Cả dịch vụ và ứng dụng đều cung cấp tài nguyên có thể được chia sẻ bởi khách hàng trong một hệ thống phân phối. Do đó, có khả năng một số khách hàng sẽ cố gắng

truy cập tài nguyên được chia sẻ cùng một lúc. Ví dụ: cấu trúc dữ liệu ghi lại giá thầu cho một cuộc đấu giá có thể được truy cập rất thường xuyên khi nó gần đến thời hạn cuối cùng.

Quá trình quản lý tài nguyên được chia sẻ có thể thực hiện một yêu cầu của khách hàng tại một thời điểm. Nhưng cách tiếp cận đó giới hạn thông lượng. Do đó, các dịch vụ và ứng dụng thường cho phép nhiều yêu cầu của khách hàng sẽ được xử lý đồng thời. Để làm cho điều này cụ thể hơn, giả sử rằng mỗi tài nguyên được đóng gói dưới dạng một đối tượng và các lệnh gọi được thực thi trong các chủ đề đồng thời. Trong trường hợp này, có thể một số luồng có thể đang thực thi đồng thời trong một đối tượng, trong trường hợp đó các hoạt động của chúng trên đối tượng có thể xung đột với nhau và tạo ra kết quả không nhất quán. Ví dụ: nếu hai giá thầu đồng thời trong một cuộc đấu giá là "Smith: 122 đô la" và "Jones: 111 đô la" và các hoạt động tương ứng là xen kẽ mà không có bất kỳ kiểm soát nào, sau đó chúng có thể được lưu trữ dưới dạng ‘Smith: $ 111’ và ‘Jones: $ 122 '.

Đạo lý của câu chuyện này là bất kỳ đối tượng nào đại diện cho một tài nguyên được chia sẻ trong một hệ thống phân tán phải chịu trách nhiệm đảm bảo rằng nó hoạt động chính xác trong một môi trường đồng thời. Điều này không chỉ áp dụng cho các máy chủ mà còn cho các đối tượng trong các ứng dụng. Do đó, bất kỳ lập trình viên nào thực hiện một đối tượng không nhằm mục đích sử dụng trong một hệ thống phân tán phải làm bất cứ điều gì cần thiết để thực hiện nó an toàn trong một môi trường đồng thời.

Để một đối tượng được an toàn trong một môi trường đồng thời, các hoạt động của nó phải được đồng bộ hóa theo cách mà dữ liệu của nó vẫn nhất quán. Điều này có thể đạt được bằng các kỹ thuật tiêu chuẩn như semaphores, được sử dụng trong hầu hết các hệ điều hành. Điều này chủ đề và phần mở rộng của nó đối với các bộ sưu tập các đối tượng được chia sẻ phân tán được thảo luận trong Chương 7 và 17.

1.5.7 Tính minh bạch

Tính minh bạch được định nghĩa là sự che giấu người dùng và ứng dụng lập trình viên của việc tách các thành phần trong một hệ thống phân tán, để hệ thống được nhìn nhận như một tổng thể hơn là một tập hợp các thành phần độc lập. Các tác động của tính minh bạch là một ảnh hưởng lớn đến thiết kế của phần mềm hệ thống.

Hướng dẫn Tham khảo ANSA [ANSA 1989] và Tổ chức Quốc tế cho Mô hình Tham chiếu của Tiêu chuẩn hóa cho Xử lý Phân tán Mở (RM-ODP) [ISO 1992] xác định tám hình thức minh bạch. Chúng tôi đã diễn giải bản gốc Định nghĩa ANSA, thay thế tính minh bạch về di chuyển của chúng bằng tính di động của chính chúng ta minh bạch, có phạm vi rộng hơn:

Tính minh bạch của quyền truy cập cho phép truy cập tài nguyên cục bộ và từ xa bằng cách sử dụng hoạt động giống hệt nhau.

Tính minh bạch về vị trí cho phép các tài nguyên có thể được truy cập mà không cần biết về vị trí vật lý hoặc mạng (ví dụ: tòa nhà hoặc địa chỉ IP).

Tính minh bạch đồng thời cho phép một số quy trình hoạt động đồng thời bằng cách sử dụng tài nguyên được chia sẻ mà không có sự can thiệp giữa chúng.

Tính minh bạch của nhân bản cho phép nhiều trường hợp tài nguyên được sử dụng để tăng độ tin cậy và hiệu suất mà người dùng không biết về các bản sao hoặc lập trình viên ứng dụng.

Tính minh bạch của lỗi không cho phép che giấu lỗi, cho phép người dùng và các chương trình ứng dụng để hoàn thành nhiệm vụ của chúng bất chấp lỗi phần cứng hoặc các thành phần phần mềm.

Tính minh bạch về tính di động cho phép sự di chuyển của các nguồn lực và khách hàng trong một hệ thống mà không ảnh hưởng đến hoạt động của người dùng hoặc chương trình.

Tính minh bạch về hiệu suất cho phép cấu hình lại hệ thống để cải thiện hiệu suất khi tải khác nhau. Tính minh bạch về quy mô cho phép hệ thống và ứng dụng mở rộng quy mô mà không thay đổi cấu trúc hệ thống hoặc các thuật toán ứng dụng.

Hai yếu tố trong suốt quan trọng nhất là tính minh bạch về truy cập và vị trí; của chúng sự hiện diện hay vắng mặt ảnh hưởng mạnh mẽ nhất đến việc sử dụng các nguồn tài nguyên phân tán. Họ đôi khi được gọi chung là tính minh bạch của mạng.

Như một minh họa về tính minh bạch của quyền truy cập, hãy xem xét giao diện người dùng đồ họa với các thư mục này giống nhau cho dù các tệp bên trong thư mục là cục bộ hay từ xa. Khác ví dụ là một API cho các tệp sử dụng các hoạt động giống nhau để truy cập cả cục bộ và từ xa các tệp (xem Chương 12). Như một ví dụ về sự thiếu minh bạch trong truy cập, hãy xem xét một hệ thống phân tán không cho phép bạn truy cập tệp trên máy tính từ xa trừ khi bạn sử dụng chương trình ftp để làm như vậy.

Tên hoặc URL tài nguyên web rõ ràng về vị trí vì một phần của URL xác định tên miền máy chủ web đề cập đến tên máy tính trong miền, thay vì đến một địa chỉ Internet. Tuy nhiên, các URL không minh bạch về tính di động, bởi vì trang web cá nhân của ai đó không thể chuyển đến nơi làm việc mới của họ ở một nơi khác tên miền - tất cả các liên kết trong các trang khác sẽ vẫn trỏ đến trang gốc.

Nói chung, các số nhận dạng như URL bao gồm tên miền của máy tính ngăn chặn sự minh bạch sao chép. Mặc dù DNS cho phép tên miền tham chiếu đến một số máy tính, nó chỉ chọn một trong số chúng khi tìm kiếm tên. Kể từ một bản sao chương trình thường cần có khả năng truy cập tất cả các máy tính tham gia, nó sẽ cần truy cập từng mục DNS theo tên.

Như một minh họa về sự hiện diện của tính minh bạch của mạng, hãy xem xét việc sử dụng địa chỉ thư điện tử như Fred.Flintstone@stoneit.com. Địa chỉ bao gồm một tên người dùng và tên miền. Gửi thư cho một người dùng như vậy không liên quan đến việc biết vị trí thực tế hoặc mạng của họ. Cũng không phải thủ tục để gửi một tin nhắn email phụ thuộc vào vị trí của người nhận. Do đó, thư điện tử trong Internet cung cấp cả tính minh bạch về vị trí và quyền truy cập (nghĩa là tính minh bạch của mạng).

Sự không minh bạch cũng có thể được minh họa trong bối cảnh của thư điện tử, cuối cùng được phân phối, ngay cả khi máy chủ hoặc liên kết giao tiếp bị lỗi. Các lỗi là che giấu bằng cách cố gắng truyền lại tin nhắn cho đến khi chúng được gửi thành công, thậm chí nếu nó mất vài ngày. Phần mềm trung gian thường chuyển đổi các lỗi của mạng và các quy trình thành các ngoại lệ ở cấp độ lập trình (xem Chương 5 để biết giải thích).

Để minh họa tính minh bạch về tính di động, hãy xem xét trường hợp của điện thoại di động. Giả sử rằng cả người gọi và người gọi đang đi bằng tàu hỏa ở các vùng khác nhau của một quốc gia, di chuyển

từ môi trường (ô) này sang môi trường khác. Chúng tôi coi điện thoại của người gọi là khách hàng và điện thoại của callee như một tài nguyên. Hai người sử dụng điện thoại thực hiện cuộc gọi không biết về tính di động của điện thoại (máy khách và tài nguyên) giữa các ô.

Tính minh bạch ẩn và ẩn danh các tài nguyên không trực tiếp sự phù hợp với nhiệm vụ đang giao cho người dùng và người lập trình ứng dụng. Ví dụ, nó là thường mong muốn các tài nguyên phần cứng tương tự được phân bổ thay thế cho nhau thực hiện một tác vụ - danh tính của một bộ xử lý được sử dụng để thực hiện một quá trình thường bị ẩn từ người dùng và vẫn ẩn danh. Như đã chỉ ra trong Phần 1.3.2, điều này có thể không luôn luôn là những gì được yêu cầu: ví dụ: một khách du lịch gắn một máy tính xách tay vào mạng cục bộ tại mỗi văn phòng được ghé thăm nên sử dụng các dịch vụ cục bộ như gửi dịch vụ thư, sử dụng các máy chủ khác nhau tại mỗi địa điểm. Ngay cả trong một tòa nhà, nó là bình thường để sắp xếp tài liệu được in tại một máy in cụ thể, được đặt tên: thường là một gần với người dùng.

1.5.8 Chất lượng dịch vụ

Sau khi người dùng được cung cấp chức năng mà họ yêu cầu đối với một dịch vụ, chẳng hạn như dịch vụ tập tin trong một hệ thống phân tán, chúng ta có thể tiếp tục hỏi về chất lượng của dịch vụ được cung cấp. Các thuộc tính phi chức năng chính của hệ thống ảnh hưởng đến chất lượng của dịch vụ được khách hàng và người dùng trải nghiệm là độ tin cậy, bảo mật và hiệu suất. Khả năng thích ứng để đáp ứng các cấu hình hệ thống đang thay đổi và khả năng cung cấp tài nguyên đã được được công nhận là một khía cạnh quan trọng hơn nữa của chất lượng dịch vụ.

Các vấn đề về độ tin cậy và bảo mật là rất quan trọng trong thiết kế của hầu hết các hệ thống máy tính. Khía cạnh hiệu suất của chất lượng dịch vụ ban đầu được định nghĩa theo khả năng đáp ứng và thông lượng tính toán, nhưng nó đã được định nghĩa lại về khả năng đáp ứng các đảm bảo về thời gian, như được thảo luận trong các đoạn sau.

Một số ứng dụng, bao gồm các ứng dụng đa phương tiện, xử lý dữ liệu quan trọng về thời gian - các luồng dữ liệu được yêu cầu xử lý hoặc chuyển từ một quy trình sang khác với tỷ lệ cố định. Ví dụ: một dịch vụ phim có thể bao gồm một chương trình khách hàng đó là truy xuất phim từ máy chủ video và trình chiếu phim trên màn hình của người dùng. Cho một kết quả hài lòng, các khung hình liên tiếp của video cần được hiển thị cho người dùng trong một số giới hạn thời gian quy định.

Trên thực tế, chữ viết tắt QoS đã được dùng để chỉ khả năng của các hệ thống để đáp ứng các thời hạn đó. Thành tích của nó phụ thuộc vào tính khả dụng tài nguyên mạng và máy tính cần thiết vào những thời điểm thích hợp. Điều này nghĩa là một yêu cầu đối với hệ thống để cung cấp tính toán và giao tiếp được đảm bảo tài nguyên đủ để cho phép các ứng dụng hoàn thành từng tác vụ đúng hạn (đối với ví dụ: nhiệm vụ hiển thị một khung hình của video).

Các mạng thường được sử dụng ngày nay có hiệu suất cao - ví dụ: BBC iPlayer thường hoạt động ở mức chấp nhận được - nhưng khi các mạng được tải nhiều hiệu suất có thể xấu đi và không có đảm bảo nào được cung cấp. QoS áp dụng cho hoạt động hệ thống cũng như mạng. Mỗi tài nguyên quan trọng phải được các ứng dụng dành riêng yêu cầu QoS và phải có người quản lý tài nguyên cung cấp các bảo đảm. Yêu cầu đặt chỗ trước không được đáp ứng sẽ bị từ chối. Những vấn đề này sẽ được giải quyết tiếp tục trong Chương 20.

1.6 Case study: The World Wide Web

World Wide Web [www.w3.org I, Berners-Lee 1991] là một hệ thống đang phát triển để xuất bản và truy cập các tài nguyên và dịch vụ trên Internet. Thông qua các trình duyệt web phổ biến có sẵn, người dùng truy xuất và xem các tài liệu thuộc nhiều loại, nghe các luồng âm thanh và xem các luồng video cũng như tương tác với một bộ dịch vụ không giới hạn.

Web bắt đầu ra đời tại trung tâm nghiên cứu hạt nhân Châu Âu (CERN), Thụy Sĩ, vào năm 1989 như một phương tiện trao đổi tài liệu giữa một cộng đồng các nhà vật lý được kết nối bằng Internet [Berners-Lee 1999]. Đặc điểm chính của Web là nó cung cấp một cấu trúc siêu văn bản (hypertext) trong số các tài liệu mà nó lưu trữ, phản ánh yêu cầu của người dùng trong việc tổ chức kiến thức của họ. Điều này có nghĩa là các tài liệu chứa các liên kết-links (hoặc siêu liên kết- hyperlinks) - các tham chiếu đến các tài liệu và tài nguyên khác cũng được lưu trữ trong Web.

Điều cơ bản đối với trải nghiệm Web của người dùng là khi họ gặp một hình ảnh hoặc đoạn văn bản nhất định trong tài liệu, điều này thường đi kèm với các liên kết đến các tài liệu liên quan và các tài nguyên khác. Cấu trúc của các liên kết có thể phức tạp tùy ý và tập hợp các tài nguyên có thể được thêm vào là không giới hạn - ‘web’ các liên kết thực sự rộng khắp thế giới. Bush [1945] quan niệm về cấu trúc siêu văn bản hơn 50 năm trước; Với sự phát triển của Internet, ý tưởng này có thể được thể hiện trên quy mô toàn thế giới.

Web là một hệ thống mở: nó có thể được mở rộng và triển khai theo những cách mới mà không làm ảnh hưởng đến chức năng hiện có của nó . Thứ nhất, hoạt động của nó dựa trên các tiêu chuẩn truyền thông và các tiêu chuẩn tài liệu hoặc nội dung được xuất bản tự do và thực hiện rộng rãi. Ví dụ: có nhiều loại trình duyệt, mỗi loại trong nhiều trường hợp được thực hiện trên một số nền tảng; và có nhiều triển khai của máy chủ web. Bất kỳ trình duyệt tuân thủ nào cũng có thể truy xuất tài nguyên từ bất kỳ máy chủ tuân thủ nào. Vì vậy, người dùng có quyền truy cập vào các trình duyệt trên phần lớn các thiết bị mà họ sử dụng, từ điện thoại di động đến máy tính để bàn.

Thứ hai, Web mở đối với các loại tài nguyên có thể được xuất bản và chia sẻ trên đó. Nói một cách đơn giản nhất, tài nguyên trên Web là một trang web hoặc một số loại nội dung khác có thể được trình bày cho người dùng, chẳng hạn như các tệp phương tiện và tài liệu trong Định dạng tài liệu di động. Chẳng hạn, nếu ai đó phát minh ra một định dạng lưu trữ hình ảnh mới, thì hình ảnh ở định dạng này ngay lập tức có thể được xuất bản trên Web. Người dùng yêu cầu phương tiện để xem hình ảnh ở định dạng mới này, nhưng các trình duyệt được thiết kế để phù hợp với chức năng trình bày nội dung mới dưới dạng ứng dụng ‘helper’ và ‘plug-ins’.

Web đã vượt ra ngoài các tài nguyên dữ liệu đơn giản này để bao gồm các dịch vụ, chẳng hạn như mua hàng điện tử. Nó đã phát triển mà không thay đổi kiến trúc cơ bản của nó. Web dựa trên ba thành phần công nghệ tiêu chuẩn chính:

• the HyperText Markup Language (HTML), một ngôn ngữ để chỉ định nội dung và bố cục của các trang khi chúng được hiển thị bởi các trình duyệt web;

• Uniform Resource Locators (URL), còn được gọi là Uniform Resource Identifiers (URI), xác định tài liệu và các tài nguyên khác được lưu trữ như một phần của Web;

• Một kiến trúc hệ thống máy khách-máy chủ, với các quy tắc tiêu chuẩn cho tương tác (the HyperText Transfer Protocol – HTTP) mà theo đó trình duyệt và máy khách khác tìm nạp tài liệu và các tài nguyên khác từ máy chủ web. Hình 1.7 cho thấy một số máy chủ web và trình duyệt đưa ra yêu cầu đối với chúng. Đây là một tính năng quan trọng mà người dùng có thể định vị và quản lý máy chủ web của riêng họ ở bất kỳ đâu trên Internet.

Bây giờ chúng ta sẽ lần lượt thảo luận về các thành phần này và giải thích hoạt động của các trình duyệt và máy chủ web khi người dùng tìm nạp các trang web và nhấp vào các liên kết bên trong chúng.

HTML • The HyperText Markup Language [www.w3.org II] được sử dụng để chỉ định văn bản và hình ảnh tạo nên nội dung của trang web, đồng thời chỉ định cách chúng được trình bày và định dạng để trình bày cho người dùng. Một trang web chứa các mục có cấu trúc như tiêu đề, đoạn văn, bảng và hình ảnh. HTML cũng được sử dụng để chỉ định các liên kết và tài nguyên nào được liên kết với chúng.

Người dùng có thể tạo HTML bằng tay, sử dụng trình soạn thảo văn bản tiêu chuẩn, nhưng họ thường sử dụng trình biên tập ‘wysiwyg’ nhận biết HTML để tạo HTML từ bố cục mà họ tạo bằng đồ họa. Một đoạn văn bản HTML điển hình sau:

Văn bản HTML này được lưu trữ trong một tệp mà máy chủ web có thể truy cập - giả sử tệp earth.html. Trình duyệt truy xuất nội dung của tệp này từ máy chủ web - trong trường hợp này là máy chủ trên máy tính có tên www.cdk5.net. Trình duyệt đọc nội dung do máy chủ trả về và hiển thị nó thành văn bản và hình ảnh có định dạng được trình bày trên một trang web theo kiểu quen thuộc. Chỉ có trình duyệt - không phải máy chủ - thông dịch văn bản HTML. Nhưng máy chủ không thông báo cho trình duyệt về loại nội dung mà nó đang trả về, để phân biệt với một tài liệu ở Định dạng Tài liệu Di động. Máy chủ có thể suy ra loại nội dung từ phần mở rộng tên tệp ‘.html’.

Lưu ý rằng các chỉ thị HTML, được gọi là thẻ (tags), được bao bởi dấu ngoặc nhọn, chẳng hạn như <P>. Dòng 1 của ví dụ xác định tệp chứa hình ảnh để trình bày. URL của nó là http://www.cdk5.net/WebExample/Images/earth.jpg. Dòng 2 và 5 lần lượt là các lệnh để bắt đầu và kết thúc một đoạn văn. Dòng 3 và 4 chứa văn bản được hiển thị trên trang web ở định dạng đoạn văn chuẩn.

Dòng 4 chỉ định một liên kết trong trang web. Nó chứa từ ‘Moon’ được bao quanh bởi hai thẻ HTML có liên quan, <A HREF...> và </A>. Văn bản giữa các thẻ này là những gì xuất hiện trong liên kết khi nó được trình bày trên trang web. Hầu hết các trình duyệt được định cấu hình để hiển thị văn bản của các liên kết được gạch chân theo mặc định, vì vậy những gì người dùng sẽ thấy trong đoạn đó là:

Welcome to Earth! Visitors may also be interested in taking a look at the Moon.

Trình duyệt ghi lại mối liên kết giữa văn bản được hiển thị của liên kết và URL có trong thẻ <A HREF...> - trong trường hợp này:

http://www.cdk5.net/WebExample/moon.html

Khi người dùng nhấp vào văn bản, trình duyệt sẽ truy xuất tài nguyên được xác định bởi URL tương ứng và trình bày cho người dùng. Trong ví dụ, tài nguyên là một tệp HTML chỉ định một trang web về Mặt trăng.

URL • Mục đích của Bộ định vị Tài nguyên Thống nhất [www.w3.org III] là xác định một tài nguyên. Thật vậy, thuật ngữ được sử dụng trong tài liệu kiến trúc web là Định danh tài nguyên đồng nhất (URI), nhưng trong cuốn sách này, thuật ngữ URL được biết đến nhiều hơn sẽ được sử dụng khi không có sự nhầm lẫn nào có thể xảy ra. Trình duyệt kiểm tra các URL để truy cập các tài nguyên tương ứng. Đôi khi người dùng nhập URL vào trình duyệt. Thông thường hơn, trình duyệt tìm kiếm URL tương ứng khi người dùng nhấp vào liên kết hoặc chọn một trong các 'dấu trang' của họ; hoặc khi trình duyệt tìm nạp tài nguyên được nhúng trong trang web, chẳng hạn như hình ảnh.

Mọi URL, ở dạng đầy đủ, tuyệt đối, đều có hai thành phần cấp cao nhất:

scheme : scheme-specific-identifier

Thành phần đầu tiên, ' scheme ', khai báo đây là loại URL nào. URL được yêu cầu để xác định nhiều loại tài nguyên. Ví dụ: mailto: joe@anISP.net xác định địa chỉ email của người dùng; ftp://ftp.downloadIt.com/software/aProg.exe xác định tệp sẽ được truy xuất bằng Giao thức truyền tệp (FTP) chứ không phải là giao thức HTTP được sử dụng phổ biến hơn. Các ví dụ khác về lược đồ là ‘tel’ (được sử dụng để chỉ định số điện thoại để gọi, đặc biệt hữu ích khi duyệt trên điện thoại di động) và ‘tag’ (được sử dụng để xác định một thực thể tùy ý).

Web mở đối với các loại tài nguyên mà nó có thể được sử dụng để truy cập, nhờ các bộ chỉ định lược đồ trong URL. Nếu ai đó phát minh ra một loại tài nguyên ‘widget’ mới hữu ích - có lẽ với lược đồ định địa chỉ riêng để định vị các widget và giao thức riêng để truy cập chúng - thì thế giới có thể bắt đầu sử dụng các URL có dạng widget: .... Tất nhiên, các trình duyệt phải được cung cấp khả năng sử dụng giao thức ‘widget’ mới, nhưng điều này có thể được thực hiện bằng cách thêm một trình cắm thêm.

HTTP URLs được sử dụng rộng rãi nhất, để truy cập tài nguyên bằng giao thức HTTP tiêu chuẩn. Một URL HTTP có hai công việc chính: xác định máy chủ web nào duy trì tài nguyên và xác định tài nguyên nào tại máy chủ đó được yêu cầu. Hình 1.7 cho thấy ba trình duyệt đưa ra yêu cầu đối với tài nguyên được quản lý bởi ba web may chủ. Trình duyệt trên cùng đang đưa ra một truy vấn cho một công cụ tìm kiếm. Trình duyệt giữa yêu cầu trang mặc định của một trang web khác. Trình duyệt dưới cùng yêu cầu một trang web được chỉ định đầy đủ, bao gồm tên đường dẫn liên quan đến máy chủ. Các tệp cho một máy chủ web nhất định được duy trì trong một hoặc nhiều cây con (thư mục) của hệ thống tệp của máy chủ và mỗi tài nguyên được xác định bằng tên đường dẫn liên quan đến máy chủ.

Nói chung, các HTTP URLs có dạng sau:

http:// servername [:port] [/pathName] [?query] [ #fragment]

trong đó các mục trong dấu ngoặc vuông là tùy chọn. URL HTTP đầy đủ luôn bắt đầu bằng chuỗi ‘http: //’ theo sau là tên máy chủ, được biểu thị dưới dạng tên Domain Name System (DNS ) . Tên DNS của máy chủ được tùy chọn theo sau bởi số 'cổng' mà máy chủ lắng nghe các yêu cầu , theo mặc định là 80. Sau đó, xuất hiện tên đường dẫn tùy chọn của tài nguyên của máy chủ. Nếu điều này không có thì trang web mặc định của máy chủ là bắt buộc. Cuối cùng, URL tùy chọn kết thúc trong một thành phần truy vấn - ví dụ: khi người dùng gửi các mục nhập trong một biểu mẫu chẳng hạn như tìm kiếm trang truy vấn của engine – và / hoặc số nhận dạng phân đoạn, xác định một thành phần của tài nguyên.

Xem xét các URL:

http://www.cdk5.net

http://www.w3.org/standards/faq.html#conformance

http://www.google.com/search?q=obama

Chúng có thể được chia nhỏ như sau:

URL đầu tiên chỉ định trang mặc định do www.cdk5.net cung cấp. Phần tiếp theo xác định một đoạn của tệp HTML có tên đường dẫn là tiêu chuẩn / faq.html liên quan đến máy chủ www.w3.org. Số nhận dạng của phân đoạn (được chỉ định sau ký tự ‘#’ trong URL) là phần giới thiệu và trình duyệt sẽ tìm kiếm số nhận dạng phân đoạn đó trong văn bản HTML sau khi đã tải xuống toàn bộ tệp. URL thứ ba chỉ định một truy vấn cho một công cụ tìm kiếm. Đường dẫn xác định một chương trình có tên là ‘tìm kiếm’ và chuỗi sau ký tự ‘?’ Mã hóa một chuỗi truy vấn được cung cấp dưới dạng đối số cho chương trình này. Chúng tôi thảo luận về các URL xác định tài nguyên có lập trình chi tiết hơn khi chúng tôi xem xét các tính năng nâng cao hơn bên dưới.

Xuất bản tài nguyên: Mặc dù Web có một mô hình được xác định rõ ràng để truy cập nguồn từ URL của nó, nhưng các phương pháp chính xác để xuất bản tài nguyên trên Web phụ thuộc vào việc triển khai máy chủ web. Về cơ chế cấp thấp, phương pháp đơn giản nhất để xuất bản tài nguyên trên Web là đặt tệp tương ứng vào một thư mục mà máy chủ web có thể truy cập. Khi biết tên của máy chủ S và tên đường dẫn cho tệp P mà máy chủ có thể nhận ra, người dùng sau đó sẽ xây dựng URL trottp: // S / P. Người dùng đặt URL này trong một liên kết từ một tài liệu hiện có hoặc phân phối URL cho những người dùng khác, chẳng hạn như qua email.

Thông thường những mối quan tâm như vậy sẽ bị che giấu khỏi người dùng khi họ tạo nội dung. Ví dụ: ‘người viết blog’ thường sử dụng các công cụ phần mềm, được triển khai dưới dạng các trang web, để tạo bộ sưu tập các trang tạp chí có tổ chức. Các trang sản phẩm cho trang web của một công ty thường được tạo bằng hệ thống quản lý nội dung, một lần nữa bằng cách tương tác trực tiếp với trang web thông qua các trang web quản trị. Cơ sở dữ liệu hoặc hệ thống tệp mà các trang sản phẩm dựa trên đó là minh bạch.

Cuối cùng, Huang và cộng sự. [2000] cung cấp một mô hình để chèn nội dung vào Web với sự can thiệp tối thiểu của con người. Điều này đặc biệt phù hợp khi người dùng cần trích xuất nội dung từ nhiều thiết bị khác nhau, chẳng hạn như máy ảnh, để xuất bản trên các trang web.

HTTP • The HyperText Transfer Protocol [www.w3.org IV] xác định các cách thức mà trình duyệt và các loại máy khách khác tương tác với máy chủ web. Chương 5 sẽ xem xét HTTP chi tiết hơn, nhưng ở đây chúng tôi phác thảo các tính năng chính của nó (giới hạn cuộc thảo luận của chúng tôi đối với việc truy xuất tài nguyên trong tệp):

Tương tác yêu cầu trả lời: HTTP là một giao thức "yêu cầu trả lời". Máy khách gửi thông báo yêu cầu đến máy chủ chứa URL của tài nguyên được yêu cầu. Máy chủ tra cứu tên đường dẫn và nếu nó tồn tại, sẽ gửi lại nội dung của tài nguyên trong một tin nhắn trả lời cho máy khách. Nếu không, nó sẽ gửi lại một phản hồi lỗi chẳng hạn như quen thuộc '404 Not Found'. HTTP xác định một tập hợp nhỏ các hoạt động hoặc phương thức có thể được thực hiện trên một tài nguyên. Phổ biến nhất là GET, để lấy dữ liệu từ tài nguyên và POST, để cung cấp dữ liệu cho tài nguyên.

Loại nội dung: Các trình duyệt không nhất thiết có khả năng xử lý mọi loại nội dung. Khi trình duyệt đưa ra yêu cầu, trình duyệt sẽ bao gồm danh sách các loại nội dung mà trình duyệt ưa thích - ví dụ: về nguyên tắc, trình duyệt có thể hiển thị hình ảnh ở định dạng ‘GIF’ nhưng không hiển thị định dạng ‘JPEG’. Máy chủ có thể tính đến điều này khi trả về nội dung cho trình duyệt. Máy chủ bao gồm loại nội dung trong thư trả lời để trình duyệt biết cách xử lý. Các chuỗi biểu thị kiểu nội dung được gọi là kiểu MIME và chúng được chuẩn hóa trong RFC 1521 [Freed và Borenstein 1996]. Ví dụ: nếu nội dung thuộc loại ‘text / html’ thì trình duyệt sẽ diễn giải văn bản dưới dạng HTML và hiển thị nó; nếu nội dung thuộc loại ‘image / GIF’ thì trình duyệt sẽ hiển thị nội dung đó dưới dạng hình ảnh ở định dạng ‘GIF’; nếu loại nội dung là ‘application / zip’ thì đó là dữ liệu được nén ở định dạng ‘zip’ và trình duyệt sẽ khởi chạy ứng dụng trợ giúp bên ngoài để giải nén. Tập hợp các hành động mà trình duyệt sẽ thực hiện đối với một loại nội dung nhất định có thể định cấu hình và người đọc có thể quan tâm đến việc kiểm tra các cài đặt này cho trình duyệt của riêng họ.

Một tài nguyên cho mỗi yêu cầu: Khách hàng chỉ định một tài nguyên cho mỗi yêu cầu HTTP. Ví dụ: nếu một trang web chứa chín hình ảnh, thì trình duyệt sẽ đưa ra tổng cộng mười yêu cầu riêng biệt để lấy toàn bộ nội dung của trang. Các trình duyệt thường đưa ra một số yêu cầu đồng thời để giảm độ trễ tổng thể cho người dùng.

Kiểm soát truy cập đơn giản: Theo mặc định, bất kỳ người dùng nào có kết nối mạng với máy chủ web đều có thể truy cập bất kỳ tài nguyên nào đã xuất bản của nó. Nếu người dùng muốn hạn chế quyền truy cập vào một tài nguyên, thì họ có thể định cấu hình máy chủ để đưa ra 'thử thách' cho bất kỳ ứng dụng nào yêu cầu nó. Người dùng tương ứng sau đó phải chứng minh rằng họ có quyền truy cập tài nguyên, chẳng hạn bằng cách nhập mật khẩu.

Trang động • Cho đến nay, chúng tôi đã mô tả cách người dùng có thể xuất bản các trang web và nội dung khác được lưu trữ trong các tệp trên Web. Tuy nhiên, phần lớn trải nghiệm của người dùng trên Web là tương tác với các dịch vụ thay vì truy xuất dữ liệu. Ví dụ: khi mua một mặt hàng tại cửa hàng trực tuyến, người dùng thường điền vào biểu mẫu web để cung cấp chi tiết cá nhân hoặc để chỉ định chính xác những gì họ muốn mua. Biểu mẫu web là một trang web chứa các hướng dẫn cho người dùng và các tiện ích con nhập như trường văn bản và hộp kiểm. Khi người dùng gửi biểu mẫu (thường bằng cách nhấn nút hoặc phím ‘quay lại’), trình duyệt sẽ gửi một yêu cầu HTTP đến máy chủ web, chứa các giá trị mà người dùng đã nhập.

Vì kết quả của yêu cầu phụ thuộc vào đầu vào của người dùng, máy chủ sẽ xử lý đầu vào của người dùng. Do đó, URL hoặc thành phần ban đầu của nó chỉ định một chương trình trên máy chủ, không phải một tệp. Nếu đầu vào của người dùng là một tập hợp các tham số nhỏ hợp lý, nó thường được gửi dưới dạng thành phần truy vấn của URL, bằng cách sử dụng phương thức GET; cách khác, nó được gửi dưới dạng dữ liệu bổ sung trong yêu cầu bằng phương thức POST. Ví dụ: một yêu cầu chứa URL sau sẽ gọi một chương trình được gọi là 'search' tại www.google.com và chỉ định một chuỗi truy vấn là "obama": http://www.google.com/search?q=obama.

Chương trình ‘tìm kiếm’ đó tạo ra văn bản HTML làm đầu ra của nó và người dùng sẽ thấy danh sách các trang có chứa từ ‘obama’. (Người đọc có thể quan tâm đến việc nhập một truy vấn vào công cụ tìm kiếm yêu thích của họ và nhận thấy URL mà trình duyệt hiển thị khi kết quả được trả về.) Máy chủ trả về văn bản HTML mà chương trình tạo ra giống như thể nó đã truy xuất nó từ một tệp. Nói cách khác, sự khác biệt giữa nội dung tĩnh được tìm nạp từ một tệp và nội dung được tạo động là trong suốt đối với trình duyệt.

Một chương trình mà máy chủ web chạy để tạo nội dung cho khách hàng của họ được gọi là chương trình Giao diện cổng chung (CGI). Chương trình CGI có thể có bất kỳ chức năng nào dành riêng cho ứng dụng, miễn là nó có thể phân tích cú pháp các đối số mà máy khách cung cấp cho nó và tạo ra nội dung thuộc loại bắt buộc (thường là văn bản HTML). Chương trình thường sẽ tham khảo hoặc cập nhật cơ sở dữ liệu trong quá trình xử lý yêu cầu.

Mã đã tải xuống: Một chương trình CGI chạy tại máy chủ. Đôi khi các nhà thiết kế dịch vụ web yêu cầu một số mã liên quan đến dịch vụ để chạy bên trong trình duyệt, trên máy tính của người dùng. Đặc biệt, mã được viết bằng Javascript [www.netscape.com] thường được tải xuống cùng với trang web có chứa biểu mẫu, để cung cấp chất lượng tương tác tốt hơn với người dùng so với mã được hỗ trợ bởi các tiện ích con chuẩn của HTML. Trang được nâng cao bằng Javascript có thể cung cấp cho người dùng phản hồi ngay lập tức về các mục nhập không hợp lệ, thay vì buộc người dùng kiểm tra các giá trị tại máy chủ, điều này sẽ mất nhiều thời gian hơn.

Javascript cũng có thể được sử dụng để cập nhật các phần nội dung của trang web mà không cần tìm nạp phiên bản hoàn toàn mới của trang và hiển thị lại. Các cập nhật động này xảy ra do hành động của người dùng (chẳng hạn như nhấp vào liên kết hoặc nút radio) hoặc khi trình duyệt thu được dữ liệu mới từ máy chủ đã cung cấp trang web. Trong phần sau trường hợp, vì thời điểm dữ liệu đến không được kết nối với bất kỳ hành động nào của người dùng trên chính trình duyệt, nên nó được gọi là không đồng bộ. Một kỹ thuật được gọi là AJAX (Javascript và XML không đồng bộ) được sử dụng trong những trường hợp như vậy. AJAX được mô tả đầy đủ hơn trong Phần 2.3.2.

Một chương trình thay thế cho chương trình Javascript là một applet: một ứng dụng được viết bằng ngôn ngữ Java [Flanagan 2002], mà trình duyệt tự động tải xuống và chạy khi nó tìm nạp một trang web tương ứng. Applet có thể truy cập mạng và cung cấp giao diện người dùng tùy chỉnh. Ví dụ: các ứng dụng ‘trò chuyện’ đôi khi được triển khai dưới dạng các applet chạy trên trình duyệt của người dùng, cùng với một chương trình máy chủ. Các applet gửi văn bản của người dùng đến máy chủ, từ đó phân phối nó đến tất cả các applet để trình bày cho người dùng. Chúng ta sẽ thảo luận chi tiết hơn về các applet trong Phần 2.3.1.

Các dịch vụ web • Cho đến nay, chúng ta đã thảo luận về Web phần lớn từ quan điểm của người dùng vận hành một trình duyệt. Nhưng các chương trình không phải là trình duyệt cũng có thể là ứng dụng khách của Web; thực sự, truy cập có lập trình vào các tài nguyên web là phổ biến.

Tuy nhiên, HTML không đủ cho khả năng tương tác theo chương trình. Ngày càng có nhiều nhu cầu trao đổi nhiều loại dữ liệu có cấu trúc trên Web, nhưng HTML bị hạn chế ở chỗ nó không thể mở rộng cho các ứng dụng ngoài việc duyệt thông tin. HTML có một tập hợp các cấu trúc tĩnh như đoạn văn và chúng được liên kết với nhau theo cách dữ liệu sẽ được trình bày cho người dùng. Ngôn ngữ Đánh dấu Mở rộng (XML) (xem Phần 4.3.3) đã được thiết kế như một cách biểu diễn dữ liệu ở dạng chuẩn, có cấu trúc, dành riêng cho ứng dụng. Về nguyên tắc, dữ liệu được thể hiện trong XML có thể di động giữa các ứng dụng vì nó tự mô tả: nó chứa tên, kiểu và cấu trúc của các phần tử dữ liệu bên trong nó. Ví dụ, XML có thể được sử dụng để mô tả các sản phẩm hoặc thông tin về người dùng, cho nhiều dịch vụ hoặc ứng dụng khác nhau. Trong giao thức HTTP, dữ liệu XML có thể được truyền bằng các hoạt động POST và GET. Trong AJAX, nó có thể được sử dụng để cung cấp dữ liệu cho các chương trình Javascript trên các trình duyệt.

Tài nguyên web cung cấp các hoạt động dịch vụ cụ thể. Ví dụ, trong cửa hàng tại amazon.com, các hoạt động dịch vụ web bao gồm một thao tác để đặt sách và một thao tác khác để kiểm tra trạng thái hiện tại của đơn đặt hàng. Như chúng tôi đã đề cập, HTTP cung cấp một tập hợp nhỏ các hoạt động có thể áp dụng cho bất kỳ tài nguyên nào. Chúng chủ yếu bao gồm GET và Phương thức POST trên các tài nguyên hiện có và các thao tác PUT và DELETE tương ứng. để tạo và xóa tài nguyên web. Bất kỳ thao tác nào trên tài nguyên đều có thể được gọi bằng một trong các phương thức GET hoặc POST, với nội dung có cấu trúc được sử dụng để chỉ định các thông số, kết quả và phản hồi lỗi của thao tác. Cái gọi là REST Kiến trúc (REpresentational State Transfer) cho các dịch vụ web [Fielding 2000] áp dụng cách tiếp cận này trên cơ sở khả năng mở rộng của nó: mọi tài nguyên trên Web đều có URL và phản hồi với cùng một nhóm hoạt động, mặc dù việc xử lý các hoạt động có thể khác nhau rất nhiều so với tài nguyên đến tài nguyên. Mặt trái của khả năng mở rộng đó có thể là sự thiếu mạnh mẽ trong cách phần mềm hoạt động. Chương 9 mô tả thêm về REST và xem xét sâu hơn về khung dịch vụ web, cho phép các nhà thiết kế dịch vụ web mô tả cho các lập trình viên một cách cụ thể hơn về các hoạt động dành riêng cho dịch vụ có sẵn và cách khách hàng phải truy cập chúng.

Thảo luận về Web • Thành công phi thường của Web dựa trên sự dễ dàng tương đối mà nhiều nguồn tổ chức và cá nhân có thể xuất bản tài nguyên, tính phù hợp của cấu trúc siêu văn bản của nó để tổ chức nhiều loại thông tin và tính mở của kiến trúc hệ thống của nó. Các tiêu chuẩn dựa trên kiến trúc của nó rất đơn giản và chúng đã được công bố rộng rãi ở giai đoạn đầu. Chúng đã cho phép nhiều loại tài nguyên và dịch vụ mới được tích hợp.

Thành công của Web có một số vấn đề về thiết kế. Đầu tiên, mô hình siêu văn bản của nó đang thất bại ở một số khía cạnh. Nếu một tài nguyên bị xóa hoặc bị di chuyển, những liên kết được gọi là 'treo lủng lẳng' đến tài nguyên đó có thể vẫn còn, gây ra sự thất vọng cho người dùng. Và có một vấn đề quen thuộc là người dùng bị 'lạc trong không gian rộng'. Người dùng thường thấy mình bối rối, theo nhiều liên kết khác nhau, tham chiếu các trang từ một bộ sưu tập các nguồn khác nhau và về độ tin cậy đáng ngờ trong một số trường hợp.

Công cụ tìm kiếm là một giải pháp thay thế rất phổ biến cho việc theo dõi các liên kết như một phương tiện tìm kiếm thông tin trên Web, nhưng chúng không hoàn hảo trong việc tạo ra những gì người dùng có ý định cụ thể. Một cách tiếp cận cho vấn đề này, được ví dụ trong Khung mô tả tài nguyên [www.w3.org V], là tạo ra các từ vựng chuẩn, cú pháp và ngữ nghĩa để diễn đạt siêu dữ liệu về những thứ trong thế giới của chúng ta và gói siêu dữ liệu đó trong các tài nguyên web tương ứng để truy cập có lập trình. Thay vì tìm kiếm các từ xuất hiện trong các trang web, về nguyên tắc, các chương trình có thể thực hiện tìm kiếm dựa trên siêu dữ liệu để biên dịch danh sách các liên kết có liên quan dựa trên ngữ nghĩa sự phù hợp. Nói chung, web chứa các tài nguyên siêu dữ liệu được liên kết là ý nghĩa của web ngữ nghĩa.

Là một kiến trúc hệ thống, Web gặp phải các vấn đề về quy mô. Các máy chủ web phổ biến có thể gặp nhiều 'lần truy cập' mỗi giây và do đó, phản hồi cho người dùng có thể chậm. Chương 2 mô tả việc sử dụng bộ nhớ đệm trong các trình duyệt và máy chủ proxy để tăng khả năng đáp ứng và phân chia tải của máy chủ trên các cụm máy tính.

1.7 Summary

Hệ thống phân tán ở khắp mọi nơi. Internet cho phép người dùng trên khắp thế giới truy cập các dịch vụ của nó ở bất kỳ nơi nào họ có thể ở. Mỗi tổ chức quản lý một mạng nội bộ, mạng này cung cấp các dịch vụ cục bộ và dịch vụ Internet cho người dùng cục bộ và nói chung cung cấp dịch vụ cho những người dùng khác trên Internet. Các hệ thống phân tán nhỏ có thể được xây dựng từ máy tính di động và các thiết bị tính toán nhỏ khác được gắn vào mạng không dây.

Chia sẻ tài nguyên là yếu tố thúc đẩy chính để xây dựng các hệ thống phân tán. Các tài nguyên như máy in, tệp, trang web hoặc bản ghi cơ sở dữ liệu được quản lý theo máy chủ thuộc loại thích hợp. Ví dụ: máy chủ web quản lý các trang web và các tài nguyên web khác. Tài nguyên được truy cập bởi khách hàng - ví dụ, máy khách của máy chủ web thường được gọi là trình duyệt.

Việc xây dựng các hệ thống phân tán tạo ra nhiều thách thức:

Không đồng nhất:Chúng phải được xây dựng từ nhiều mạng khác nhau, hệ điều hành, phần cứng máy tính và ngôn ngữ lập trình. Các giao thức truyền thông Internet che dấu sự khác biệt trong mạng và phần mềm trung gian có thể giải quyết những khác biệt khác.

Độ mở:Hệ thống phân tán nên có thể mở rộng - bước đầu tiên là xuất bản giao diện của các thành phần, nhưng việc tích hợp các thành phần được viết bởi các lập trình viên khác nhau là một thách thức thực sự.

Bảo mật: Mã hóa có thể được sử dụng để cung cấp bảo vệ đầy đủ các tài nguyên được chia sẻ và giữ bí mật thông tin nhạy cảm khi được truyền trong tin nhắn qua

mạng lưới. Từ chối các cuộc tấn công dịch vụ vẫn là một vấn đề.

Khả năng mở rộng:Một hệ thống phân tán có thể mở rộng nếu chi phí thêm người dùng là một số tiền không đổi về tài nguyên phải được thêm vào. Các thuật toán được sử dụng để truy cập dữ liệu được chia sẻ nên tránh tắc nghẽn hiệu suất và dữ liệu nên được cấu trúc theo thứ bậc để có được thời gian truy cập tốt nhất. Dữ liệu được truy cập thường xuyên có thể được sao chép.

Xử lý lỗi: Bất kỳ quá trình, máy tính hoặc mạng nào cũng có thể bị lỗi một cách độc lập với khác. Do đó, mỗi thành phần cần phải nhận thức được các cách có thể xảy ra mà các thành phần phụ thuộc vào có thể bị lỗi và được thiết kế để đối phó với từng hư hỏng đó một cách thích hợp.

Đồng thời: Sự hiện diện của nhiều người dùng trong một hệ thống phân tán là một nguồn của các yêu cầu đồng thời đối với tài nguyên của nó. Mỗi tài nguyên phải được thiết kế để an toàn trong một môi trường đồng thời. Tính minh bạch: Mục đích là làm cho một số khía cạnh của phân phối trở nên vô hình đối với người lập trình ứng dụng để họ chỉ cần quan tâm đến thiết kế của ứng dụng cụ thể của họ. Ví dụ, họ không cần quan tâm đến vị trí của nó hoặc chi tiết về cách các hoạt động của nó được các thành phần khác truy cập, hoặc liệu nó có được sao chép hoặc di chuyển. Ngay cả các lỗi của mạng và quy trình có thể được trình bày cho các lập trình viên ứng dụng dưới dạng ngoại lệ - nhưng chúng phải được xử lý. Chất lượng dịch vụ. Cung cấp quyền truy cập vào các dịch vụ trong hệ thống phân tán là không đủ. Đặc biệt, điều quan trọng là phải cung cấp các đảm bảo về các phẩm chất liên quan đến việc tiếp cận dịch vụ đó. Ví dụ về các phẩm chất đó bao gồm các thông số liên quan đến hiệu suất, bảo mật và độ tin cậy.